Coupling isogeometric analysis with deep learning for stability evaluation of rectangular tunnels

Toan Nguyen-Minh [a], Tram Bui-Ngoc [b,f], Jim Shiau [c], Tan Nguyen [a,d,*], Trung Nguyen-Thoi [e,f]

[a] Faculty of Civil Engineering, Ton Duc Thang University, Ho Chi Minh City, Viet Nam
[b] Laboratory for Computational Mechanics, Institute for Computational Science and Artificial Intelligence, Van Lang University, Ho Chi Minh City, Viet Nam
[c] School of Engineering, University of Southern Queensland, Toowoomba 4350, QLD, Australia
[d] Smart Computing in Civil Engineering Research Group, Ton Duc Thang University, Ho Chi Minh City, Viet Nam
[e] Laboratory for Applied and Industrial Mathematics, Institute for Computational Science and Artificial Intelligence, Van Lang University, Ho Chi Minh City, Viet Nam
[f] Faculty of Mechanical - Electrical and Computer Engineering, School of Technology, Van Lang University, Ho Chi Minh City, Viet Nam

A R T I C L E   I N F O

A B S T R A C T

The problem considered in this paper is for the stability evaluation of a rectangular tunnel in undrained clay during lining process. The approach adopted involved the use of isogeometric analysis (IGA) and the upper bound limit analysis formulation for the stability analysis. For the geometrical representation, B-spline basis functions are used to generate a set of B-spline surfaces that define the boundary of the soil domain, allowing for the exact representation of the tunnel geometry. The upper bound limit analysis is then formulated as a second-order cone program (SOCP), which can be solved by using a numerical optimization algorithm. The accuracy and reliability of the proposed method are validated by comparing the results with those published in previous studies. Furthermore, a large dataset is generated by randomly varying the input parameters, and a deep learning model is trained to learn the dataset. The deep learning model is trained using the mean squared error (MSE) metric, which yields an MSE as small as 10^{-6}, indicating the high accuracy and precision of the proposed approach. Feature analysis and Partial Dependence Plots (PDPs) were used to gain insights into physical behavior by identifying important variables and understanding their individual and collective effects. In conclusion, the coupling of IGA and upper bound limit analysis provides a comprehensive and reliable solution to the stability of rectangular tunnels. The results obtained from this approach are accurate and can be achieved with reduced computational cost, making it an attractive approach for practical engineering applications. This approach can be used as a basis for future research on tunnel and tunnel stability analysis and may be extended to other types of soil structures under complex geometries.

1. Introduction

Tunnels have become a vital infrastructure for transportation, storage, and underground resource protection. As they are built underground, their stability is of utmost importance during both the design and construction phases. The tunnel stability refers to its ability to maintain its shape and support its own weight, as well as any loads imposed on it. Therefore, understanding the behavior of tunnels in undrained soils and developing appropriate measures to ensure their stability is of paramount importance to engineers and designers (Kim and Yoo, 2005). Unlike the comprehensive assessment of a tunnel's lifelong service, which incorporates time-dependent analyses of displacement

and stress perspectives, collapses that occur during the construction phase are rapid and instant. As a result, there is a paramount need for conducting stability evaluations specifically in this construction phase to optimize the methodology and mitigate potential hazards.

A considerable amount of research has been dedicated to investigating the stability of tunnels or tunnels, with scholars employing various approaches. These include analytical approaches (Wu et al., 2022), numerical methods (Gioda and Swoboda, 1999), experimental approaches (Meguid et al., 2008), and hybrid methods that combine experimental, analytical, and/or numerical approaches (Shahbazi et al., 2021; Tsinidis et al., 2016), among others. By utilizing these different techniques, researchers have gained a better understanding of the
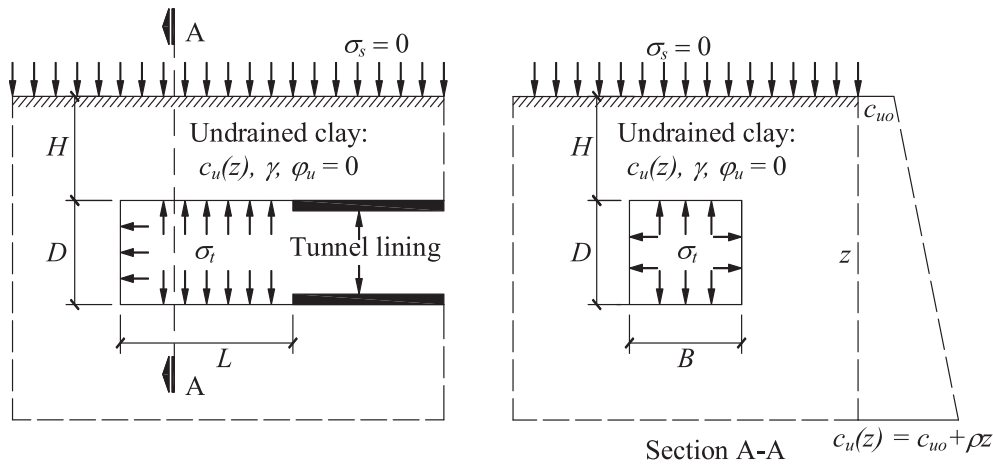
Fig. 1. Plane strain rectangular tunnel.

factors that influence tunnel stability and have developed more effective ways of assessing it. In recent times, significant progress has been achieved in enhancing the safety aspects of tunnel design. Noteworthy contributions include the study conducted by Guo et al. (2021) on estimating the minimum cover depth for underwater shield tunnels, the investigation by (Wang et al., 2019) on the seismic response of tunnel lining structures in thick expansive soil strata, the exploration of blast-induced vibration by Wang et al. (2022b), and the examination of the responses of existing tunnels to nearby excavations as discussed by Meng et al. (2022), and among others.

Among the various methods used to investigate tunnel stability, the finite element limit analysis (FELA) approach has proven to be particularly effective (Tschuchnigg et al., 2015). This is due to its ability to determine failure mechanisms through an optimization process as well as its ability to determine both upper-bound and lower-bound solutions, so that the exact results can bracketed to within a few percentages. The FELA approach was pioneered by Sloan in 1988 and 1989, and since then, numerous studies have been conducted to gain insight into the failure mechanisms of tunnels under different scenarios. These studies have resulted in a vast body of literature on the subject (Wilson et al., 2015; Wilson et al., 2017).

Finite element limit analysis (FELA) is a popular approach for investigating the stability of tunnels. FELA has two main components (Sloan, 1988, 1989): discretizing tools, such as the finite element method (FEM) or smooth FEM (Nguyen and Nguyen-Son, 2022; Vo-Minh and Nguyen, 2022), and optimizing methods, such as linear programming, non-linear programming, or conic programming. Linear finite element is often used for discretizing tunnels; however, it has limitations in accurately modeling complex geometries with curvature boundaries, owing to its nature in linear approximation with straight lines. To overcome this limitation, IsoGeometric Analysis (IGA) was introduced by Hughes et al. (2005), which can precisely model any complex geometries, just like in Computer-Aided Design (CAD). The superiority of IGA has been realized by many researchers, and it has become increasingly popular in the field of FEA. Several noteworthy advancements within the framework of the Isogeometric Analysis (IGA) platform have been proposed in recent times. These advancements include the development of precise geometry representation models specifically tailored for complex geometries, as discussed by Ortiz-Puerta et al. (2022). Furthermore, Mishra (2023) has made significant contributions by addressing three-dimensional (3D) fracture problems within the context of IGA. Additionally, Wang et al. (2022a) have introduced a planar parameterization technique that holds immense potential for IGA applications. In computational geomechanics, IGA has been widely adopted for various topics, such as unsaturated soil mechanics (Shahrokhabadi et al., 2019), poroelasticity (Irzal et al., 2013),

load transfer mechanism of granular material (Nguyen and Tran, 2021), and tunneling (Nguyen, 2014; Vo-Minh et al., 2021).

In this study, IGA is adopted as a discretizing tool for limit analysis, which uses upper-bound theorems to revisit the problems of the undrained stability of rectangular tunnels. Through this approach, this study aims to overcome the limitation of FELA in accurately model complex geometries, providing a more comprehensive understanding of the stability of tunnels under undrained soil conditions. Recent advancements in deep neural networks have shown tremendous potential in solving complex engineering problems (Nguyen et al., 2022). By leveraging large datasets to identify patterns and relationships, these models can significantly improve the accuracy and reliability of analytical tools in engineering. The specific objectives of the study are to: (1) compare the accuracy and reliability of the upper-bound approach based on IGA and SOCP with the results of a previous study; (2) generate a large dataset with 5000 runs for various random input parameters; (3) utilize deep learning techniques, particularly deep feedforward neural networks (DFNNs), to train and learn the dataset; (4) demonstrate the potential of machine learning in combination with traditional analytical tools to provide a more comprehensive and accurate understanding of complex engineering problems, particularly in the field of tunnel stability. In sum, this study is dedicated to facilitating the safe and economical design of the construction process during tunnel lining, enhancing the overall understanding of stability considerations in this critical phase of tunnel construction.

2. Problem definition

The paper deals with rectangular tunnels that possess a width B, height D, and depth H, as shown in Fig. 1. The problem replicates bored tunnels in soft soil, where the excavation progresses by inserting a rigid lining, and an unlined heading with a length L is supported by internal pressure σ_t. The actual practice supports the unlined heading with either compressed air or clay slurry. The heading collapse is driven by the surcharge pressure σ_s and the soil unit weight γ. The assumption of plane strain holds valid when the length L is considerably greater than the width B ($L \gg B$). The soil is deemed as a heterogeneous Tresca material having an undrained cohesion at the ground surface c_{uo} and a strength factor ρ that defines the rate of strength increase with depth. The undrained strength of the soil at the depth z can be represented by Eq. (1).

$$c_u(z) = c_{uo} + \rho z \tag{1}$$

Broms and Bennermark (1967) proposed a way to assess the stability of a tunnel face using a parameter called the stability number. The stability number is defined as in Eq. (2):
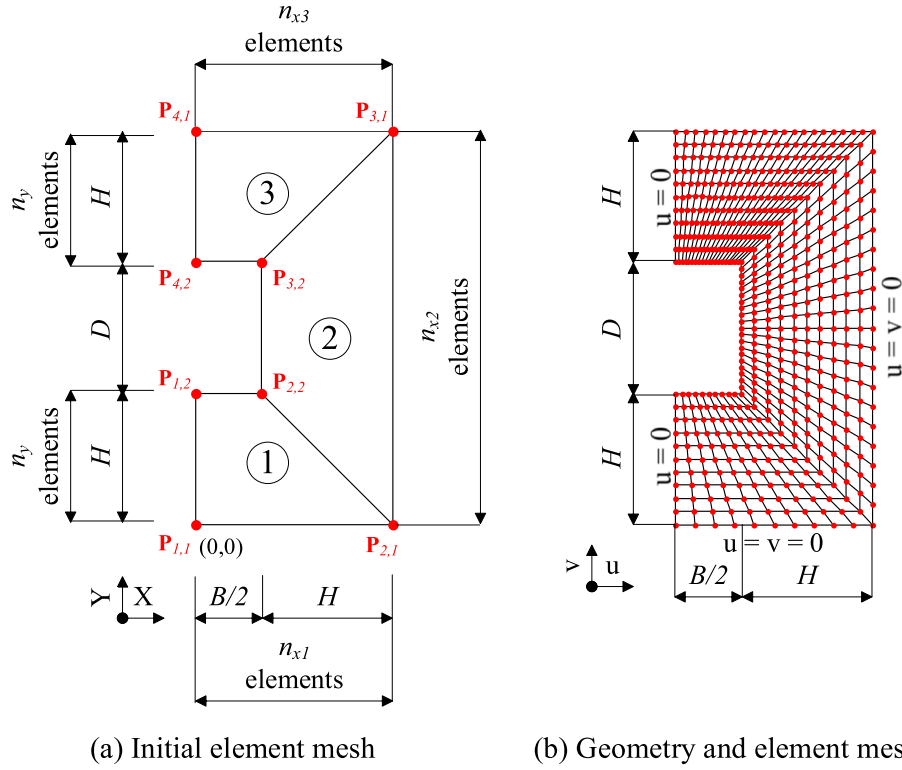
(a) Initial element mesh

(b) Geometry and element mesh

Fig. 2. Schematic of mesh. (a) The initial element with 3 elements. (b) Using knot insertion routine for refinement of the mesh.

Table 1
The coordinates of initial control points for Fig. 2a example.

i	$\mathbf{P}_{i,1}(x,y)$	$\mathbf{P}_{i,2}(x,y)$
1	(0,0)	(0, H)
2	($B/2 + H$, 0)	($B/2$, H)
3	($B/2 + H$, $D + 2H$)	($B/2$, $D + H$)
4	(0, $D + 2H$)	(0, $D + H$)

$$N = \frac{(\sigma_s - \sigma_t)}{c_{uo}} \tag{2}$$

The stability analysis of shallow square tunnels was studied in detail by Assadi and Sloan (1991), while Abbo et al. (2013) used finite element limit analysis methods to analyze the stability of wide rectangular tunnels. Later, Sloan and Assadi (1991) and Wilson et al. (2013) extended this research to include tunnels in soils where undrained shear strength increases linearly with depth. Recent works on the development of tunneling analysis is due to the contribution from Shiau and Al-Asadi (2020); Shiau and Keawsawasvong (2022); Shiau et al. (2022) using three bearing capacity factors.

This study extends the upper bound limit analysis method using isogeometric analysis (IGA) and second-order cone programming (SOCP) to calculate the stability number of rectangular tunnels. The formulations of upper bound limit analysis using simplex strain elements and SOCP were presented by Makrodimopoulos and Martin (2007), while the concept of isogeometric analysis was introduced by Hughes et al. (2005). Our approach utilizes B-spline basis functions to model the exact geometry, which also serves as the basis for the solution space of the numerical method, in accordance with the isoparametric concept. More details will be discussed next.

3. Numerical framework – IsoGeometric analysis (IGA)

Isogeometric analysis (IGA) and finite element analysis (FEA) both use the isoparametric concept. However, IGA reverses the traditional approach by selecting a basis that can exactly represent the geometry and uses it as a basis for the fields that require approximation. In a traditional FEA, geometry is followed by the analysis, while in IGA, geometry determines the analysis. Hughes et al. (2005) proposed the concept of isogeometric analysis using B-spline basis functions that can be used to construct an exact geometric model, thereafter, resulting in a higher-order approach for finite element analysis. A brief discussion of B-spline components in two dimensions is provided in Appendix A1.

3.1. The element analysis

The bivariate B-spline basis functions $R_I(\xi,\eta) = R_{i,j}(\xi,\eta)$ are also used to approximate the displacement fields. This is the same concept as the shape functions in a traditional FEA:

$$\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix} = \sum_{I=1}^{ncp} R_I(\xi,\eta) \begin{bmatrix} u_I \\ v_I \end{bmatrix} = \mathbf{R}(\xi,\eta) \times \mathbf{d};$$
$$I = 1, 2, ..., ncp = n \times m \tag{3}$$

where (u_I, v_I) denote the values of the displacements at the control point $\mathbf{P}_I$ ($\mathbf{P}_{i,j}$) and $ncp = n \times m$ is the number of control points.

The displacement vector $\mathbf{d}$ of control points is stored in the following order:

$$\mathbf{d} = \begin{bmatrix} u_1 & u_2 & ... & u_{ncp} & v_1 & v_2 & ... & v_{ncp} \end{bmatrix}^T \tag{4}$$

Strains are given by:

$$\varepsilon(\xi,\eta) = \partial \mathbf{R}(\xi,\eta) \times \mathbf{d} = \mathbf{B}(\xi,\eta) \times \mathbf{d} \tag{5}$$

where $\mathbf{B}(\xi,\eta) = \partial\mathbf{R}(\xi,\eta)$ is the strain–displacement matric, which is not constant and changes with the value of $\mathbf{R}(\xi,\eta)$.

The number of control points relate to a single two-dimensional element is $(p + 1)^2$, where there are $(1 + 1)^2 = 4$ control points for an element in rectangular tunnel (polynomial order 1). In isogeometric analysis, the Jacobian matrix is used to map the element from physical
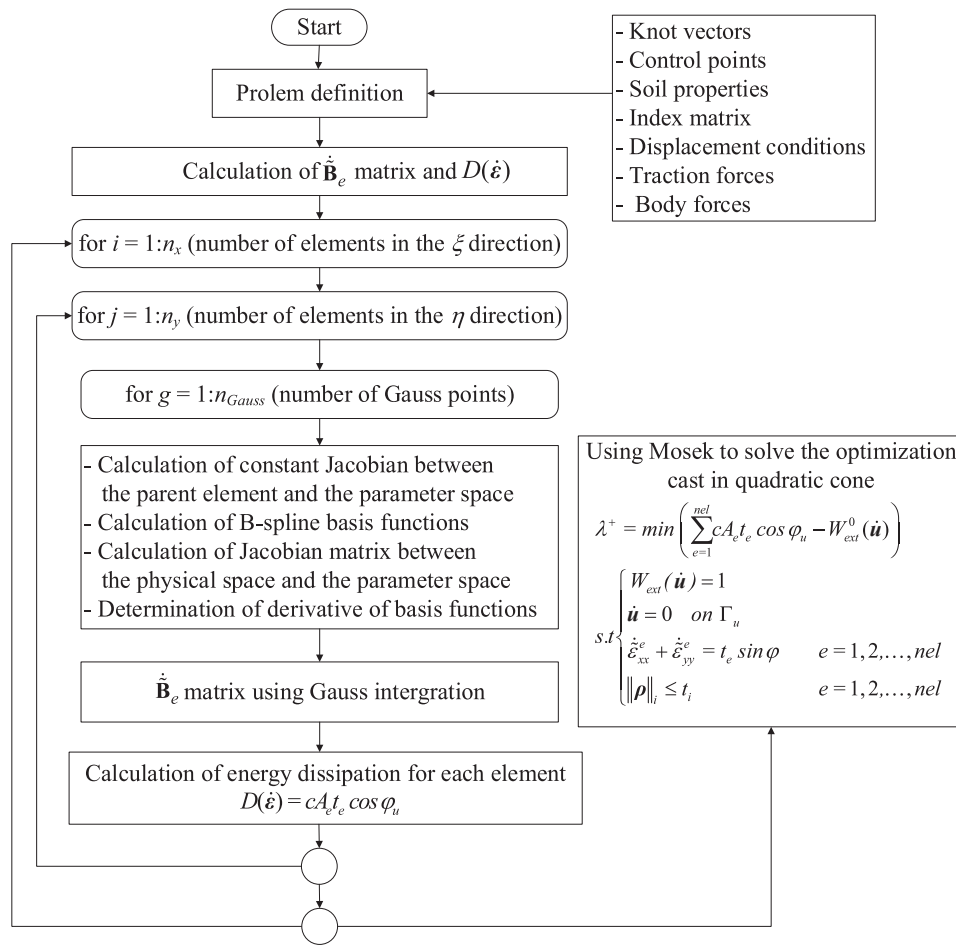
Fig. 3. Illustration of upper bound limit analysis using IGA and SOCP.

Table 2
Comparison of stability number, convergence rate, and CPU time for rectangular tunnels with uniform soil shear strength ($\rho D / c_{uo} = 0$).

H/D	B/D	$\gamma D/c_{uo}$	Stability numbers (N), Abbo et al. (2013)		Stability number (N), this paper[#]; CPU times (seconds)[*]				
			LB	UB	nel = 500	nel = 2,000	nel = 4,500	nel = 8,000	nel = 12,500
1	2	0	0.98	0.98	0.997	0.990	0.987	0.985	0.984
					0.9 s	3.8 s	13.2 s	84.3 s	242.0 s
1	4	0	0.42	0.43	0.438	0.429	0.426	0.424	0.423
					1.0 s	4.4 s	30.8 s	102.3 s	255.6 s
3	3	2	−4.24	−4.18	−4.134	−4.159	−4.175	−4.185	−4.191
					1.5 s	4.2 s	12.2 s	35.7 s	113.1 s
3	4	2	−4.59	−4.56	−4.536	−4.540	−4.549	−4.556	−4.560
					0.9 s	3.8 s	12.6 s	62.2 s	206.3 s
6	4	3	−15.80	−15.68	−15.548	−15.643	−15.667	−15.679	−15.688
					1.1 s	3.6 s	12.4 s	40.8 s	165.2 s
10	3	5	−48.19	−48.00	−47.350	−47.819	−47.933	−47.981	−48.005
					0.7 s	3.7 s	13.2 s	38.6 s	177.3 s

[#] nel: number of IGA elements.
[*] Processor: Intel Core i5 (8 CPUs); Memory: 8192 MB RAM.

space to the parameter space. As the basis functions span the parameter space, which comprises multiple elements, a mapping between each parent element and the parameter space is also required. The parent element in IGA is similar to the element in FEA but without nodes. It is at the parent element level that Gaussian quadrature is employed for numerical integration. Since the parameter space is rectangular, the mapping between the parent element and the parameter space involves a constant Jacobian and linear relationships between the Gauss points and their corresponding parameter values:

$$\xi = \xi_i + \left(\hat{\xi} + 1\right) \frac{\xi_{i+1} - \xi_i}{2}; \quad \eta = \eta_i + \left(\hat{\eta} + 1\right) \frac{\eta_{i+1} - \eta_i}{2} \tag{6}$$

where the hat indicates coordinates on the parent element. The mapping processes in isogeometric analysis are illustrated in Fig. A1.1.

3.2. Numerical simulation

To exploit the symmetry of the problem, the analysis is limited to half of the region. This approach ensures that rigid elements emerge along

Table 3
Comparison of stability number, convergence rate, and CPU time for square tunnels ($B/D = 1$) where the shear strength increases linearly with depth.

H/D	$\rho D/c_{u0}$	$\gamma D/c_{u0}$	Stability numbers (N), Wilson et al. (2013)		Stability numbers (N), this paper[#]; CPU times (seconds)*				
			LB	UB	nel = 500	nel = 2,000	nel = 4,500	nel = 8,000	nel = 12,500
4	0	1	−0.30	−0.12	0.040	−0.062	−0.102	−0.124	−0.138
					0.8 s	3.9 s	13.4 s	41.0 s	176.1 s
4	1	1	12.38	13.06	13.993	13.454	13.229	13.099	13.016
					0.8 s	4.0 s	14.3 s	42.0 s	161.9 s
7	0	3	−17.46	−17.13	−16.657	−17.020	−17.109	−17.148	−17.171
					0.7 s	3.9 s	16.6 s	44.2 s	183.9 s
7	1	3	10.17	12.11	15.286	13.362	12.697	12.353	12.137
					0.8 s	4.6 s	15.9 s	52.4 s	198.5 s

[#] nel: number of IGA elements.
* Processor: Intel Core i5 (8 CPUs); Memory: 8192 MB RAM.
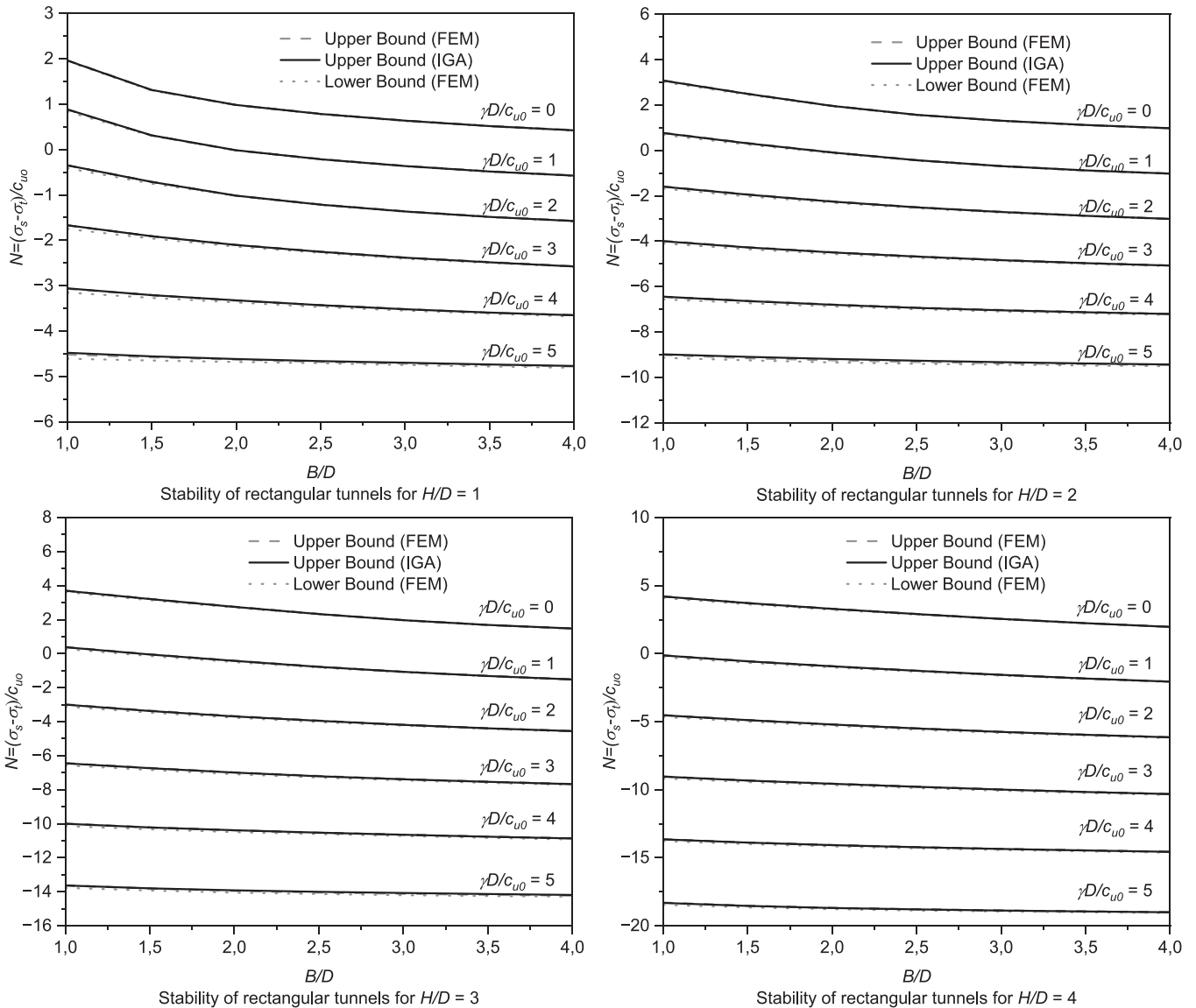
Fig. 4. Comparison of the stability of rectangular tunnels in uniform undrained shear strength between the present study and those given by Abbo et al. (2013) for $H/D = 1$–4.

the complete boundary, as illustrated in Fig. 2. The rectangular tunnel is modeled using the same polynomial orders of B-spline basis functions for both directions, $p = q = 1$. The total number of elements, *nel*, is determined as the product of the number of elements in the ξ and η directions, represented by n_x and n_y, respectively. The number of initial

elements is $nel = 3 \times 1 = 3$, the number of initial control points are $(3 + p) \times (1 + q) = 8$, with $p = q = 1$. The coordinates of initial control points (the red points in Fig. 2a) are shown in Table 1. Also, the initial knot vectors are shown in Eq. (7).
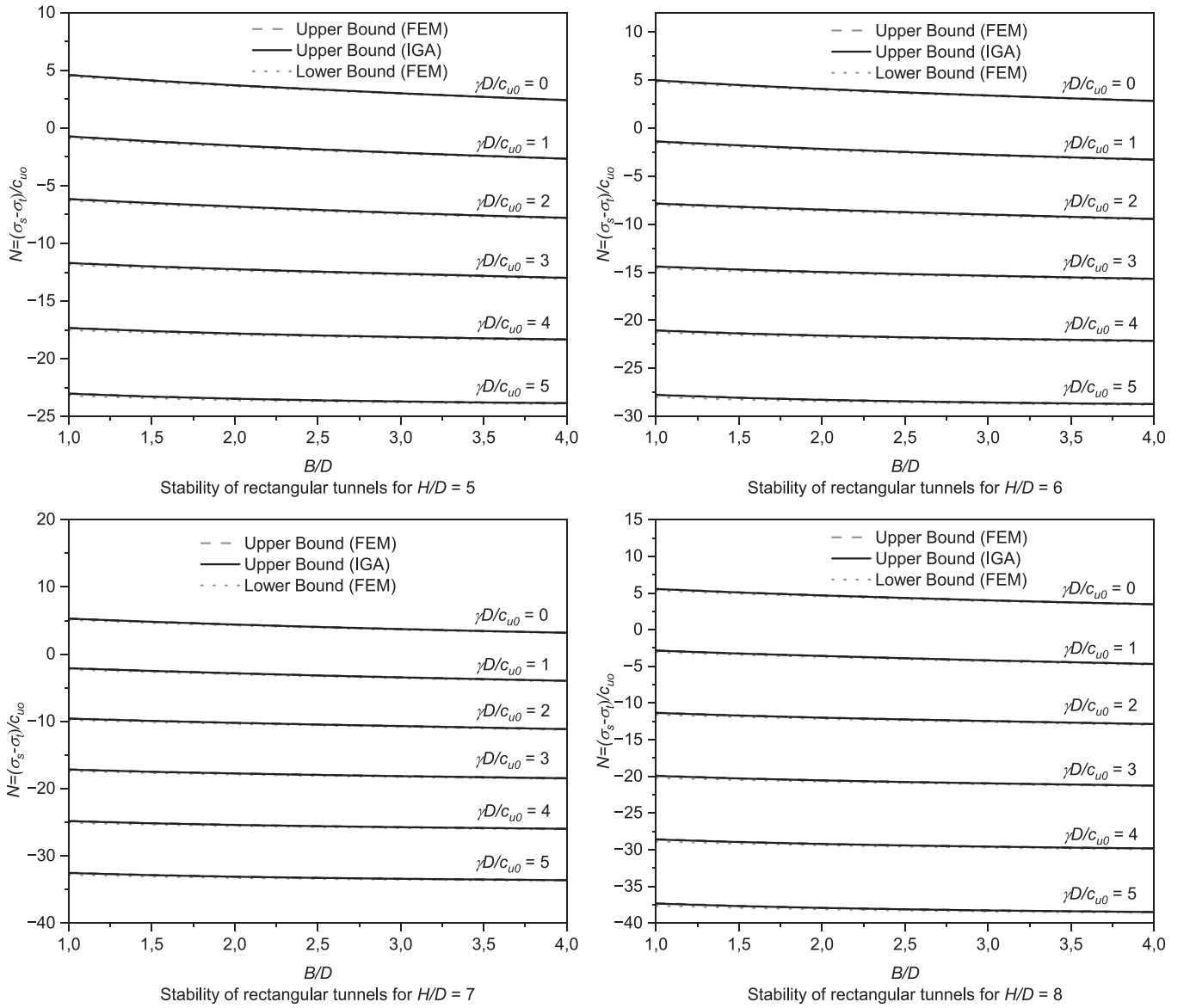
Fig. 5. Comparison of the stability of rectangular tunnels in uniform undrained shear strength between the present study and those given by Abbo et al. (2013) for $H/D = 5$–8.

ξ *direction*: $\Xi = \{0,0,n_{x1},(n_{x1} + n_{x2}),n_x,n_x\}$ (7)

η *direction*: $H = \{0, 0, n_y, n_y\}$.

where $n_x = n_{x1} + n_{x2} + n_{x3}$ (with n_{x1}, n_{x2}, n_{x3} are illustrated in Fig. 2a).

The alignment of control points when the mesh is refined is then determined from a knot insertion routine. Since the formulas are based on a single knot inserted, the routine must be repeated to give the desired number of elements. The total number of elements is $nel = n_x \times n_y = (50 + 2 \times 50 + 2 \times 50) \times 50 = 250 \times 50 = 12,500$ elements.

The global knot vectors expand as knots are inserted:

ξ *direction*: $\Xi = \{0, 0, 1, 2, …,248, 249, 250, 250\}$ (8)

η *direction*: $H = \{0, 0, 1, 2, …, 48, 49, 50, 50\}$.

Lastly, an element in parameter space has an area of $[\xi_i \xi_{i+1}] \times [\eta_j \eta_{j+1}] = [0,1] \times [0,1]$. Therefore, a mapping between each parent element and the parameter space is necessary for using numerical integration.

3.3. Upper bound analysis as second-order cone programming

The upper bound analysis as second-order cone programming formulated by Makrodimopoulos and Martin (2007) is provided in Appendix A2. Introducing an approximation of the displacement and using the smoothed strains, the upper bound limit analysis problem for plane strain can be formulated as:

$$\lambda^+ = min \left(\sum_{e=1}^{nel} cA_e t_e cos\varphi - W_{ext}^0(\dot{\boldsymbol{u}}) \right) \tag{9a}$$

subject to.

$$\begin{cases} W_{ext}(\dot{\boldsymbol{u}}) = 1 \\ \dot{\boldsymbol{u}} = 0 \quad on \; \Gamma_u \\ \dot{\bar{\varepsilon}}_{xx}^e + \dot{\bar{\varepsilon}}_{yy}^e = t_e sin\varphi \quad e = 1, 2, …, nel \\ \|\boldsymbol{\rho}\|_i \leqslant t_i \quad e = 1, 2, …, nel \end{cases} \tag{9b}$$

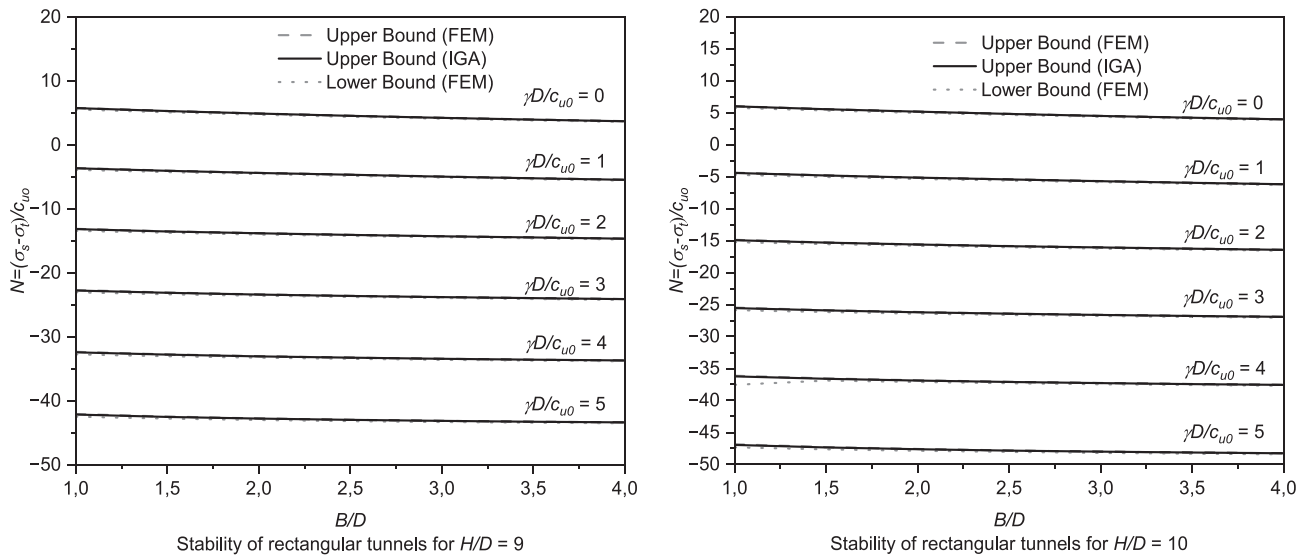where *nel* is the number of elements in the whole investigated domain.

Fig. 6. Comparison of the stability of rectangular tunnels in uniform undrained shear strength between the present study and those given by Abbo et al. (2013) for $H/D = 9$–10.
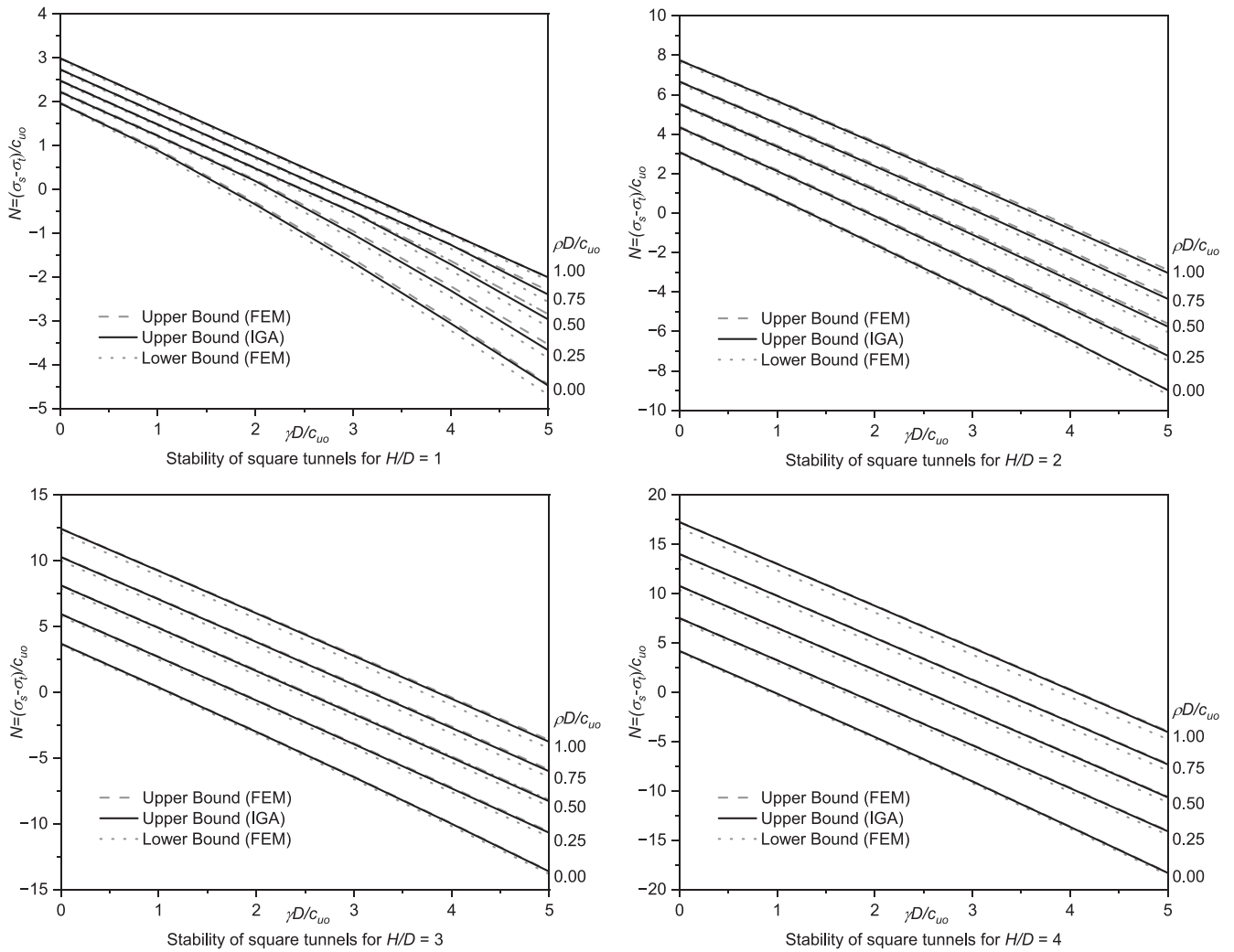

Fig. 7. Comparison of the stability of square tunnels $B/D = 1$ between the present study and those given by Wilson et al. (2013) for $H/D = 1$–4.
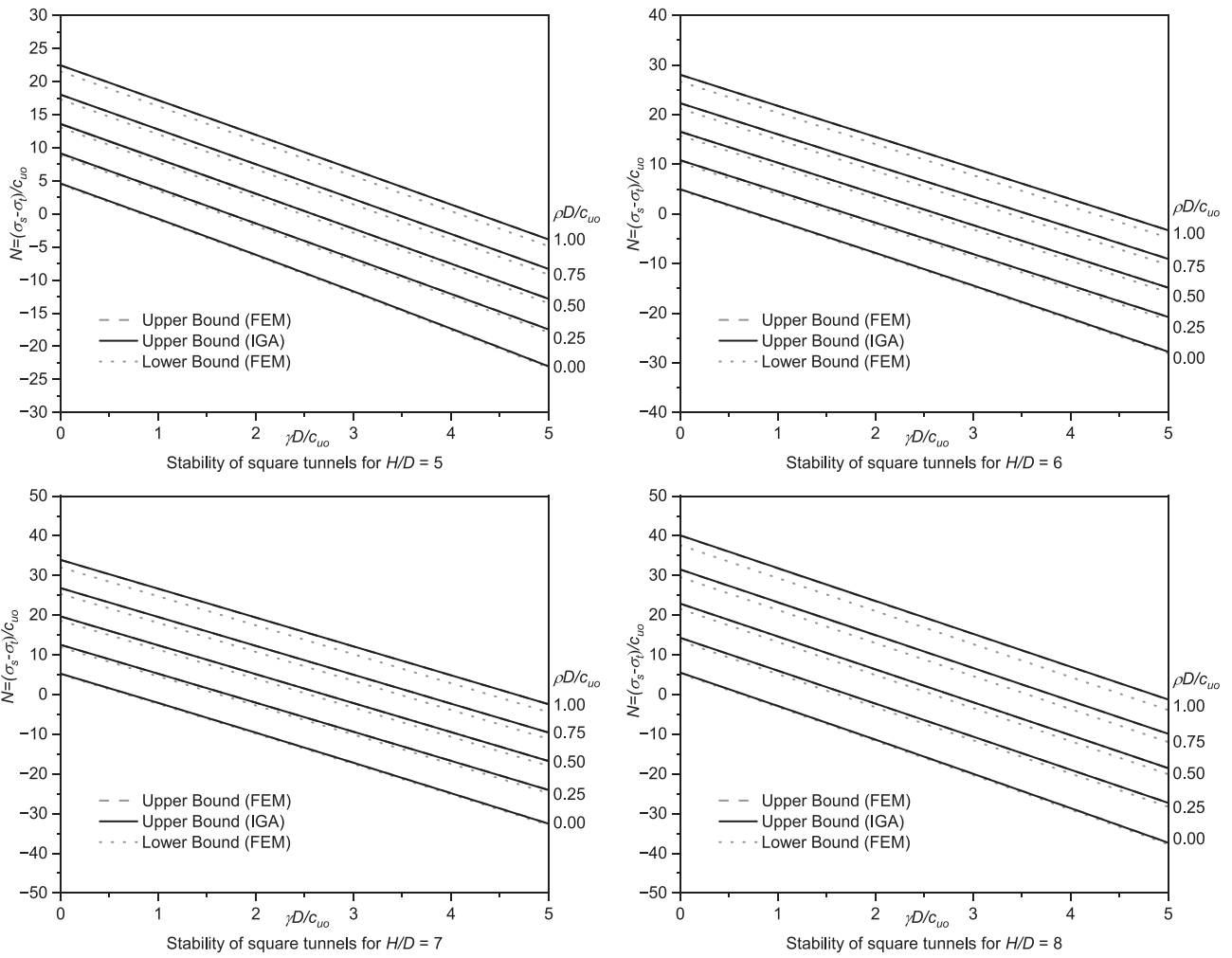
Fig. 8. Comparison of the stability of square tunnels $B/D = 1$ between the present study and those given by Wilson et al. (2013) for $H/D = 5$–8.
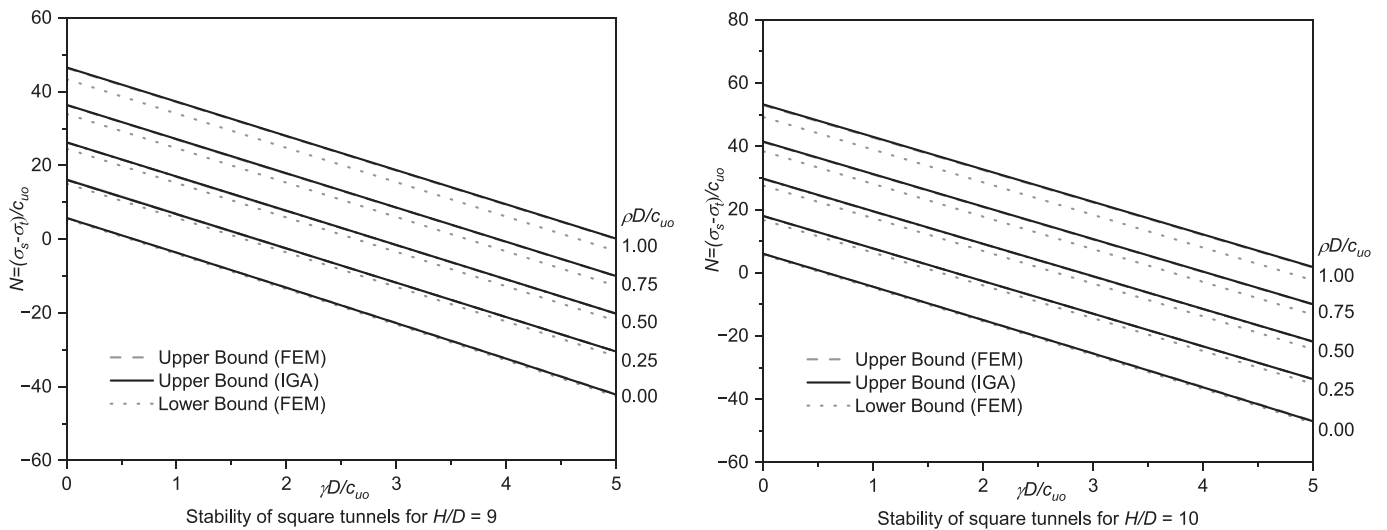

Fig. 9. Comparison of the stability of square tunnels $B/D = 1$ between the present study and those given by Wilson et al. (2013) for $H/D = 9$–10.

(a) $H/D = 1$, $B/D = 2$, $\gamma D/c_u = 0$

(b) $H/D = 1$, $B/D = 4$, $\gamma D/c_u = 0$

(c) $H/D = 3$, $B/D = 3$, $\gamma D/c_u = 2$

(d) $H/D = 3$, $B/D = 4$, $\gamma D/c_u = 2$

(e) $H/D = 6$, $B/D = 4$, $\gamma D/c_u = 3$

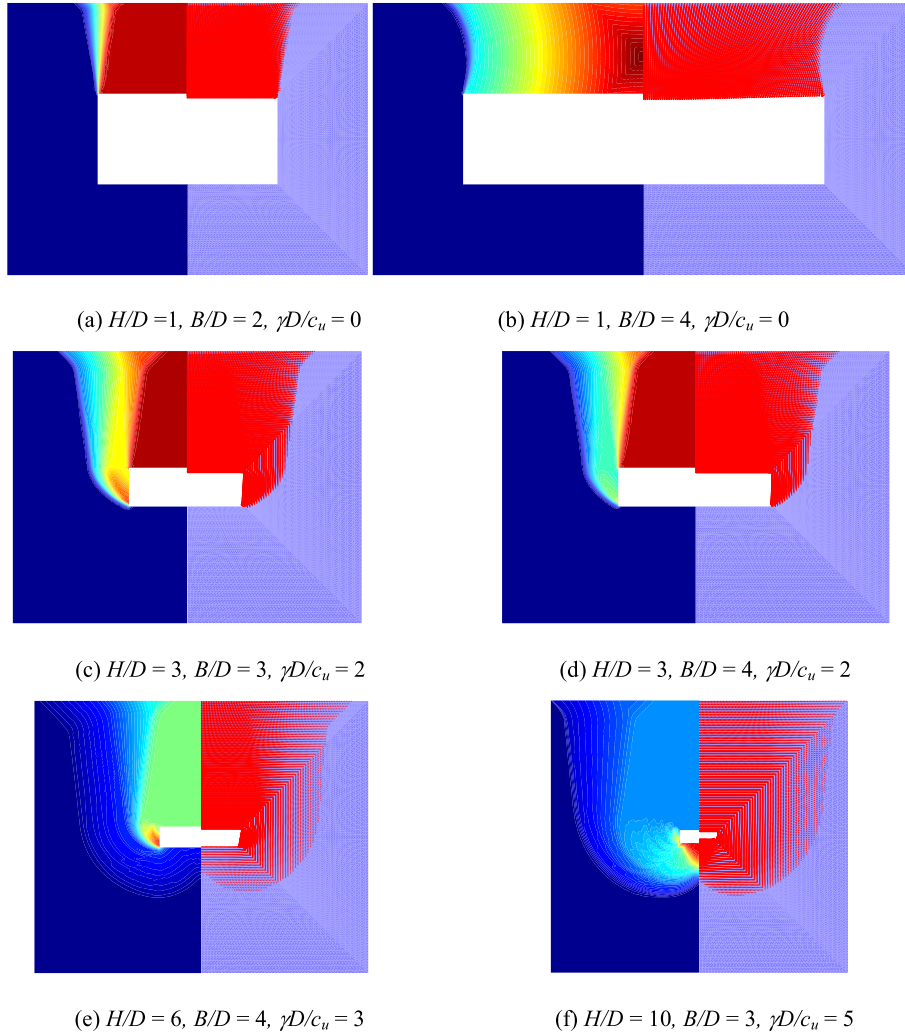(f) $H/D = 10$, $B/D = 3$, $\gamma D/c_u = 5$

Fig. 10. The failure mechanisms of rectangular tunnels in uniform undrained shear strength.

And the fourth constraint in Equation (9b), resulting optimization problem is cast in the form of a second-order cone programming (SOCP) problem so that a large-scale problem can be solved efficiently.

Since strains change with the value of $\mathbf{R}(\xi,\eta)$ (see Eq. (5)), Nguyen (2014) has proposed a framework to obtain the smoothed strains of IGA element e as below:

$$\dot{\tilde{\varepsilon}}_e = \begin{bmatrix} \dot{\tilde{\varepsilon}}_{xx}^e & \dot{\tilde{\varepsilon}}_{yy}^e & \dot{\tilde{\varepsilon}}_{xy}^e \end{bmatrix}^T = \dot{\tilde{B}}_e d_e (d_e = const) \qquad (10)$$

where

$$\dot{\tilde{B}}_e = \begin{bmatrix} \dot{\tilde{R}}_{1,x} & \dot{\tilde{R}}_{2,x} & \dot{\tilde{R}}_{3,x} & \dot{\tilde{R}}_{4,x} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dot{\tilde{R}}_{1,y} & \dot{\tilde{R}}_{2,y} & \dot{\tilde{R}}_{3,y} & \dot{\tilde{R}}_{4,y} \\ \dot{\tilde{R}}_{1,y} & \dot{\tilde{R}}_{2,y} & \dot{\tilde{R}}_{3,y} & \dot{\tilde{R}}_{4,y} & \dot{\tilde{R}}_{1,x} & \dot{\tilde{R}}_{1,x} & \dot{\tilde{R}}_{3,x} & \dot{\tilde{R}}_{4,x} \end{bmatrix} \qquad (11)$$

with

$$\dot{\tilde{R}}_{I,x} = \frac{1}{A_e} \int_{\Omega_e} \dot{R}_{I,x} d\Omega_e \quad \dot{\tilde{R}}_{I,y} = \frac{1}{A_e} \int_{\Omega_e} \dot{R}_{I,y} d\Omega_e$$

$$A_e = \int_{\Omega_e} d\Omega_e; \ I = 1, 2, 3, 4 \qquad (12)$$

In this study, isogeometric analysis is employed to compute the internal power dissipation, the undrained stability number for a domain Ω divided into $nel = n_x \times n_y$ elements. It is worth to mention that the last

constraint in Equation (9b) was formed in the quadratic, allowing the conic interior point optimizer of the academic Mosek to be adopted for solving the optimization problem. The procedure of upper bound limit analysis using IGA and SOCP is illustrated in Fig. 3.

4. Numerical results and validation

The stability number in Equation (2) is described conveniently by the dimensionless parameters. Because the function f is not the precise form, it is necessary to determine discrete values which define it.

$$N = \frac{(\sigma_s - \sigma_t)}{c_{uo}} = f\left(\frac{H}{D}, \frac{B}{D}, \frac{\gamma D}{c_{uo}}, \frac{\rho D}{c_{uo}}\right) \qquad (13)$$

For undrained analysis, the solution is independent of the loading direction (Shiau and Al-Asadi, 2020) and the problem can be simplified by setting the surcharge pressure σ_s to zero. Additionally, assumptions of unity are adopted for both the tunnel height (D) and the undrained shear strength at the ground surface (c_{uo}) in this study. By using dimensionless parameters H/D, B/D, $\gamma D/c_{uo}$, and $\rho D/c_{uo}$, parametric studies are conducted by varying the tunnel depth H, width B, unit weight γ, and soil strength factor ρ to investigate the stability of the rectangular tunnels. For upper bound limit analysis, this study sets $H/D = 1–10$, $B/D = 1–4$, $\gamma D/c_{uo} = 0–5$, and $\rho D/c_{uo} = 0–1$.

In contrast to the previous study conducted by Nguyen (2014) that showcased the superiority of the Isogeometric Analysis (IGA) in modeling circular tunnels, the efficacy of IGA in the context of
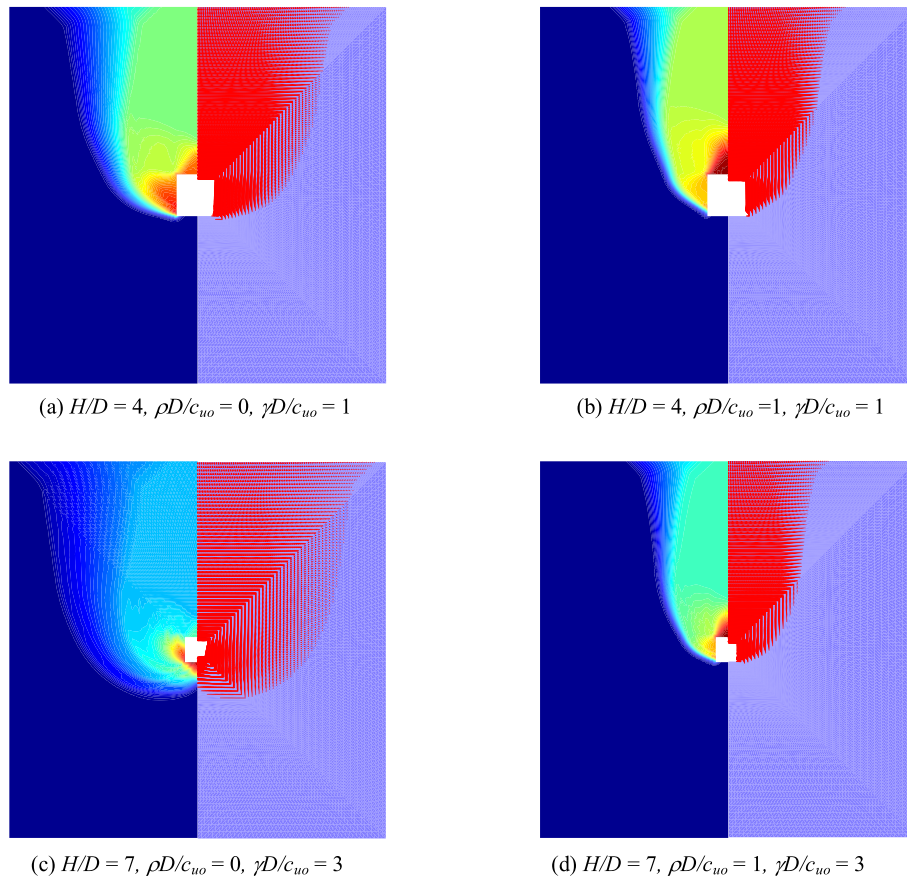
(a) $H/D = 4$, $\rho D/c_{uo} = 0$, $\gamma D/c_{uo} = 1$

(b) $H/D = 4$, $\rho D/c_{uo} = 1$, $\gamma D/c_{uo} = 1$

(c) $H/D = 7$, $\rho D/c_{uo} = 0$, $\gamma D/c_{uo} = 3$

(d) $H/D = 7$, $\rho D/c_{uo} = 1$, $\gamma D/c_{uo} = 3$

Fig. 11. The failure mechanisms of square tunnels where the shear strength increases linearly with depth.
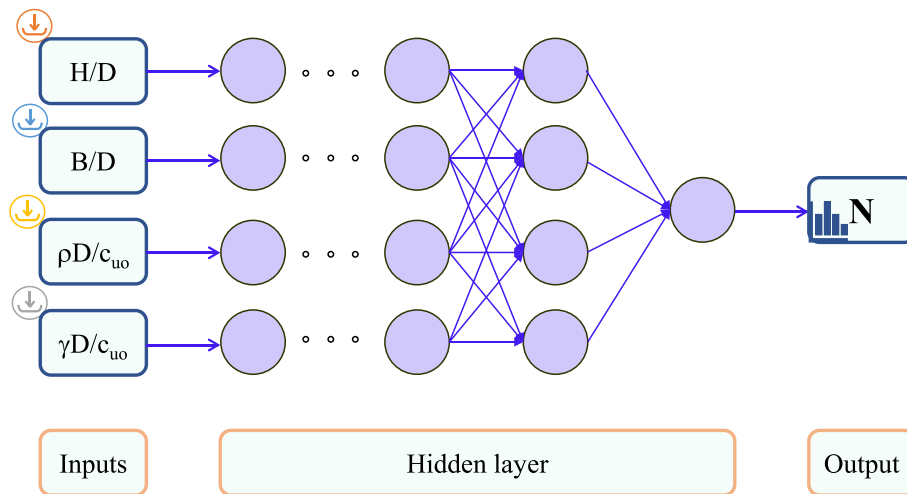

Fig. 12. Typical topology of deep feedforward neural networks.

rectangular tunnels remains unproven. However, the results presented in Tables 2 and 3 indicate that IGA-LA model exhibits promising speed. This advancement holds significant potential for efficiently processing large datasets in machine learning computations, thereby enhancing overall computational efficiency.

It is important to note that a negative stability number N indicates a compressive normal stress that must be applied to the tunnel faces to maintain stability, while a positive stability number means that no tunnel support is required to prevent collapse. A series of convergence study using the present method is shown in Tables 2 and 3. Numerical

results have shown that the number of *nel* would improve the solution accuracy. When the number of elements is 12,500, the stability numbers are between the upper and lower bound results of Abbo et al. (2013) and Wilson et al. (2013). The CPU time for such a refined study with 12,500 elements is approximately 200 s.

The IGA method provides accurate solutions, and they are compatible with those produced by finite element limit analysis, as shown in Figs. 4–9. IGA upper bound solutions are almost the same as those FELA upper bounds. The average stability number difference is approximately 1.10% and 5.92%, respectively, compared to Abbo's and Wilson's

Table 4

Effect of the number of samples in the dataset on the performance model with the 4–10–10–10-1 architecture, Tanh activation, Adam optimization, epoch 500, bath size = 32.

Number of data	MSE ($\times 10^{-5}$)		MAE ($\times 10^{-3}$)		R^2		Time (s)
	Training	Testing	Training	Testing	Training	Testing	
1000	5.81 ± 3.20	6.45 ± 3.53	5.43 ± 1.40	5.59±1.30	0.99764 ± 0.00121	0.99733 ± 0.00196	48.8
2000	1.91 ± 0.85	2.14 ± 0.86	3.29 ± 0.77	3.44 ± 0.77	0.99928 ± 0.00027	0.99916 ± 0.00026	76.3
3000	1.58 ± 0.81	1.68 ± 0.95	3.00 ± 0.93	3.06 ± 0.99	0.99940 ± 0.00013	0.99938 ± 0.00019	81.1
4000	1.27 ± 0.72	1.31 ± 7.24	2.68 ± 0.89	2.71 ± 0.86	0.99952 ± 0.00014	0.99949 ± 0.00019	107.3
5000	**0.48 ± 0.19**	**0.49 ± 0.19**	**1.67 ± 0.34**	**1.69 ± 0.33**	**0.99979 ± 0.00009**	**0.99978 ± 0.00009**	**176.8**

Table 5

The statistical description of the input and output variables.

Variables	Unit	Notation	Min	Mean	Std	Skewness	Max
H/D	–	X_1	1.001	5.512	2.580	−0.002	9.997
B/D	–	X_2	1.000	2.481	0.848	0.036	3.998
$\rho D/c_{u0}$	–	X_3	0.001	0.501	0.287	−0.011	1.000
$\gamma D/c_{u0}$	–	X_4	0.000	2.553	1.456	−0.025	4.994
N	–	Y	−40.562	−2.944	10.722	0.038	40.858
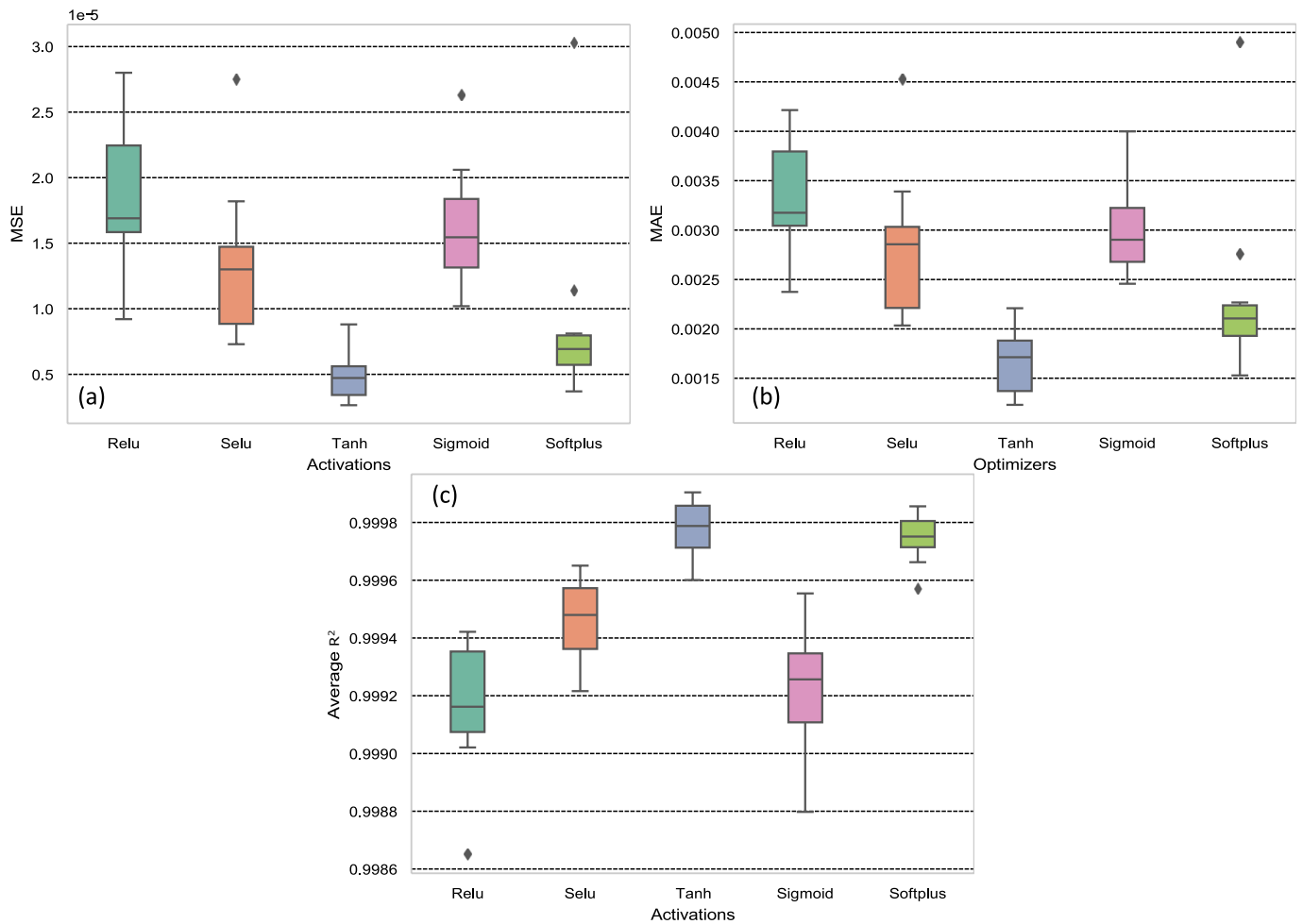
Fig. 13. Comparison of the peformance model of the different activation functions with the 4–10–10–10–1 architecture trained with Adam optimizer algorithm, after 500 epochs, batch size = 32: (a) MSE, (b) MAE, (c) R^2.

models. Moreover, isogeometric analysis uses only *nel* = 12,500 1st order elements, whereas Abbo's model in finite element method (FEM) uses, 199,400 continuum and discontinuity elements, and Wilson's model in finite element method (FEM) uses 100,000 continuum and discontinuity elements. The IGA could significantly reduce the size of

the optimization problem but also minimizes computational resources.

Selected failure mechanisms are shown in Figs. 10 and 11. For homogeneous soil ($\rho D/c_{u0} = 0$), the failure mode is dependent on both the tunnel depth and the unit weight. For shallow tunnel and low unit weight in Fig. 10 (a) and (b), the failure mechanism is more like a roof

failure of the tunnel. For moderate depth and large strength ratio in Fig. 10 (c), (d), (e), (f), the failure mechanisms become wider and includes heaving of the roof, walls, and floor. For the shear strength increases linearly with depth, the failure mechanism is more localized when compared with the uniform strength case. This is illustrated in Fig. 11, where the failure zone is much reduced when $\rho D/c_{uo} = 1$, and there is no plastic deformation below the floor of the tunnel, as observed in Fig. 11 (c) and (d).

## 5. Deep Feedforward neural networks (DFNN)

Deep feedforward neural networks (DFNN) have emerged as a prominent class of deep neural networks owing to their ability to model non-linear relationships, flexibility to accommodate diverse tasks and scalability for handling large datasets. In addition, their ability to train in parallel results in faster training times. Therefore, deep feedforward neural networks have been adopted as the preferred method of investigation in this study for the design of rectangular tunnels. The comparison of our results with those of a prior study is conducted to showcase the accuracy and reliability of the upper-bound approach utilizing Isogeometric Analysis (IGA) and Second-Order Cone Programming (SOCP). Following this, a substantial dataset comprising 5000 runs is generated, encompassing diverse random input parameters, and deep learning techniques are employed to train and analyze the dataset. By utilizing DFNNs, the study aims to harness the power of deep learning techniques to provide accurate and efficient solutions for

practical uses.

5.1. Numerical framework of DFNN model

The Deep Feedforward Neural Network (DFNN) is a fundamental model in the field of deep learning (DL) (Goodfellow et al., 2016). Its purpose is to learn a sophisticated and intricate representation of data in a hierarchical fashion by passing it through multiple layers of transformation (Najafabadi et al., 2015). The architecture of the DFNN typically consists of three parts: the input layer, hidden layer, and output layer, each containing interconnected processing units (Fig. 12). The mathematical description of DFNN is given in Appendix A3.

In the context of a regression problem, the selection of an appropriate activation function is essential in determining the accuracy of the output data for a given neural network architecture. Traditionally, the sigmoid and hyperbolic tangent activation functions have been widely utilized in defining specific output values of the network. However, as the number of layers in the network increases, the use of these smooth nonlinear activation functions fails to capture useful gradient information of the loss function. This phenomenon, known as the vanishing gradient issue, arises when the gradient of the loss function is back-propagated through the network and used to modify internal parameters. The use of these activation functions in hidden layers leads to gradients that approach zero, making it difficult to determine the direction in which the parameters should be adjusted to improve the cost function (Goodfellow et al., 2016). To address this issue, various
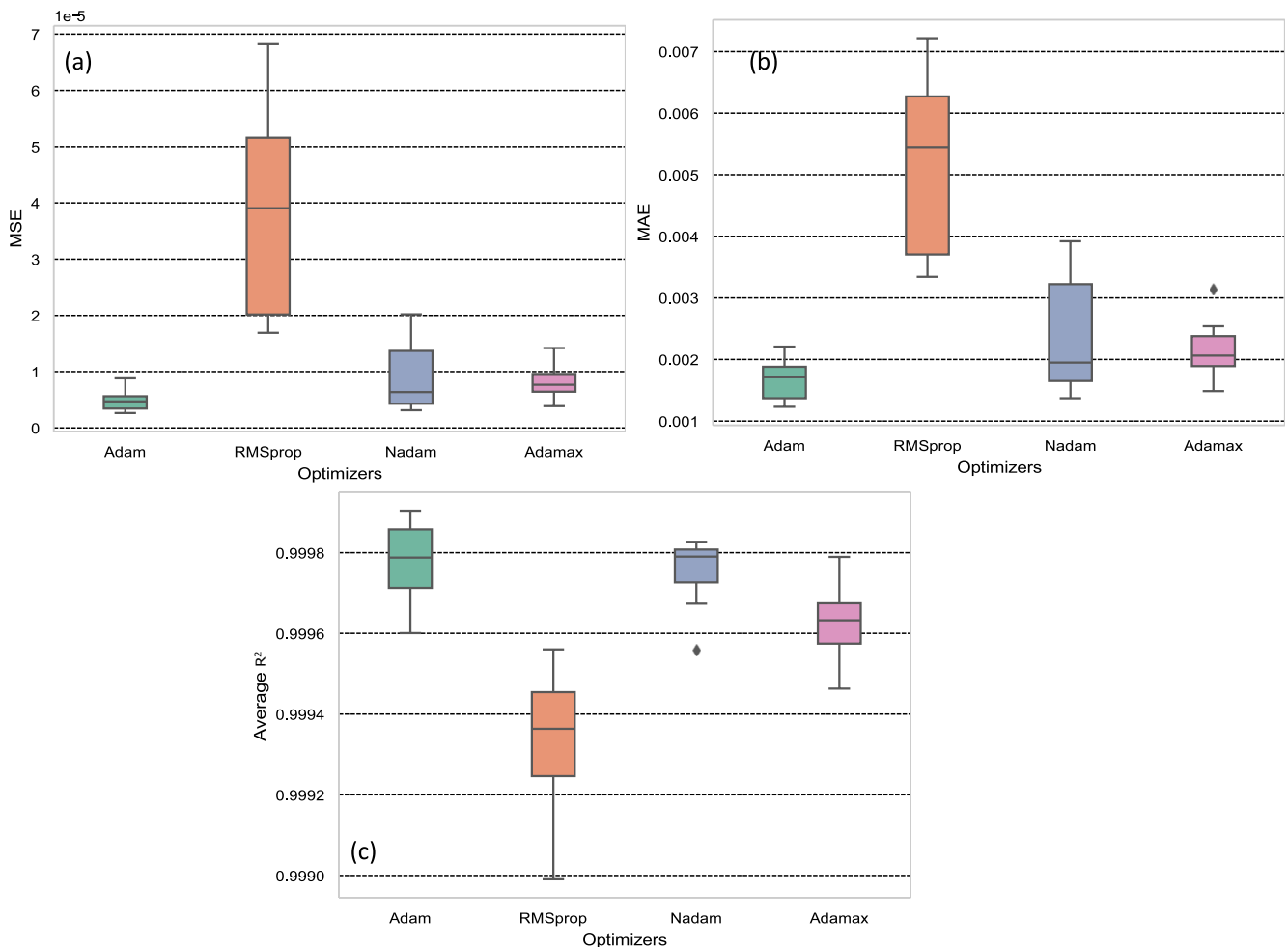
Fig. 14. Comparison of the peformance model of the different optimizer althgorithms with the 4–10–10–10–1 architecture trained with Tanh activation, after 500 epochs, batch size = 32: (a) MSE, (b) MAE, (c) R^2.

activation functions were adopted to find the most appropriate activation function that provides superior performance for the DFNN.

To evaluate the accuracy of the output model's prediction, it is crucial to train the Deep Feedforward Neural Network (DFNN) to minimize the loss function. This study implements the Mean Square Error (MSE) as the loss function for the regression problem, which is a popular technique used to measure the discrepancy between predicted and actual values. MSE is often used in training DFFN models due to its simplicity, computational efficiency, sensitivity to outliers, and alignment with the assumption of Gaussian error distribution commonly used in traditional DNN models (Wani et al., 2020). Additionally, to assess the precision of the model's prediction, this study employs the Mean Absolute Error (MAE) and Coefficient of Determination (R^2) as evaluation metrics (see Appendix A3). These metrics are widely used to evaluate the accuracy of prediction results and provide a comprehensive understanding of the model's performance. By using these evaluation metrics, valuable insights can be obtained regarding the model's predictive accuracy, thereby enabling the identification of potential areas for further improvement if needed.

During the network training process, a specific gradient descent algorithm is used to minimize the MSE loss function. Numerous algorithms have been developed to search for optimal network parameters that can minimize the loss function (Nguyen et al., 2023), including stochastic gradient descent (SGD) (Robbins and Monro, 1951), adaptive gradient algorithm (Adagrad) (Duchi et al., 2011), adaptive moment estimation (Adam) (Kingma and Ba, 2014), RMSprop (D. Gutierrez, 2017), and Adadelta (Zeiler, 2012). While they are similar methods that

work well in similar circumstances (Ruder, 2016), there is no single algorithm that is the most efficient as it depends on the specific problem and conditions.

When optimizing weights in a network, batch gradient descent calculates the error for each sample in the training set, but the model is updated only after evaluating all training samples. This can make model updates and training speed slow for large datasets. To address this issue, the mini-batch technique, introduced by Hinton (2012), is adopted in this study. The mini-batch gradient descent helps split the training dataset into small batches that are used to compute model error and update model parameters (Truong et al., 2020). DFNN networks may encounter an overfitting problem during training. To mitigate this issue, the dropout method proposed by Srivastava et al. (2014) is applied in this study.

5.2. Constructing DFNN model

5.2.1. Dataset

It is important to take into account the impact of the size of a dataset on the performance of a deep feedforward neural network (DFNN) model, particularly when dealing with models that have a specific architecture, such as the one analyzed in this study. The study conducted on this model reveals that the number of 5000 samples is an appropriate choice to obtain reliable and accurate results, as demonstrated by the excellent performance metrics presented in Table 4. The analysis conducted on this dataset provides confidence in the predictive capabilities of the model. The selected sample size has been carefully considered to
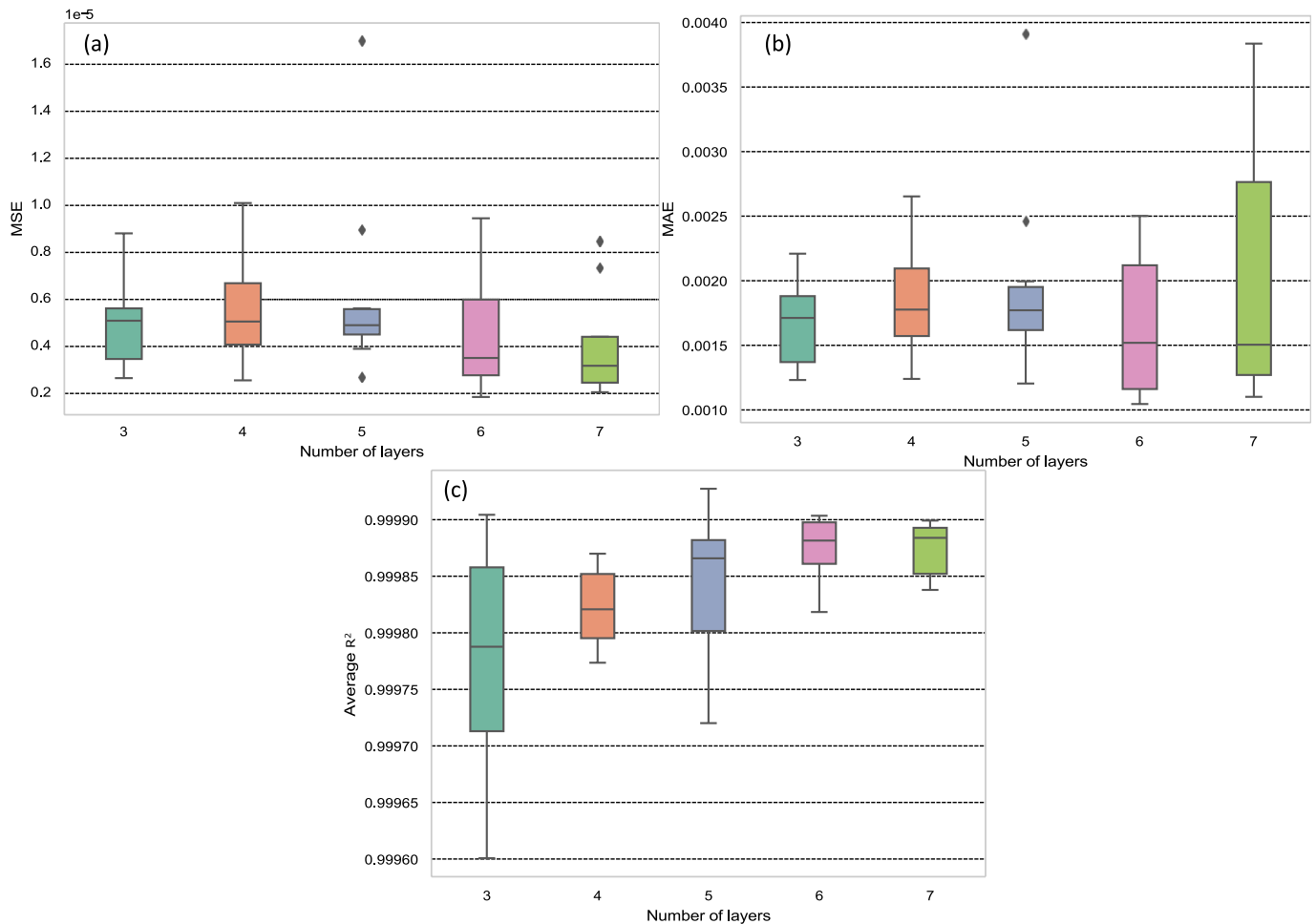

Fig. 15. Effect the number of layers on the performance model trained with Tanh activation, Adam optimizer, after 500 epochs, batch size = 32: (a) MSE, (b) MAE, (c) R^2.

ensure sufficient representation of the variability and characteristics of the system, allowing for robust predictions of stability. Based on this analysis and the achieved performance metrics, it can be concluded that the chosen sample size of 5000 samples is appropriate for obtaining reliable and accurate results in predicting the stability of rectangular tunnels. Therefore, it is recommended to consider this sample size when working with similar DFNN models to achieve optimal performance. The dataset has been split into two subsets: 80% for the training set and 20% for the testing set.

The dataset comprising 5000 samples was automatically generated using the IGA-LA model. While this approach may allow for a broad exploration of the parameter space, it can also introduce biases or gaps in the data that may not accurately represent real-world scenarios. It is therefore important to consider the implications of using the dataset that was generated through random parameter variation. A list of the statistical description of the input and output variables is shown in Table 5. Noted that the stability number N defined in Eq. (2) is the output of the proposed DFNN model.

The following steps are outlined for constructing DFNN.

Step 1. Investigate activation function.

The effectiveness of several activation functions, including ReLU, SeLU, Tanh, Sigmoid, and Softplus, was tested in deep learning models. The performance of each function was evaluated based on MSE, MAE, and R^2 values for both training and test data. Among these functions, Tanh consistently outperformed the others across all metrics. It achieved the lowest MSE and MAE scores and the highest R^2 values for both the

training and test data sets (see Fig. 13). Consequently, the Tanh activation function was chosen for the proposed DFNN model to ensure superior results. These findings highlight the crucial role of selecting appropriate activation functions in designing deep learning models.

Step 2. Investigate gradient descent optimization algorithms:

To achieve accurate predictions in deep learning, the selection of an appropriate gradient descent optimization algorithm is critical. Several widely used optimization algorithms, including Adam, RMSprop, Nadam, and Adamax, were compared to assess their effectiveness. The performance of these algorithms was evaluated based on MSE, MAE, and R^2 values for both the training and test data sets (see Fig. 14).

Fig. 14 also presents results of four performance models. It is concluded that the Adam optimizer consistently outperformed the other algorithms in terms of all metrics. Specifically, it achieved the lowest MSE and MAE scores and the highest R^2 values for both the training and test data sets. Consequently, the optimization algorithm selected for our proposed Deep Feedforward Neural Network (DFNN) model was Adam, chosen to ensure optimal performance. This finding emphasizes the critical role of selecting an appropriate gradient descent optimization algorithm in deep learning. By choosing the most effective algorithm, researchers and practitioners can ensure the highest levels of accuracy and predictive power in their models. The anticipated outcome of this study is to offer valuable insights to researchers and practitioners in the field, thus contributing to the advancement of deep learning applications.
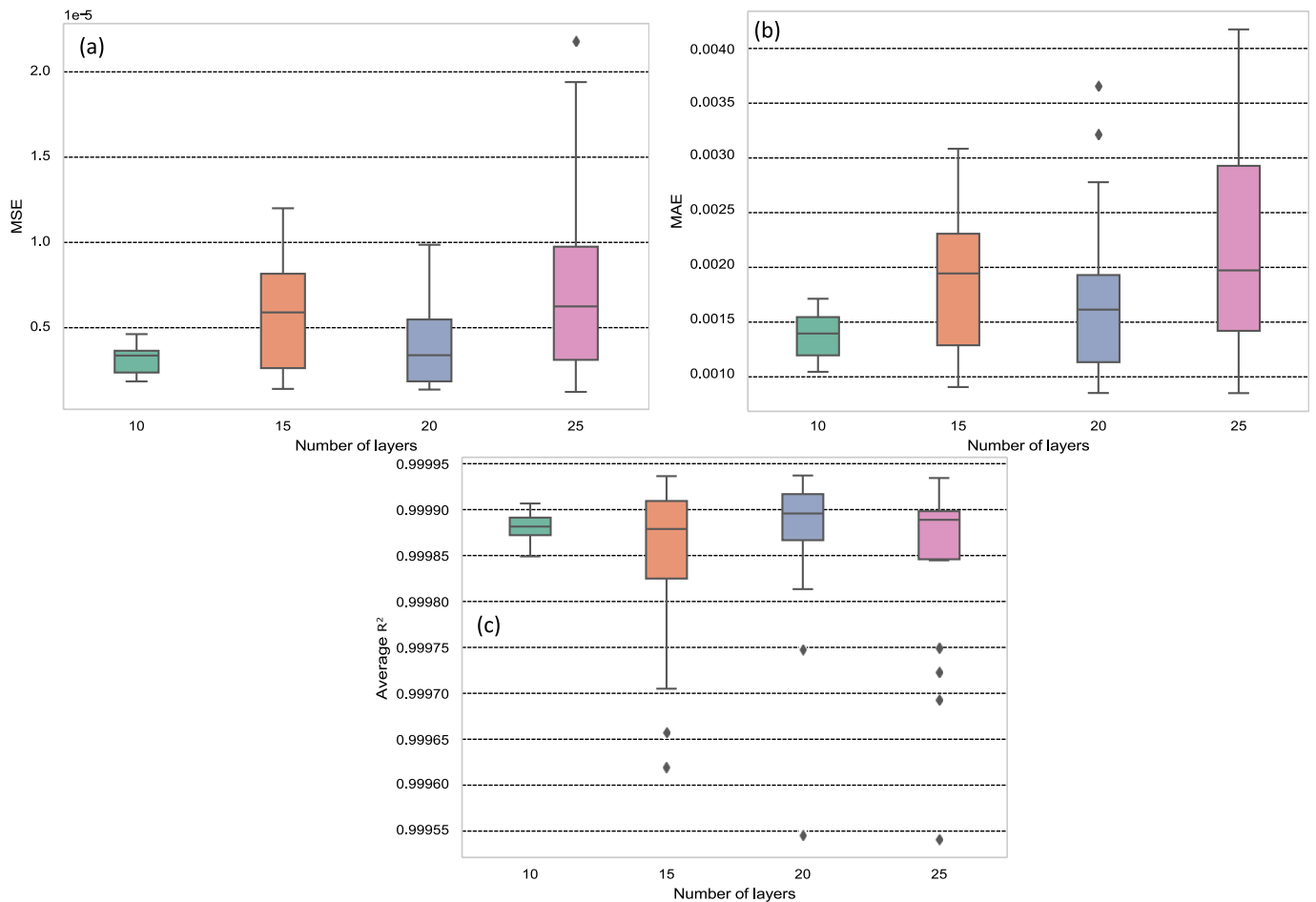

Fig. 16. Effect the number of hidden neurons on the performance model trained with Tanh activation, Adam optimizer, after 500 epochs, batch size = 32: (a) MSE, (b) MAE, (c) R^2.
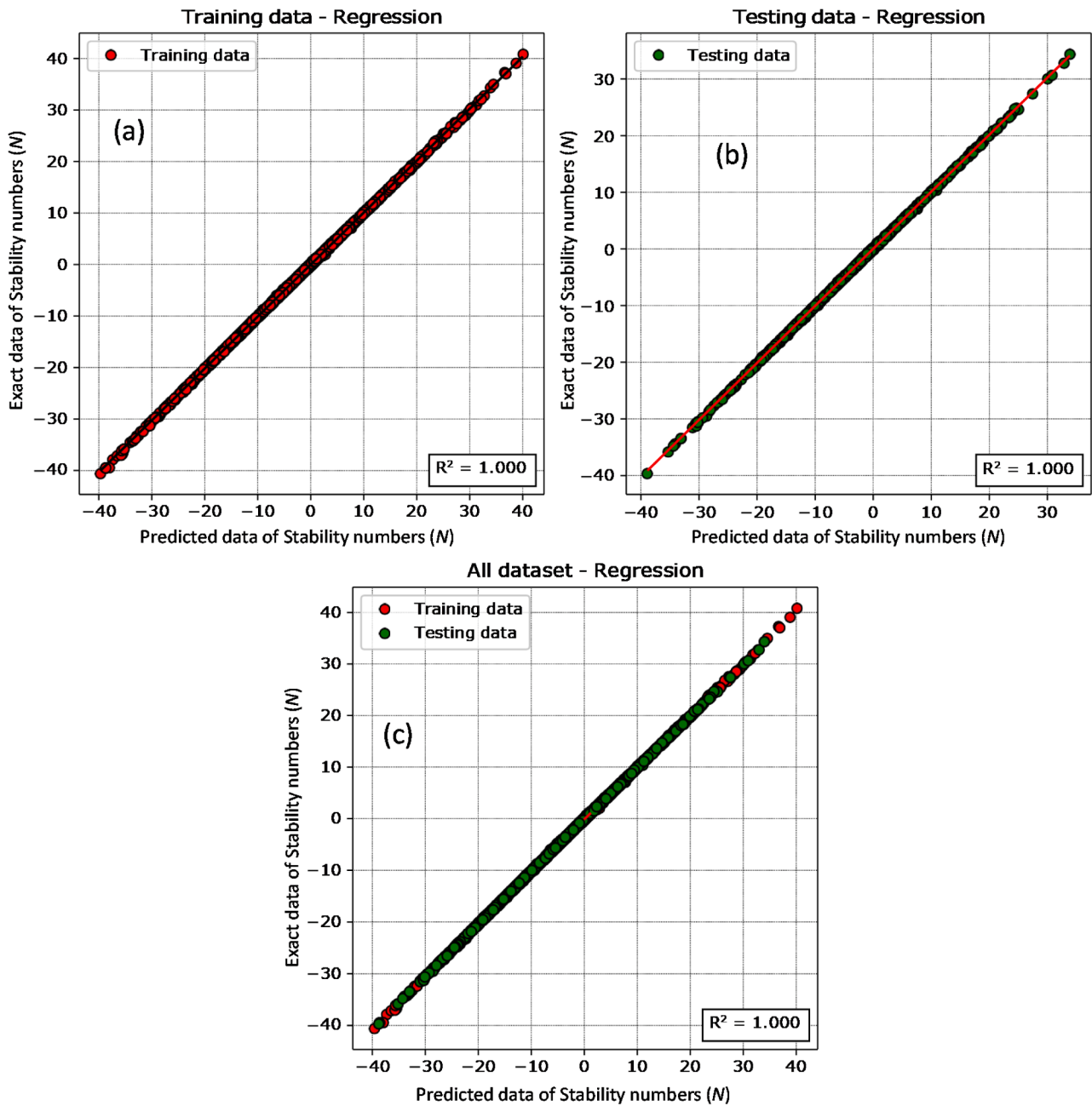
Fig. 17. Regression plot of the best DFNN model for: a) Training dataset; b) Testing data; c) All data.

Table 6
Compare the DFNN model with LSTM, CNN, SVM, and XGB.

Methods	MSE (x10^{-4})		MAE (x10^{-3})		R^2		Time (s)	Ranking
	Training	Testing	Training	Testing	Training	Testing		
DFNN	**0.026**	**0.026**	**1.215**	**1.231**	**0.9999**	**0.9999**	181.8	1
LSTM	0.189	0.200	3.545	3.686	0.9995	0.9995	305.1	3
CNN	2.612	3.614	10.799	11.300	0.9855	0.9800	131.8	4
SVM	10.27	9.95	26.76	26.60	0.9523	0.9568	12.0	5
XGB	0.15	0.12	2.31	6.82	0.9993	0.9929	0.440	2

Step 3. Determining the number of layers, hidden neurons.

The impact of varying the number of layers and hidden neurons on the performance of Deep Feedforward Neural Network (DFNN) models was also analyzed in this study. The results, as presented in Figs. 15 and 16, highlight the performance of different architectures in terms of MSE, MAE, and R^2 values. The architecture comprising 4 input parameters, 6 hidden layers with 10 units and 1 output (4–10–10–10–10–10–10–1) consistently outperformed the other architectures, achieving the lowest MSE and MAE scores and the highest R^2 values. As a result, this
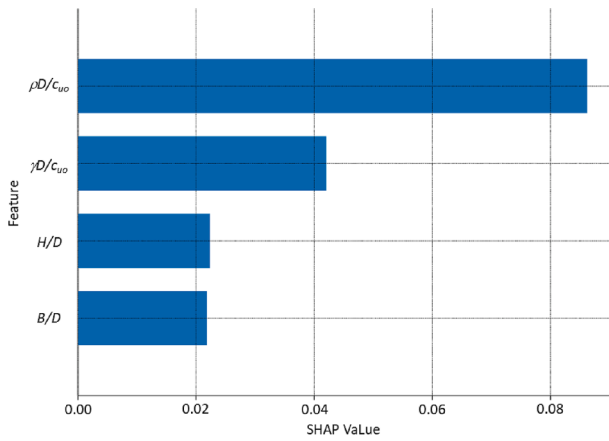
Fig. 18. SHAP values of feature.

architecture was identified as the optimal model for the proposed DFNN.

Detailed evaluation of the performance of the best-performing DFNN model is provided in Fig. 17. These findings suggest that careful selection of the number of layers and hidden neurons can help design DFNN models that achieve high levels of accuracy and predictive power, and they are valuable for researchers and practitioners working in the field of deep learning. It is also anticipated to significantly contribute to the development of more effective DFNN models in the future.

5.3. Summary

The optimal architecture for the proposed Deep Feedforward Neural Network (DFNN) model was identified based on our analysis of varying the number of layers and hidden neurons. This study's findings suggest that the selected architecture is highly effective, as indicated by the coefficient of determination (R^2) values. Specifically, the performance of the constructed model is outstanding, with an R^2 value of 1.0 for both the training and testing phases. This result indicates that the model explains all the variability in the data and has excellent predictive power.

This study's results underscore the importance of selecting an appropriate architecture for DFNN models to achieve optimal performance. These findings are expected to contribute to the development of more effective deep learning models and help advance the field's understanding of neural networks. Note that the performance of the optimal architecture for the proposed Deep Feedforward Neural Network (DFNN) model significantly outperformance other machine learning models such as Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), Support Vector Machine (SVM), and eXtreme Gradient Boosting (XGB) as shown in Table 6.

This table presents the performance comparison of five different models, including DFNN, LSTM, CNN, SVM, and XGB, based on their mean squared error (MSE), mean absolute error (MAE), coefficient of determination (R2), and execution time. The results show that the DFNN model achieved the best performance, with the lowest MSE and MAE
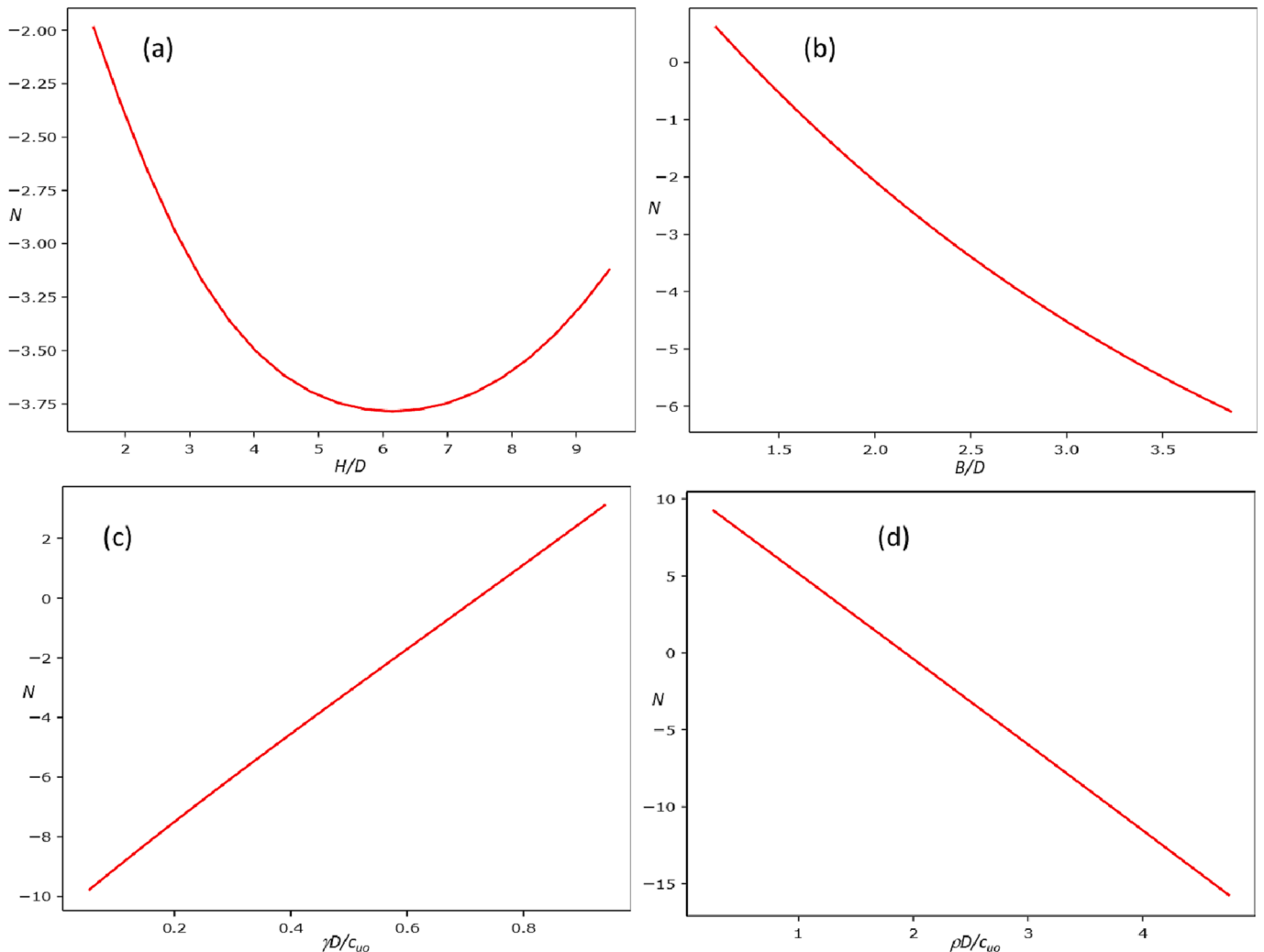

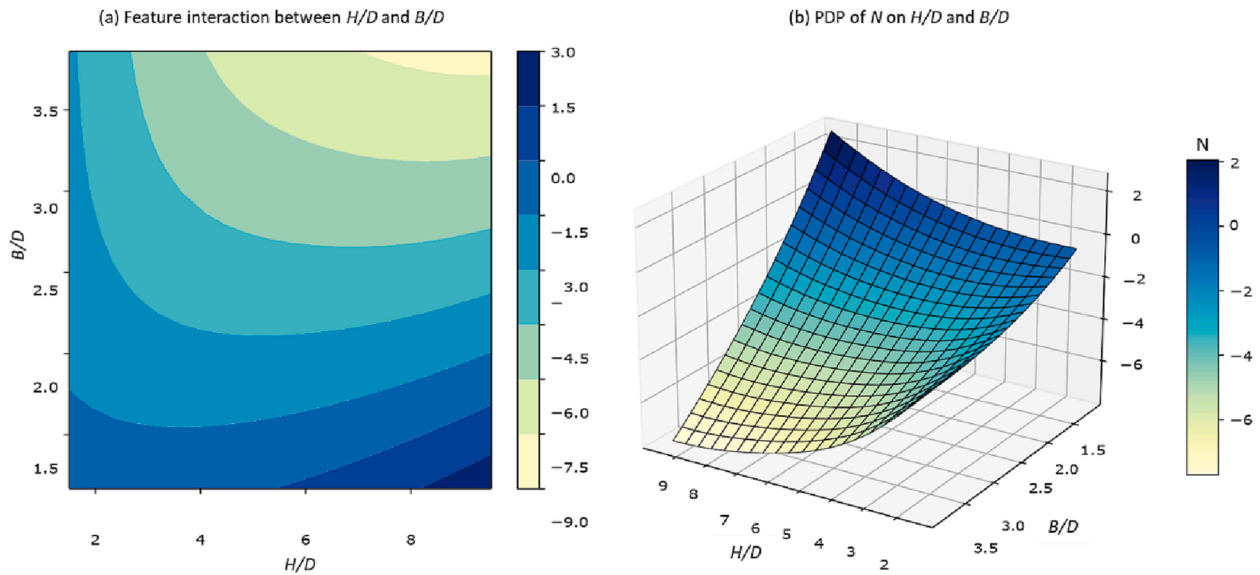**Fig. 19.** Partial Dependence of each input of features on the output.

Fig. 20. Partial dependence of N on (H/D and B/D): (a) 2D features interaction; (b) 3D features-output relations.
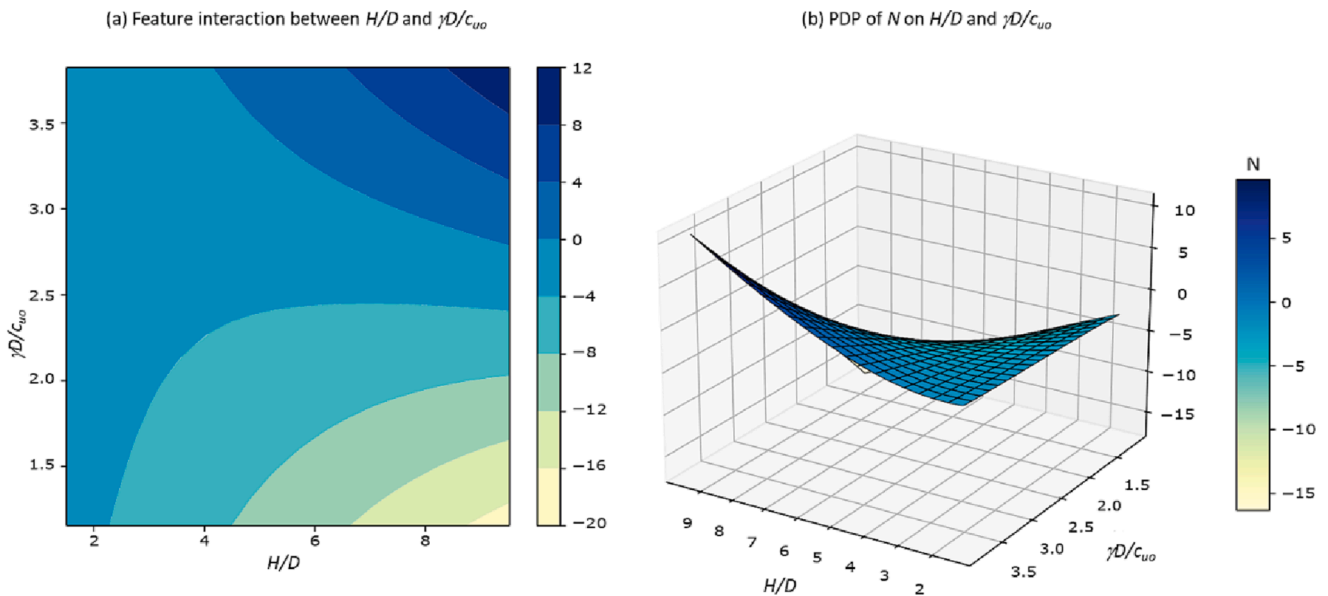

Fig. 21. Partial dependence of N on (H/D and $\gamma D/c_{uo}$): (a) 2D features interaction; (b) 3D features-output relations.

values and the highest R2 values for both the training and testing phases. The XGB model also performed well, with the second-highest ranking based on its R2 values, although it had relatively high MAE values. The LSTM, CNN, and SVM models had lower rankings due to their relatively high MSE and MAE values and lower R^2 values. Finally, the table also provides information on the execution time of each model, with the DFNN and LSTM models taking the longest time to train and test, and the XGB model being the fastest.

6. Feature Analysis and Partial Dependence Plots.

Explainable Machine Learning techniques like SHAP and Partial Dependence Plots (PDPs) help us understand the contribution of each feature to a model's prediction by facilitating both local and global analysis of the dataset. The shape value of a feature shows how it affects the prediction, and SHAP values can be used to compare feature importance and improve our understanding of the model.

The SHAP (Shapley Additive exPlanations) values presented in

Fig. 18 clearly demonstrate that the factor of $\rho D/c_{uo}$ has the most significant impact on the model's predictions. This means that variations in the value of $\rho D/c_{uo}$ have a relatively higher influence on the predicted outcomes of the model as compared to other input features. Additionally, the factor of $\gamma D/c_{uo}$ is found to have the second most substantial effect on the model's predictions. On the other hand, the input features H/D and B/D are shown to have similar and relatively low SHAP values, indicating their comparatively lesser contribution to the model's predictions. By understanding the relative importance of each input feature, insights into the model's behavior can be gained, and this knowledge can be leveraged to enhance the accuracy of predictions.

To provide a more detailed understanding of the impact of each parameter on the output, a Partial Dependence Plot (PDP) analysis was conducted and is presented in Fig. 19. The results show a significant drop in the magnitude of the N value when the $\rho D/c_{uo}$ parameter is increased. Conversely, the N value increases as the $\gamma D/c_{uo}$ parameter is increased. It follows that the parameters B/D causes a reduction in the value of N, with a decrease from 0 to minus 6. In addition, for H/D values
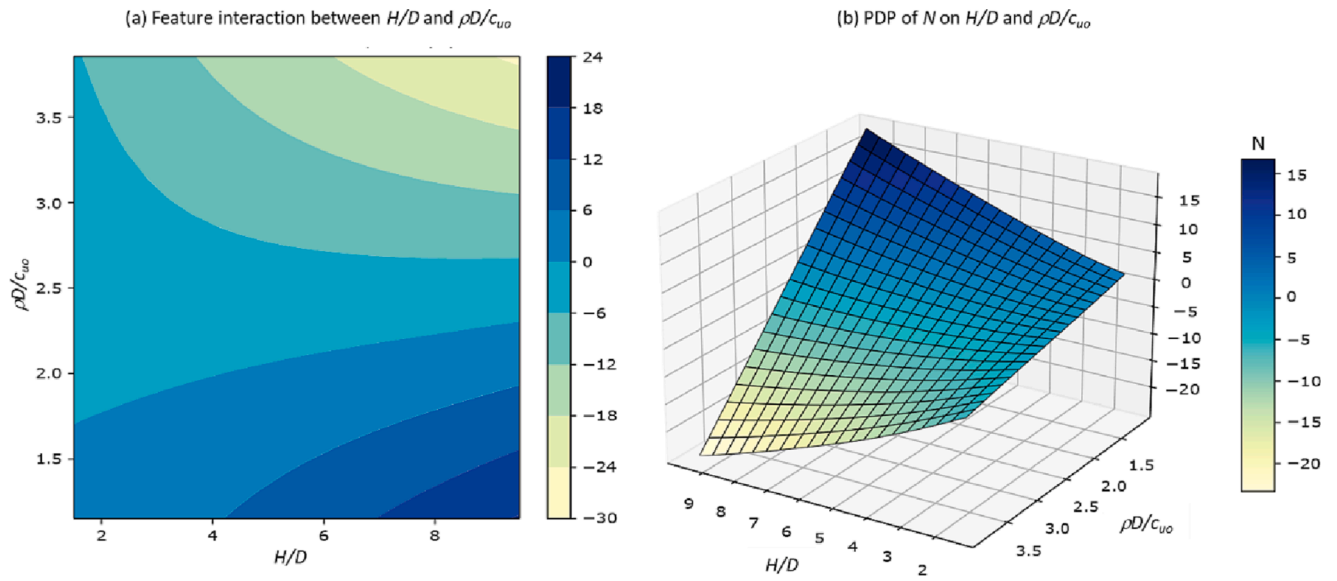
Fig. 22. Partial dependence of N on (H/D and $\rho D/c_{uo}$): (a) 2D features interaction; (b) 3D features-output relations.
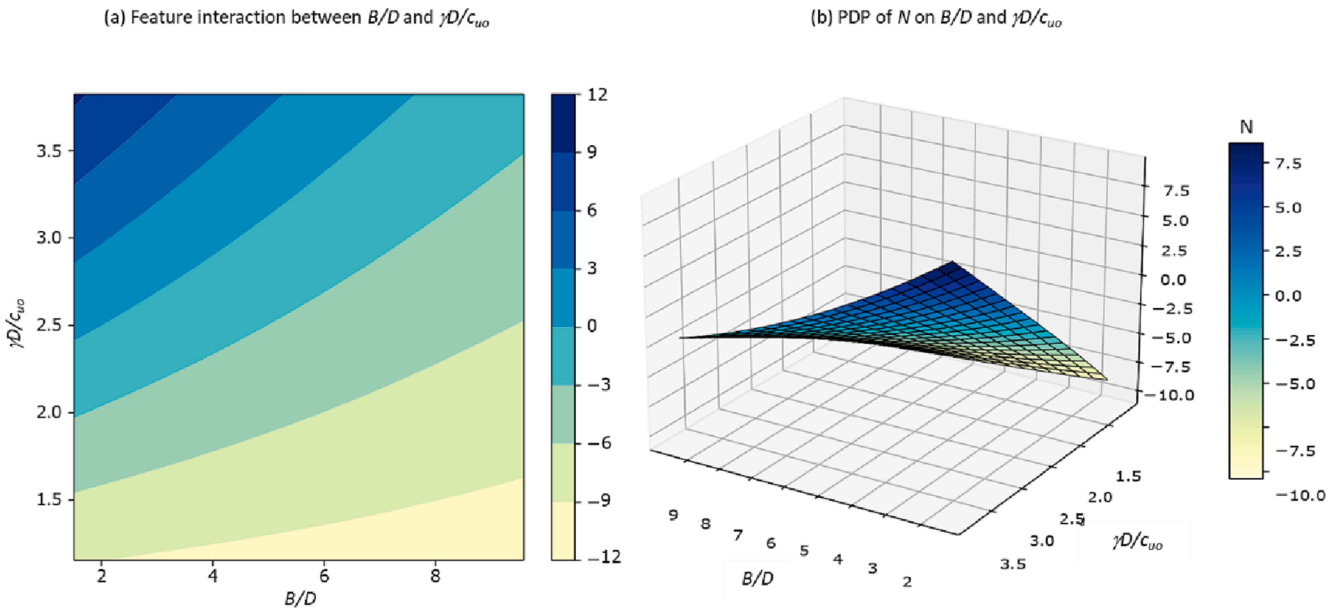

Fig. 23. Partial dependence of N on (B/D and $\gamma D/c_{uo}$): (a) 2D features interaction; (b) 3D features-output relations.

less than 6, it leads to a decrease in N from -2 to -3.75, whereas for H/D values greater than 6, it causes an increase in N. These findings provide a more comprehensive understanding of the influence of each parameter on the output, enabling more informed decisions when designing the system.

In addition to analyzing the individual effects of various parameters on N, this study also investigates the partial dependence of N on the simultaneous variation of two parameters. This approach allows for a more comprehensive understanding of the underlying relationships between the input variables and the response variable. The impact of various parameter pairs on N is specifically analyzed, including (H/D and B/D), (H/D and $\gamma D/c_u$), (H/D and $\rho D/c_{uo}$), (B/D and $\gamma D/c_{uo}$), (B/D and $\rho D/c_{uo}$), and ($\gamma D/c_{uo}$ and $\rho D/c_{uo}$). These pairs are explored in-depth, and the results are visualized in Figs. 20–25, providing valuable insights into the behavior of N in relation to multiple variables.

In Fig. 20, numerical results have shown that N value changes from positive to negative as both (H/D and B/D) increase. Note that the gradient of change is more gentle for H/D than for B/D (see Fig. 20a and b). As discussed before, a positive value of N may indicate a strong curvert (need external pressure to fail the system), while a weak one (need internal support) can be identified when a negative value of N is presented. Phyically, that means the stability of tunnel declines sharply as the value of B/D increases. Similar observation is made for other pairs. In particular, the linear relationships found for the pairs of (B/D and $\gamma D/c_{uo}$), (B/D and $\rho D/c_{uo}$), and ($\gamma D/c_{uo}$ and $\rho D/c_{uo}$) in Figs. 21–25. Through the analysis of the partial dependence of N on these parameter pairs, a deeper understanding of the intricate interactions between the input variables and the response variable can be obtained. This knowledge serves to enhance modeling and prediction accuracy in future studies.
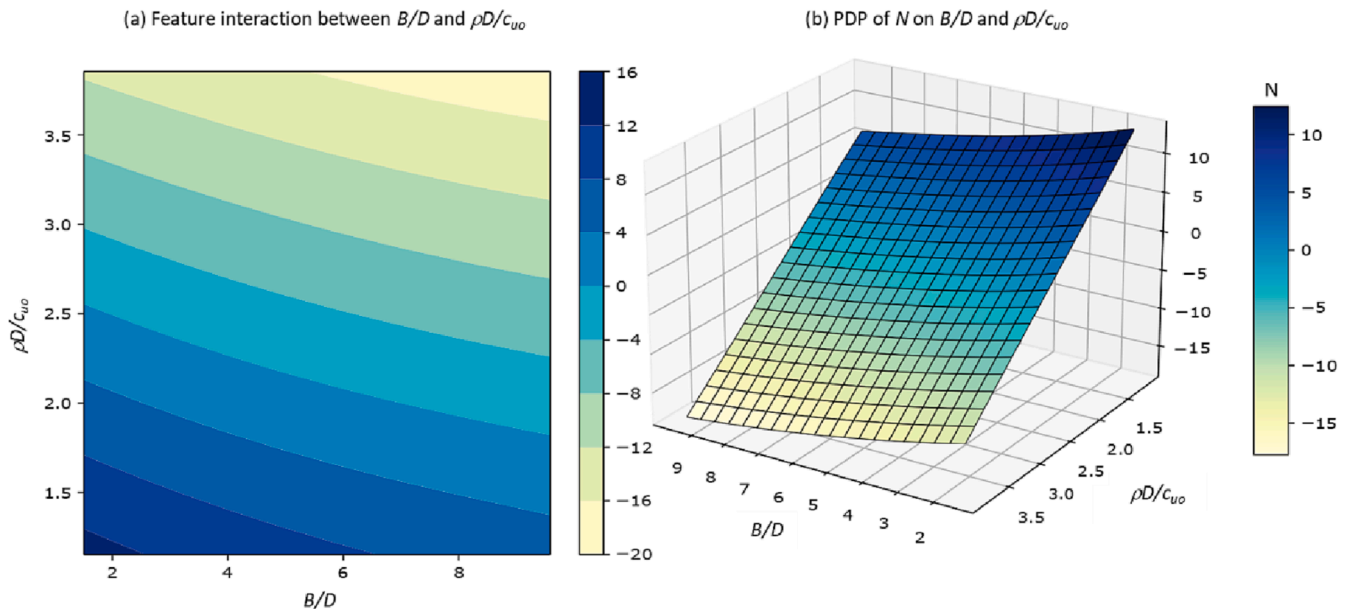
Fig. 24. Partial dependence of N on (B/D and $\rho D/c_{uo}$): (a) 2D features interaction; (b) 3D features-output relations.
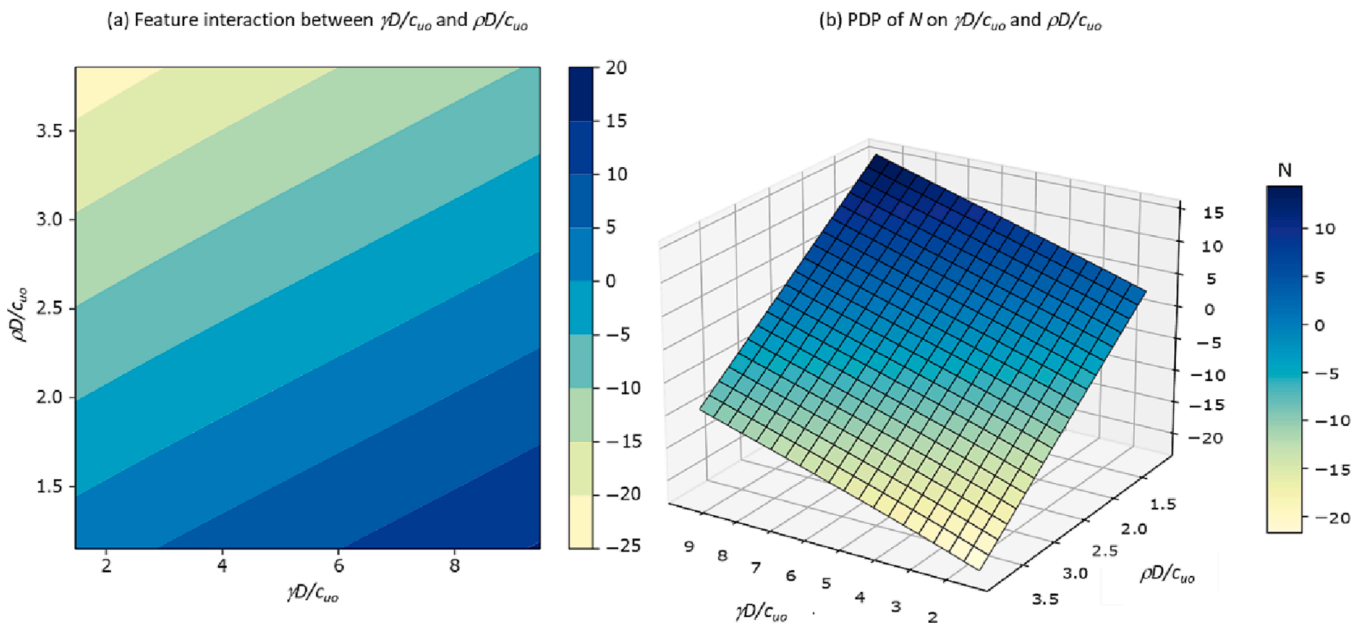

Fig. 25. Partial dependence of N on ($\gamma D/c_{uo}$ and $\rho D/c_{uo}$): (a) 2D features interaction; (b) 3D features-output relations.

7. Conclusion

A novel coupled approach based on upper bound isogeometric analysis and deep learning was proposed to investigate the undrained stability of rectangular tunnels. The coupled approach was validated by comparing numerical results with those of previous studies. Furthermore, a big dataset was randomly generated with 5000 runs for various input parameters, and a deep learning algorithm was employed to learn the dataset. The results showed that the proposed approach could accurately predict the undrained stability of rectangular tunnels with a small mean squared error of 10^{-6}. The proposed approach has several advantages over traditional methods. First, the use of isogeometric analysis provides a higher level of accuracy in modeling the geometry of the tunnels. Second, the use of upper-bound limit analysis ensures that the calculated results are safe and conservative. Third, the use of deep

learning allows for the development of accurate and efficient prediction models for complex problems. Our findings indicate that the soft computing model using deep learning is highly effective in predicting the stability of tunnels with a high level of accuracy.

Feature analysis and Partial Dependence Plots (PDPs) were conducted in this study to gain valuable insights into the physical behavior of the system under investigation. Through feature analysis, the study identified and prioritized the important variables and factors influencing the observed outcomes, providing a deeper understanding of the relationships and interactions among different variables in the system. Additionally, the utilization of Partial Dependence Plots (PDPs) allowed for visualizing and analyzing the individual and collective effects of selected features on the target variable. These plots facilitated a comprehensive examination of how changes in specific features impact the overall behavior of the system, revealing significant trends,
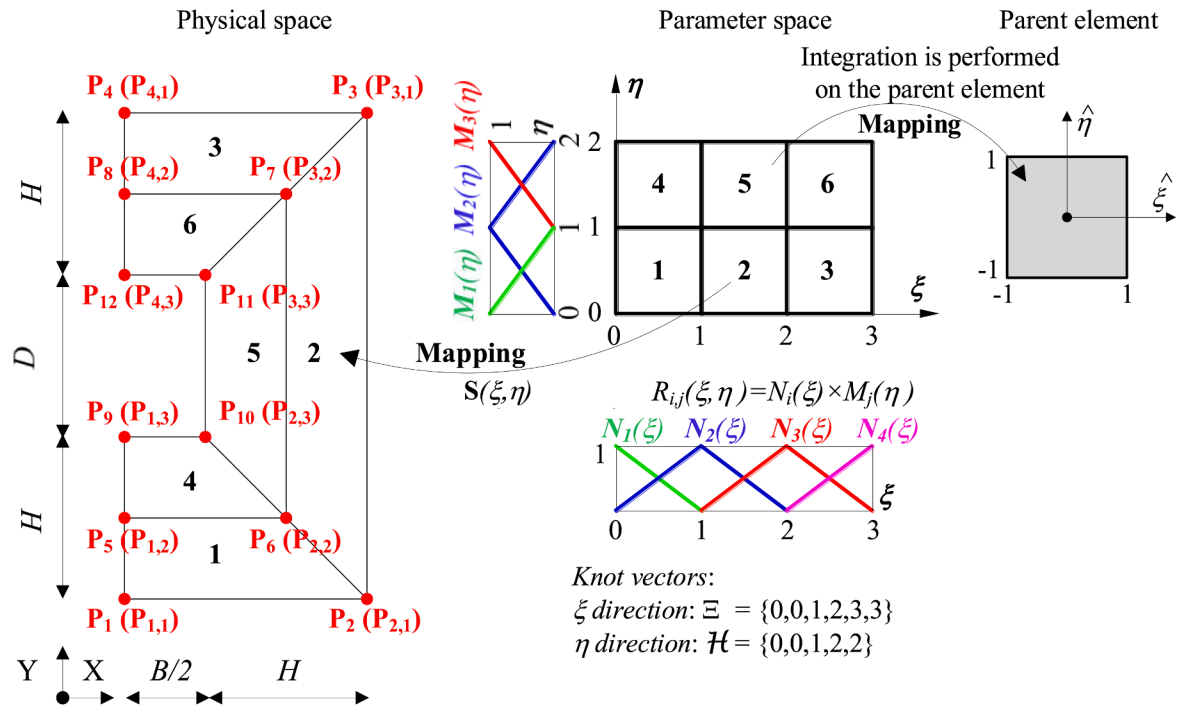
Fig. A1. B-spline surface $S(\xi,\eta)$ in case 6 elements, the polynomial order $p = q = 1$.

dependencies, and non-linear relationships between variables. These findings offer valuable support for optimizing the construction process, enabling cost minimization while achieving the desired predesigned stability factor (N). Stakeholders can leverage these insights to make informed decisions, enhancing construction efficiency and ensuring the attainment of the desired stability levels within budgetary constraints.

In conclusion, the present study can be a useful tool for engineers and researchers to evaluate the undrained stability of rectangular tunnels in various geotechnical conditions during the construction process. Overall, it highlights the potential of machine learning in combination with traditional analytical tools to provide a more comprehensive and accurate understanding of complex engineering problems. The combination of isogeometric analysis and deep learning can also be extended to other engineering problems, offering a promising avenue for future research.

Limitations and research outlooks.

In addition to the aforementioned merits, it is important to acknowledge the limitations of this study, which are outlined below:

- This study primarily concentrates on the undrained shear strength of the ground, overlooking the incorporation of tunnel structural properties, thereby impeding a comprehensive understanding of the interplay between the ground and the tunnel structure. Furthermore, a common assumption of rigid-perfectly plastic material for undrained clay, as seen in other limit analysis studies, is considered inadequate.
- Unlike the availability of field data during a tunnel's lifelong service, field observation regarding tunnel stability during the construction process is scarce. As a result, the present study solely relies on computational simulation data, without incorporating or validating the findings with field data. Furthermore, this study is limited to the deep learning modeling of undrained stability of rectangular tunnels, without fully exploring the potential of AI modeling for more demanding applications with high computational costs, such as reliability analysis. To facilitate further advancements in this area, a supplementary file has been provided, containing a substantial dataset of 5000 samples.

- The simplicity of the rectangular tunnel geometry prevented the full utilization of one of the primary advantages of Isogeometric Analysis (IGA), which is its ability to precisely model complex geometries. The potential of IGA in accurately representing tunnels with curved shapes was not fully explored in this study.

It is important to acknowledge these limitations as they provide opportunities for future research to address and overcome these challenges, thus contributing to a more comprehensive and advanced understanding of tunnel engineering.

Declaration of Competing Interest

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

Data availability

Data will be made available on request.

Acknowledgement

greatly valuable comments and feedback that have significantly   improved the quality and clarity of the paper.

Appendix A. B-Spline components

Knot vectors

In the context of computational geometry, a knot vector is defined as an ordered sequence of parameter values in a specific direction. The sequence, denoted as $\Xi = \{\xi_1, \xi_2,..., \xi_i,..., \xi_{n+p+1}\}$ where ξ_i is the ith knot, i is the knot index, n is the number of basis functions, and p is the polynomial order. The knot vector divides the parametric space into intervals usually referred to as knot spans.

Basis functions

The Cox-de Boor recursion formula is used to define the basis functions once a knot vector has been selected:

For p = 0 :

$$N_{i,0}(\xi) = \begin{cases} 1 & \xi_i \leqslant \xi < \xi_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

For p ⩾ 1:

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi)$$

(A1.1)

B-spline curves

Constructing a piecewise-polynomial B-spline curve involves using a set of n control points $\mathbf{P}_i \in \mathrm{R}^2$ and n B-spline basis functions $N_{i,p}(\xi)$. The curve $\mathbf{C}(\xi)$ is then constructed as a linear combination of the control points, where the coefficients of the linear combination are the basis functions:

$$\mathrm{C}(\xi) = \begin{bmatrix} x \\ y \end{bmatrix}(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi)\mathrm{P}_i = \sum_{i=1}^{n} N_{i,p}(\xi)\begin{bmatrix} x_i \\ y_i \end{bmatrix}$$

(A1.2)

B-spline surfaces

Given two knot vectors (one for each parametric direction) $\Xi = \{\xi_1, \xi_2, ..., \xi_{n+p+1}\}$ and $H = \{\eta_1, \eta_2, ..., \eta_{m+q+1}\}$ and a control net $\mathbf{P}_{i,j} \in \mathrm{R}^2$, a tensor-product B-spline surface $\mathbf{S}(\xi,\eta)$ is defined as:

$$\mathrm{S}(\xi,\eta) = \begin{bmatrix} x \\ y \end{bmatrix}(\xi,\eta) = \sum_{i=1}^{n} \sum_{j=1}^{m} N_{i,p}(\xi) \times M_{j,q}(\eta) \times \mathrm{P}_{i,j} = ...$$

$$... = \sum_{i=1}^{n} \sum_{j=1}^{m} R_{i,j}(\xi,\eta) \times \mathrm{P}_{i,j} = \sum_{i=1}^{n} \sum_{j=1}^{m} R_{i,j}(\xi,\eta)\begin{bmatrix} x_{i,j} \\ y_{i,j} \end{bmatrix}$$

(A1.3)

where $R_I(\xi,\eta) = R_{i,j}(\xi,\eta) = N_{i,p}(\xi) \times M_{j,q}(\eta)$ are the bivariate B-spline basis functions. Fig A1. illustrates B-spline surface $\mathbf{S}(\xi,\eta)$ as a rectangular tunnel with 6 elements, the polynomial order is $p = q = 1$.

The local support of a basis function reads:

$$N_{i,p}(\xi) \times M_{j,q}(\eta) = [\xi_i, \xi_{i+p+1}] \times [\eta_j, \eta_{j+q+1}]$$

(A1.4)

meaning that the support of the bivariate function $N_{i,p}(\xi) \times M_{j,q}(\eta)$ extends over the area restricted by the knot values $[\xi_i, \xi_{i+p+1}] \times [\eta_j, \eta_{j+q+1}]$, where i, j is the knot index and p,q is the polynomial order in ξ and η direction, respectively.

Appendix A2

According to Makrodimopoulos and Martin (2007), the structure will collapse if and only if there exists a kinematically admissible displacement field $\dot{\mathbf{u}} = [u,v]^{\mathrm{T}}$ such that:

$$W_{int}(\dot{\boldsymbol{\varepsilon}}) < \lambda^+ W_{ext}(\dot{\boldsymbol{u}}) + W_{ext}^0(\dot{\boldsymbol{u}})$$

(A2.1)

where $W_{int}(\dot{\boldsymbol{\varepsilon}})$ is the internal plastic dissipation; λ^+ is the collapse load multiplier; $W_{ext}^0(\boldsymbol{u})$ is the work of any additional loads f_0, g_0 not subjected to the multiplier; $W_{ext}(\boldsymbol{u})$ is the external work rate of a rigid perfectly plastic body of area $\Omega \in \mathrm{R}^2$ with boundary Γ, which is subjected to body forces $\boldsymbol{f}$ and to surface tractions $\boldsymbol{g}$ such that:

$$W_{ext}(\boldsymbol{u}) = \int_{\Omega} \boldsymbol{f}^T \boldsymbol{u} \ \mathrm{d}\Omega + \int_{\Gamma_t} \boldsymbol{g}^T \boldsymbol{u} \ \mathrm{d}\Gamma \tag{A2.2}$$

The internal plastic dissipation of the two-dimensional domain Ω can be written as:

$$W_{int}(\dot{\boldsymbol{\varepsilon}}) = \int_{\Omega} D(\dot{\boldsymbol{\varepsilon}}) \ \mathrm{d}\Omega \tag{A2.3}$$

where the plastic dissipation $D(\dot{\boldsymbol{\varepsilon}})$ is defined by:

$$D(\dot{\boldsymbol{\varepsilon}}) = \max_{\psi(\boldsymbol{\sigma}) \leqslant 0} \boldsymbol{\sigma} \cdot \dot{\boldsymbol{\varepsilon}} \equiv \boldsymbol{\sigma_{\varepsilon}} \cdot \dot{\boldsymbol{\varepsilon}} \tag{A2.4}$$

With $\boldsymbol{\sigma}$ represents the admissible stresses contained within the convex yield surface $\psi(\sigma)$ and $\boldsymbol{\sigma_{\varepsilon}}$ represents the stresses on the yield surface associated to any strain rates $\dot{\varepsilon}$ through the plasticity condition.

If defining $C = \{\dot{\boldsymbol{u}} \in U \mid W_{ext}(\dot{\boldsymbol{u}}) = 1\}$, the collapse load multiplier λ^+ can be determined by the following mathematical programming:

$$\lambda^+ = \min_{\dot{\boldsymbol{u}} \in C} \int_{\Omega} D(\dot{\boldsymbol{\varepsilon}}) \ \mathrm{d}\Omega - W_{ext}^0(\dot{\boldsymbol{u}}) \tag{A2.5}$$

For this study, the behavior of soil is assumed to adhere to the characteristics of rigid-perfectly plastic materials, with increments of plastically admissible strain followed by the normality rule. The Mohr-Coulomb failure criterion is expressed in terms of the cohesion and the internal friction angle of soil such that:

$$\psi(\sigma) = \sqrt{(\sigma_{xx} - \sigma_{yy})^2 + 4\tau_{xy}^2} + (\sigma_{xx} + \sigma_{yy})\sin\varphi - 2c\cos\varphi \tag{A2.6}$$

Hence, the power of dissipation can be formulated as a function of strain rates for each element e as Makrodimopoulos and Martin (2007):

$$D(\dot{\boldsymbol{\varepsilon}}) = cA_e t_e \cos\varphi \tag{A2.7}$$

$$\begin{cases} \|\boldsymbol{\rho}\|_e \leqslant t_e \\ \boldsymbol{\rho} = \begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix} = \begin{bmatrix} \dot{\tilde{\varepsilon}}_{xx}^e - \dot{\tilde{\varepsilon}}_{yy}^e \\ \dot{\tilde{\gamma}}_{xy}^j \end{bmatrix} \\ \dot{\tilde{\varepsilon}}_{xx}^e + \dot{\tilde{\varepsilon}}_{yy}^e = t_e \sin\varphi \end{cases} \tag{A2.8}$$

Appendix A3. Mathematical description of DFNN and model's evaluation metric

Mathematical Description of DFNN

Nonlinear transformations are applied to the input at each layer, resulting in a unique representation at the output. Let us assume that there are V (same as stability number) layers in the neural network, and the output signal of the lth layer is expressed as follows:

$$z_j^l = f\left(w_j^T a_j^{l-1} + b_j\right), \quad l = 1, 2, 3, 4, ..., V, \tag{A3.1}$$

The output signal of the $(l)^{\text{th}}$ layer in the DFNN is determined by a combination of factors. Firstly, the activation function f is applied to the input. Additionally, the weight vector w_j^T describes the influence of all units in the same hidden layer, while a_j^{l-1} represents the output signal of the previous $(l\text{-}1)^{\text{th}}$ layer. Finally, b_j represents the bias parameter of the jth unit in the current $(l)^{\text{th}}$ layer. Together, these factors contribute to the overall output signal of the DFNN at each layer, and ultimately to the final prediction made by the model.

Model's evaluation metric.

The MSE can be formulated as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (p_i - y_i)^2 \tag{A3.2}$$

The MAE can be formulated as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} (|p_i - y_i|) \tag{A3.3}$$

Coefficient of determination (R^2).

$$R^2 = 1 - \frac{\sum_{i-1}^{n} (p_i - y_i)^2}{\sum_{i-1}^{n} (y_i - \overline{y_i})^2} \tag{A3.4}$$

where n represents the sample size in the training set; y_i denotes the actual output; p_i corresponds to the predicted value of the model; and $\overline{y_i}$ is the average of the actual values.

Appendix B. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.tust.2023.105330.

References

Abbo, A.J., Wilson, D.W., Sloan, S.W., Lyamin, A.V., 2013. Undrained stability of wide rectangular tunnels. Comput. Geotech. 53, 46–59. https://doi.org/10.1016/j.compgeo.2013.04.005.

Assadi, A., Sloan, S.W., 1991. Undrained stability of shallow square tunnel. J. Geotech. Eng. 117 (8), 1152–1173. https://doi.org/10.1061/(asce)0733-9410(1991)117:8 (1152).

Broms, B.B., Bennermark, H., 1967. Stability of clay at vertical opening. J. Soil Mech. Found. Div. 93 (1), 71–94.

Duchi, J., Hazan, E., Singer, Y., 2011. Adaptive subgradient methods for online learning and stochastic optimization. J. Mach. Learn. Res. 12 (7).

Gioda, G., Swoboda, G., 1999. Developments and applications of the numerical analysis of tunnels in continuous media. Int. J. Numer. Anal. Meth. Geomech. 23 (13), 1393–1405. https://doi.org/10.1002/(sici)1096-9853(199911)23:13<1393::aid-nag933>3.0.co;2-z.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

Guo, P., Gong, X., Wang, Y., Lin, H., Zhao, Y., 2021. Minimum cover depth estimation for underwater shield tunnels. Tunn. Undergr. Space Technol. 115, 104027 https://doi.org/10.1016/j.tust.2021.104027.

Gutierrez, D., 2017. RMSprop Optimization Algorithm for Gradient Descent with Neural Networks. <https://insidebigdata.com/2017/09/24/rmsprop-optimization-algorithm-gradient-descent-neural-networks/>.

Hinton, G.E., 2012. A practical guide to training restricted Boltzmann machines. Neural Networks: Tricks of the Trade: Second Edition, pp. 599–619.

Hughes, T.J.R., Cottrell, J.A., Bazilevs, Y., 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comput. Methods Appl. Mech. Eng. 194 (39–41), 4135–4195. https://doi.org/10.1016/j.cma.2004.10.008.

Irzal, F., Remmers, J.J.C., Verhoosel, C.V., de Borst, R., 2013. Isogeometric finite element analysis of poroelasticity. Int. J. Numer. Anal. Meth. Geomech. 37 (12), 1891–1907. https://doi.org/10.1002/nag.2195.

Kim, K., Yoo, C.H., 2005. Design loading on deeply buried box culverts. J. Geotech. Geoenviron. Eng. 131 (1), 20–27. https://doi.org/10.1061/(asce)1090-0241(2005) 131:1(20).

Kingma, D.P., Ba, J., 2014. Adam: A Method for Stochastic Optimization. arXiv preprint arXiv:1412.6980.

Makrodimopoulos, A., Martin, C.M., 2007. Upper bound limit analysis using simplex strain elements and second-order cone programming. Int. J. Numer. Anal. Meth. Geomech. 31 (6), 835–865. https://doi.org/10.1002/nag.567.

Meguid, M.A., Saada, O., Nunes, M.A., Mattar, J., 2008. Physical modeling of tunnels in soft ground: a review. Tunn. Undergr. Space Technol. 23 (2), 185–198. https://doi.org/10.1016/j.tust.2007.02.003.

Meng, F.-Y., Chen, R.-P., Xu, Y., Wu, K., Wu, H.-N., Liu, Y., 2022. Contributions to responses of existing tunnel subjected to nearby excavation: a review. Tunn. Undergr. Space Technol. 119, 104195 https://doi.org/10.1016/j.tust.2021.104195.

Mishra, R., 2023. 3D fracture study of cracked functionally graded biological materials by XIGA approach using Bézier extraction of NURBS. Theor. Appl. Fract. Mech. 124, 103818 https://doi.org/10.1016/j.tafmec.2023.103818.

Najafabadi, M.M., Villanustre, F., Khoshgoftaar, T.M., Seliya, N., Wald, R., Muharemagic, E., 2015. Deep learning applications and challenges in big data analytics. J. Big Data 2 (1). https://doi.org/10.1186/s40537-014-0007-7.

Nguyen, M.-T., 2014. Limit Analysis Of Geotechnical Problems Using Isogeometric Analysis And Second Order Cone Programming. Ho Chi Minh City University of Technology (HCMUT).

Nguyen, T.-H., Nguyen, T., Truong, T.T., Doan, D.T.V., Tran, D.-H., 2023. Corrosion effect on bond behavior between rebar and concrete using Bayesian regularized feed-forward neural network. Structures 51, 1525–1538. https://doi.org/10.1016/j.istruc.2023.03.128.

Nguyen, H.C., Nguyen-Son, L., 2022. A stable CS-FEM for the static and seismic stability of a single square tunnel in the soil where the shear strength increases linearly with depth. J. Rock Mech. Geotech. Eng. 14 (4), 1253–1265. https://doi.org/10.1016/j.jrmge.2022.01.006.

Nguyen, T., Tran, L.V., 2021. Arching effect in sand piles under base deflection using geometrically non-linear isogeometric analysis. Geomech. Eng. 26 (4), 369–384. https://doi.org/10.12989/gae.2021.26.4.369.

Nguyen, T., Truong, T.T., Nguyen-Thoi, T., Van Hong Bui, L., Nguyen, T.-H., 2022. Evaluation of residual flexural strength of corroded reinforced concrete beams using convolutional long short-term memory neural networks. Structures 46, 899–912. https://doi.org/10.1016/j.istruc.2022.10.103.

Ortiz-Puerta, D., Cox, A., Hurtado, D.E., 2022. Snakes Isogeometric Analysis (SIGA): Towards accurate and flexible geometrical models of the respiratory airways. Comput. Methods Appl. Mech. Eng. 394, 114841 https://doi.org/10.1016/j.cma.2022.114841.

Robbins, H., Monro, S., 1951. A stochastic approximation method. Ann. Math. Stat. 400–407.

Ruder, S., 2016. An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747.

Shahbazi, A., Chesnaux, R., Saeidi, A., 2021. A new combined analytical-numerical method for evaluating the inflow rate into a tunnel excavated in a fractured rock mass. Eng. Geol. 283, 106003 https://doi.org/10.1016/j.enggeo.2021.106003.

Shahrokhabadi, S., Cao, T.D., Vahedifard, F., 2019. Thermo-hydro-mechanical modeling of unsaturated soils using isogeometric analysis: model development and application to strain localization simulation. Int. J. Numer. Anal. Meth. Geomech. 44 (2), 261–292. https://doi.org/10.1002/nag.3015.

Shiau, J., Al-Asadi, F., 2020. Two-dimensional tunnel heading stability factors F, F and F. Tunn. Undergr. Space Technol. 97, 103293 https://doi.org/10.1016/j.tust.2020.103293.

Shiau, J., Keawsawasvong, S., 2022. Producing undrained stability factors for various tunnel shapes. Int. J. Geomech. 22 (8) https://doi.org/10.1061/(asce)gm.1943-5622.0002487.

Shiau, J., Chudal, B., Keawsawasvong, S., 2022. Three-dimensional sinkhole stability of spherical cavity. Acta Geotech. 17 (9), 3947–3958. https://doi.org/10.1007/s11440-022-01522-8.

Sloan, S.W., 1988. Lower bound limit analysis using finite elements and linear programming. Int. J. Numer. Anal. Meth. Geomech. 12 (1), 61–77. https://doi.org/10.1002/nag.1610120105.

Sloan, S.W., 1989. Upper bound limit analysis using finite elements and linear programming. Int. J. Numer. Anal. Meth. Geomech. 13 (3), 263–282. https://doi.org/10.1002/nag.1610130304.

Sloan, S.W., Assadi, A., 1991. Undrained stability of a square tunnel in a soil whose strength increases linearly with depth. Comput. Geotech. 12 (4), 321–346. https://doi.org/10.1016/0266-352x(91)90028-e.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (1), 1929–1958.

Truong, T.T., Dinh-Cong, D., Lee, J., Nguyen-Thoi, T., 2020. An effective deep feedforward neural networks (DFNN) method for damage identification of truss structures using noisy incomplete modal data. J. Build. Eng. 30, 101244 https://doi.org/10.1016/j.jobe.2020.101244.

Tschuchnigg, F., Schweiger, H.F., Sloan, S.W., Lyamin, A.V., Raissakis, I., 2015. Comparison of finite-element limit analysis and strength reduction techniques. Géotechnique 65 (4), 249–257. https://doi.org/10.1680/geot.14.P.022.

Tsinidis, G., Rovithis, E., Pitilakis, K., Chazelas, J.L., 2016. Seismic response of box-type tunnels in soft soil: experimental and numerical investigation. Tunn. Undergr. Space Technol. 59, 199–214. https://doi.org/10.1016/j.tust.2016.07.008.

Vo-Minh, T., Nguyen, H.C., 2022. Seismic stability of a circular tunnel in cohesive-frictional soils using a stable node-based smoothed finite element method. Tunn. Undergr. Space Technol. 130, 104606 https://doi.org/10.1016/j.tust.2022.104606.

Vo-Minh, T., Nguyen-Son, L., Nguyen-Van, G., Thai-Phuong, T., 2021. Upper bound limit analysis of circular tunnel in cohesive-frictional soils using isogeometric analysis based on Bézier extraction. Tunn. Undergr. Space Technol. 114, 103995 https://doi.org/10.1016/j.tust.2021.103995.

Wang, X., Li, J., Zhao, X., Liang, Y., 2022b. Propagation characteristics and prediction of blast-induced vibration on closely spaced rock tunnels. Tunn. Undergr. Space Technol. 123, 104416 https://doi.org/10.1016/j.tust.2022.104416.

Wang, S., Ren, J., Fang, X., Lin, H., Xu, G., Bao, H., Huang, J., 2022a. IGA-suitable planar parameterization with patch structure simplification of closed-form polysquare. Comput. Meth. Appl. Mech. Eng. 392, 114678 https://doi.org/10.1016/j.cma.2022.114678.

Wang, Y.X., Shan, S.B., Zhang, C., Guo, P.P., 2019. Seismic response of tunnel lining structure in a thick expansive soil stratum. Tunn. Undergr. Space Technol. 88, 250–259. https://doi.org/10.1016/j.tust.2019.03.016.

Wani, M.A., Bhat, F.A., Afzal, S., Khan, A.I., 2020. Advances in Deep Learning. Springer.

Wilson, D.W., Abbo, A.J., Sloan, S.W., Lyamin, A.V., 2013. Undrained stability of a square tunnel where the shear strength increases linearly with depth. Comput. Geotech. 49, 314–325. https://doi.org/10.1016/j.compgeo.2012.09.005.

Wilson, D.W., Abbo, A.J., Sloan, S.W., Lyamin, A.V., 2015. Undrained stability of dual square tunnels. Acta Geotech. 10 (5), 665–682. https://doi.org/10.1007/s11440-014-0340-1.

Wilson, D.W., Abbo, A.J., Sloan, S.W., Yamamoto, K., 2017. Undrained stability of rectangular tunnels where shear strength increases linearly with depth. Can. Geotech. J. 54 (4), 469–480. https://doi.org/10.1139/cgj-2016-0072.

Wu, K., Shao, Z., Sharifzadeh, M., Chu, Z., Qin, S., 2022. Analytical approach to estimating the influence of shotcrete hardening property on tunnel response. J. Eng. Mech. 148 (1) https://doi.org/10.1061/(asce)em.1943-7889.0002052.

Zeiler, M.D. 2012. Adadelta: an adaptive learning rate method. arXiv preprint arXiv: 1212.5701.