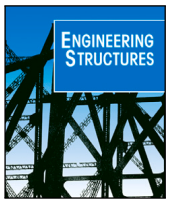
Engineering Structures 346 (2026) 121623

Contents lists available at ScienceDirect

Engineering Structures

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



Prediction of pipeline strain demand under ground movement using a neural network-enhanced finite difference method

Beilei Ji [a,*], Qipei Mei [a], Nader Yoosef-Ghodsi [b], Samer Adeeb [a]

[a] Department of Civil & Environmental Engineering, University of Alberta, Edmonton, Alberta, Canada
[b] Enbridge Pipelines Inc., Edmonton, Alberta, Canada

ARTICLE INFO

Keywords:
Pipeline strain demand
Permanent ground displacement
Finite difference method
Neural network surrogate
Nonlinear hardening model

ABSTRACT

Pipelines are vulnerable to permanent ground movement induced by geohazards, which can lead to excessive strain and potential structural failure. Accurate prediction of strain demand is essential for ensuring pipeline safety and supporting geohazard risk assessment. A recently developed finite difference method (FDM)-based framework for pipeline strain analysis has demonstrated strong potential as a fast and practical tool. However, this framework relies on a bilinear stress–strain assumption that allows closed-form expressions for internal forces. For more general nonlinear hardening models, such closed-form solutions are not available, and internal forces must instead be evaluated through numerical integration, resulting in a substantial increase in computational cost. This study proposes a neural network-enhanced finite difference method (NN-FDM) to address this limitation. Neural network surrogates are trained to replace numerical integration in the evaluation of internal forces, thereby enabling the incorporation of general nonlinear hardening behavior within the FDM framework. Additional robustness enhancement strategies are implemented to ensure stable iterative convergence. Two case studies with both bilinear and nonlinear hardening conditions are conducted to examine the effectiveness of NN-FDM. Results demonstrate that NN-FDM maintains comparable accuracy to original FDM for bilinear cases while showing significant advantages for nonlinear hardening scenarios, achieving high accuracy and substantially improved computational performance. The method's applicability is further validated against four existing analytical solutions for buried pipelines subjected to strike-slip faulting. The proposed method provides a simple and general framework for evaluating pipeline strain demand across a wide range of constitutive models and shows strong potential as an efficient tool for rapid assessment and pre-screening applications in geohazard-prone areas.

1. Introduction

Linear infrastructure systems such as pipelines, railways, roads and power lines exhibit similar vulnerability characteristics due to their long path and exposure to diverse geological conditions. Among these linear assets, oil and gas pipelines are particularly critical as they transport essential resources over long distances. These pipelines face significant risks from permanent ground deformation (PGD) induced by geohazards including landslides, earthquakes, fault movements, soil liquefaction, and permafrost thaw. Such ground movements can generate excessive strains in pipeline, potentially resulting in structural failures with significant consequences. Historical incidents demonstrate the magnitude of these threats. The 1994 Northridge earthquake in California caused significant damage to pipeline systems among other infrastructure, with total repair costs exceeding $21 million [1]. In 2018, the Peace River Regional District landslides in British Columbia damaged the Enbridge natural gas pipeline, causing significant supply disruptions throughout western Canada [2]. Pipeline and Hazardous Materials Safety Administration data indicated that ground movement-related pipeline incidents in North America between 2010 and 2020 resulted in property damage exceeding $500 million [3]. These consequences underscore the necessity for effective methodologies to predict pipeline strain under geohazard conditions.

In response to these critical safety concerns, numerous analytical models for pipeline behavior prediction have been developed over the past few decades. Earliest effort was made by Newmark and Hall [4], who developed a small-deflection analytical framework that estimated pipeline elongation by relating soil slip friction to the static earth pressure, without accounting for passive soil resistance. Kennedy et al. [5] later extended this model by incorporating large-deflection theory and

* Corresponding author.
E-mail address: beilei@ualberta.ca (B. Ji).

https://doi.org/10.1016/j.engstruct.2025.121623
Received 17 July 2025; Received in revised form 13 September 2025; Accepted 16 October 2025
Available online 22 October 2025

passive soil pressure, idealizing the pipeline as a flexible cable dominated by axial tension. Building upon these early developments, Wang and Yeh [6,7] introduced a hybrid model in which the large-deflection segment of the pipeline was represented by a constant-curvature arc, while the remaining segments were treated as semi-infinite beams on an elastic foundation. Chiou et al. [8] further refined this approach by modeling the large-deflection portion as an elastica, addressing limitations inherent in the constant-curvature assumption. More recent studies, including those by Takada et al. [9], Karamitros et al. [10],Trifonov and Cherniy [11,12] and Sarvanis and Karamanos [13], have expanded analytical frameworks to incorporate cross-sectional deformation, varying boundary conditions, nonlinear soil-pipeline interaction, and operating loads. Although analytical models offer valuable insights with relatively low computational effort, they often rely on simplifying assumptions regarding geometry, material properties, and boundary conditions. These limitations restrict their applicability in complex conditions.

Given these constraints, numerical modeling approaches have increasingly been adopted for simulating pipeline deformation under PGD. Among these, the finite element method (FEM) has become a widely used and powerful tool, commonly implemented through commercial software such as ABAQUS and ANSYS. In FEM simulations, two key aspects are essential for accurately capturing pipeline response: the representation of the pipeline and the modeling of pipe–soil interaction. For pipeline representation, beam, shell [14], or hybrid elements [15,16] are commonly used depending on the required level of detail and computational cost. Pipe–soil interaction is typically modeled using either contact algorithms [17–19] or nonlinear soil spring models recommended by the American Lifelines Alliance (ALA) [20–22]. Numerous studies have demonstrated that FEM provides a robust and accurate simulations of pipeline response under complex loading and ground deformation scenarios. However, its application in routine engineering practice remains limited due to several inherent challenges. The development of FEM models demands significant effort, including detailed mesh generation, the specification of boundary and loading conditions, and the calibration of constitutive models for both pipe and soil materials. These tasks are time intensive and require specialized expertise, which may not be readily available in many pipeline operating organizations. To address these challenges and reduce the reliance on specialized tools and expertise, alternative numerical methods have been explored for practical implementation in pipeline engineering. The finite difference method (FDM) represents one such alternative. Owing to its algorithmic simplicity and computational efficiency, FDM is particularly attractive for rapid assessments and integration into decision-support tools [23]. Recently, Zheng et al. [24,25] proposed an FDM-based framework in which the pipeline is idealized as an Euler–Bernoulli beam. The governing differential equations are discretized and formulated as a nonlinear system, which is then solved using built-in solvers in Python. This approach yields strain demand predictions that are in good agreement with FEM benchmarks, while significantly reducing computational time and implementation complexity. Compared to traditional analytical methods discussed earlier, the FDM framework offers improved prediction accuracy and greater modeling flexibility, making it a practical tool for pipeline assessments under geohazard conditions. Despite its advantages, the applicability of this method is currently restricted to cases where a bilinear stress–strain relationship is assumed. Under this assumption, the axial force N and bending moment M, which are required in the governing equations, can be explicitly calculated. However, when nonlinear hardening models are employed, such explicit solutions are no longer available. In these cases, the internal forces must be evaluated through numerical integration, which significantly increases the computational cost.

Machine learning techniques, particularly neural networks (NNs), have been widely demonstrated as effective surrogate models for addressing computationally intensive tasks, offering a flexible, data-driven framework for capturing complex relationships in scientific and engineering problems [26–31]. The use of NNs as surrogate modeling tools can be traced back to the early 1990s [32–35]. Since then, interest in this area has grown substantially, driven by the increasing availability of powerful computing resources such as graphics processing units (GPUs) [36], as well as the development of modern machine learning libraries including TensorFlow [37] and PyTorch [38], which have made neural network implementation more accessible to researchers and engineers. Among the many applications of NN-based surrogate models, their use in constitutive modeling has received particular attention. One major application of NN-based constitutive models lies in material behavior modeling. In this context, NNs excel at capturing implicit and complex relationships that challenges traditional constitutive models. For instance, Al-Haik et al. [39] proposed an NN to predict relaxation in polymeric matrix composites using strain, temperature, and process time as inputs. Their model demonstrated superior performance compared to explicit conventional viscoelastic models, particularly near the glass transition temperature. Similarly, Linka et al. [40] developed a data-driven surrogate model integrating stress–strain responses with theoretical and microstructural information to enable efficient and generalizable material modeling. Yamanaka et al. [41] created a surrogate model of computational homogenization for elastoplastic composite materials that serves as a homogenized constitutive law for decoupled two-scale analysis. Dettmer et al. [42] presented a NN-based framework for modeling rate-independent inelastic materials, employing recurrent architectures to capture path-dependent behavior without relying on traditional phenomenological equations. By effectively modeling complex material behavior, NN-based constitutive models offer a promising pathway to accelerate numerical simulations by replacing computationally intensive constitutive calculations. A growing body of research has successfully embedded such surrogates within finite element frameworks. Stöcker et al. [43] developed a recurrent neural network (RNN)-based constitutive model that efficiently predicts anisotropic damage softening behavior, replacing costly multiscale simulations while ensuring stable FE analysis through implicit gradient regularization. Pang et al. [44] proposed a deep neural network (DNN)-enhanced finite element scheme for metallic glasses, where the DNN accurately predicts free-volume–based constitutive behavior, effectively bypassing explicit model evaluations and significantly accelerating FEM simulations. Parallel advancements [45–48] further demonstrate the capacity of NNs to enhance simulation efficiency in FEM-based studies. Despite these promising developments, the integration of NNs into finite difference methods for constitutive modeling remains relatively underexplored.

To improve the flexibility of the FDM in handling nonlinear hardening models while preserving its computational advantages, this paper proposes an enhanced approach that integrates NNs into the finite difference scheme for evaluating pipeline strain demand under PGD. The NN serves as a surrogate model for capturing complex nonlinear relationships between internal forces and the pipe displacement derivatives, replacing the need for computationally intensive numerical integration typically required by general nonlinear hardening models. The proposed methodology maintains the computational efficiency and accessibility that make the FDM attractive for practical engineering applications while extending its applicability to more complex constitutive behaviors.

2. Methodology

2.1. Problem statement

Pipelines subjected to PGD can be conceptually illustrated in Fig. 1. In this configuration, the pipeline is buried through a geohazard zone, where the middle segment lies within the moving ground, while the two end segments remain in stable ground. The lengths of the left, middle, and right sections are denoted as L_1, L_2, and L_3, respectively. The angle β denotes the intersection angle between the pipeline axis and the

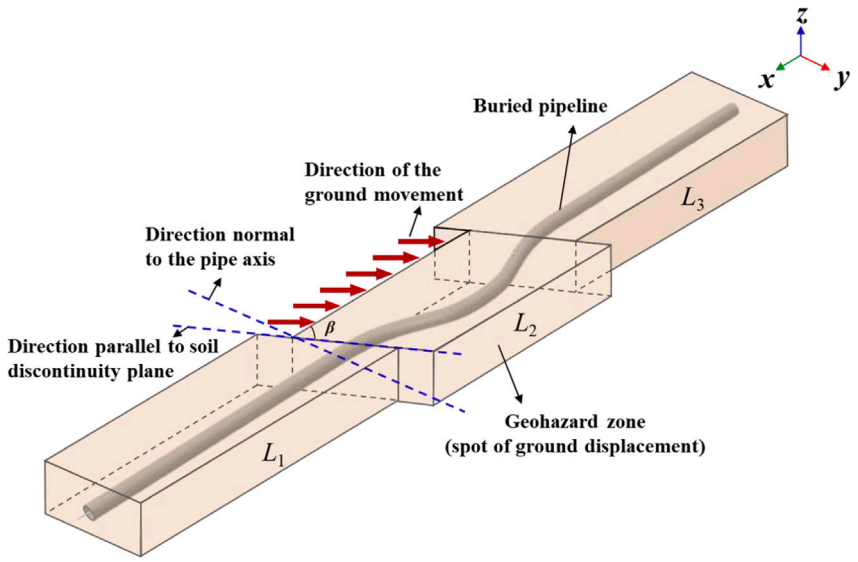


Fig. 1. Schematic representation of the pipe subjected to ground movement.

direction of ground movement. For the sake of simplicity, the pipeline is idealized as an Euler–Bernoulli beam subjected to distributed loads, with axial load density f and lateral load density q. The governing equations are given in Eq. (1).

$$\begin{cases} \dfrac{dN}{dx} + f\,(\delta\cos\beta - u) = 0 \\ \dfrac{d^2M}{dx^2} - \dfrac{d}{dx}\left(N\cdot\dfrac{dv}{dx}\right) - q\,(\delta\sin\beta - v) = 0 \end{cases} \tag{1}$$

where u and v represent the axial and lateral displacements of the pipe, while δ corresponds to the magnitude of PGD. The axial and lateral distributed loads f and q result from the interaction between the pipe and the surrounding soil, modeled using soil springs following the ALA Guidelines [20].

2.2. A neural network enhanced finite difference method

2.2.1. Overview of finite difference framework

In numerical analysis, FDM are widely used for solving differential equations by approximating their derivatives with finite differences. In previous studies, Zheng et al. [24,25] developed an FDM-based framework to solve the governing differential equations presented in Eq. (1). To support the development of the proposed method, a brief overview of the finite difference framework for predicting the strain demand in pipelines subjected to PGD is provided below.

a. Derivation of axial force N and bending moment M

Based on the Euler–Bernoulli beam theory and the assumption that plane sections remain plane and always perpendicular to the axis of the beam during deformation, the longitudinal strain ε_l in the pipe can be expressed as:

$$\varepsilon_l = \varepsilon_{\text{axial}} + \varepsilon_{\text{bending}} = \left(u' + \frac{1}{2}v'^2\right) - zv'' \tag{2}$$

Here u', v', and v'' are the first and second derivatives of the pipe displacement with respect to the axial coordinate. The term $u' + \frac{1}{2}v'^2$ represents the axial strain, while $-zv''$ accounts for the bending strain at a distance z from the neutral axis.

The axial force N and bending moment M, required in the governing equations, are calculated by integrating the longitudinal stress σ_l over the pipe cross-section as:

$$N = \int_A \sigma_l\,\mathrm{d}A, \quad M = -\int_A \sigma_l z\,\mathrm{d}A \tag{3}$$

For an elastic pipe, the corresponding longitudinal stress σ_l is determined using Hooke's law, assuming a linear elastic relationship between stress and strain. Substituting $\sigma_l = E\varepsilon_l$ into Eq. (3) yields:

$$N = EA\left(u' + \frac{1}{2}v'^2\right), \quad M = EIv'' \tag{4}$$

Here E is Young's modulus, A is the cross-sectional area, and I is the second moment of area of the pipe section.

For inelastic pipes, the stress no longer follows a linear relationship with strain. Instead, the axial force and bending moment must be computed by numerically integrating the nonlinear stress distribution across the pipe cross-section. To reduce computational cost while capturing essential inelastic behavior, a bilinear stress–strain relationship is assumed, as defined in Eq. (5).

$$\sigma_l = \begin{cases} E\varepsilon_l, & \text{if } |\varepsilon_l| \le \varepsilon_Y \\ E\varepsilon_Y + E_p\left(\varepsilon_l - \text{sign}(\varepsilon_l)\varepsilon_Y\right), & \text{if } |\varepsilon_l| > \varepsilon_Y \end{cases} \tag{5}$$

Here E_p is the plastic modulus, ε_Y is the yield strain.

Based on this simplified constitutive model, four distinct cases are considered to represent different strain profiles, as illustrated in Fig. 2. Cases 1 and 2 represent longitudinal strain distributions without bending, while Cases 3 and 4 include both axial and bending strain components. In each case, the axial force N and bending moment M are expressed as functions of the unknown axial and lateral displacements u and v, allowing their incorporation into the finite difference formulation of the governing equations. A detailed description of these cases, along with the corresponding formulations of N and M can be found in the referenced publication [24].

b. Definition of the pipe–soil interaction

In the finite difference framework, the external loads f and q in the governing equations, arising from pipe–soil interaction, are modeled using discrete soil springs connected to each node along the pipe length. As shown in Fig. 3, the actual relationship between soil reaction force and pipe displacement in the axial, lateral, and vertical directions follows a hyperbolic curve, shown as a dashed line. For computational efficiency, these curves are commonly approximated by bilinear spring models, which capture the peak resistance(T_u, P_u, Q_u, Q_d) and associated yield displacement(Δ_t, Δ_p, Δq_u, and Δq_d). These parameters can be estimated using empirical relationships provided by ALA [20].

Since this study considers a one-dimensional problem, the external force terms f and q in the governing equations are represented by the soil spring responses in either the axial and lateral directions or the axial and vertical directions, depending on the plane in which ground movement occurs. As shown in Fig. 1, when the displacement takes

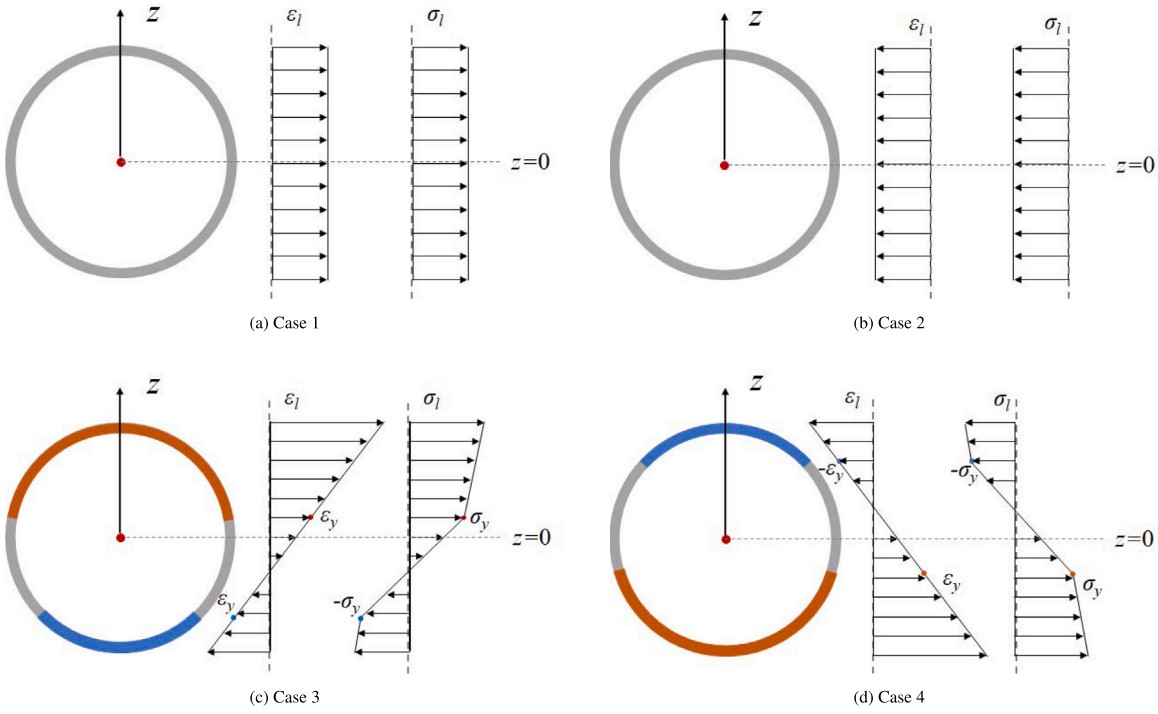


Fig. 2. Strain distribution profiles for the four representative cases.

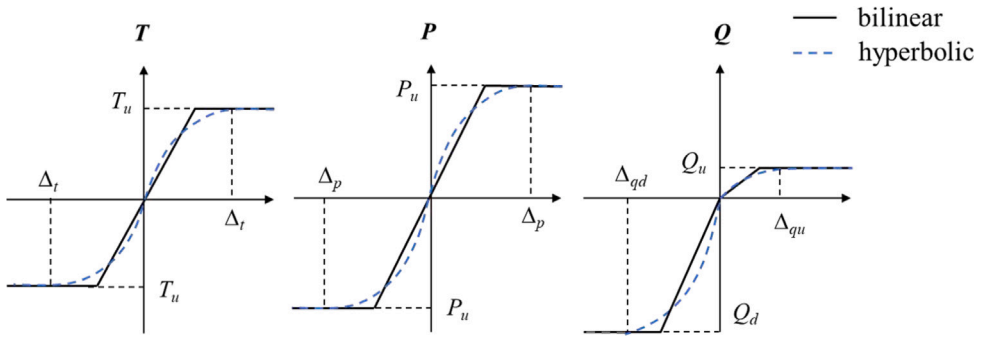


Fig. 3. Soil springs.

place in the horizontal plane (x–y plane), axial and lateral soil springs are activated, in which the soil resistance is assumed to be symmetric. In contrast, if the movement occurs in the vertical plane (x–z plane), axial and vertical springs are applied, where the vertical soil resistance is typically asymmetric due to reduced uplift capacity. In the case of horizontal ground movement, the axial and lateral load densities are formulated as functions of the relative displacements between the ground and pipeline, denoted as $f(r)$ and $q(s)$, respectively, as shown in Eqs. (6) and (7).

$$f(r)=\begin{cases}T_u, & r>\Delta_t\\ \frac{T_u}{\Delta_t}\cdot r, & |r|\le\Delta_t\\ -T_u, & r<-\Delta_t,\end{cases}\quad \text{where } r=\delta\cos\beta-u \tag{6}$$

$$q(s)=\begin{cases}P_u, & s>\Delta_p\\ \frac{P_u}{\Delta_p}\cdot s, & |s|\le\Delta_p\\ -P_u, & s<-\Delta_p,\end{cases}\quad \text{where } s=\delta\sin\beta-v \tag{7}$$

c. Development of finite-difference equations

By incorporating the explicit expressions for the axial force, bending moment, and soil–pipe interaction into the governing equations, the system can be reformulated in terms of the unknown pipe deformations u and v, along with their derivatives u', u'', v', v'', and v'''. These derivatives are approximated at each node using the finite difference method. As a result, the governing equations are transformed into a system of nonlinear equations, where the primary unknowns are the nodal displacements u and v.

d. Solution of pipe displacement and strain demand

A built-in nonlinear equation solver in Python is employed to solve the resulting system of nonlinear equations, incorporating specified boundary conditions such as fixed-end constraints and continuity of displacements and their derivatives. Consequently, the pipe displacements at each predefined grid point can be obtained. Based on these displacements, the strain distribution along the pipe can be evaluated, enabling the determination of the corresponding strain demand.

2.2.2. Neural network constitutive model

In the original FDM framework described in Section 2.2.1, the axial force N and bending moment M are computed from the pipe displacement derivatives using explicit expressions derived under the bilinear stress–strain assumption. This formulation enables fast and straightforward implementation; however, it is only valid for a limited class of idealized material behaviors. When more general nonlinear hardening models are considered, internal forces N and M must be obtained through numerical integration using Eq. (3). This procedure is

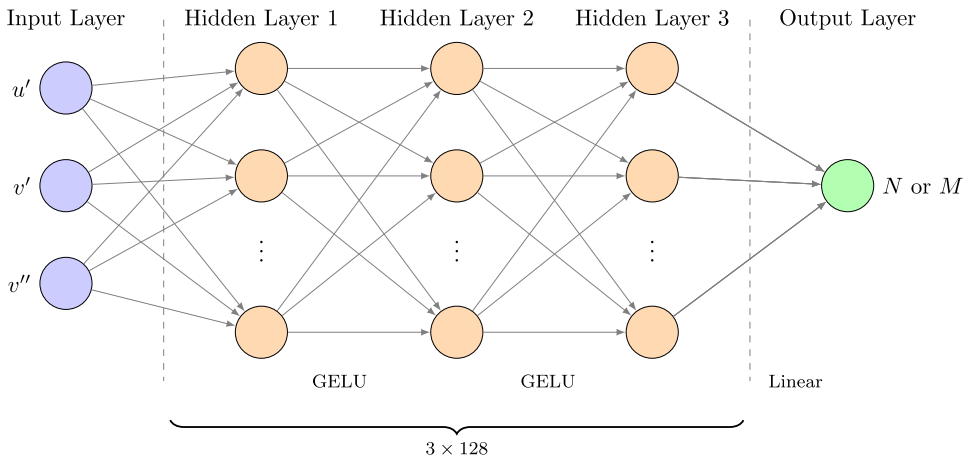


Fig. 4. NN architecture.

computationally expensive, particularly when embedded in an iterative solver that evaluates internal forces at every node and every iteration.

To address this challenge, this section proposes the use of NN as a surrogate model to approximate the internal forces computation. Specifically, two independent NNs are trained to approximate the nonlinear constitutive relationship between displacement derivatives (u', v', v'') and the internal forces (N, M). Once trained, the surrogate model is embedded in the FDM framework and used to predict N and M during the optimization process, replacing the need for costly numerical integration. The integration of the NN constitutive model forms the foundation of the proposed neural network-enhanced finite difference method (NN-FDM). The following subsections outline the data preparation, model development, training, and validation procedures for the NN surrogate models.

a. Database Preparation

A dataset consisting of input features (u', v', v'') and corresponding outputs(N, M) is generated through numerical integration based on Eq. (3), assuming a bilinear stress–strain relationship representative of X52 pipe steel. To ensure comprehensive coverage of the input domain, the ranges of the input variables were determined through extensive numerical experiments using the finite difference model under a variety of loading scenarios. These ranges were slightly expanded to improve model robustness, resulting in: $u' \in [-0.05, 0.05]$, $v' \in [-0.5, 0.5]$, and $v'' \in [-0.001, 0.001]$. Several sampling strategies can be adopted to construct the training dataset, including random sampling [49], Latin hypercube sampling [50], or full factorial sampling [51]. In this study, to achieve uniform and systematic coverage of the input space, each input variable was independently sampled and a full factorial approach was then adopted to generate all possible input combinations. As a result, a total of 100,000 input–output pairs were generated and used to train the NN surrogate models. It is important to note that the training data was generated for a fixed pipe configuration. Specifically, the pipe diameter, wall thickness, Young's modulus, yield strength, and hardening modulus were all held constant. Therefore, the trained surrogate models are applicable only to this configuration and must be retrained if applied to pipelines with different geometries or material properties.

b. NN model setup

To approximate the nonlinear relationships between the derivatives and internal forces, two separate NNs were developed: One for predicting the axial force (N) and the other for the bending moment (M). To balance predictive accuracy and computational efficiency, a sensitivity analysis was conducted by evaluating eight network architectures with varying layer numbers and sizes. For each case, the mean absolute error (MAE) on the testing dataset was recorded for both N and M predictions, along with the average training time of the two models (Table 1). The results indicated that increasing the network size from 32 to 128 neurons substantially improved prediction accuracy. However, a further increase to 256 neurons resulted in higher test errors and longer training times, which are likely attributable to overfitting. Based on these results, both models were configured as a three-layer multilayer perceptron (MLP) with 128 neurons in each hidden layer. All hidden layers employ the GELU activation function [52], selected for its smooth activation, allowance of negative outputs, and superior performance observed in the tests, while the output layer adopts a linear activation. The overall NN architecture is illustrated in Fig. 4.

Table 1
Sensitivity analysis of NN architectures.

Cases	Test MAE		Average time (min)
	N (kN)	M (kN m)	
32–32	44.6	5.5	4.8
32-32-32	25.0	2.9	8.6
64-64	23.3	3.2	5.6
64-64-64	20.5	1.7	11.9
128-128	15.2	2.8	10.4
128-128-128	**8.7**	**1.4**	**16.7**
256–256	11.6	2.3	14.2
256-256-256	9.4	1.0	21.8

Prior to training, all input features were standardized using z-score normalization to enforce zero mean and unit variance. The complete dataset was randomly split into training and testing subsets using an 80:20 ratio. Additionally, 20% of the training data was reserved for validation during model training. Both networks were trained using the Adam optimizer with an initial learning rate of 0.01. To enhance training efficiency, a learning rate scheduler was applied to automatically reduce the learning rate when the validation loss plateaued. The loss function used was mean absolute error (MAE), defined as:

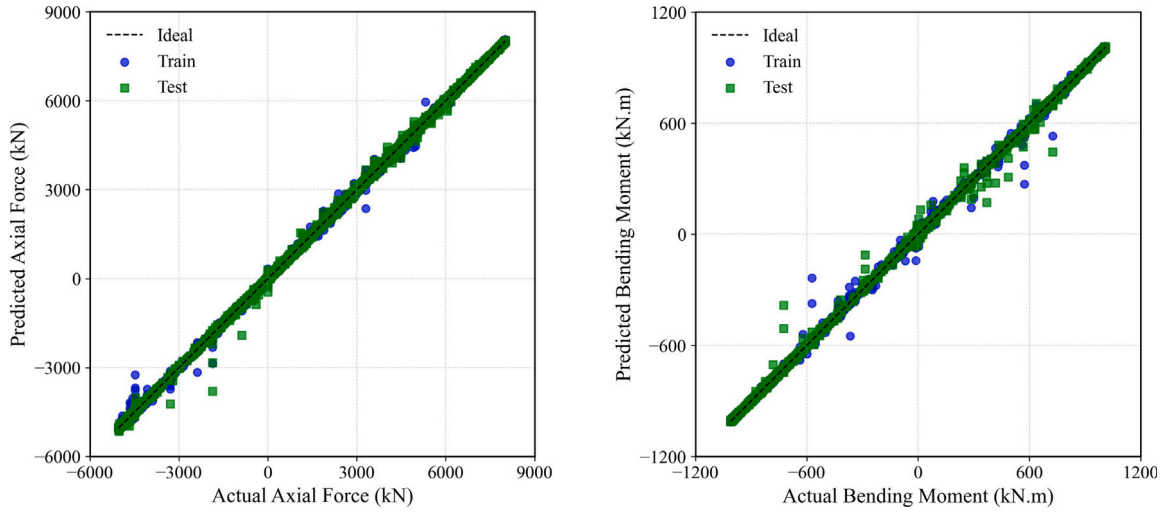
$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|$$

where y_i is the reference value, $\hat{y}_i$ is the predicted value, and n is the total number of samples.

c. Training results

The NN models were implemented using TensorFlow with GPU acceleration. Fig. 5 presents the comparison between predicted and reference values for both the axial force and bending moment models. The data points of training and testing align both closely with the ideal $y = x$ line, demonstrating the high approximation accuracy achieved by the NN surrogates.

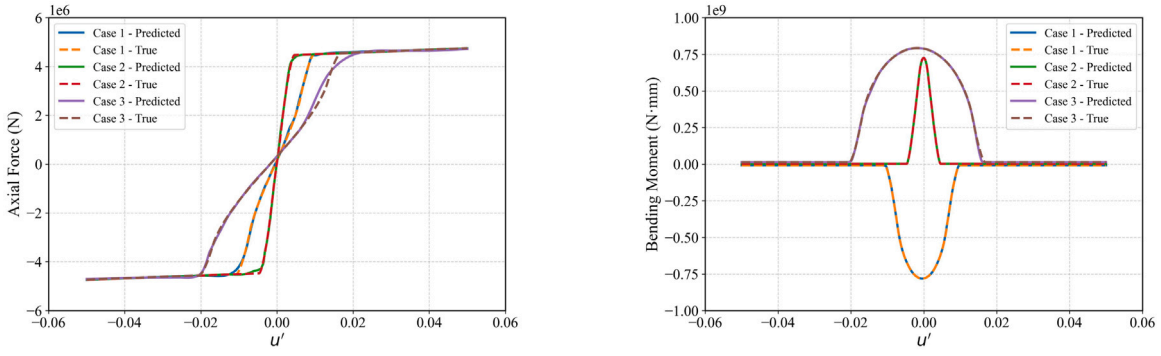
To further evaluate the predictive performance of the surrogate models, a set of tests was conducted in which each input variable



(a) Axial force (N) prediction performance

(b) Bending moment (M) prediction performance

Fig. 5. Predicted versus reference values for the NN models.



(a) N vs. u'

(b) M vs. u'

Fig. 6. Predicted and reference results of N and M versus u', under three input settings: Case 1: $v' = -0.03$, $v'' = -3 \times 10^{-5}$; Case 2: $v' = 0.01$, $v'' = 1 \times 10^{-5}$; Case 3: $v' = 0.06$, $v'' = 6 \times 10^{-5}$.

Table 2
R^2 values for predicted internal forces under different test cases.

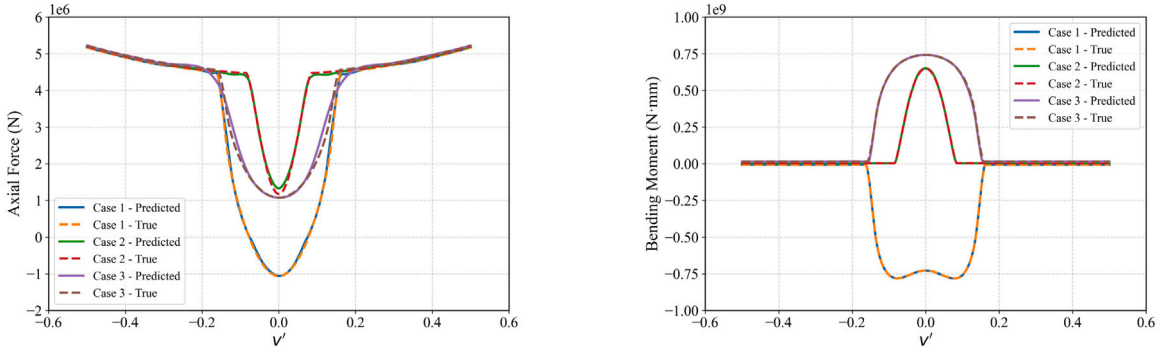
Test case	N vs. u'	M vs. u'	N vs. v'	M vs. v'	N vs. v''	M vs. v''
Case 1	0.9999	1.0000	0.9995	1.0000	0.9975	0.9999
Case 2	1.0000	1.0000	0.9982	1.0000	0.9966	0.9998
Case 3	0.9992	0.9998	0.9926	0.9998	0.9992	0.9998

was varied across its full range, while the other two inputs were held constant at three representative combinations. Figs. 6–8 show the predicted and reference results for both N and M with respect to u', v', and v'', with corresponding R^2 values listed in Table 2. In all cases, the NN predictions exhibit excellent agreement with the reference results obtained from numerical integration, confirming the model's capability to accurately capture the nonlinear relationships between displacement derivatives and internal force response.

2.2.3. The implementation procedure

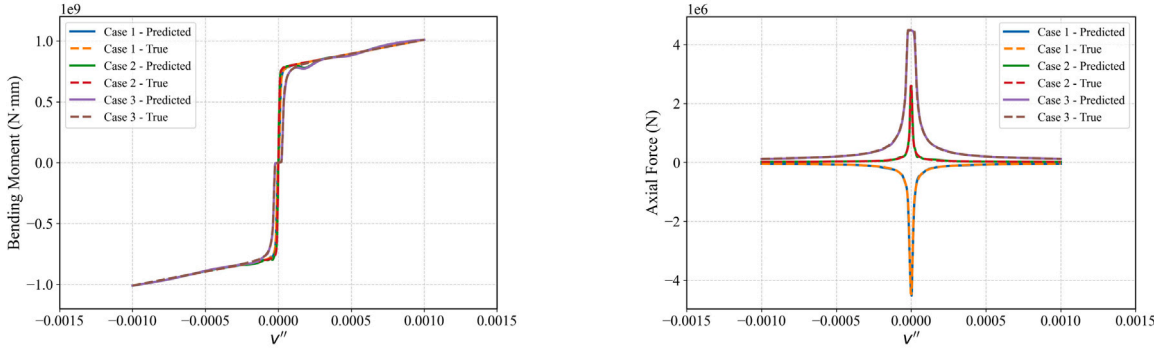
Once the NN models are properly trained, they can be integrated into the FDM framework to predict the axial force and bending moment required in the governing equations. This section outlines the overall implementation procedure of the NN-FDM approach. As shown in Fig. 9, the pipeline is divided into three segments, where the middle segment affected by ground movement, and the adjacent segments remain in stable ground. The implementation procedure consists of the following steps:

1. **Domain Discretization:** The pipeline is discretized into a series of grid points along its axial direction. Each grid point corresponds to a node where the unknown axial and lateral displacements (u_i, v_i) are defined.
2. **Initial Guess Assignment:** At the first PGD step, the nodal displacements are initialized to zero. For subsequent PGD steps, the converged solution from the previous step is used as the initial guess to enhance solver stability and efficiency.
3. **Derivative Evaluation:** Based on the current estimate of nodal displacements, the first and second spatial derivatives (u_i', v_i', v_i'') are computed using finite difference approximations.
4. **Neural Network Prediction:** A pre-trained NN constitutive model is used to predict the axial force N_i and bending moment M_i at each node, using the local displacement derivatives (u_i', v_i', v_i'') as inputs.
5. **Equation Formulation:** At each interior node, the governing residual equations $R_i(u, v)$ are constructed based on the predicted internal forces and imposed external loading. These equations, together with boundary conditions, form a nonlinear system.
6. **Nonlinear System Solution:** The system of nonlinear equations is solved using the Dogbox algorithm implemented in SciPy's



(a) N vs. v' (b) M vs. v'

Fig. 7. Predicted and reference results of N and M versus v', under three input settings: Case 1: $u' = -0.003$, $v'' = -3 \times 10^{-5}$; Case 2: $u' = 0.001$, $v'' = 1 \times 10^{-5}$; Case 3: $u' = 0.006$, $v'' = 6 \times 10^{-5}$.



(a) N vs. v'' (b) M vs. v''

Fig. 8. Predicted and reference results of N and M versus v'', under three input settings: Case 1: $u' = -0.003$, $v' = -0.03$; Case 2: $u' = 0.001$, $v' = 0.01$; Case 3: $u' = 0.006$, $v' = 0.06$.

`least_squares` function [53]. At each iteration, the residuals $R_i(u, v)$ are evaluated:

- If $|R_i(u, v)| < \epsilon$, convergence is achieved and the solution is obtained.
- Otherwise, the nodal displacements are updated and the process is repeated until convergence.

More details on the solver behavior, implementation challenges, and strategies used to improve convergence are provided in Section 2.2.4.

2.2.4. Solver behavior and convergence challenges

The proposed NN-FDM introduces NN surrogates directly into the numerical solution of a nonlinear system. Unlike the explicit, physics-based equations, NNs approximate relationships based on learned data patterns. While this allows for greater flexibility and modeling of complex behavior, it also introduces approximation errors that can propagate through a chain of dependencies in the numerical algorithm. Specifically, inaccuracies in NN predictions can lead to errors in residual evaluation, which in turn affect the finite-difference Jacobian approximation. This may result in incorrect descent directions and unstable solver updates.

The solver behavior resulting from this issue is illustrated in Fig. 10(a), which compares the performance of the original FDM and the NN-FDM for the case of $\delta = 1.0\,\text{m}$ and $\beta = 60°$. Both methods were solved using a nonlinear solver (the `root` function in the `scipy.optimize` library). The original FDM exhibits smooth convergence, whereas the NN-FDM fails to converge, resulting in a nearly flat solution trajectory and a persistently high residual. The following subsections outline the numerical challenges underlying this behavior and the strategies adopted to address them.

a. Implementation of a trust-region based nonlinear solver

One of the main reasons for selecting the Dogbox method is its use of a trust-region strategy, which enhances solver stability during the solution of nonlinear systems with noisy or approximate gradients. Unlike solvers that follow the steepest descent or Newton direction and may take full steps regardless of function behavior, the Dogbox method restricts each update step to remain within a bounded region around the current solution estimate. This region is adaptively adjusted based on the agreement between predicted and actual residual reductions. Table 3 summarizes the Dogbox trust-region procedure, including the mechanism used to update the trust-region radius.

In the NN-FDM framework, where residuals are indirectly influenced by NN predictions, such conservative step control is particularly beneficial. NNs may exhibit local non-smoothness and weak sensitivity to near-zero inputs, leading to unstable descent directions if not properly regulated. The trust-region formulation mitigates this issue by prioritizing stability over aggressiveness, ensuring that each update remains within a domain where the model behaves reliably. This conservative strategy was found to consistently improve solver convergence across a wide range of cases in this study.

b. Adjustment of finite difference step size for Jacobian approximation

In the Dogbox method, the Jacobian matrix is approximated using finite differences by perturbing each input variable with a fixed step size ϵ, as described in Table 3. By default, the solver selects the perturbation size based on the scale of each variable. In the implementation of the NN-FDM framework, this automatic scaling often resulted in perturbation sizes on the order of 10^{-8}, which proved insufficient to

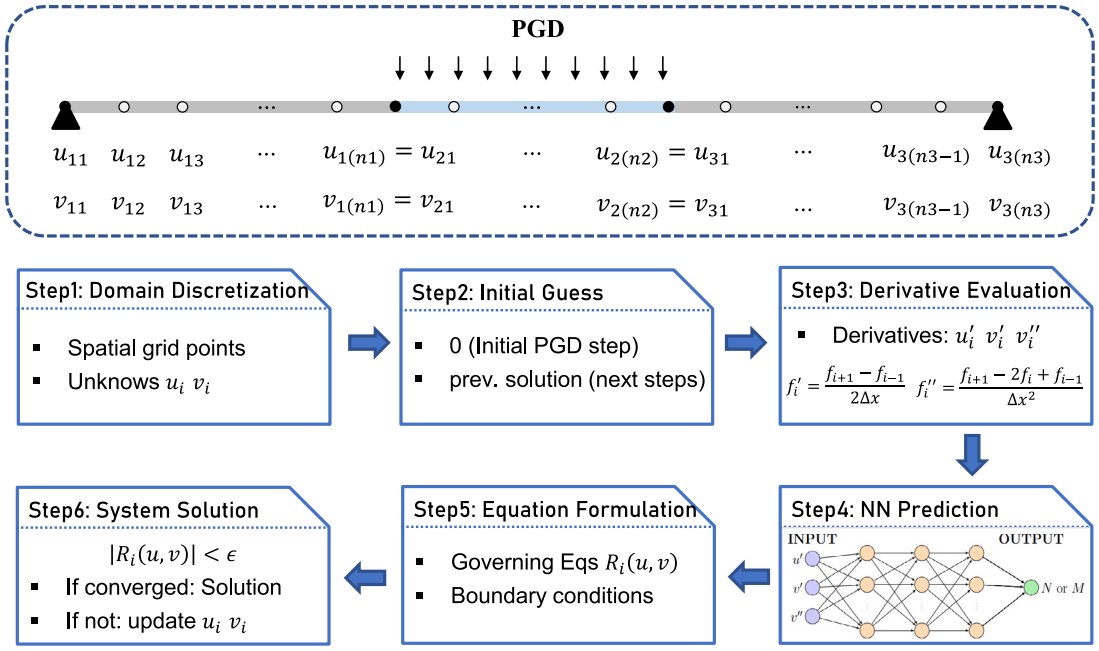
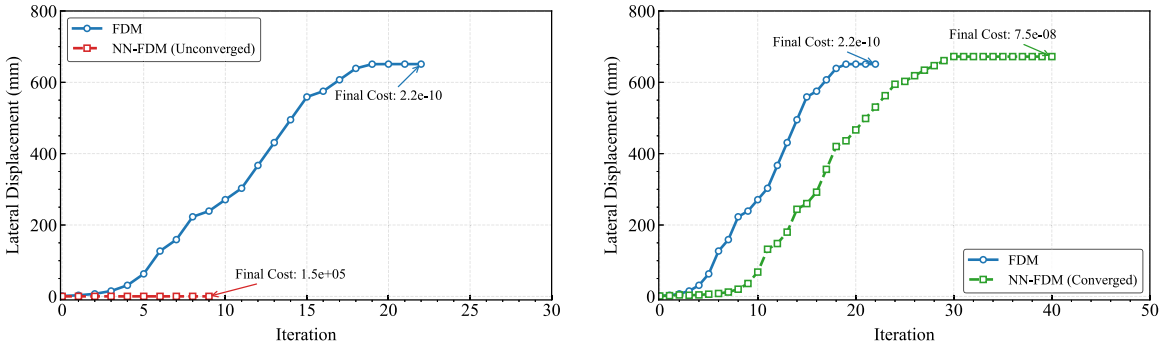


Fig. 9. Schematic view of the calculation procedure of the proposed method.



(a) NN-FDM without enhancements (Unconverged).

(b) NN-FDM with solver enhancements (Converged).

Fig. 10. Solver behavior of NN-FDM for the case $\delta = 1.0\,\text{m}, \beta = 60°$.

Table 3
Workflow of the nonlinear solver using Dogbox method [54].

Algorithm: Dogbox Trust Region Method for Solving $\mathbf{R}(\mathbf{x}) = 0$
1. Define the initial guess $\mathbf{x}_0$ and evaluate the initial residual $\mathbf{R}_0 = \mathbf{R}(\mathbf{x}_0)$
2. Estimate the Jacobian matrix $\mathbf{J}$ using finite difference approximation: $\mathbf{J}_{:,j} \approx \frac{\mathbf{R}(\mathbf{x}_0 + \epsilon \mathbf{e}_j) - \mathbf{R}(\mathbf{x}_0)}{\epsilon}$ where $\mathbf{e}_j$ is the unit vector in the jth direction and ϵ is the finite difference step size.
3. Determine the update step $\Delta\mathbf{x}$ within the trust region by solving: $\min_{\Delta\mathbf{x}} \left\|\mathbf{J}\Delta\mathbf{x} + \mathbf{R}(\mathbf{x}_0)\right\|^2$ subject to $\|\Delta\mathbf{x}\| \le \Delta$
4. Update the solution: $\mathbf{x}_{\text{new}} = \mathbf{x}_0 + \Delta\mathbf{x}$ and compute $\mathbf{R}_{\text{new}} = \mathbf{R}(\mathbf{x}_{\text{new}})$
5. Evaluate the reduction ratio ρ between actual and predicted residual decrease: Actual reduction: $\frac{1}{2}\|\mathbf{R}(\mathbf{x}_0)\|^2 - \frac{1}{2}\|\mathbf{R}(\mathbf{x}_{\text{new}})\|^2$ Predicted reduction: $-\mathbf{R}_0^T\mathbf{J}\Delta\mathbf{x} - \frac{1}{2}\Delta\mathbf{x}^T\mathbf{J}^T\mathbf{J}\Delta\mathbf{x}$ $\rho = \frac{\text{Actual reduction}}{\text{Predicted reduction}}$
6. Update the trust region radius Δ based on ρ and the current step norm $\|\Delta\mathbf{x}\|$: • If $\rho < 0.25$, $\Delta_{\text{new}} = 0.25\|\Delta\mathbf{x}\|$ • If $\rho > 0.75$, $\Delta_{\text{new}} = 2\Delta$ • Otherwise, $\Delta_{\text{new}} = \Delta$
7. Set $\mathbf{x}_0 \leftarrow \mathbf{x}_{\text{new}}$ and $\Delta \leftarrow \Delta_{\text{new}}$, then repeat steps 2–6 until convergence criteria are met

produce measurable changes in the NN outputs due to their limited sensitivity to near-zero inputs. As a result, the computed Jacobian matrices became poorly conditioned, leading to inaccurate descent directions and convergence failure.

To address this issue, the finite difference step size was manually fixed at 1×10^{-3}, a value determined through empirical testing. This magnitude was found to be sufficiently large to induce a detectable change in the neural network output, while remaining small enough to satisfy the local linearity assumption required for accurate Jacobian approximation. The effectiveness of this adjustment is demonstrated in Table 4. Step sizes smaller than 1×10^{-3} consistently failed to converge, while larger values increased the iteration count or led to inaccurate convergence. In contrast, a step size of 1×10^{-3} achieved both rapid and accurate convergence, with a minimal final residual and the fewest iterations among all tested configurations.

The strategies described in Section 2.2.4(a) and (b) were found to significantly improve the convergence performance of the NN-FDM method. Fig. 10(b) shows the improved convergence behavior for the same test case presented in Fig. 10(a) ($\delta = 1.0\,\text{m}, \beta = 60°$). With the proposed modifications, the NN-FDM method is able to converge reliably, although it requires more iterations than the original FDM due to the conservative nature of the Dogbox trust-region strategy. This conservative behavior prioritizes stability over rapid descent, ensuring

Table 4
Performance summary of the solver for different step sizes (Case: δ = 1.0 m, β = 60°)

Step size	Initial cost	Final cost	Iterations	Converged
1.00×10^{-5}	1.47×10^{5}	8.62×10^{4}	173	False
1.00×10^{-4}	1.47×10^{5}	4.29×10^{4}	169	False
1.00×10^{-3}	**1.47×10^{5}**	**7.51×10^{-8}**	**40**	**True**
1.00×10^{-2}	1.47×10^{5}	8.72×10^{-8}	61	True
1.00×10^{-1}	1.47×10^{5}	4.83×10^{4}	191	False

Table 5
Soil properties and soil spring parameters.

Parameter	Symbol	Value
Internal friction angle (°)	ϕ	34
Unit weight (kg/m^3)	γ	1760
Burial depth (m)	–	1.8
Axial friction factor	–	0.6
Axial soil resistance (N/mm)	T_u	14
Axial yield displacement (mm)	Δ_t	5
Lateral soil resistance (N/mm)	P_u	204
Lateral yield displacement (mm)	Δ_p	46

that the solver progresses within a region where the model response is well-behaved. As a result, the method achieves a low final residual and effectively mitigates the convergence issues associated with NN prediction errors.

3. Verification of proposed method

Two case studies were designed to evaluate the performance of the proposed method under different material constitutive models. The pipeline considered in both cases is a 559 mm diameter X52 steel pipe with a wall thickness of 7.14 mm. According to the API 5L Specification [55], the material has a yield stress of 360 MPa and an ultimate stress of 460 MPa, with corresponding yield and ultimate strains of 0.18% and 21%, respectively. The total pipeline length is 210 m, consisting of a 10 m geohazard zone located between two 100 m segments. In both case studies, the pipeline is subjected to ground movement occurring in the horizontal plane, characterized by the ground displacement magnitude δ and the intersection angle β. According to the pipe–soil interaction model described in Section 2.2.1, only the axial and lateral soil springs are activated in this scenario. The parameters used to calculate the soil spring properties, as well as the resulting soil resistances and displacements, are provided in Table 5.

Section 3.1 shows the results obtained under a bilinear stress–strain relation, while Section 3.2 extends the analysis by incorporating a nonlinear hardening constitutive model that provides a smooth transition from the elastic to the plastic state. In both cases, the proposed NN-FDM is compared against the original FDM and a FEM benchmark across various ground displacement magnitudes δ and the intersection angle β. Note that the FEM results were obtained using the commercial software ABAQUS, where the pipeline was modeled using PIPE32 elements and the pipe–soil interaction was represented by nonlinear springs implemented with PSI36 elements.

3.1. Bilinear stress–strain relation

In this section, the bilinear stress–strain model is adopted to describe the material behavior of the pipe. The corresponding NN surrogate, trained under this bilinear model, is detailed in Section 2.2.2. To evaluate the performance of the proposed method (NN-FDM), three mesh configurations are considered: Case 1 (31–11–31 nodes), Case 2 (61–25–61 nodes), and Case 3 (101–35–101 nodes), corresponding to coarse, medium, and fine discretizations, respectively. For each case, both the NN-FDM and the FDM are applied using the corresponding mesh, while FEM simulations are performed using the finest mesh and serve as a benchmark for comparison.

Table 6
Mean relative error (MRE) between NN-FDM and reference methods for TSD and CSD.

Cases	Angle	NN-FDM vs. FDM		NN-FDM vs. FEM	
	(deg)	TSD (%)	CSD (%)	TSD (%)	CSD (%)
Case1: 31 11 31	30	3.29	3.81	45.19	51.59
	60	1.83	2.32	26.69	29.83
	90	2.08	2.79	21.19	30.03
Case2: 61 25 61	30	7.62	15.26	22.97	20.60
	60	7.88	17.57	24.33	23.98
	90	10.12	16.60	33.26	34.35
Case3: 101 35 101	30	21.86	17.17	18.42	22.74
	60	17.46	12.42	18.39	12.41
	90	13.13	10.44	26.85	10.52

Figs. 11–13 illustrate the tensile strain demand (TSD, solid lines) and compressive strain demand (CSD, dashed lines) under varying intersection angles for each mesh case. Table 6 provides a quantitative summary of the mean relative errors (MRE) between NN-FDM and the reference methods (original FDM and FEM). In case 1, the NN-FDM shows nearly identical results to the original FDM (Fig. 11), with a maximum MRE of only 3.81%. This confirms the effectiveness of the NN surrogate in reproducing internal force predictions based on displacement derivatives. This high consistency also validates the successful integration of the surrogate model into the finite difference scheme. As expected, both the NN-FDM and the original FDM exhibit deviations from the FEM benchmark due to the relatively low spatial resolution employed in Case 1.

As the mesh density increases in Cases 2 and 3, as shown in Figs. 12 and 13, both NN-FDM and the original FDM exhibit progressively improved agreement toward the FEM benchmark. Although discrepancies between the two approaches become more apparent at higher intersection angles and larger displacements, their predictions remain in close agreement overall. These differences primarily result from the accumulation of prediction errors from the NN model within the iterative solver framework. While the prediction error in each iteration might be minor, the cumulative effect becomes increasingly pronounced as the number of nodes grows. Additionally, the Dogbox solver used in this study employs a trust-region algorithm, which tends to select conservative update steps when faced with noisy gradients introduced by NN outputs. Although this conservative approach enhances stability, it can limit the solver's effectiveness in minimizing the residual, resulting in convergence at a higher residual level compared to the coarse-mesh cases. Consequently, slight deviations in the final strain predictions of the NN-FDM are observed.

It is further noted that, with mesh refinement, the FDM results for compressive strain demand (CSD) align increasingly well with the FEM benchmark, whereas deviations in tensile strain demand (TSD) become more noticeable. This discrepancy may reflect an inherent limitation of the Euler–Bernoulli beam-based formulation adopted in the FDM framework. As strain levels increase, the simplifying assumptions of the theory may become inadequate. Consequently, the FDM may lose accuracy in capturing the full strain response under high-deformation scenarios.

3.2. Nonlinear hardening stress–strain relation

To further assess the generalizability of the proposed NN-FDM method, a nonlinear hardening model is adopted in this section [56]. The constitutive relation used follows a strain hardening law that transitions smoothly from elastic to plastic behavior. The stress σ as

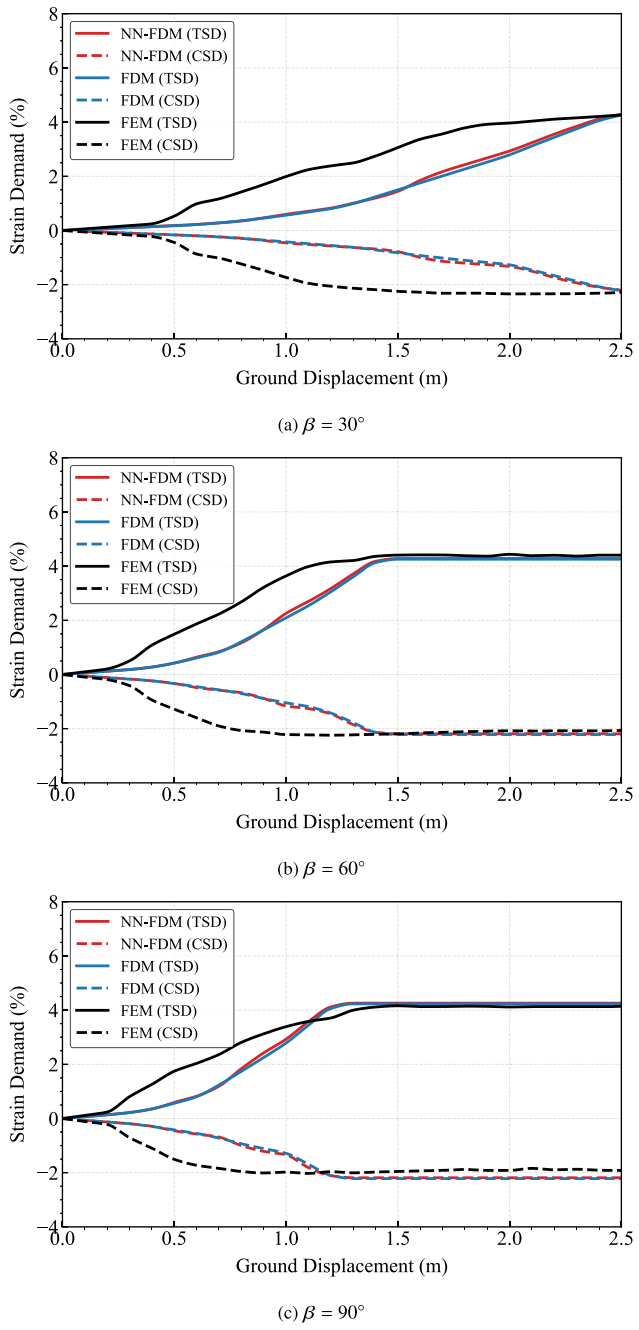


Fig. 11. Strain demand results for Case1 under a bilinear stress–strain relation.

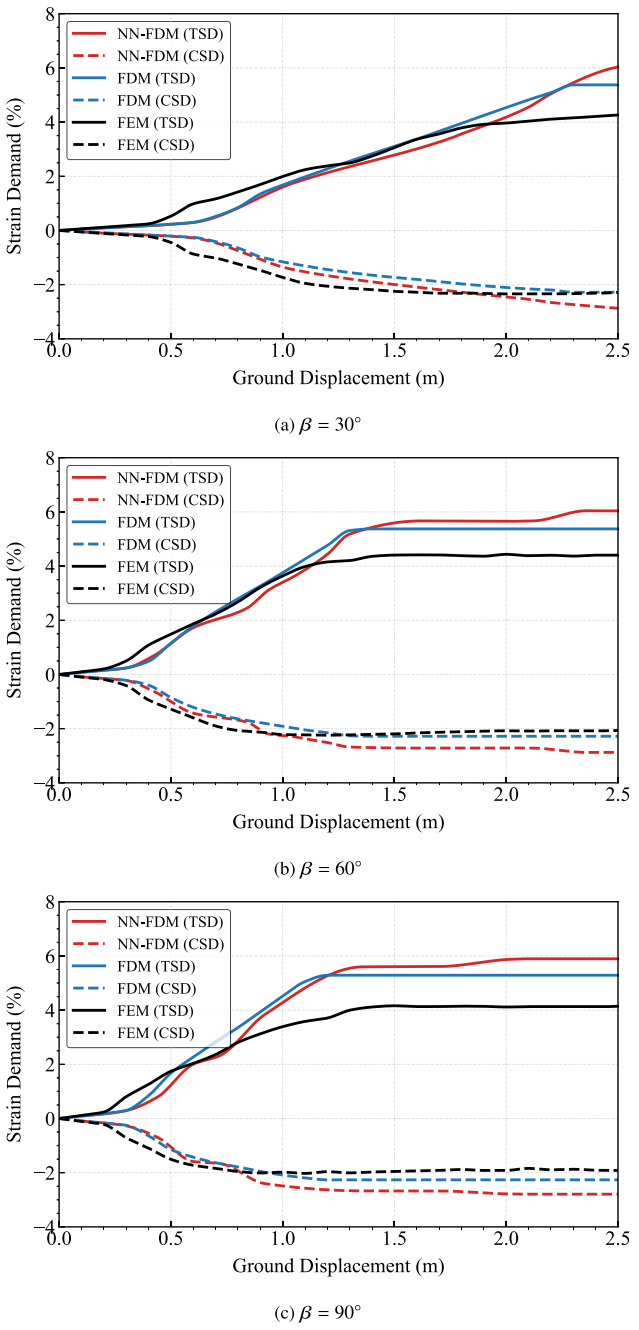


Fig. 12. Strain demand results for Case2 under a bilinear stress–strain relation.

a function of strain ε is expressed as:

$$\sigma = E\varepsilon\left[b + \frac{1-b}{\left(1+\left(\frac{E\varepsilon}{\sigma_y}\right)^R\right)^{1/R}}\right] \tag{8}$$

In this study, $b = 0.002$ and $R = 2$ are used to represent the material behavior of X52 pipe steel.

To incorporate this constitutive law into the NN-FDM framework, a separate neural network surrogate was retrained on a new dataset, in which the internal forces N and M were obtained from displacement derivatives using the nonlinear hardening model defined by Eq. (8). The same sampling strategy and network architecture as described in Section 2.2.2 were employed. The retrained surrogate was subsequently integrated into the FDM framework, and the performance of the method was evaluated against both the original FDM and a FEM benchmark. In contrast to the bilinear case, the internal forces in the original FDM were calculated using Eq. (8) through numerical integration at each solver iteration.

As illustrated in Figs. 14–16 and Table 7, NN-FDM remains in perfect agreement with the original FDM in the coarse mesh case (Case 1), demonstrating the surrogate model's capability to accurately reproduce nonlinear constitutive responses. With increasing mesh density (Cases 2 and 3), deviations between the NN-FDM and FDM become more noticeable, particularly at the tensile strain region. This divergence is consistent with earlier observations under the bilinear model and can be primarily attributed to the compounding effect of NN approximation

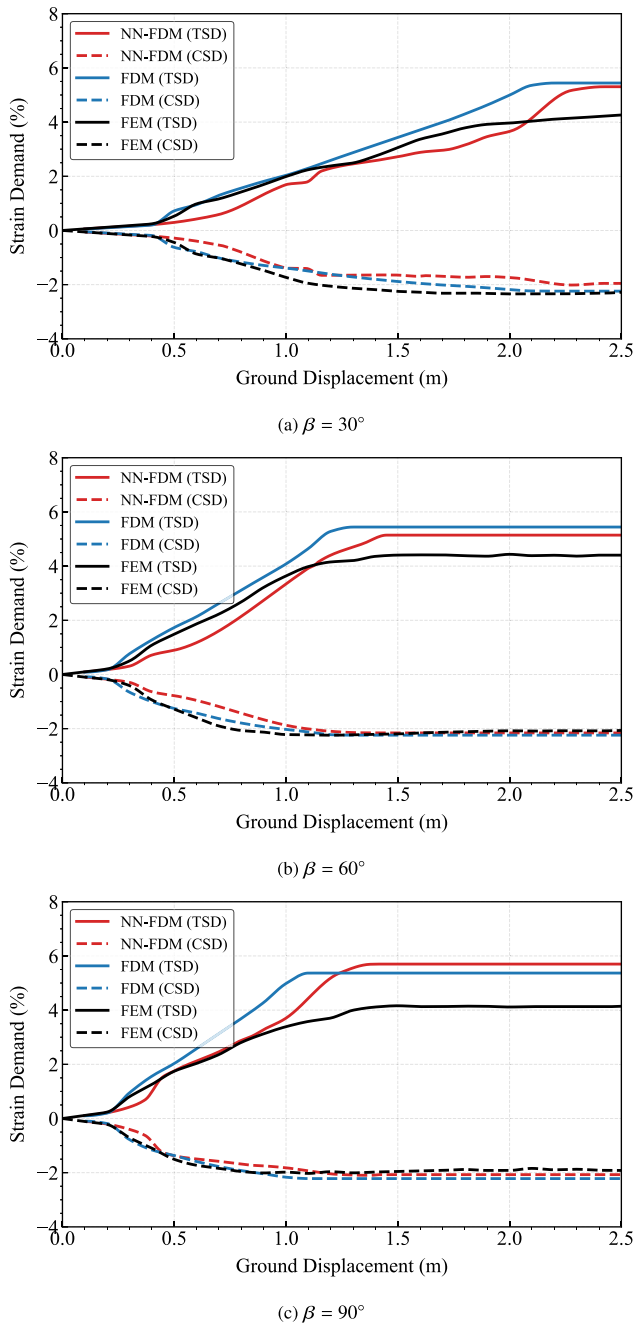


Fig. 13. Strain demand results for Case3 under a bilinear stress–strain relation.

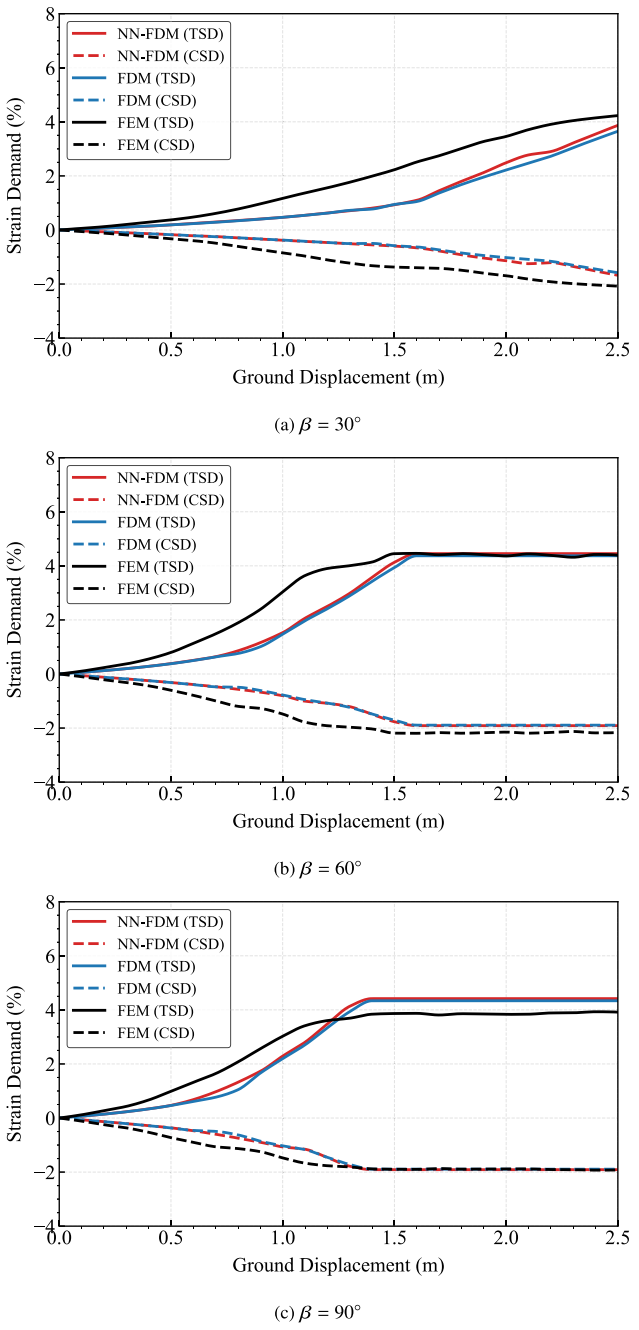


Fig. 14. Strain demand results for Case1 under a nonlinear hardening relation.

errors and the conservative update strategy employed by the Dogbox solver.

To highlight the computational advantage of the proposed method, Fig. 17 presents a comparison of the average runtime between NN-FDM and the original FDM under three intersection angles (30°, 60°, and 90°) across different mesh configurations. All simulations were performed on a desktop equipped with a 13th Gen Intel® Core™ i7-13700 processor and 32 GB of RAM. As shown in the figure, the NN-FDM consistently achieves significantly lower runtimes compared to the original FDM across all cases. The difference in computational cost becomes increasingly pronounced as the mesh is refined. In the fine mesh case (Case 3), the original FDM with numerical integration requires, on average, over 25 min per simulation, whereas NN-FDM completes the same task in under 1.5 min. This efficiency gain is primarily attributed to the replacement of repeated numerical integration with NN surrogates, which greatly accelerates the evaluation of internal forces during the iterative solution process.

4. Comparison of the proposed method with existing analytical methods

To further evaluate the performance of the proposed NN-FDM, it is compared with existing analytical methods through a case study of a buried pipeline subjected to strike-slip faulting. The present case adopts the same parameters as those reported in Zheng et al. [24], which included results from several analytical solutions (Newmark and Hall [4], Kennedy et al. [5], Karamitros et al. [10], Sarvanis and Karamanos [13]) together with FEM and FDM simulations. In this study, these reported results are extracted from the graphical outputs in [24] and used as reference data to validate the proposed method.

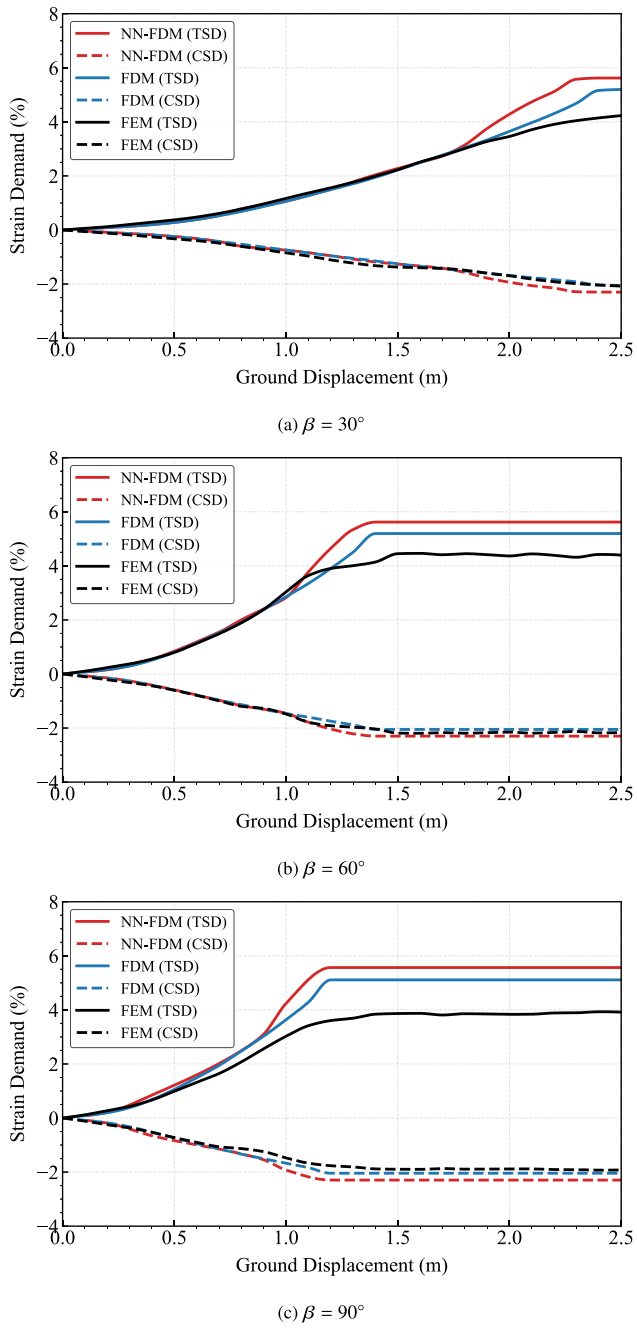


Fig. 15. Strain demand results for Case2 under a nonlinear hardening relation.

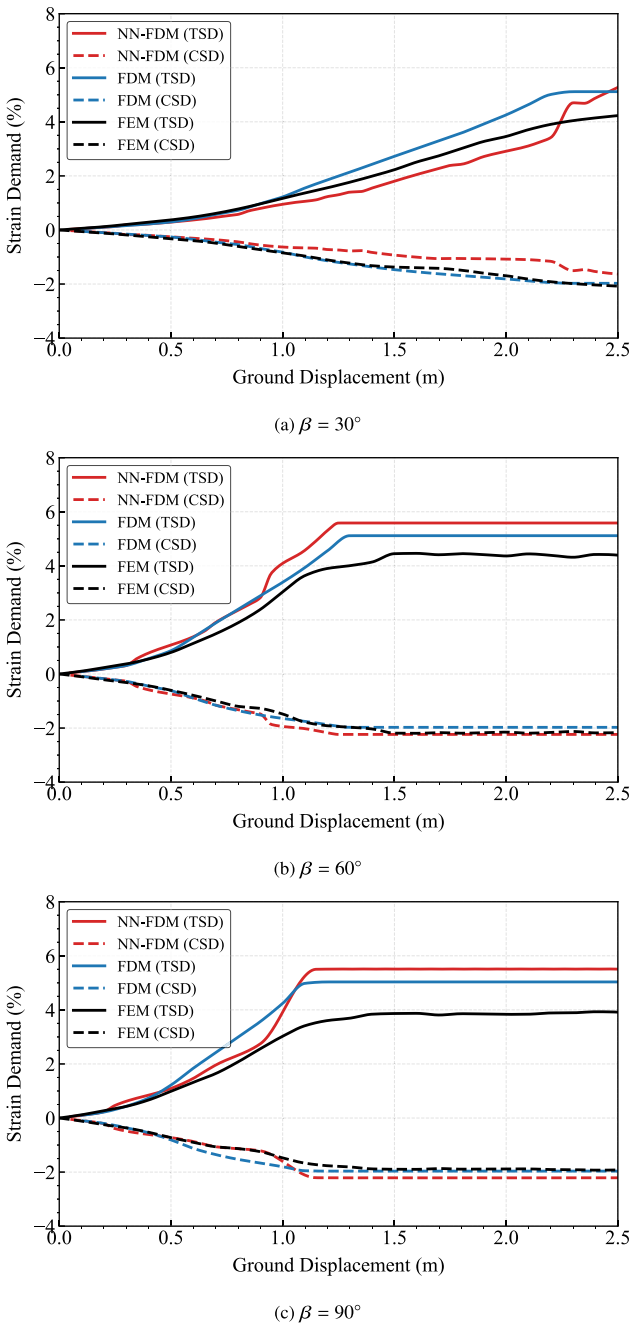


Fig. 16. Strain demand results for Case3 under a nonlinear hardening relation.

The modeled pipeline is 1000 m in length, with 500 m on each side of the fault. The right end of the pipe is free to move while the left end is fixed. Ground displacement is applied in the horizontal plane at the fault zone, and three pipe–fault intersection angles (β = 30°, 60°, 90°) are considered. The ground displacement magnitude ranges from 0 to 3 m, and tensile strain demand is selected as the performance metric. A uniform mesh size of 1 m is used in the NN-FDM simulation, consistent with the reference FEM and FDM models. Fig. 18 presents the comparison results, where the proposed NN-FDM predictions (solid green line) are shown alongside analytical solutions and numerical results extracted from the literature. As shown in Fig. 18, the proposed NN-FDM method exhibits excellent agreement with both the original FDM and the FEM benchmark across all intersection angles and fault displacement levels. At small fault displacements, where the pipeline response remains elastic, the NN-FDM results closely follow the FEM benchmark, capturing the initial linear trend. As fault displacement increases and plastic deformation develops, NN-FDM continues to accurately reproduce the onset of yielding and the subsequent strain plateau, maintaining excellent agreement with the FEM predictions. In addition, the close consistency between NN-FDM and the original FDM results further confirms the effectiveness of the neural network model in replacing the conventional constitutive formulation within the finite difference framework. By comparison, the analytical solutions show varying degrees of discrepancy from the FEM results, primarily due to simplified assumptions such as neglecting lateral pipe–soil interaction (Newmark and Hall), modeling the pipeline as a flexible cable (Kennedy et al.), or prescribing predefined displacement shapes (Sarvanis and Karamanos). Among these, the method by Karamitros

Table 7
Mean relative error (MRE) between NN-FDM and reference methods for TSD and CSD (Nonlinear).

Cases	Angle	NN-FDM vs. FDM		NN-FDM vs. FEM	
	(deg)	TSD (%)	CSD (%)	TSD (%)	CSD (%)
Case1: 31 11 31	30	4.23	4.12	44.18	44.51
	60	3.04	2.19	25.65	30.35
	90	4.38	2.81	25.03	18.85
Case2: 61 25 61	30	6.39	6.16	16.55	13.85
	60	7.41	8.28	20.13	7.73
	90	9.91	11.53	36.86	21.32
Case3: 101 35 101	30	21.17	25.08	19.47	26.81
	60	9.67	11.34	27.39	11.08
	90	12.23	14.26	34.30	14.81

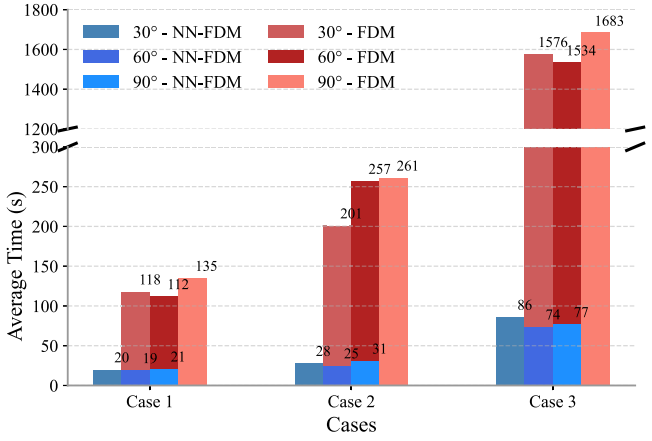


Fig. 17. Runtime comparison between NN-FDM and FDM.

et al. performs well overall, but overestimates strain demand at large fault displacements for $\beta = 90°$. This discrepancy is attributed to the neglect of axial strain contributions and their interaction with bending deformation under this specific loading condition.

5. Conclusion

A neural network-enhanced finite difference method (NN-FDM) has been developed to evaluate the strain demand of pipelines subjected to permanent ground displacement, extending the original FDM framework to accommodate nonlinear hardening models. In this approach, two NNs are trained to directly map displacement derivatives to internal force responses, enabling an efficient and data-driven representation of nonlinear material behavior.

Several numerical challenges were encountered in the integration of NN surrogates into the FDM framework. In particular, the weak input sensitivity to near-zero inputs and local smoothness of NN outputs were found to hinder convergence in certain cases. To improve solver robustness, the solver incorporated a trust-region update strategy along with a manually adjusted step size for Jacobian estimation. These strategies proved effective in maintaining stability and accuracy during iterative solution procedures. The findings are expected to provide practical guidance for the development of hybrid physics–machine learning frameworks, particularly when NNs are embedded within nonlinear solvers.

The proposed method is validated through two case studies involving both bilinear and nonlinear hardening stress–strain relationships. Results indicate that NN-FDM exhibits excellent agreement with the original FDM under coarse mesh discretization, while maintaining acceptable accuracy relative to the FEM benchmark as the mesh is refined. Deviations observed in fine-mesh cases are primarily due to solver

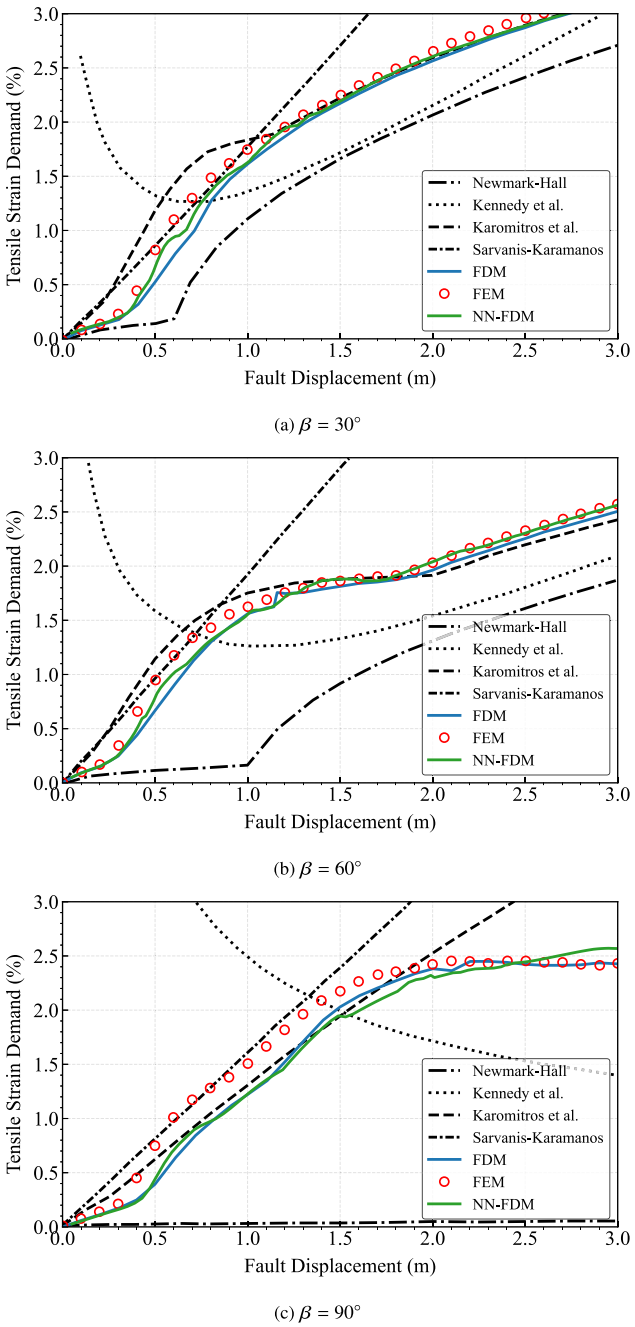


Fig. 18. Comparison of NN-FDM with analytical and numerical benchmarks.. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

sensitivity to the approximation characteristics of the NN model under highly nonlinear conditions. Nevertheless, the overall predictive performance remains sufficient for practical engineering applications. In terms of computational efficiency, NN-FDM offers a notable improvement in computational efficiency compared to the original FDM, which requires numerical integration when handling a nonlinear hardening model. By replacing integration procedures with NN surrogates, the proposed method significantly accelerates the process of solving the nonlinear equation system. To further demonstrate its applicability, a separate benchmark case involving a buried pipeline subjected to strike-slip faulting was considered, where NN-FDM was compared with several published analytical solutions. The results confirm that, while

analytical models provide useful closed-form estimates under simplified assumptions, they deviate significantly from FEM benchmarks. In contrast, NN-FDM consistently achieves high accuracy across different intersection angles and displacement levels, thereby highlighting both its advantage over analytical methods and its robustness to diverse pipeline configurations.

To conclude, the proposed NN-FDM method offers an effective and efficient framework for predicting pipeline strain demand under PGD across a wide range of constitutive models. Its computational advantages and modeling flexibility makes it particularly suitable for rapid assessments and pre-screening applications in geohazard-prone regions. Although minor deviations are observed under fine-mesh conditions, these are generally acceptable for preliminary design and engineering use. However, for scenarios involving strong nonlinearity, such as long pipeline segments or highly refined discretizations, the current implementation may face limitations in accuracy and convergence. Addressing this limitation may require the use of more advanced and robust solvers, which are expected to enhance convergence stability and allow the method to handle highly nonlinear scenarios. It is also important to note that the present framework is designed for global strain demand prediction and does not explicitly account for local instability modes, such as cross-sectional buckling. More advanced approaches such as full 3D finite element modeling may be needed to resolve local cross-sectional deformation in practical application. In addition, its application has been limited to one-dimensional pipeline configurations due to the use of Euler–Bernoulli beam theory. Nevertheless, the underlying mechanism of NN-FDM is transferable, and with appropriate higher-dimensional theories, the framework can be extended to more complex geometries and loading conditions.

CRediT authorship contribution statement

Beilei Ji: Writing – original draft, Visualization, Validation, Methodology, Formal analysis. **Qipei Mei:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Nader Yoosef-Ghodsi:** Writing – review & editing, Project administration, Conceptualization. **Samer Adeeb:** Writing – review & editing, Supervision, Methodology, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Beilei Ji reports financial support was provided by China Scholarship Council. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the program of the China Scholarship Council [No. CSC202308180007].

Data availability

Data will be made available on request.

References

[1] O'Rourke TD. Lessons learned from earthquake damage to passive systems and hazardous materials. Buffalo, NY: U.S. National Center for Earthquake Engineering Research; 1999, NCEER-99-0003.

[2] Geertsema M, Menounos B, Chow W, Battle J. The 2018 Peace River Regional district landslide swarm, Northeastern British Columbia, Canada. Landslides 2022;19(3):613–30.

[3] Pipeline and Hazardous Materials Safety Administration. Pipeline incident 20 year trends. Tech. rep., U.S. Department of Transportation; 2021, [Accessed 15 March 2024].

[4] Newmark NM, Hall WJ. Pipeline design to resist large fault displacement. In: Proc., U.S. National conference on earthquake engineering. 1975, p. 416–25.

[5] Kennedy RP, Chow AT, Williamson RA. Fault movement effects on buried oil pipeline. J Transp Eng Div ASCE 1977;103(5):617–33. http://dx.doi.org/10.1061/TPEJAN.0000659.

[6] Wang LRL, Yeh YH. A refined analysis of buried pipeline response to fault movement. Earthq Eng Struct Dyn 1985;13(1):75–96. http://dx.doi.org/10.1002/eqe.4290130109.

[7] Wang LRL, Yeh YH. Seismic design of pipeline for fault movement effects. ASME J Press Vessel Technol 1986;108:202–8. http://dx.doi.org/10.1115/1.3264770.

[8] Chiou YJ, Chi SY, Chang HY. A study on buried pipeline response to fault movement. J Press Vessel Technol 1994;116(1):36–41. http://dx.doi.org/10.1115/1.2929556.

[9] Takada S, Hassani N, Fukuda K. A new proposal for simplified design of buried steel pipes crossing active faults. Earthq Eng Struct Dyn 2001;30(8):1243–57. http://dx.doi.org/10.1002/eqe.62.

[10] Karamitros DK, Bouckovalas GD, Kouretzis GP. Stress analysis of buried steel pipelines at Strike-Slip fault crossings. Soil Dyn Earthq Eng 2007;27(3):200–11. http://dx.doi.org/10.1016/j.soildyn.2006.08.001.

[11] Trifonov OV, Cherniy VP. A Semi-Analytical approach to a nonlinear Stress–Strain analysis of buried steel pipelines crossing active faults. Soil Dyn Earthq Eng 2010;30(11):1298–308. http://dx.doi.org/10.1016/j.soildyn.2010.06.002.

[12] Trifonov OV, Cherniy VP. Elastoplastic Stress–Strain analysis of buried steel pipelines subjected to fault displacements with account for service loads. Soil Dyn Earthq Eng 2012;33(1):54–62. http://dx.doi.org/10.1016/j.soildyn.2011.10.001.

[13] Sarvanis GC, Karamanos SA. Analytical model for the strain analysis of continuous buried pipelines in geohazard areas. Eng Struct 2017;152:57–69. http://dx.doi.org/10.1016/j.engstruct.2017.08.060.

[14] Yan X, Zhang L, Yang X. Strain response study of oil-gas pipelines crossing earthquake faults based on the pipe-soil coupling and the large deformation shell model. China Civ Eng J 2010;43:132–9.

[15] Liu X, Sun S. Strain based design of buried pipelines crossing faults. Spec Struct 2005;22(2):81–5. http://dx.doi.org/10.3969/j.issn.1001-3598.2005.02.028.

[16] Gu X, Zhang H. Research on aseismatic measures of gas pipeline crossing a fault for strain-based design. In: Proceedings of the ASME pressure vessels and piping conference. 2009, p. 43680. http://dx.doi.org/10.1115/PVP2009-77987.

[17] Vazouras P, Dakoulas P, Karamanos S. Pipe–soil interaction and pipeline performance under strike–slip fault movements. Soil Dyn Earthq Eng 2015;72:48–65. http://dx.doi.org/10.1016/j.soildyn.2015.01.014.

[18] Sarvanis G, Karamanos S, Vazouras P, Mecozzi E, Lucci A, Dakoulas P. Permanent earthquake-induced actions in buried pipelines: numerical modeling and experimental verification. Earthq Eng Struct Dyn 2018;47(4):966–87. http://dx.doi.org/10.1002/eqe.3001.

[19] Vazouras P, Karamanos S, Dakoulas P. Finite element analysis of buried steel pipelines under strike-slip fault displacements. Soil Dyn Earthq Eng 2010;30(11):1361–76. http://dx.doi.org/10.1016/j.soildyn.2010.06.011.

[20] American Lifelines Alliance (ALA). Guidelines for the design of buried steel pipe. American Society of Civil Engineers; 2001.

[21] Liu A, Hu Y, Zhao F, Li X, Takada S, Zhao L. An equivalent-boundary method for the shell analysis of buried pipelines under fault movement. Acta Seismol Sin 2004;17(1):150–6. http://dx.doi.org/10.1007/s11589-004-0078-1, Supp -0150-07.

[22] Liu X, Zhang H, Han Y, Xia M, Zheng W. A semi-empirical model for peak strain prediction of buried X80 steel pipelines under compression and bending at strike-slip fault crossings. J Nat Gas Sci Eng 2016;32:465–75. http://dx.doi.org/10.1016/j.jngse.2016.04.054.

[23] Grossmann C, Roos H-G, Stynes M. Numerical treatment of partial differential equations. Universitext, 1st ed.. Springer Berlin, Heidelberg; 2007, http://dx.doi.org/10.1007/978-3-540-71584-9.

[24] Zheng Q, Li Y, Yoosef-Ghodsi N, Fowler M, Kainat M, Adeeb S. A finite difference-based approach for strain demand prediction of inelastic pipes subjected to permanent ground displacements. Eng Struct 2022;273:115072. http://dx.doi.org/10.1016/j.engstruct.2022.115072.

[25] Zheng Q, Graf-Alexiou L, Li Y, Yoosef-Ghodsi N, Fowler M, Kainat M, Adeeb S. Strain demand of elastic pipes subjected to permanent ground displacements using the finite difference method. J Pipeline Sci Eng 2021;1:176–86. http://dx.doi.org/10.1016/j.jpse.2021.06.001.

[26] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521(7553):436–44. http://dx.doi.org/10.1038/nature14539.

[27] Cichy RM, Kaiser D. Deep neural networks as scientific models. Trends Cogn Sci 2019;23(4):305–17. http://dx.doi.org/10.1016/j.tics.2019.01.009.

[28] Liu C, Han Y, Gao S. Research on stability and application of neural network. Procedia Comput Sci 2025;262:1180–6. http://dx.doi.org/10.1016/j.procs.2025.05.158.

[29] Dornheim J, Morand L, Nallani HJ, Helm D. Neural networks for constitutive modeling: From universal function approximators to advanced models and the integration of physics. Arch Comput Methods Eng 2024;31:1097–127. http://dx.doi.org/10.1007/s11831-023-10009-y.

[30] Taraghi P, Li Y, Adeeb S. Prediction of ground movement-induced pipe responses considering variable PGD magnitudes using physics-informed neural networks and transfer learning. Eng Struct 2025;332:120018. http://dx.doi.org/10.1016/j.engstruct.2025.120018.
[31] Taraghi P, Li Y, Adeeb S. Application of physics-informed neural networks for nonlinear analysis of buried steel pipelines to support pipe reinforcement against ground movement. Comput Geotech 2025;186:107389. http://dx.doi.org/10.1016/j.compgeo.2025.107389.
[32] Ghaboussi J, Garret JH, Wu X. Knowledge-based modeling of material behavior with neural networks. J Eng Mech 1991;117(1):132–53. http://dx.doi.org/10.1061/(ASCE)0733-9399(1991)117:1(132).
[33] Kortesis S, Panagiotopoulos PD. Neural networks for computing in structural analysis: methods and prospects of applications. Internat J Numer Methods Engrg 1993;36(13):2305–18. http://dx.doi.org/10.1002/nme.1620361310.
[34] Yagawa G, Okuda H. Finite element solutions with feedback network mechanism through direct minimization of energy functionals. Internat J Numer Methods Engrg 1996;39(5):867–83. http://dx.doi.org/10.1002/(SICI)1097-0207(19960315)39:5<867::AID-NME886>3.0.CO;2-Q.
[35] Papadrakakis M, Papadopoulos V, Lagaros N. Structural reliability analysis of elastic–plastic structures using neural networks and Monte Carlo simulation. Comput Methods Appl Mech Engrg 1996;136:145–63. http://dx.doi.org/10.1016/0045-7825(96)01016-3.
[36] Bergstra J, Breuleux O, Bastien F, Lamblin P, Pascanu R, Desjardins G, Turian J, Warde-Farley D, Bengio Y. Theano: a CPU and GPU math compiler in Python. In: Proceedings of the 9th python in science conference (sciPy). 2010, p. 1–7.
[37] Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado G, Davis A, Dean J, Devin M, et al. TensorFlow: Large-scale machine learning on heterogeneous distributed systems. 2016, arXiv preprint arXiv:1603.04467.
[38] Paszke A, Gross S, Chintala S, Chanan G. PyTorch. 2017, https://pytorch.org.
[39] Al-Haik M, Hussaini M, Garmestani H. Prediction of nonlinear viscoelastic behavior of polymeric composites using an artificial neural network. Int J Plast 2006;22(7):1367–92. http://dx.doi.org/10.1016/j.ijplas.2006.01.001.
[40] Linka K, Hillgärtner M, Abdolazizi K, Aydin R, Itskov M, Cyron C. Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning. J Comput Phys 2021;429:110010. http://dx.doi.org/10.1016/j.jcp.2020.110010.
[41] Yamanaka Y, Matsubara S, Hirayama N, Moriguchi S, Terada K. Surrogate modeling for the homogenization of elastoplastic composites based on RBF interpolation. Comput Methods Appl Mech Engrg 2023;415:116282. http://dx.doi.org/10.1016/j.cma.2023.116282.
[42] Dettmer WG, Muttio EJ, Alhayki R, Perić D. A framework for neural network based constitutive modelling of inelastic materials. Comput Methods Appl Mech Engrg 2024;420:116672. http://dx.doi.org/10.1016/j.cma.2023.116672.
[43] Stöcker JP, Platen J, Kaliske M. Introduction of a recurrent neural network constitutive description within an implicit gradient enhanced damage framework. Comput Struct 2023;289:107162. http://dx.doi.org/10.1016/j.compstruc.2023.107162.
[44] Pang B, Long Z, Long T, Deng R, Su Z. Accelerating finite element simulation of metallic glasses using Data-Driven constitutive models trained by deep neural networks. Mater Today Commun 2025;45:112252. http://dx.doi.org/10.1016/j.mtcomm.2025.112252.
[45] Le BA, Yvonnet J, He Q-C. Computational homogenization of nonlinear elastic materials using neural networks. Internat J Numer Methods Engrg 2015;104(12):1061–84. http://dx.doi.org/10.1002/nme.4953.
[46] Liu Z, Wu CT, Koishi M. A deep material network for multiscale topology learning and accelerated nonlinear modeling of heterogeneous materials. Comput Methods Appl Mech Engrg 2019;345:1138–68. http://dx.doi.org/10.1016/j.cma.2018.09.020.
[47] Lu X, Giovanis DG, Yvonnet J, Papadopoulos V, Detrez F, Bai J. A data-driven computational homogenization method based on neural networks for the nonlinear anisotropic electrical response of graphene/polymer nanocomposites. Comput Mech 2019;64:307–21. http://dx.doi.org/10.1007/s00466-018-1643-0.
[48] Reimann D, Nidadavolu K, ul Hassan H, Vajragupta N, Glasmachers T, Junker P, Hartmaier A. Modeling macroscopic material behavior with machine learning algorithms trained by micromechanical simulations. Front Mater 2019;6:181. http://dx.doi.org/10.3389/fmats.2019.00181.
[49] Wu Z, Wang Z, Chen J, You H, Yan M, Wang L. Stratified random sampling for neural network test input selection. Inf Softw Technol 2024;165:107331. http://dx.doi.org/10.1016/j.infsof.2023.107331.
[50] Sket K, Potocnik D, Berus L, Hernavs J, Ficko M. Optimizing laser cutting of stainless steel using Latin hypercube sampling and neural networks. Opt Laser Technol 2025;182:112220. http://dx.doi.org/10.1016/j.optlastec.2024.112220.
[51] Farid Z, Assimeddine M, Abdennouri M, Barka N, Sadiq M. Flotation enhancement of sedimentary phosphate ores by cornstarch as an environmental depressant: modeling and analysis using full factorial design (ffd) and artificial neural network (ann) approaches. Environ Funct Mater 2023;2(3):243–54. http://dx.doi.org/10.1016/j.efmat.2024.03.001.
[52] Hendrycks D, Gimpel K. Gaussian error linear units (GELUs). 2016, arXiv preprint arXiv:1606.08415.
[53] Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, van der Walt SJ. Scipy 1.0: fundamental algorithms for scientific computing in python. Nature Methods 2020;17:261–72. http://dx.doi.org/10.1038/s41592-019-0686-2.
[54] Voglis C, Lagaris IE. A rectangular Trust-Region approach for unconstrained and box-constrained optimization problems. In: International conference of computational methods in sciences and engineering (ICCMSE 2004). 1st ed.. CRC Press; 2004, p. 4.
[55] API 5L: Specification for line pipe. 46th ed.. API Specification 5L; 2020.
[56] Menegotto M, Pinto PE. Method of analysis for cyclic loaded r.c. Plane frame including changes in geometry and non-elastic behaviour of elements under combined normal force and bending. In: Proceedings of IABSE symposium on resistance and ultimate deformability of structures acted on by well defined repeated loads. vol. 11, 1973, p. 15–22.