PHYSICS-INFORMED NEURAL NETWORKS FOR SYSTEM IDENTIFICATION OF STRUCTURAL SYSTEMS WITH A MULTIPHYSICS DAMPING MODEL

Tong Liu[1] and Hadi Meidani, Ph.D.[2,*]

[1]Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign. Email: tongl5@illinois.edu

[2]Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign. Email: meidani@illinois.edu

ABSTRACT

Structural system identification is critical in resilience assessments and structural health monitoring, especially following natural hazards. Among the nonlinear structural behaviors, structural damping is a complex behavior that can be modeled as a multiphysics system wherein the structure interacts with an external thermal bath and undergoes thermalization. In this paper, we propose a novel physics-informed neural network approach for nonlinear structural system identification and demonstrate its application in multiphysics cases where the damping term is governed by a separated dynamics equation. The proposed approach, called PIDynNet, improves the estimation of the parameters of nonlinear structural systems by integrating auxiliary physics-based loss terms, one for the structural dynamics and one for the thermal transfer. These physics-based loss terms form the overall loss function in addition to a supervised data-based loss term. To ensure effective learning during the identification process, sub-sampling and early stopping strategies are developed. The proposed framework also has the generalization capability to predict nonlinear responses for unseen ground excitations. Two numerical experiments of nonlinear systems are conducted to demonstrate the comparative performance of PIDynNet.

INTRODUCTION

Structural health monitoring plays a critical role in the life-cycle assessment and reliability analysis of civil infrastructure systems. Structural system identification (SSI) is one of the essential components of structural health monitoring in which the parameters of the mathematical model of the structural response are estimated. These parameters typically include stiffness, damping, mode shapes, frequencies, and the resulting identified model can predict the structural response given any excitation (Sirca Jr and Adeli 2012). SSI can also be used to identify damages in infrastructure systems, such as buildings (Lee and Park 2011; Abazarsa et al. 2013), bridges (Yang and Yang 2018; Eshkevari et al. 2020), and tunnels (Alonso-Rodriguez et al. 2018). Numerous SSI algorithms have been proposed over the past decades and can be categorized into time domain versus frequency domain methods, and parametric versus non-parametric approaches (for theoretical foundation and comprehensive surveys see (Sirca Jr and Adeli 2012)).

Structural damping modeling posed a challenge in structural dynamics. Specifically, the damping force is commonly expressed as a function of the damping coefficient and velocity. However, the damping coefficient also depends on complex mechanisms that are not yet fully quantifiable in the design stage (Adhikari 2013). From a mechanics perspective, damping is a consequence of energy dissipation when a structural system undergoes vibration. The process can also be considered as a thermalization process between a structural system and an external bath. To provide an energy-based characterization of this process, (Louhghalam et al. 2018) formulated the damping and energy dissipation in conjunction with the Nosé-Hoover thermostat (Nosé 1984; Hoover 1985), originally applied in the field of molecular dynamics. The structural system exchanges heat with the external bath until they ultimately reach the thermal equilibrium. Furthermore, the heat exchange rate between the structure and the external bath can be characterized as the damping coefficient, which evolves through time (Louhghalam et al. 2018). This effectively forms a handshake between classical structure damping and the statistical thermalization model.

Time-domain SSI methods include restoring forces surface (RFS), sparse identification of nonlinear dynamics (SINDy), and Kalman Filter-based approach. RFS is a simple and efficient

method that approximates the surface with a polynomial series or parametric expression for the nonlinear system (Villani et al. 2020). The drawback of the RFS method is that it requires more terms for non-smooth nonlinear cases (Ceravolo et al. 2013). The sparse identification approach (SINDy) is a non-parametric identification approach that approximates the dynamics equation by a sparse polynomial representation using the least absolute shrinkage and selection operator (LASSO) (Quade et al. 2018; Leylaz et al. 2021). SINDy is effective and efficient in recovering the dynamics, but it requires the synchronous data of displacement, velocity, and acceleration, which is a strict requirement for measurement (Lai and Nagarajaiah 2019a; Lai and Nagarajaiah 2019b). Kalman Filter has also been widely used in system identification and state estimation using incoming measurement data, state equations, and a noise model (Nguyen and Goulet 2018; Li and Wang 2020; Karimi et al. 2020).

The methods based on artificial neural networks have been recently demonstrated as promising tools in many complex problems (Gladstone et al. 2022; Liu and Meidani 2022). The data-driven approaches have demonstrated the ability to capture underlying nonlinear input-output relationships for complex systems. Multi-layer perceptrons (MLP) have been applied in predicting responses under static and dynamic loading (Eshkevari et al. 2021; Huang and Chen 2021). Furthermore, the recurrent neural network methods, including Gated recurrent units and Long Short-Term Memory, have been particularly successful in nonlinear sequence-to-sequence models and time series forecasting (Gonzalez and Yu 2018). Recent studies show that recurrent neural network has the potential for structural dynamic response modeling (Wang 2017; Zhang et al. 2019b).

In training of neural network models, insufficient training data can pose a significant challenge. One approach to compensate for data insufficiency is to integrate model-guided constraints. Specifically, the use of physics-based loss functions in the form of the residual of the governing differential equations has been proposed to enable the neural network to account for physical principles in the modeling process. This idea was first introduced to apply neural networks for initial value problems (Dissanayake and Phan-Thien 1994). The advances in computing hardware and optimization

algorithms have resulted in wide use of neural networks with physics-based loss function. The physics-informed neural network (PINN) in particular has been successfully applied to deterministic and random differential equations (Lagaris et al. 1998; Raissi et al. 2017; Nabian and Meidani 2018; Zhong and Meidani 2023). Nonetheless, the applicability of PINN to high dimensional and high temporal dependency problems remains an open question that requires further investigation.

Recent studies (Zhang et al. 2019a; Zhang et al. 2020; Eshkevari et al. 2021) show that neural networks with physics-based loss have the potential to be effectively used for structural modeling and model updating. These works didn't treat structural system identification and structural response prediction simultaneously under the same framework, and the generalization capability of the trained neural network for structural response prediction wasn't studied. To address the aforementioned issues, we develop PIDynNet, an ODE-constrained neural network for nonlinear structure system identification. Specifically, our approach integrates knowledge of the governing differential equations into the network training process, achieving comparable performance with fewer training data. Our methodology also enables accurate prediction of structural system response under unseen earthquake scenarios. For numerical demonstration, we apply PIDynNet to identify the parameters of thermalizing structural systems under earthquake excitations.

The paper is organized as follows. Section 2 introduces the basic idea of physics-informed training of neural networks and structural system identification with the Nosé-Hoover thermostat. Section 3 introduces the formulation of the identification problem and implementation of PIDynNet and techniques for fast and robust training. Section 4 studies the generalization capability of PIDynNet under unseen ground excitations. Section 5 presents two numerical experiments for nonlinear structural system identification imposing different types of nonlinearity with the Nosé-Hoover thermostat. Section 6 includes a summary and conclusions about the performance of the proposed method.

METHODOLOGY

This section includes a brief technical background of physics-informed deep learning and system identification with the Nosé-Hoover thermostat, followed by the introduction of the proposed

framework.

Physics-Informed Neural Network

In the context of PINN, the neural network is trained to approximate the solution of the ODE with given initial conditions. A generical ODE formulation can be given by

$$\begin{aligned} \boldsymbol{u}_t(\boldsymbol{x}, t) &= 0, \quad t \in [0, T] \subseteq \mathcal{T}, \boldsymbol{x} \in \Omega \subseteq \mathbb{R}^d \\ \boldsymbol{u}(\boldsymbol{x}, 0) &= \boldsymbol{h}(\boldsymbol{x}), \quad \boldsymbol{x} \in \Omega \subseteq \mathbb{R}^d, \end{aligned} \tag{1}$$

where $\boldsymbol{u}(\boldsymbol{x}, t)$ is the solution of the ODE with initial values $\boldsymbol{h}(\boldsymbol{x})$, Ω and $\mathcal{T}$ represents the computational domain and time domain, and $\boldsymbol{x} \in \mathbb{R}^d$ and $t \in \mathbb{R}$ are spatial and temporal coordinates of the system, respectively. This formulation can be easily generalized to higher-order ODEs since Eq. 1 can be written as the system of first-order ODEs. Besides, $\boldsymbol{u}(\boldsymbol{x}, t)$ and $\boldsymbol{h}(x)$ can consist of both linear and nonlinear terms. The solution $\boldsymbol{u}(\boldsymbol{x}, t)$ is then approximated by a neural network $\boldsymbol{u}(\boldsymbol{x}, t; \boldsymbol{\theta})$ with additional trainable parameters $\boldsymbol{\theta}$. Two residual-based loss terms are formed based on Eq. 1 indicating the discrepancy between the neural network prediction and the observation. These loss terms are given by

$$\begin{aligned} r_{\mathcal{N}}(\boldsymbol{\theta}) &= \int_{x \in \Omega} \int_{t \in \mathcal{T}} |\boldsymbol{u}_t(\boldsymbol{x}, t; \boldsymbol{\theta})|^2 \mathrm{d}t \mathrm{d}x, \\ r_{\mathcal{I}}(\boldsymbol{\theta}) &= |\boldsymbol{u}(\boldsymbol{x}, 0; \boldsymbol{\theta}) - \boldsymbol{h}(\boldsymbol{x})|^2, \end{aligned} \tag{2}$$

where $r_{\mathcal{N}}$ and $r_{\mathcal{I}}$ are the residuals of governing equations and initial conditions, respectively. In numerical calculation of Eq. 2, collocations points are sampled from the domain. The PINN ensures that the neural network solution satisfies the underlying governing equations throughout the domain by enforcing the governing physics equations at the collocation points. In that way, the Eq. 2 is transformed into a discrete form

$$\begin{aligned} r_{\mathcal{N}}(\boldsymbol{\theta}) &= \frac{1}{N_b} \sum_{i=1}^{N_b} |\boldsymbol{u}_t(\boldsymbol{x}^i, t^i; \boldsymbol{\theta})|^2, \\ r_{\mathcal{I}}(\boldsymbol{\theta}) &= |\boldsymbol{u}(\boldsymbol{x}^0, t^0; \boldsymbol{\theta}) - \boldsymbol{h}(\boldsymbol{x}^0)|^2, \end{aligned} \tag{3}$$

where N_b is the number of collocation points, $\{\boldsymbol{x}^i, t^i\}_{i=0}^{N_b}$ represents the collocation point of $\boldsymbol{u}(\boldsymbol{x}, t)$, which are in the interior part of the domain and on the boundary. The network parameters $\boldsymbol{\theta}$ can then be estimated by minimizing loss function as the weighted sum of Eq. 3, which is expressed as follows

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, w_{\mathcal{N}} r_{\mathcal{N}}(\boldsymbol{\theta}) + w_{\mathcal{I}} r_{\mathcal{I}}(\boldsymbol{\theta}), \tag{4}$$

where $w_{\mathcal{N}}$ and $w_{\mathcal{I}}$ denote the assigned weights for each loss term. These weights can be considered as hyperparameters and calculated during the training. This loss function consists of derivatives with respect to time which is an explicit input to the neural network approximation of the response. Therefore, these derivatives can be efficiently calculated using automatic differentiation (Raissi et al. 2017).

Structural System Identification with Nosé-Hoover thermostat

Structural system identification is an inverse problem with a forward model typically in the form of a multiple degrees of freedom (MDOF) linear dynamical systems with the following general form

$$\boldsymbol{M}\ddot{\boldsymbol{x}}(t) + \boldsymbol{K}\boldsymbol{x}(t) = 0, \tag{5}$$

where $\boldsymbol{M}, \boldsymbol{K} \in \mathbb{R}^{N\times N}$ denote the mass and inter-story stiffness matrices, respectively; $\boldsymbol{x}^T = [x_1, \ldots, x_n]^T$ and $\ddot{\boldsymbol{x}}^T = [\ddot{x}_1, \ldots, \ddot{x}_n]^T$ are displacement and acceleration vectors of the n-degree of freedom (DOF) system. For the sake of simplicity, let us consider a single degree of freedom (SDOF) system in the following discussion. From the Lagrangian mechanics point of view (Goncalves Salsa et al. 2018), the Euler-Lagrange equation of motion for an undamped SDOF system is given by

$$\begin{aligned} -\frac{\partial}{\partial t}\frac{\partial \mathcal{L}_s}{\partial \dot{x}} + \frac{\partial \mathcal{L}_s}{\partial x} &= 0, \\ \ddot{x}(t) + kx(t) &= 0, \end{aligned} \tag{6}$$

where $\mathcal{L}_s = \frac{1}{2}\dot{x}^2 - \frac{1}{2}kx^2$ is the Lagrangian function for the SDOF system with unit mass. The formulation in Eq. 6 is for an NVE–ensemble (Kraska 2006) where the mass is conserved (N is constant), the system size is fixed (V is constant), and the energy is conserved (E is constant). To consider damping in a structural system, whose motion is governed by Eq. 6, we consider the system to be in contact with an external bath at a prescribed temperature T_∞. The extended system with the external bath is an NVT-ensemble (Labík and Smith 1994). The Lagrangian of the structure-bath system, $\mathcal{L}_{sys}$, is then given by $\mathcal{L}_{sys} = \mathcal{L}_s + \mathcal{L}_b$. The Lagrangian of the thermal bath, $\mathcal{L}_b$, is the difference between the kinetic and potential energy of the bath (Louhghalam et al. 2018). Using the Nosé-Hoover thermostat, the Lagrangian of the bath thermalized at temperature T_∞, which is the difference between the bath's kinetic energy and potential energy, is given by

$$\mathcal{L}_b = \frac{1}{2}Qc^2 - RT_\infty \ln s, \tag{7}$$

where Q and c are the fictitious mass and velocity of the bath, and R is the product of the Boltzmann constant and the degree of freedom, s is a generalized coordinate that defines the stretch in time scale between the time of the bath and the structural time, i.e. $s = \frac{\mathrm{d}\tau}{\mathrm{d}t}$ (Louhghalam et al. 2018), and velocity, c, which measures the thermal transfer between the structure and the external bath, is given by $c = \frac{\mathrm{d}s}{\mathrm{d}\tau}$. We can then rewrite the Lagrangian of the structure-bath system as follows

$$\mathcal{L}_{sys} = \frac{1}{2}s^2(\frac{\mathrm{d}x}{\mathrm{d}\tau})^2 - \frac{1}{2}kx^2 + \frac{1}{2}Qc^2 - RT_\infty \ln s. \tag{8}$$

Using the structural and bath variables, x and s, the Euler-Lagrange equations of the system are obtained as

$$\begin{aligned} -\frac{\partial}{\partial\tau}\frac{\partial\mathcal{L}_{sys}}{\partial(\mathrm{d}x/\mathrm{d}\tau)} + \frac{\partial\mathcal{L}_{sys}}{\partial x} = 0, \\ -\frac{\partial}{\partial\tau}\frac{\partial\mathcal{L}_{sys}}{\partial(\mathrm{d}s/\mathrm{d}\tau)} + \frac{\partial\mathcal{L}_{sys}}{\partial s} = 0, \end{aligned} \tag{9}$$

based on which, the governing equations of the structure-bath system with the Nosé-Hoover ther-

mostat are given by

$$\begin{aligned} &\ddot{x} + c\dot{x} + kx = 0, \\ &\dot{c} = \Gamma(\frac{T(t)}{T_0} - \frac{T_\infty}{T_0}), \end{aligned} \tag{10}$$

where $\Gamma = RT_0/Q$ represents the bath-to-water ratio and $T(t)/T_0 = \dot{x}^2/RT_0$ represents the temperature evolution, and c appears as the damping coefficient in the structural dynamics equation. Eq. 10 indicates that the structure-bath system introduces nonlinearity into structural damping and that the structural damping itself evolves through time and has its own dynamics.

PIDYNNET FOR NONLINEAR STRUCTURAL SYSTEM IDENTIFICATION

We first formulate PIDynNet for dynamic systems with rate-dependent behavior, which is particularly challenging for structural system identification. This is because the rate-dependent variable is typically unobserved. A general form for the governing equations of a N-DOF dynamic system with rate-dependent behavior under external ground motion is given by

$$\begin{aligned} M\ddot{\boldsymbol{x}}(t) + f(\Phi(\boldsymbol{x}(t)), \Psi(\boldsymbol{z}(t)); \boldsymbol{\theta}_p) &= -M\vec{1}\ddot{x}_g(t), \\ \dot{\boldsymbol{z}}(t) &= g(\boldsymbol{x}(t), \boldsymbol{z}(t); \boldsymbol{\theta}_z), \\ x(0) = x_0, \;\; \dot{x}(0) = \dot{x}_0, \;\; z(0) &= z_0, \end{aligned} \tag{11}$$

where $M \in \mathbb{R}^{N\times N}$ is the mass matrix, $\boldsymbol{x} \in \mathbb{R}^N$ and $\boldsymbol{z} \in \mathbb{R}^N$ denote the observable displacement and rate-dependent behavior, $\Phi(\cdot) : \mathbb{R}^N \to \mathbb{R}^N$ and $\Psi(\cdot) : \mathbb{R}^N \to \mathbb{R}^N$ are nonlinear functions mapping taking displacement and rate-dependent behavior as inputs, respectively. Also, $f(\cdot,\cdot) : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^N$ is a nonlinear function mapping to restoring force where $\boldsymbol{\theta}_p \in \mathbb{R}^{N_p}$ denotes the properties parameterized $f(\cdot,\cdot)$; $g(\cdot) : \mathbb{R}^N \to \mathbb{R}^N$ is a nonlinear differential equation controlling the rate-dependent and hysteretic behavior where $\boldsymbol{\theta}_z \in \mathbb{R}^{N_z}$ denotes the system properties that parameterize $g(\cdot)$. The rate-dependent behavior $\boldsymbol{z}$ is generally a latent variable and cannot be measured externally. The task of the nonlinear system identification problem is to identify the

unknown parameter $\boldsymbol{\theta}_p$ and $\boldsymbol{\theta}_z$ given the explicit observations $\{\boldsymbol{x}, \dot{\boldsymbol{x}}, \ddot{\boldsymbol{x}}\}$, and external ground motion $\ddot{x}_g$.

The framework of the proposed PIDynNet is shown in Figure 1. It consists of a response prediction network (RPN) and a latent neural network (LNN). The RPN and LNN are built with fully connected networks with skip connections, trying to capture the underlying dynamics of function mappings Φ and Ψ, respectively. Without loss of generality, RPN and LNN are chosen to be fully connected networks. Other networks such as recurrent neural network (Zhang et al. 2020) or one-dimensional convolutional neural network (1D-CNN) (Kiranyaz et al. 2021) can also be used. The model parameters that need to be identified are constructed as trainable parameters in the neural networks, which could be optimized by minimizing the objective function.

The collocation points that serve as inputs for the RPN and LNN are obtained by sampling from given time interval and are represented as $\boldsymbol{I} = [t, \ddot{x}_g] \in \mathbb{R}^{2\times 1}$. The input consists of the sampled time and the corresponding ground acceleration. The input is resampled and fed into RPN and LNN for each iteration. The ground excitation corresponding to the sampled time is computed by linear interpolation from the available ground motion record. The task of RPN is to predict the observable responses by mapping the input vector into the output vector $\tilde{\boldsymbol{x}} \in \mathbb{R}^{N_o\times 1}$, e.g., $\tilde{\boldsymbol{x}} = \text{RPN}(\boldsymbol{I}; \boldsymbol{\theta}_R)$, where $\boldsymbol{\theta}_R \in \mathbb{R}^{N_R}$ and $N_o \in \mathbb{R}$ denotes the number of trainable parameters and cardinality of output in RPN. LNN is trained to predict the additional latent variables in rate-dependent behavior $\tilde{\boldsymbol{z}} \in \mathbb{R}^{N_o\times 1}$, where the predictions are $\tilde{\boldsymbol{z}} = \text{LNN}(\boldsymbol{I}; \boldsymbol{\theta}_L)$, with $\boldsymbol{\theta}_L \in \mathbb{R}^{N_L}$ being the trainable weights and biases in LNN. Furthermore, the outputs of RPN and LNN are passed into a graph-based automatic differentiator by taking the derivative with respect to time, denoted by $\boldsymbol{S} = \{\dot{\tilde{\boldsymbol{x}}}, \ddot{\tilde{\boldsymbol{x}}}, \dot{\tilde{\boldsymbol{z}}}\}$. The differentiation and the unknown structural parameters $\boldsymbol{\theta}_p$ and $\boldsymbol{\theta}_z$ are then used to calculate the physics-based loss and latent physics-based loss term $L_{\mathcal{N}}$ and $L_{\mathcal{Z}}$. The system identification is accomplished by learning the structural parameters ($\boldsymbol{\theta}_p$ and $\boldsymbol{\theta}_z$) and neural network parameters ($\boldsymbol{\theta}_R$ and $\boldsymbol{\theta}_L$) simultaneously by solving the following optimization problem

$$\boldsymbol{\theta}_R^*, \boldsymbol{\theta}_L^*, \boldsymbol{\theta}_p^*, \boldsymbol{\theta}_z^* = \underset{\boldsymbol{\theta}_R, \boldsymbol{\theta}_L, \boldsymbol{\theta}_p, \boldsymbol{\theta}_z}{\text{argmin}} \quad w_{\mathcal{N}} L_{\mathcal{N}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_p) + w_{\mathcal{S}} L_{\mathcal{S}}(\boldsymbol{\theta}_R) + w_{\mathcal{Z}} L_{\mathcal{Z}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_L, \boldsymbol{\theta}_z). \tag{12}$$

where $w_{\mathcal{N}}$ and $w_{\mathcal{S}}$ are weight parameters for the physics-based loss $L_{\mathcal{N}}$ and supervised loss $L_{\mathcal{S}}$, respectively. $w_{\mathcal{Z}}$ is the weight parameters for the latent physics-based loss $L_{\mathcal{L}}$ for rate-dependent dynamics. For instance, the physics-based loss term $L_{\mathcal{N}}$ for the SDOF system governed by Eq. 9, where $\boldsymbol{\theta}_p = [m, c, k]$, is given by

$$L_{\mathcal{N}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_p) = \mathbb{E}_{\boldsymbol{I}\sim p(\boldsymbol{I})}\left[L_{\mathcal{N}}\left(\boldsymbol{\theta}_R, \boldsymbol{\theta}_p | \boldsymbol{I}\right)\right] \approx \sum_{i=1}^{N_b} \left\| m\ddot{\tilde{\boldsymbol{x}}}^i(\boldsymbol{\theta}_R) + c\dot{\tilde{\boldsymbol{x}}}^i(\boldsymbol{\theta}_R) + k\tilde{\boldsymbol{x}}^i(\boldsymbol{\theta}_R) + m\ddot{\boldsymbol{x}}_g^i \right\|_2^2 . \quad (13)$$

where N_b is the batch size. This physics-based loss encourages the identified structural parameters and the predicted response to conform simultaneously to the governing equations. The second term $L_{\mathcal{S}}$ in the loss function, which is the data-driven "supervised" loss function, is given by

$$L_{\mathcal{S}}(\boldsymbol{\theta}_R) = \mathbb{E}_{\boldsymbol{I}\sim p(\boldsymbol{I})}\left[L_{\mathcal{S}}\left(\boldsymbol{\theta}_R | \boldsymbol{x}, \dot{\boldsymbol{x}}, \boldsymbol{I}\right)\right] \approx \sum_{i=1}^{N_b} \left\| \tilde{\boldsymbol{x}}^i(\boldsymbol{\theta}_R) - \boldsymbol{x}^i \right\|_2^2 + \sum_{i=1}^{N_b} \left\| \dot{\tilde{\boldsymbol{x}}}^i(\boldsymbol{\theta}_R) - \dot{\boldsymbol{x}}^i \right\|_2^2 . \quad (14)$$

This term measures the difference between the neural network prediction and the ground truth signal, and is a function of only the RPN parameters $\boldsymbol{\theta}_R$. The third term $L_{\mathcal{Z}}$ is given by

$$L_{\mathcal{Z}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_L, \boldsymbol{\theta}_z) = \sum_{i=1}^{N_b} \left\| \dot{\tilde{\boldsymbol{z}}}^i(\boldsymbol{\theta}_L) - g\left(\tilde{\boldsymbol{x}}^i(\boldsymbol{\theta}_R), \tilde{\boldsymbol{z}}^i(\boldsymbol{\theta}_L); \boldsymbol{\theta}_z\right) \right\|_2^2 , \quad (15)$$

where $\dot{\tilde{\boldsymbol{z}}}^i$ and other derivatives in $g(\cdot)$, are calculated using automatic differentiation of LNN.

The PIDynNet method is shown in Algorithm 1. The network makes predictions on $\tilde{\boldsymbol{x}}$ and latent variable $\tilde{\boldsymbol{z}}$ separately, then aggregates all the variables in the loss function with unknown parameters. Instead of using data from the whole response time period, PIDynNet seeks to minimize the loss function by sequentially training the network and structural parameters over consecutive subperiods. Specifically, first, the subperiods are created by equally dividing the whole response time period. Then, using data from each subperiod, PIDynNet is trained sequentially where the structural parameters are calculated and serve as the initial values in the next subperiod. Within each subperiod, there are variations in the calculated parameters over the iterations of the algorithm. As

more subperiods are used, one expects these variations to reduce. Therefore, after each subperiod, we require the next structural parameters to be constrained in the range $\{\boldsymbol{p}^* - 3\boldsymbol{\sigma_p}, \boldsymbol{p}^* + 3\boldsymbol{\sigma_p}\}$, where $\boldsymbol{p}^*$ is the latest estimate for the parameters, and $\boldsymbol{\sigma_p}$ is the standard deviation of the parameters that are estimated through iterations in the current subperiod (calculated using the second half of the iterations.). The rationale behind the inclusion of $3\boldsymbol{\sigma_p}$ is to provide an approximation of the range of variation for the state variables, which is adaptively updated during the identification process. This approach is similar to other adaptive updating methods (Li and Wang 2020; Song et al. 2020) and the range of $3\boldsymbol{\sigma_p}$ was determined from a parametric study that considered both the identification accuracy and computational time.

Algorithm 1: PIDynNet algorithm for structural system identification

Input : $\boldsymbol{x}, \dot{\boldsymbol{x}}, \ddot{\boldsymbol{x}}, \ddot{\boldsymbol{x}}_g, \boldsymbol{t}$;
Separate the time range into $\{t_0, t_1, \ldots, t_n\}$;
Initialize parameter $\boldsymbol{\theta}_R, \boldsymbol{\theta}_p$;
for $i \leftarrow 0$ **to** $n-1$ **do**
  Generate the sampling pool $\boldsymbol{T}_i \subset \{t | t_i \le t \le t_{i+1}\}$;
  Update uncertainty constrain of $\boldsymbol{\theta}_p$ and $\boldsymbol{\theta}_z$;
  $n \leftarrow 0$;
  while $n \le MaxIter$ **do**
    Generate normalized sample $\tilde{\boldsymbol{T}} \subset \boldsymbol{T}_i$;
    Compute the ground motion $\ddot{\tilde{\boldsymbol{x}}}_g$ corresponding to $\tilde{\boldsymbol{T}}$;
    $\boldsymbol{I} \leftarrow \{\tilde{\boldsymbol{T}}, \ddot{\tilde{\boldsymbol{x}}}_g\}$;
    $\tilde{\boldsymbol{x}}, \dot{\tilde{\boldsymbol{x}}} \leftarrow \mathrm{RPN}(\boldsymbol{I}; \boldsymbol{\theta}_R)$;
    $\tilde{\boldsymbol{z}}, \dot{\tilde{\boldsymbol{z}}} \leftarrow \mathrm{LNN}(\boldsymbol{I}; \boldsymbol{\theta}_L)$;
    $L \leftarrow w_{\mathcal{N}} L_{\mathcal{N}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_p) + w_{\mathcal{S}} L_{\mathcal{S}}(\boldsymbol{\theta}_R) + w_{\mathcal{Z}} L_{\mathcal{Z}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_L, \boldsymbol{\theta}_z)$;
    $\boldsymbol{\theta}_R \leftarrow \boldsymbol{\theta}_R - \alpha \Delta L_{\boldsymbol{\theta}_R}$;
    $\boldsymbol{\theta}_L \leftarrow \boldsymbol{\theta}_L - \alpha \Delta L_{\boldsymbol{\theta}_L}$;
    $\boldsymbol{\theta}_p \leftarrow \boldsymbol{\theta}_p - \alpha \Delta L_{\boldsymbol{\theta}_p}$;
    $\boldsymbol{\theta}_z \leftarrow \boldsymbol{\theta}_z - \alpha \Delta L_{\boldsymbol{\theta}_z}$;
    if $\boldsymbol{\theta}_p$ *and* $\boldsymbol{\theta}_z$ *meets the stopping criteria* **then**
      Go to the next time range;
      $i \leftarrow i + 1$;
      break;
    else
      Continue;
  Update $\boldsymbol{\theta}_p$ and $\boldsymbol{\theta}_z$ from the identification history;
Output: $\boldsymbol{\theta}_R, \boldsymbol{\theta}_p$

Furthermore, we implement a sub-sampling method to extract samples from the sampling pool to control the size of collocation points. The sub-sampling method utilizes the equivalent amount of data compared with other identification methods, for instance, the Kalman filter. It circumvents the situation where the neural network overuses the training data. To maintain the value of time on a regular scale, the time vector will be normalized after subsampling. The parameters of the neural network and unknown structural parameters are optimized by stochastic gradient descent. During the iteration in the training process, the value of identified parameters is checked every N_c iteration, and an early stopping strategy is introduced if the unknown parameters $\boldsymbol{\theta}_p$ do not change beyond the threshold ϵ for n successive evaluation, i.e.,

$$\left\|\boldsymbol{\theta}_p^{N_0+iN_c} - \boldsymbol{\theta}_p^{N_0}\right\| / \left\|\boldsymbol{\theta}_p^{N_0}\right\| < \epsilon \quad \forall i = 1, \ldots, n. \tag{16}$$

GENERALIZATION CAPABILITY OF PIDYNNET

After identifying the unknown structural model parameter using the response time history to a given ground excitation, the neural network may have the generalization capability to predict the response time history under an unseen ground excitation. Instead of retraining the PIDynNet from scratch for a new ground motion, we just re-train the RPN and LNN components for a few 'fine-tuning' steps. PIDynNet is expected to make an accurate prediction after fine-tuning, during which the structural parameters, $\boldsymbol{\theta}_p$, and $\boldsymbol{\theta}_z$, are kept fixed, and only the neural network parameters, θ_R, and θ_L, are retrained. The fine-tuning only uses physics-based loss functions with the input ground excitations, which is given by

$$\boldsymbol{\theta}_R^{**}, \boldsymbol{\theta}_L^{**} = \underset{\boldsymbol{\theta}_R, \boldsymbol{\theta}_L}{\operatorname{argmin}} \; L_{\text{fine-tuning}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_L, \boldsymbol{\theta}_p^*, \boldsymbol{\theta}_z^*) = \underset{\boldsymbol{\theta}_R, \boldsymbol{\theta}_L}{\operatorname{argmin}} \; w_{\mathcal{N}} L_{\mathcal{N}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_p^*) + w_{\mathcal{Z}} L_{\mathcal{Z}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_L, \boldsymbol{\theta}_z^*), \tag{17}$$

where the weight assignment is the same as Eq. 12. The number of iterations in the fine-tuning phase is substantially smaller. In our experiments, this number is more than one order of magnitude less than the training iterations. The fine-tuning process of PIDynNet is shown in Algorithm 2.

Algorithm 2: PIDynNet algorithm for use in generalization cases

Input : $\ddot{\boldsymbol{x}}_g, \boldsymbol{t}, \boldsymbol{\theta}_R, \boldsymbol{\theta}_L, \boldsymbol{\theta}_p^*, \boldsymbol{\theta}_z^*$;

Separate the time range into $\{t_0, t_1, \ldots, t_n\}$;
for $i \leftarrow 0$ **to** $n-1$ **do**
  Generate the sampling pool $\boldsymbol{T}_i \subset \{t | t_i \leq t \leq t_{i+1}\}$;
  $n \leftarrow 0$;
  while $n \leq MaxIter$ **do**
    Generate normalized sample $\tilde{\boldsymbol{T}} \subset \boldsymbol{T}_i$;
    Compute the ground motion $\ddot{\tilde{\boldsymbol{x}}}_g$ corresponding to $\tilde{\boldsymbol{T}}$;
    $\boldsymbol{I} \leftarrow \{\tilde{\boldsymbol{T}}, \ddot{\tilde{\boldsymbol{x}}}_g\}$;
    $\tilde{\Phi}(\boldsymbol{x}) \leftarrow \mathrm{RPN}(\boldsymbol{I}; \boldsymbol{\theta}_R)$;
    $\tilde{\Psi}(\boldsymbol{z}) \leftarrow \mathrm{LNN}(\boldsymbol{I}; \boldsymbol{\theta}_L)$;
    $L \leftarrow L_{\text{fine-tuning}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_p^*)$ or $L_{\text{fine-tuning}}(\boldsymbol{\theta}_R, \boldsymbol{\theta}_L, \boldsymbol{\theta}_p^*, \boldsymbol{\theta}_z^*)$;
    $\boldsymbol{\theta}_R \leftarrow \boldsymbol{\theta}_R - \alpha \Delta L_{\boldsymbol{\theta}_R}$;
    $\boldsymbol{\theta}_L \leftarrow \boldsymbol{\theta}_L - \alpha \Delta L_{\boldsymbol{\theta}_L}$;

Output : $\boldsymbol{\theta}_R, \boldsymbol{\theta}_p$

NUMERICAL RESULTS

In this section, we consider problems with nonlinear damping and inelasticity subject to ground excitation. In particular, two nonlinear systems are considered: the cubic stiffness system and the Bouc-Wen hysteretic system. All experiments are performed using an Intel E5-2620 CPU and 2 NVIDIA Tesla A100 GPUs. The neural network is implemented and trained using the PyTorch (Paszke et al. 2019).

Cubic Stiffness System

The cubic stiffness system, commonly referred to as the Duffing system, has been successfully used to model various physical processes such as stiffening springs, and beam buckling. In this case study, we consider the Duffing oscillator where a cubic spring is located between the first mass and the ground. In particular, the Duffing oscillator can be interpreted as a forced oscillator with nonlinear (3^{rd} order) elastic stiffeners whose governing equation for story i in the MDOF with DOF equal to N is written as:

$$
\begin{aligned}
&m_i\ddot{x}_i + c_i\dot{x}_i - c_{i+1}\dot{x}_{i+1} + k_i x_i - k_{i+1}x_{i+1} + k_{nl}x_i^3 = -m_i\ddot{x}_g, && i = 1\\
&m_i\ddot{x}_i - c_i\dot{x}_{i-1} + (c_i + c_{i+1})\dot{x}_i - c_{i+1}\dot{x}_{i+1} - k_i x_{i-1} + (k_i + k_{i+1})x_i - k_{i+1}x_{i+1} = -m_i\ddot{x}_g, && i = 2, \cdots, N-1\\
&m_i\ddot{x}_i - c_i\dot{x}_{i-1} + (c_{i-1} + c_i)\dot{x}_i + k_i x_{i-1} - (k_{i-1} + k_i)x_i = -m_i\ddot{x}_g, && i = N\\
&\dot{c}_i = \tfrac{\Gamma}{RT_0}\dot{x}_i^2, && i = 1, \cdots, N
\end{aligned}
\tag{18}
$$

where $m_i, c_i, k_i \in \mathbb{R}$ denote the mass, damping, and linear inter-story stiffness of story i, respectively; $k_{nl} \in \mathbb{R}$ represent the cubic stiffness. The first three equations govern the dynamic response of the MDOF system and the last equation governs the damping evolution within the Nosé-Hoover thermostat, where the temperature ratio T_∞/T_0 between the bath and the structure is assumed to be zero. This represents the evolution of the damping ratio in time, due to the evolution of the kinetic energy of the structures compared to its initial kinetic energy. If the cubic stiffness is positive, stiffness hardening effects are present. On the contrary, stiffness-softening effects are present if the cubic stiffness is negative. In our example, a 10-DOF system is considered and the structural parameters are set as $m_i = 1\text{kg}$ for $i = 1, \cdots, 10$, $c_i = 1.5\text{N}\cdot\text{s/m}$ for $i = 1, \cdots, 4$, $c_i/c_1 = 1.25$ for $i = 5, \cdots, 7$, $c_i/c_1 = 3/2$ for $i = 8, \cdots, 10$, $k_i = 35\text{N/m}$ for $i = 1, \cdots, 3$, $k_i/k_1 = 7/6$ for $i = 4, \cdots, 6$, $k_i/k_1 = 3/2$ for $i = 7, \cdots, 10$, and $k_{nl} = 150\text{N/m}^3$. The structure–to–bath mass ratio is captured by the kinetic coefficient, set to be $\Gamma/RT_0 = 10$. The first-mode period of the system is 1.19 s when the effect of the nonlinear term is negligible. The Kobe Earthquake ground motion from the NGA-West2 ground motion database is used as the excitation. A scaled version of the 40-second ground motion record is adopted to let the response spectrum match with the target spectrum within the range of 0.2 T_1 to 1.5 T_1, where T_1 is the first-mode structural period. The training data, which includes displacement, velocity, and acceleration time histories, are obtained from the third-order Runge-Kutta Method with a sampling frequency of 1000Hz. The signal-to-noise ratio of the response time history is 7.28dB. The original data is reshaped to have input sizes of [40001, 10] in order to be compatible with the data format for PIDynNet. The architecture of RPN and LNN is a five-layer MLP with an embedding size of 64. The input and output dimensions

of RPN and LNN are 2 and 10, respectively. The batch size N_b is chosen as 500 for the case study, which is determined from the empirical parametric study to consider both the identification accuracy and computational time. The weights for supervised, physics-based, and latent physics-based losses are chosen as 1, 1, and 10, respectively. The weight for latent physics-based loss is higher because there is no observed data for latent state variables, which forces the LNN to identify the latent state variables.

Due to the unobservability of the latent variable, one can expect the training of LNN and the identification to be slow. Thus, we adopt a transfer learning-based initialization, where only the supervised loss is used and the RPN is pre-trained. After pretraining the RPN, we transfer the parameters from RPN into the LNN. The intuition behind this transfer learning-based initialization is that if random initialization is implemented in LNN, latent variable prediction is also random at the beginning of identification. In this case, the range of derivatives of latent variables is uncertain or unbounded. Thus, the latent physics-based loss decreases very slowly, which decelerates the identification process. Even though the LNN still needs to be trained after adopting the transfer learning-based initialization, the differentiation is now controlled in a bounded range, which accelerates the identification process.

In the pretraining process and identification process, adaptive momentum estimation (Adam) (Kingma and Ba 2014) is selected as the optimizer with a learning rate of 0.001 and a decay rate of 0.1. The number of epochs for the pretraining process and identification process are 1,000 and 10,000, respectively. The sampling pool size in each subperiod is 1,000. The parameters are evaluated for $N_{es} = 200$ iterations, and the default value of $n_{es} = 3$ is chosen in the early stopping strategy. The early stopping threshold is $\epsilon = 10^{-4}$. All subsequent case studies use the same hyperparameters as the cubic stiffness case.

The trainable parameters of RPN and LNN are randomly initialized, which introduces uncertainty into the identification process. Furthermore, the sub-sampling also introduces uncertainty since different identification processes might use different sampling pools. To quantify the uncertainty introduced by the aforementioned issues, the PIDynNet algorithm is run 10 times with

different initializations, and the mean value and standard deviation over 10 evaluations are calculated for the identification results of each unknown parameter. The identification results from the PIDynNet are compared with the ground truth and also with results from the unscented Kalman filter (UKF) in Figure 2 and Table 1. Table 1 compares the identified modal parameters obtained from PIDyNet and UKF, showing that PIDynNet offers relatively better prediction. Specifically, the average estimation error by PIDynNet is 0.94% while the average error by UKF is 2.94%. The largest average identification error of PIDynNet is less than 2%. Except for the cubic stiffness term, the largest standard deviation of error by PIDynNet is less than 1.5%, indicating the variation introduced by randomized initialization and sub-sampling is not significant.

Figure 2 shows the identification history of PIDynNet for each subperiod compared with UKF. Both the identification history of each randomized initialization (gray curve) and the average of all identification histories are reported. Figures in the left column indicate that the convergence of PIDynNet is faster than that of UKF for a number of parameters. Also, the figures on the right show the advantage of PIDynNet in terms of relative errors for different parameters. It can be seen that both approaches were not able to identify the parameters in the first several seconds since the structural response is small and has not exhibited nonlinear behaviors. When the system starts to exhibit nonlinear behavior, the structural parameters in the identification block of PIDynNet begin to converge to the ground truth values that minimize the physics-informed loss.

Figure 3 compares the hysteretic curve under noise-free conditions. The hysteretic curve of PIDynNet is calculated based on the identified response history and structural parameters. The gray curves indicate the hysteretic loop of identification results with different initializations, whereas the red curve is calculated using the mean value of multiple identification results. In order to measure the goodness-of-fit and degree of similarity of the hysteretic loop between the PIDynNet and the ground truth, we use the Pearson product-moment correlation coefficient γ, which is given by

$$\gamma = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}, \tag{19}$$

where $\{x_i\}_{i=1}^{T}$ and $\{y_i\}_{i=1}^{T}$ are time series that need to be measured. The correlation coefficient

between PIDynNet prediction and the ground truth is 0.972. This indicates that the PIDynNet can both captures the governing dynamics and unknown parameters, as well as response predictions.

The sensitivity analysis of the subperiod is conducted to accelerate the training while maintaining the same level of accuracy. If the subperiod is set to be too short, more subperiods and more computational time are required. On the other side, if the subperiod is too long, the training needs more iterations to converge. The proper subperiod length should be chosen to reduce the computational time. To conduct the sensitivity analysis, multiple experiments are performed with different subperiods. In particular, the chosen subperiod sizes include 2s, 3s, 5s, 10s. The PIDynNets with different subperiod are used on the same response history to identify the unknown parameter and latent state variable. The identification results of PIDynNet with different subperiod are shown in Table 2. As the table illustrates, the identification error increases when the subperiod range is too short and too long. The average identification error is minimized when the subperiod is 5s. As a result, we chose the subperiod of size 5s in this and the subsequent case study.

In order to investigate the robustness of PIDynNet, multiple experiments with different noise levels are conducted. Zero-mean Gaussian noises are added to the original signal to investigate the robustness. In the case study, the standard deviations of added Gaussian noise are 2.5% and 5% of the standard deviation of the original signal. The identification results under different noise levels are shown in Tables 3. With the noise level at 5%, the average error of PIDynNet is only 1.42% and the average error of UKF is 3.07%, which indicates that PIDynNet can still produce relatively accurate results at higher noise levels.

From the perspective of computational time, PIDynNet is tested with cubic stiffness systems of different DOF compared with UKF. The DOF is chosen as 2, 4, 8, 10, 16, 20, and 32. For each case, a separated model is trained since the output dimension of RPN and LNN is changed. The average error of all the parameters is reported in Figure 4. It can be seen that when the system size is small, the computational time of UKF is less than PIDynNet. However, when the DOF increases to 10 or higher, PIDynNet is more computationally efficient than UKF. When the degree of freedom is 10, the computational time for PIDynNet is 432s while UKF takes 892s. The average

error of PIDynNet is relatively smaller than UKF, which indicates that PIDynNet is more accurate and computationally effective than UKF on large systems.

As discussed in Section 4, a trained PIDynNet can also be used for structural response prediction, and here we seek to assess the generalization capability of the response prediction part of PIDynNet. To achieve this objective, the first step involves gathering ground excitation records that have not been previously used in the identification process. These excitations are generated by selecting ground motions from NGA-West2 that match a target response spectrum distribution (Baker and Lee 2018). After the ground motion selection, the input with unseen ground motions is fed into trained PIDynNet with fixed identified structural parameters according to Algorithm 2. The response time history of unseen ground excitation is calculated through fine-tuning, which is shown in Figure 5. The similarity of predicted responses between the reference and the PIDynNet predictions is measured with Pearson correlation coefficient according to Eq. 19. The histogram in Figure 6 presents the distribution of the correlation coefficient under different ground motions. We observed that about 90% of the prediction responses have a correlation coefficient $\gamma \geq 0.95$. We can conclude that for the unseen ground motions, the captured underlying dynamics by PIDynNet can still help predict accurate nonlinear responses without any training data from unseen records.

Bouc-Wen Hysteretic System

The Bouc-Wen model is one of the widely used hysteretic models originally proposed by Bouc (Bouc 1967) and later developed by Wen (Wen 1976), which involves considering implicit, latent hysteretic displacement to calculate the restoring force. The latent hysteretic displacement in the restoring force has dynamics which is governed by a nonlinear differential equation that depends on the structure's velocity. The hysteretic dynamics equation of the Bouc-Wen model for a N-DOF system can be expressed as:

$$
\begin{aligned}
&m_i\ddot{x}_i + c_i\dot{x}_i - c_{i+1}\dot{x}_{i+1} + k_i z_i - k_{i+1}z_{i+1} = -m_i\ddot{x}_g, && i = 1\\
&m_i\ddot{x}_i - c_i\dot{x}_{i-1} + (c_i + c_{i+1})\dot{x}_i - c_{i+1}\dot{x}_{i+1} - k_i z_{i-1} + (k_i + k_{i+1})z_i - k_{i+1}z_{i+1} = -m_i\ddot{x}_g, && i = 2, \cdots, N-1\\
&m_i\ddot{x}_i - c_i\dot{x}_{i-1} + (c_{i-1} + c_i)\dot{x}_i + k_i z_{i-1} - (k_{i-1} + k_i)z_i = -m_i\ddot{x}_g, && i = N\\
&\dot{c}_i = \tfrac{\Gamma}{RT_0}\dot{x}_i^2, && i = 1, \cdots, N\\
&\dot{z}_i = \dot{x}_i - \beta\,|\dot{x}_i|\,|z_i|^{n-1} z_i - \gamma\dot{x}_i|z_i|^n, && i = 1\\
&\dot{z}_i = (\dot{x}_i - \dot{x}_{i-1}) - \beta\,|\dot{x}_i - \dot{x}_{i-1}|\,|z_i|^{n-1} z_i - \gamma\,(\dot{x}_i - \dot{x}_{i-1})\,|z_i|^n, && i = 2, \cdots, N
\end{aligned}
\tag{20}
$$

where N is the number of stories and m_i, c_i, k_i are the lumped mass, dampings and inter-story stiffness at story i, respectively; x_i and z_i represent the observed displacement and the latent hysteretic displacement, respectively; β, γ, and n are dimensionless parameters that control the hysteretic behavior and can generate a large variety of hysteretic loops.

PIDynNet is applied on the 3-DOF Bouc-Wen system. the normalized masses are set as $m_1 = m_2 = m_3 = 1\text{kg}$, and the structural parameters are $c_1 = 1.5\text{N}\cdot\text{s/m}$, $c_2/c_1 = 1.0$, $c_3/c_1 = 0.8$, $k_1 = 50\text{N/m}$, $k_2/k_1 = 0.9$ and $k_3/k_1 = 0.9$, and the system parameters are set as $\beta = 2.0$, $\gamma = 2.0$ and $n = 1.0$. The damping history and latent hysteretic displacements are unobserved, and thus are predicted through LNN. The initial structural displacements and initial hysteretic displacements in each subperiod are computed from its previous subperiod. The hyperparameters in the identification process are identical to the previous case study. Table 4 and Figure 7 compare the accuracy using PIDynNet and UKF. Similar to the same setting in cubic stiffness model, 10 experiments with different randomized initialized parameters are conducted to quantify the uncertainty of randomness introduced by random initialization and sub-sampling. The average identification error by PIDynNet is 1.58 % and the average error by UKF is 4.53 %. Additionally, the largest estimation error by PIDynNet is 2.98% while the largest error by UKF is 12.86%, underscoring the overall better performance of PIDynNet.

The hysteretic displacement z of the Bouc-Wen model could be predicted from the LNN during the identification stage. Figure 8 demonstrates the identification results of hidden displacement

by PIDynNet. It can be seen that even though no training data and observation are available for the latent variable, the neural network can still successfully identify and capture its dynamics. The average correlation coefficient of hysteretic displacement between PIDynNet identification and the ground truth of all three stories is 0.992. Additionally, the hysteretic diagram could be plotted with the identified displacement and structural parameters. Figure 9 plots the hysteretic diagram of identified Bouc-Wen model on the first floor. The correlation coefficient between ground truth and PIDynNet identification result is 0.962. In order to assess the robustness of the model, similarly to the previous case study, multiple experiments are conducted under different measurement noise levels. The identification results under 2.5% and 5% noise are shown in Table 5. The PIDynNet model has yielded an average identification error of less than 1.5% at both levels of noise. In contrast, the UKF model has exhibited a prediction error greater than 50% as compared to PIDynNet. It indicates that PIDynNet can offer relatively accurate identification even for noisy data, with an average identification error lower than that of UKF.

Furthermore, to validate the generalization capability of the neural network unseen ground motions from NGA-West2 is used as the excitation. Figure 10 and 11 show the prediction of hysteric displacement time history and hysteretic loop under unseen ground motions. As can be seen, PIDynNet can produce an accurate prediction of the displacement time history and the hysteretic behavior. The correlation coefficients between the prediction and reference response are $\gamma = 0.961$. This shows that the response prediction part of PIDynNet can be generalized as a forecast model to predict the structural response to unseen earthquakes.

DISCUSSION AND CONCLUSION

In this paper, we proposed PIDynNet, a novel ODE-constrained neural network structural identification framework, and showed how it can be applied to multiphysics problems. One of the key strengths of PIDynNet is its ability to handle complex and nonlinear structural dynamics problems. In particular, we considered structural identification problems where the structural damping was modeled as a thermalization process between the structural system and the external environmental bath. PIDynNet utilizes latent neural networks that capture rate-dependent state

variables or unobserved variables and uses physics-based losses, which are derived from the governing equation. Along with the supervised loss, the physics-based loss is embedded in the overall loss function, which is considered an auxiliary constraint to enforce the architecture to capture the governing dynamics. In the training phase, a sub-sampling strategy and an early stopping criterion are used to accelerate learning and identification. Using numerical examples, we showed that PIDynNet is effective in identifying the parameters of nonlinear systems and outperforms the state-of-the-art identification method. Two numerical case studies, including the cubic stiffness model and the Bouc-Wen model, demonstrate the effectiveness and efficiency of PIDynNet identification performance. In addition to identification, we showed that PIDynNet has generalization capability when used as a structural response prediction for unseen earthquakes. We also demonstrated the robustness of PIDynNet by studying cases where observations were noisy. While the proposed model has been tested only on numerical experiments so far, it has the potential to be extended to real structures. The ability to accurately identify the parameters of real structures is important for structural health monitoring, which is critical for ensuring the safety of civil infrastructure. Additionally, the extension of the proposed framework to include non-uniform heat transfer would provide insights into the applicability of more realistic situations.

Data Availability Statement

Some or all data, models, or code that support the findings of this study are available from the corresponding author upon reasonable request (training dataset, code).

Acknowledgments

This material is based in part upon work supported by the National Science Foundation under Grant No. CMMI-1752302 and USDOT under Grant No. 69A3551747105.

REFERENCES

Abazarsa, F., Ghahari, S., Nateghi, F., and Taciroglu, E. (2013). "Response-only modal identification of structures using limited sensors." *Structural Control and Health Monitoring*, 20(6), 987–1006.

Adhikari, S. (2013). *Structural dynamic analysis with generalized damping models: analysis*. John Wiley & Sons.

Alonso-Rodriguez, A., Nikitas, N., Knappett, J., Kampas, G., Anastasopoulos, I., and Fuentes, R. (2018). "System identification of tunnel response to ground motion considering a simplified model." *Frontiers in Built Environment*, 4, 39.

Baker, J. W. and Lee, C. (2018). "An improved algorithm for selecting ground motions to match a conditional spectrum." *Journal of Earthquake Engineering*, 22(4), 708–723.

Bouc, R. (1967). "Forced vibrations of mechanical systems with hysteresis." *Proc. of the Fourth Conference on Nonlinear Oscillations, Prague, 1967*.

Ceravolo, R., Erlicher, S., and Fragonara, L. Z. (2013). "Comparison of restoring force models for the identification of structures with hysteresis and degradation." *Journal of Sound and Vibration*, 332(26), 6982–6999.

Dissanayake, M. and Phan-Thien, N. (1994). "Neural-network-based approximations for solving partial differential equations." *communications in Numerical Methods in Engineering*, 10(3), 195–201.

Eshkevari, S. S., Matarazzo, T. J., and Pakzad, S. N. (2020). "Bridge modal identification using acceleration measurements within moving vehicles." *Mechanical Systems and Signal Processing*, 141, 106733.

Eshkevari, S. S., Takáč, M., Pakzad, S. N., and Jahani, M. (2021). "Dynnet: Physics-based neural architecture design for nonlinear structural response modeling and prediction." *Engineering Structures*, 229, 111582.

Gladstone, R. J., Nabian, M. A., and Meidani, H. (2022). "Fo-pinns: A first-order formulation for physics informed neural networks." *arXiv preprint arXiv:2210.14320*.

Goncalves Salsa, R., Kawano, D. T., Ma, F., and Leitmann, G. (2018). "The inverse problem of linear lagrangian dynamics." *Journal of Applied Mechanics*, 85(3).

Gonzalez, J. and Yu, W. (2018). "Non-linear system modeling using lstm neural networks." *IFAC-PapersOnLine*, 51(13), 485–489.

Hoover, W. G. (1985). "Canonical dynamics: Equilibrium phase-space distributions." *Physical review A*, 31(3), 1695.

Huang, P. and Chen, Z. (2021). "Deep learning for nonlinear seismic responses prediction of subway station." *Engineering Structures*, 244, 112735.

Karimi, P., Butala, M., and Kamalabadi, F. (2020). "Efficient model selection in switching linear dynamic systems." *arXiv preprint arXiv:2012.04543*.

Kingma, D. P. and Ba, J. (2014). "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980*.

Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., and Inman, D. J. (2021). "1d convolutional neural networks and applications: A survey." *Mechanical systems and signal processing*, 151, 107398.

Kraska, T. (2006). "Molecular-dynamics simulation of argon nucleation from supersaturated vapor in the nve ensemble." *The Journal of chemical physics*, 124(5), 054507.

Labík, S. and Smith, W. (1994). "Scaled particle theory and the efficient calculation of the chemical potential of hard spheres in the nvt ensemble." *Molecular Simulation*, 12(1), 23–31.

Lagaris, I. E., Likas, A., and Fotiadis, D. I. (1998). "Artificial neural networks for solving ordinary and partial differential equations." *IEEE transactions on neural networks*, 9(5), 987–1000.

Lai, Z. and Nagarajaiah, S. (2019a). "Semi-supervised structural linear/nonlinear damage detection and characterization using sparse identification." *Structural Control and Health Monitoring*, 26(3), e2306.

Lai, Z. and Nagarajaiah, S. (2019b). "Sparse structural system identification method for nonlinear dynamic systems with hysteresis/inelastic behavior." *Mechanical Systems and Signal Processing*, 117, 813–842.

Lee, H. and Park, H. (2011). "Gage-free stress estimation of a beam-like structure based on terrestrial laser scanning." *Computer-Aided Civil and Infrastructure Engineering*, 26(8), 647–658.

Leylaz, G., Ma, S. F., and Sun, J.-Q. (2021). "An optimal model identification algorithm of nonlinear dynamical systems with the algebraic method." *Journal of Vibration and Acoustics*, 143(2).

Li, D. and Wang, Y. (2020). "Parameter identification of a differentiable bouc-wen model using constrained extended kalman filter." *Structural Health Monitoring*, 1475921720929434.

Liu, T. and Meidani, H. (2022). "Graph neural network surrogate for seismic reliability analysis of highway bridge system." *arXiv preprint arXiv:2210.06404*.

Louhghalam, A., Pellenq, R. J.-M., and Ulm, F.-J. (2018). "Thermalizing and damping in structural dynamics." *Journal of Applied Mechanics*, 85(8).

Nabian, M. A. and Meidani, H. (2018). "A deep neural network surrogate for high-dimensional random partial differential equations." *Journal of Environmental Sciences (China) English Ed.*

Nguyen, L. H. and Goulet, J.-A. (2018). "Anomaly detection with the switching kalman filter for structural health monitoring." *Structural Control and Health Monitoring*, 25(4), e2136.

Nosé, S. (1984). "A unified formulation of the constant temperature molecular dynamics methods." *The Journal of chemical physics*, 81(1), 511–519.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). "Pytorch: An imperative style, high-performance deep learning library." *Advances in neural information processing systems*, 32.

Quade, M., Abel, M., Nathan Kutz, J., and Brunton, S. L. (2018). "Sparse identification of nonlinear dynamics for rapid model recovery." *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(6), 063116.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2017). "Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations." *arXiv preprint arXiv:1711.10561*.

Sirca Jr, G. and Adeli, H. (2012). "System identification in structural engineering." *Scientia Iranica*, 19(6), 1355–1364.

Song, M., Astroza, R., Ebrahimian, H., Moaveni, B., and Papadimitriou, C. (2020). "Adaptive kalman filters for nonlinear finite element model updating." *Mechanical Systems and Signal Processing*, 143, 106837.

Villani, L. G., Da Silva, S., and Cunha, A. (2020). "Application of a stochastic version of the restoring force surface method to identify a duffing oscillator." *Nonlinear Dynamics of Structures, Systems and Devices*, Springer, 299–307.

Wang, Y. (2017). "A new concept using lstm neural networks for dynamic system identification." *2017 American Control Conference (ACC)*, IEEE, 5324–5329.

Wen, Y.-K. (1976). "Method for random vibration of hysteretic systems." *Journal of the engineering mechanics division*, 102(2), 249–263.

Yang, Y. and Yang, J. P. (2018). "State-of-the-art review on modal identification and damage detection of bridges by moving test vehicles." *International Journal of Structural Stability and Dynamics*, 18(02), 1850025.

Zhang, D., Lu, L., Guo, L., and Karniadakis, G. E. (2019a). "Quantifying total uncertainty in physics-informed neural networks for solving forward and inverse stochastic problems." *Journal of Computational Physics*, 397, 108850.

Zhang, R., Chen, Z., Chen, S., Zheng, J., Büyüköztürk, O., and Sun, H. (2019b). "Deep long short-term memory networks for nonlinear structural seismic response prediction." *Computers & Structures*, 220, 55–68.

Zhang, R., Liu, Y., and Sun, H. (2020). "Physics-informed multi-lstm networks for metamodeling of nonlinear structures." *Computer Methods in Applied Mechanics and Engineering*, 369, 113226.

Zhong, W. and Meidani, H. (2023). "Pi-vae: Physics-informed variational auto-encoder for stochastic differential equations." *Computer Methods in Applied Mechanics and Engineering*, 403, 115664.

List of Tables

TABLE 1. The parameter identification and errors of 10-DOF cubic stiffness system with noise-free data. The mean and standard deviation of the PIDynNet identifications are reported based on 10 different evaluations with different randomized initializations.

Parameter	Ground truth	UKF value	UKF error (%)	PIDynNet value (Mean ± SD)	PIDynNet error (Mean ± SD)(%)
$k_1(\mathrm{N/m})$	35	36.06	3.04	34.33 ± 0.45	**1.90** ± 1.28
$k_2(\mathrm{N/m})$	35	35.42	1.21	34.88 ± 0.16	**0.34** ± 0.47
$k_3(\mathrm{N/m})$	35	35.56	1.60	34.74 ± 0.07	**0.73** ± 0.21
$k_4(\mathrm{N/m})$	30	30.41	1.36	29.67 ± 0.13	**1.09** ± 0.43
$k_5(\mathrm{N/m})$	30	30.42	1.39	29.59 ± 0.20	**1.38** ± 0.66
$k_6(\mathrm{N/m})$	30	30.46	1.53	29.67 ± 0.19	**1.10** ± 0.64
$k_7(\mathrm{N/m})$	25	25.40	1.60	24.80 ± 0.08	**0.79** ± 0.34
$k_8(\mathrm{N/m})$	25	25.55	2.19	24.88 ± 0.06	**0.48** ± 0.26
$k_9(\mathrm{N/m})$	25	25.87	3.49	24.87 ± 0.07	**0.52** ± 0.29
$k_{10}(\mathrm{N/m})$	25	27.11	8.44	24.90 ± 0.07	**0.41** ± 0.28
$k_{nl}(\mathrm{N/m^3})$	150	140.11	6.59	147.55 ± 5.68	**1.63** ± 3.79

TABLE 2. Error comparisons in the identified parameters of 10-DOF cubic stiffness system via PIDynNet with different subperiod durations.

Parameter	T = 2s		T = 3s		T = 5s		T = 10s	
	PIDynNet	Error (%)	PIDynNet	Error (%)	PIDynNet	Error (%)	PIDynNet	Error (%)
k_1(N/m)	34.64	1.02	34.66	0.98	34.69	0.88	32.87	6.10
k_2(N/m)	34.57	1.24	34.78	0.63	34.77	0.65	34.35	1.86
k_3(N/m)	34.67	0.95	34.67	0.95	34.86	0.41	34.73	0.78
k_4(N/m)	29.63	1.24	29.65	1.18	29.86	0.45	29.51	1.63
k_5(N/m)	29.62	1.27	29.58	1.39	29.69	1.02	29.45	1.82
k_6(N/m)	29.60	1.35	29.54	1.54	29.91	0.29	29.58	1.41
k_7(N/m)	24.76	0.97	24.58	1.68	24.97	0.11	24.66	1.35
k_8(N/m)	24.81	0.76	24.65	1.42	24.99	0.04	24.63	1.50
k_9(N/m)	24.76	0.95	24.62	1.50	25.02	0.10	24.54	1.82
k_{10}(N/m)	24.78	0.89	24.70	1.18	24.95	0.21	25.06	0.23
k_{nl}(N/m^3)	148.19	1.21	146.33	2.44	152.99	1.99	171.56	14.38
Average		1.08		1.35		0.56		2.99

TABLE 3. Comparison of computed parameters of the Cubic stiffness model identified using UKF and PIDynNet with respect to the ground truth under different noise levels.

Parameter	Ground Truth	2.5% Noise				5% Noise			
		UKF	Error (%)	PIDynNet	Error (%)	UKF	Error (%)	PIDynNet	Error (%)
k_1(N/m)	35.0	35.52	1.48	34.24	2.16	36.52	4.36	34.56	1.25
k_2(N/m)	35.0	35.44	1.25	34.52	1.38	35.57	1.64	34.69	0.90
k_3(N/m)	35.0	35.66	1.88	34.48	1.49	35.37	1.07	34.56	1.26
k_4(N/m)	30.0	30.16	0.52	29.43	1.89	30.53	1.76	29.42	1.94
k_5(N/m)	30.0	30.71	2.36	29.36	2.14	30.70	2.33	29.45	1.85
k_6(N/m)	30.0	29.66	1.13	29.56	1.46	29.95	0.18	29.55	1.51
k_7(N/m)	25.0	25.34	1.36	24.72	1.13	25.43	1.72	24.70	1.21
k_8(N/m)	25.0	25.53	2.13	24.69	1.23	26.13	4.51	24.74	1.04
k_9(N/m)	25.0	25.14	0.55	24.71	1.16	25.34	1.38	24.72	1.13
k_{10}(N/m)	25.0	26.04	4.14	24.84	0.64	26.71	6.84	24.94	0.26
k_{nl}(N/m^3)	150.0	142.68	4.88	145.98	2.68	137.97	8.02	145.06	3.29
Average			1.97		1.58		3.07		1.42

TABLE 4. Parameter Identification of 3-DOF Bouc-Wen Model with noise-free data. The mean and standard deviation of the PIDynNet identification are reported based on 10 different evaluations with different randomized initializations.

Parameter	Ground Truth	UKF Value	UKF Error (%)	PIDynNet Value (Mean ± SD)	PIDynNet Error (Mean ± SD) (%)
$k_1(\mathrm{N/m})$	50	51.77	3.54	49.21 ± 0.47	1.57 ± 0.94
$k_2(\mathrm{N/m})$	45	46.71	3.80	44.64 ± 0.52	0.79 ± 1.15
$k_3(\mathrm{N/m})$	45	46.83	4.08	45.08 ± 0.48	0.17 ± 1.06
β	2	2.26	12.86	1.97 ± 0.01	1.59 ± 0.57
γ	2	2.04	2.16	1.98 ± 0.01	1.17 ± 0.45
n	1	1.01	0.73	0.99 ± 0.04	0.22 ± 0.37

TABLE 5. Comparison of computed parameters of the Bouc-Wen model identified using UKF and PIDynNet with respect to the ground truth under different noise levels.

Parameter	Ground Truth	2.5% Noise				5% Noise			
		UKF	Error (%)	PIDynNet	Error (%)	UKF	Error (%)	PIDynNet	Error (%)
k_1(N/m)	50.0	51.965	3.930	51.158	2.316	51.768	3.535	50.812	1.624
k_2(N/m)	45.0	46.980	4.399	45.961	2.135	46.711	3.803	45.679	1.509
k_3(N/m)	45.0	46.814	4.031	44.198	1.781	46.835	4.077	43.801	2.664
β	2.0	2.261	13.074	2.023	1.136	2.257	12.863	2.019	0.929
γ	2.0	2.011	0.526	2.008	0.385	2.043	2.156	2.011	0.566
n	1.0	0.998	0.202	0.993	0.680	1.007	0.727	0.997	0.284
Average			4.360		1.406		4.527		1.263

List of Figures

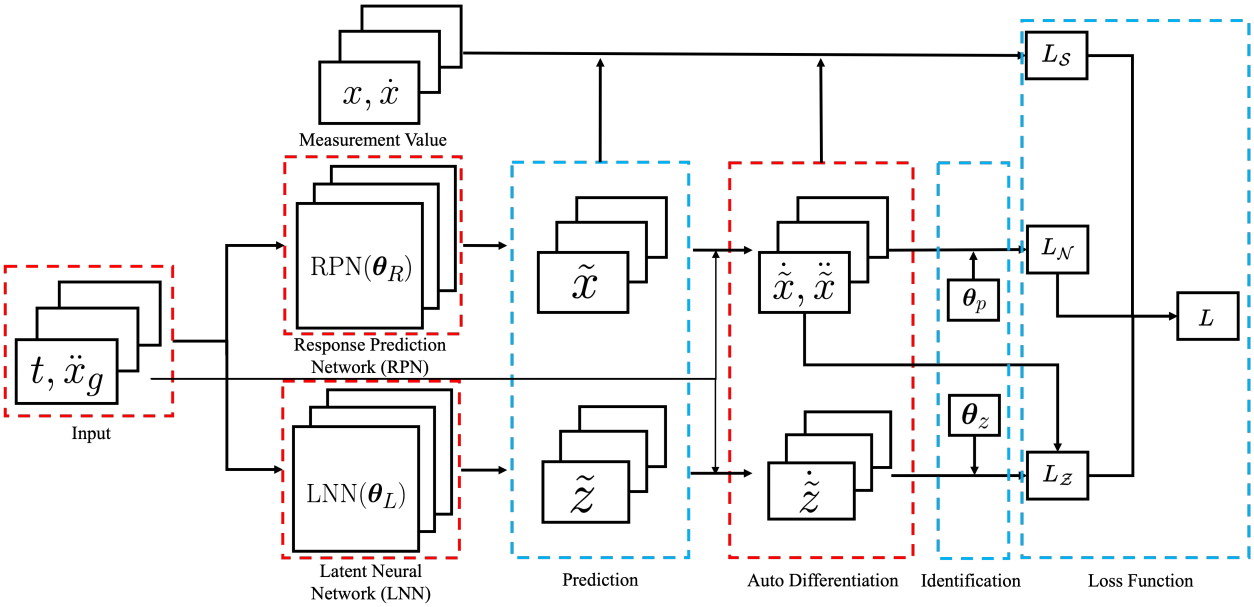

Fig. 1. Framework of PIDynNet with RPN and LNN. RPN and LNN predict the response and latent response separately. The parameters of RPN and LNN are optimized simultaneously. Three type of loss function are included: supervised loss $L_{\mathcal{S}}$, physics-based loss $L_{\mathcal{N}}$, and latent physics-based loss $L_{\mathcal{Z}}$

.

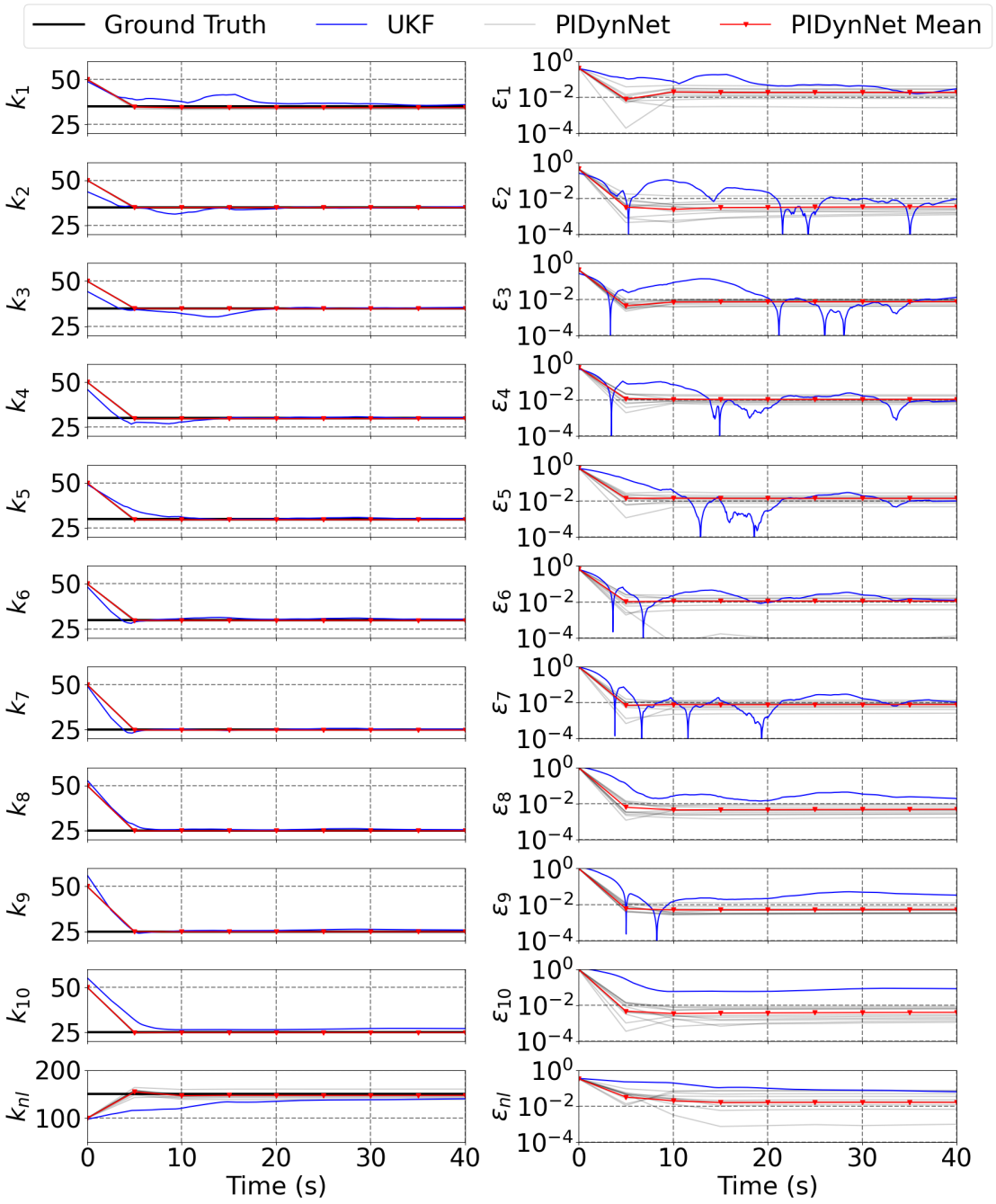

Fig. 2. Convergence of parameter updating and errors on 10-DOF cubic stiffness system. Ten independent identifications are conducted with different neural network randomized initialization. The gray curve represents the identification history of each randomized initialization and the red curve represents the average result of all randomized initialization.

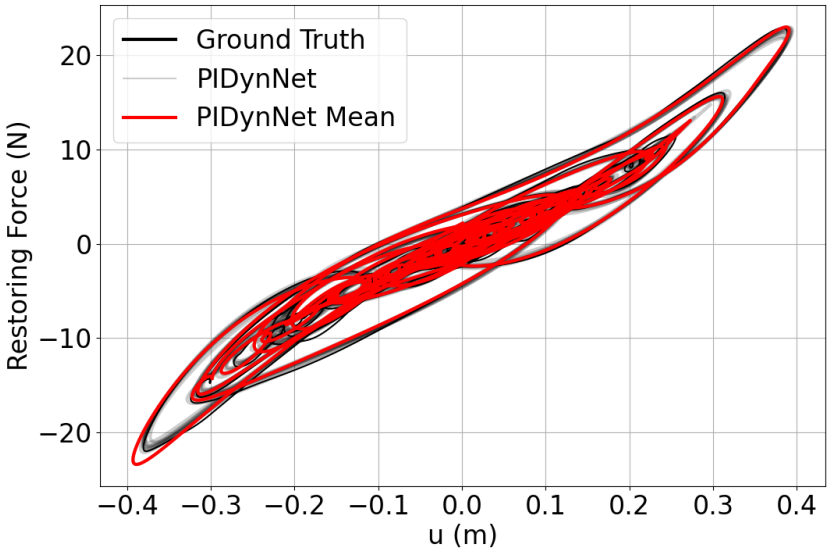

Fig. 3. Hysteresis diagram at the first story for the identified cubic stiffness system. The gray curve represents the identification history of each randomized initialization, and the red curve represents the hysteresis diagram with the average of the identified parameters of all randomized initialization

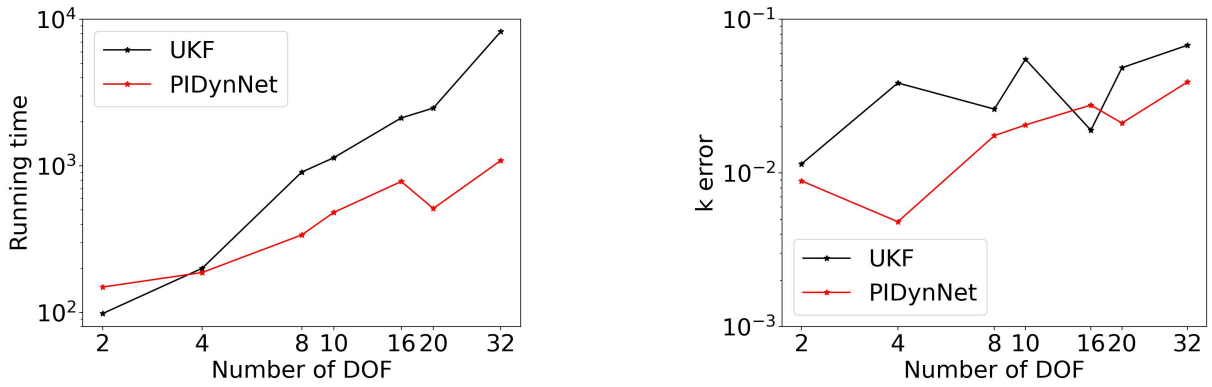

Fig. 4. Computational time and error analysis under different sizes of cubic stiffness system. The left figure compares the running time between UKF and PIDynNet under systems with different DOF. The right figure compares the average identification error between UKF and PIDynNet under systems with different numbers of DOF.

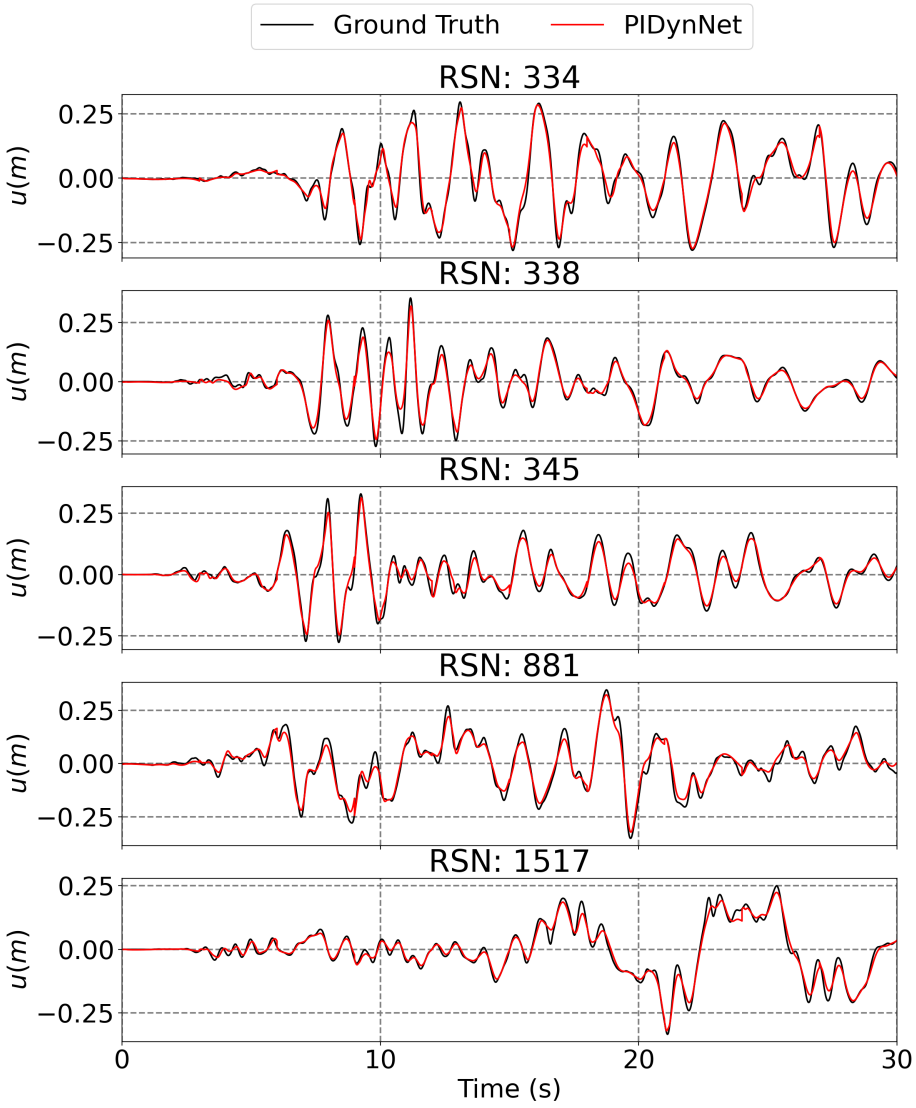

Fig. 5. Prediction of first story nonlinear displacements of cubic stiffness system under unseen ground motions.

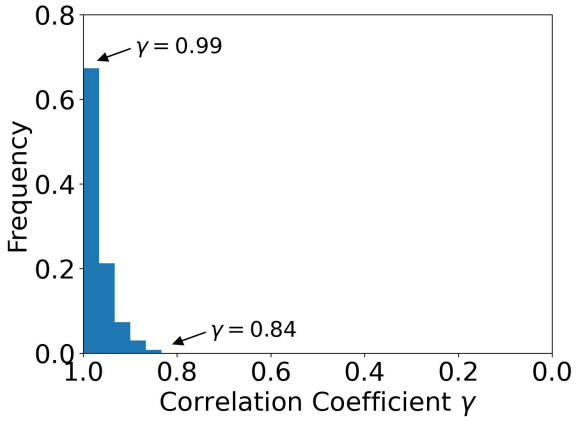

Fig. 6. Distribution of correlation coefficient under unseen ground motions. The ground motions are selected from NGA-West2 and the number of ground motions is 30.

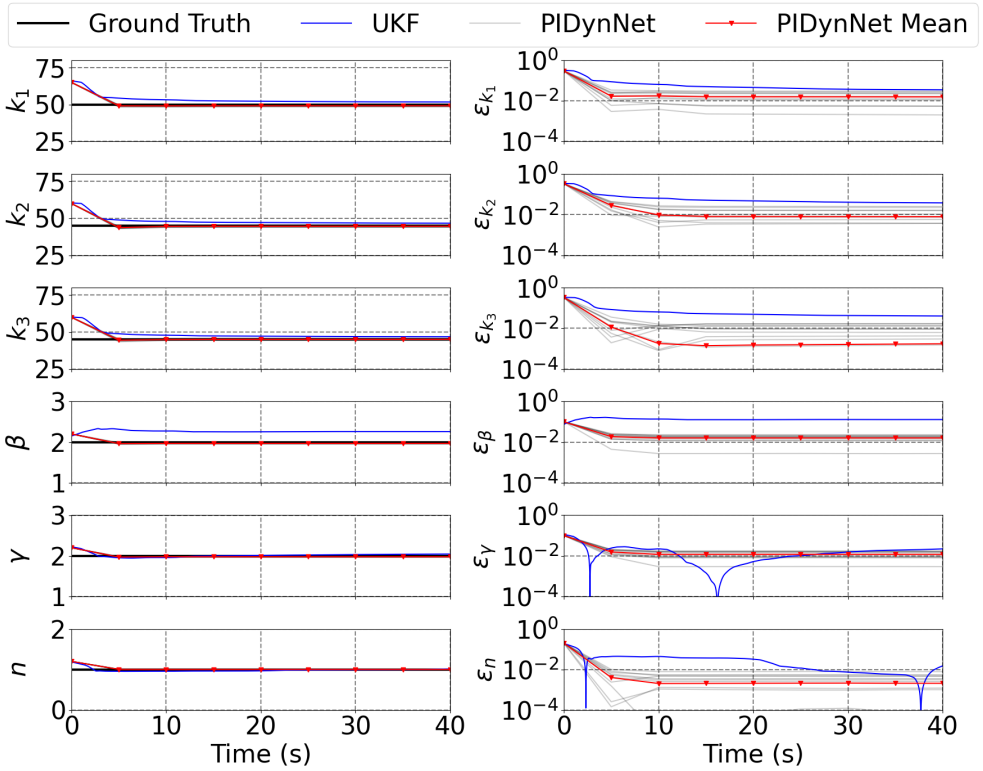

Fig. 7. Convergence of parameter updating and error of 3-DOF Bouc-Wen model using UKF and the PIDynNet. Ten independent identifications are conducted with different neural network randomized initialization. The gray curve represents the identification history of each randomized initialization, and the red curve represents the average result of all randomized initialization.

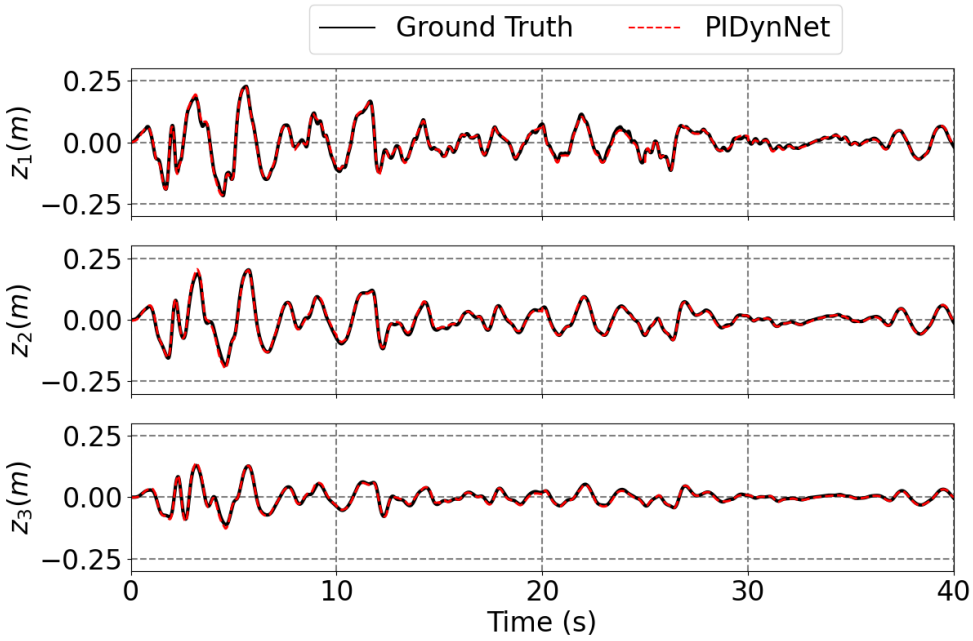

Fig. 8. Identified hysteretic displacement response at three stories of 3-DOF Bouc-Wen model.

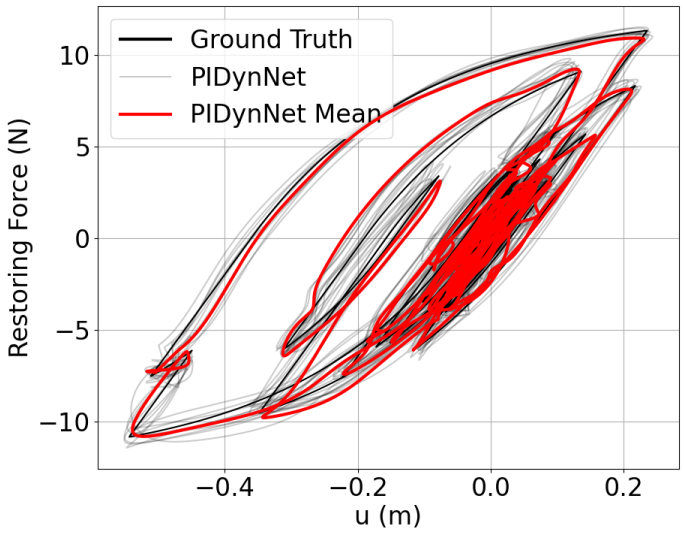

Fig. 9. Identified hysteresis diagram at the first story of 3-DOF Bouc-Wen model by PIDynNet. The gray curve represents the identification history of each randomized initialization, and the red curve represents the hysteresis diagram with the average of the identified parameters of all randomized initialization.

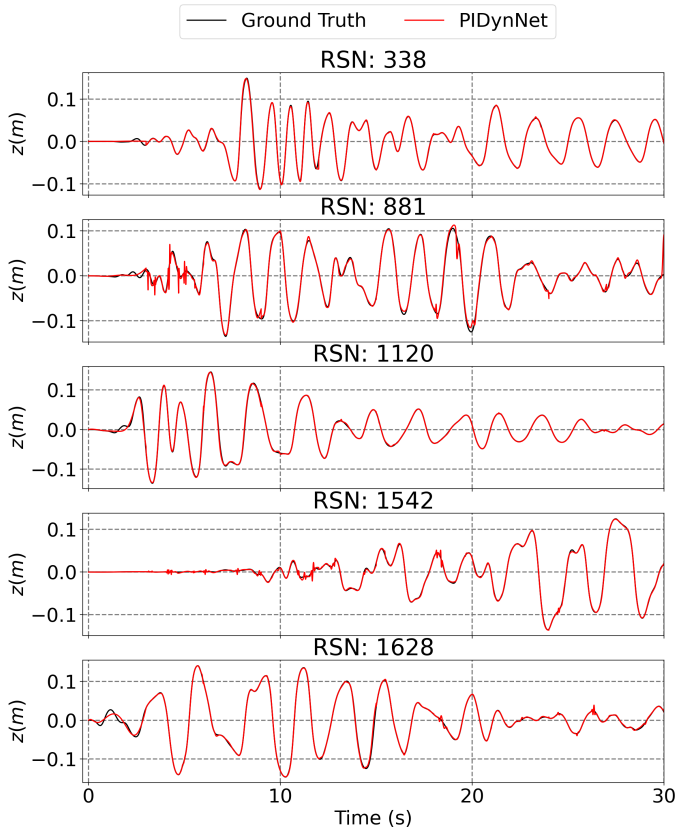

Fig. 10. Prediction of hysteresis displacement at the first story of 3-DOF Bouc-Wen model under unseen ground motions.

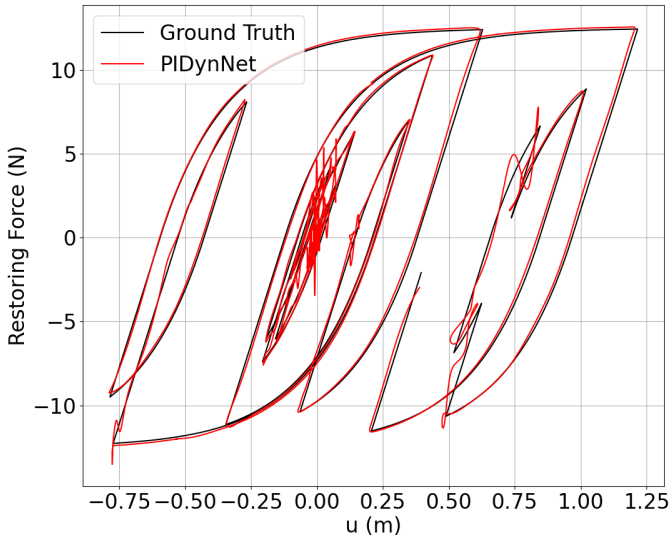

Fig. 11. Prediction of hysteresis diagram at the first story of 3-DOF Bouc-Wen model under unseen ground motions.