POLITECNICO DI TORINO
Repository ISTITUZIONALE

Nonlinear System Identification Using Audio-Inspired WaveNet Deep Neural Networks

Original
Nonlinear System Identification Using Audio-Inspired WaveNet Deep Neural Networks / Yuan, W., Zhu, R., Xiang, T., Marchesiello, S., Anastasio, D., Fei, Q.. - In: AIAA JOURNAL. - ISSN 1533-385X. - ELETTRONICO. - (2023), pp. 1-9. [10.2514/1.J062860]

Availability:
This version is available at: 11583/2979296 since: 2023-06-09T08:47:06Z

Publisher:
American Institute of Aeronautics and Astronautics

Published
DOI:10.2514/1.J062860

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
AIAA preprint/submitted version e/o postprint/Author's Accepted Manuscript

(Article begins on next page)

10 September 2026

POST-PRINT / AUTHOR'S ACCEPTED MANUSCRIPT

Nonlinear system identification using Audio-Inspired WaveNet Deep Neural Networks

Weixuan Yuan, Rui Zhu , and Tao Xiang
Technical University of Munich, Munich, 85748, Germany

Stefano Marchesiello, and Dario Anastasio
Politecnico di Torino, Torino, 10129, Italy

Qingguo Fei
Southeast University, Nanjing, Jiangsu Province, 210000, China

https://doi.org/10.2514/1.J062860

Cite as:

W. Yuan, R. Zhu, T. Xiang, S. Marchesiello, D. Anastasio, Q. Fei, Nonlinear System Identification Using Audio-Inspired WaveNet Deep Neural Networks, AIAA Journal 0 0:0, 1-9. doi: 10.2514/1.J062860

Nonlinear system identification using Audio-Inspired WaveNet Deep Neural Networks

Weixuan Yuan [1], Rui Zhu [2], and Tao Xiang [3]
Technical University of Munich, Munich, 85748, Germany

Stefano Marchesiello [4], and Dario Anastasio [5]
Politecnico di Torino, Torino, 10129, Italy

Qingguo Fei [6]
Southeast University, Nanjing, Jiangsu Province, 210000, China

Nonlinear system identification is a challenging task that requires accurate estimation of the structural model from observations of nonlinear behavior. The WaveNet, originally a neural network architecture for audio processing, has been modified and first introduced to the analysis of mechanical signals to capture long-term dependencies in mechanical systems and generate high-quality signals. A novel nonlinear system identification method has been proposed using a modified WaveNet-based approach that constructs the relationship between the vibration response and the nonlinear elements in the inverse model without the need for a definite structural model. This approach utilizes dilated convolution for feature extraction and a multi-layer perceptron for feature transition, with the addition of average pooling along the time dimension for adaptive processing of varying length data, which is more computationally efficient and widely applicable. The 13-layer modified WaveNet models have been designed and applied to the problem. Comparisons with other baseline models were made to demonstrate the method's superiority in terms of accuracy, effectiveness, and robustness. Additionally, the method has been applied to predict composite models of friction and elastic curves, demonstrating its ability to handle diverse and complex problems.

[1] Master student, Technical University of Munich, weixuan.yuan@tum.de

[2] Postdoc researcher, School of Engineering and Design, Corresponding author, r.zhu@tum.de

[3] Master student, Technical University of Munich, tao.xiang@tum.de

[4] Professor, Dipartimento di Ingegneria Meccanica ed Aerospaziale, stefano.marchesiello@polito.it

[5] Research fellow, Dipartimento di Ingegneria Meccanica ed Aerospaziale, dario.anastasio@polito.it

[6] Professor, School of Mechanical Engineering, qgfei@seu.edu.cn

Nomenclature

(Nomenclature entries should have the units identified)

$\boldsymbol{A}$ = dynamical system matrix of the state-space model

A_c = dynamical system matrix of the continuous-time state-space model

B_c = input matrix of the continuous-time state-space model

BN= batch normalization

b^i = trainable parameter

C = output matrix

C_v = viscous damping matrix

D = direct feedthrough matrix of the state-space model

$F(t)$ = force vector

H = underlying frequency response function matrix

H_E = "extended" frequency response function matrix

K = stiffness matrix

M = mass matrix

z = displacement vector

λ = coefficient of the nonlinear element

σ = nonlinear activation

1. Introduction

Nonlinear system identification [1][2] is the process of determining the characteristics and behavior of a nonlinear system by analyzing its response to various excitations. It plays a critical role in understanding the behavior and performance of nonlinear systems and has various applications, including control, signal processing, and structural analysis [3][4]. For instance, it identifies the nonlinear aerodynamic characteristics of aircraft and helps design control systems to handle these nonlinearities. In the design of aircraft structures, nonlinear system identification is employed to determine the nonlinear behavior of structural components under different loads and optimize the design for improved performance and reliability.

There are several different classification schemes that group nonlinear system identification methods. One way to classify these methods is based on the type of system being modeled. For example, some methods are specifically designed for modeling continuous systems [5], while others are more suited for discrete systems [6]. Additionally, nonlinear system identification methods can also be classified based on the approach they take to identify the system. Some methods use statistical techniques, such as regression analysis or maximum likelihood estimation [7], while others are based on more deterministic approaches such as optimization or gradient descent [8]. The development status of nonlinear system identification has been thoroughly summarized in the literature [9][10]. Marchesiello et al. [11] proposed the nonlinear subspace identification based on the idea that a nonlinear system can be approximated by a linear combination of nonlinear functions, referred to as "basis functions". Building on previous research, Zhu et al.

[12][13] introduced the restoring force surface method and Bayesian model selection into subspace identification to enable the identification of nonlinear stiffness and damping. An automatic nonlinear subspace identification method [14] is proposed to avoid the phenomenon of virtual mode or omission. The similarity coefficient and distance function are introduced to cluster the identified modal results, the poles of the false modes are removed to obtain the cluster stabilization diagram, and the best order of the system is received. In addition, deep learning methods [15][16] have been widely used in the field of system identification as they can directly model the relationship between vibrational responses and nonlinear elements. This makes them particularly powerful for understanding and predicting the behavior of nonlinear systems. Brunton et al.[17] combine sparsity-promoting techniques and machine learning with nonlinear dynamical systems to discover governing equations from noisy measurement data. One approach is to use a deep neural network to learn a mapping from input data to output data for a nonlinear system, which can work well for systems with a complex functional relationship between the inputs and outputs. Another approach is to use a recurrent neural network [18], which can learn temporal dependencies in data and is well-suited for modeling dynamic systems. For instance, a long short-term memory (LSTM) network can be used to model a nonlinear system by learning to predict its future behavior given its past behavior [19]. A reduction scheme based on the identification of continuous time recursive neural networks [20] is proposed to identify the parameters of a nonlinear aerodynamic model from input-output data obtained through high-fidelity simulations.

In this paper, the system identification method based on WaveNet is proposed to improve the accuracy of nonlinear parameter estimation without requiring the evaluation of the structural model. WaveNet [21] is a deep learning architecture developed by DeepMind, a subsidiary of Alphabet Inc, designed specifically for analyzing and synthesizing audio signals. WaveNet is based on dilated convolution, which allows for a large receptive field with a small number of parameters. As a result, WaveNet can capture long-range dependencies in data and generate high-quality audio samples. As both the response signals and sound signals in the mechanical field are time-based signal sequences, the WaveNet algorithm, which is specifically designed for analyzing and synthesizing time-based signal sequences, can be a useful tool for nonlinear system identification in the mechanical field. The ability of WaveNet to capture long-range dependencies in the data and generate high-quality signals makes it well-suited for modeling and predicting the nonlinear behavior of mechanical systems.

We propose a novel nonlinear system identification based on WaveNet. Unlike traditional methods that rely on evaluating the nonlinear structural model, the new method is an end-to-end approach that directly establishes the relationship between the response and the nonlinear element. It utilizes dilated convolution to obtain large receptive fields on vibration data and captures features at all levels for identifying the nonlinear coefficient. Furthermore, average pooling along the time dimension allows for the adaptive processing of data of varying lengths. The robustness of the proposed method is evaluated through adding different levels of noise and comparing the results with other baseline methods.

2. Nonlinear parameter identification framework

2.1 Basic description

A multilayer building with nonlinear stiffness elements is investigated in **Fig. 1**. The dynamic equation of the system can be expressed as:

$$M\ddot{z}(t) + C_v\dot{z}(t) + Kz(t) + f_n(z,\dot{z}) = F(t) \tag{1}$$

Where M represents the mass matrix, C_v represents the viscous damping matrix, K represents the linear stiffness matrix, f_n represents the nonlinear force, and F represents the excitation. It is important to note that f_n is expressed as a sum of nonlinear basis functions multiplied by nonlinear parameters $\lambda=[\lambda_1, \lambda_2,\ldots\lambda_n]$ and the location vector.

The nonlinear subspace identification (NSI) method is one approach for identifying nonlinear systems. This method begins by selecting an appropriate transformation of the system's inputs and outputs, so that the transformed system can be approximated by a linear model. Subsequently, the state-space matrices A, B, C, and D of the linear model are estimated using techniques from linear system identification. After the matrices have been estimated, the linear model is transformed back to the original space to obtain a nonlinear model of the system. The model order, which is the number of states in the model, and the values of the system matrices are crucial parameters that must be estimated accurately to obtain a good model of the system.

Therefore, the extended frequency response function H_E of the nonlinear system can be expressed as

$$H_E(\omega) = D + C\left(j\omega I - A_c\right)^{-1} B_c \tag{2}$$

Where A_c is the dynamical system matrix of the continuous-time state-space model, B_c is the input matrix of the continuous-time state-space model. H is the underlying linear frequency response function.

$$H = \left(K + j\omega C_v - \omega^2 M\right)^{-1} \tag{3}$$

The relationship between H_E and H can be obtained in Ref. [13]

$$H_E = \begin{bmatrix} H & H\lambda_1 & \cdots & H\lambda_n \end{bmatrix} \tag{4}$$

λ_i contains the coefficients and location information associated with nonlinear elements. To accurately identify the nonlinear parameters of a system, it is necessary to accurately measure the input excitation and output response at various frequencies, and use this information to estimate the extended frequency response function. For nonlinear systems, evaluating the system model or corresponding parameters can be challenging. To solve this problem, we use the WaveNet neural network to directly construct the model of the response and the nonlinear unit, avoiding the prior determination of the nonlinear structural model

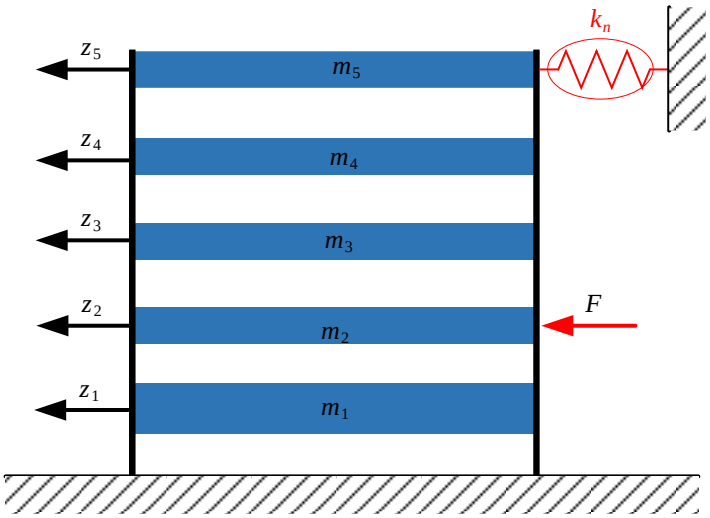

Fig. 1 Multilayer building with nonlinearity.

2.2 Nonlinear system identification based on novel WaveNet

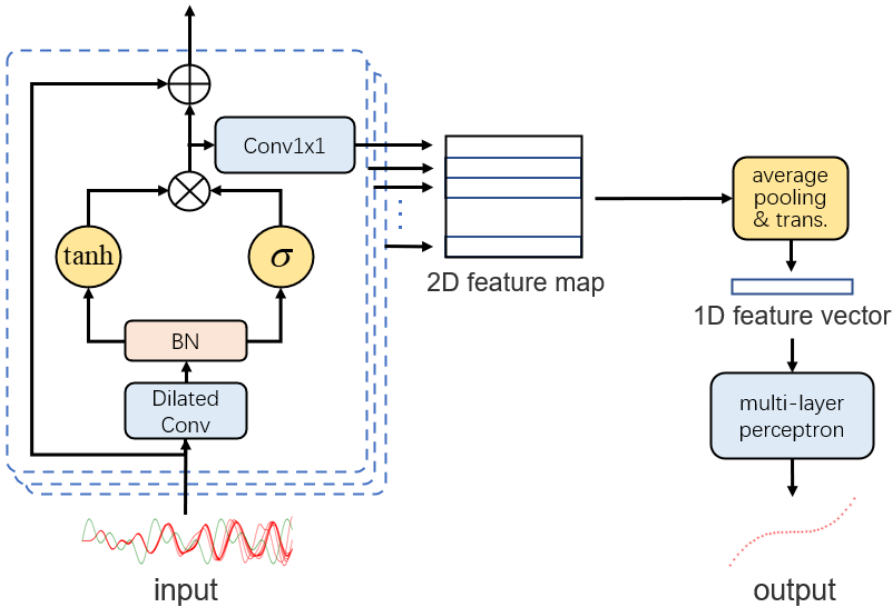

Fig. 2 WaveNet architecture with a feature extractor, an average pooling, and a multi-layer perceptron.

We propose a deep learning-based approach for reconstructing the mapping from elastic deformation to elasticity. Specifically, a neural network takes mechanical signal sequences as input and infers the discrete sampling sequence of the elasticity curve. The sampling sequence is then converted into a continuous function that describes the mapping through regression.

In this work, we introduce a WaveNet-based architecture, as depicted in **Fig. 2**, and demonstrate its advantages in accuracy and size through comparisons with a few baseline architectures, accompanied by visualized results.

2.2.1 The WaveNet-based model

The WaveNet-based architecture is divided into three modules. First, a feature extractor based on dilated convolutions, represented by the blue rectangles in **Fig. 3**, converts the input X_0 into a two-dimensional feature map *S*. Then, features under different receptive fields are merged through average pooling along the time dimension, resulting in a one-dimensional feature vector, with size independent of the input. Finally, a multi-layer perceptron translates the transposed feature vectors into an output vector of length 256, representing discrete sampling points of the nonlinear elastic and nonlinear damping curve. In the following, more detailed descriptions of each part will be discussed.

1) Feature extractor

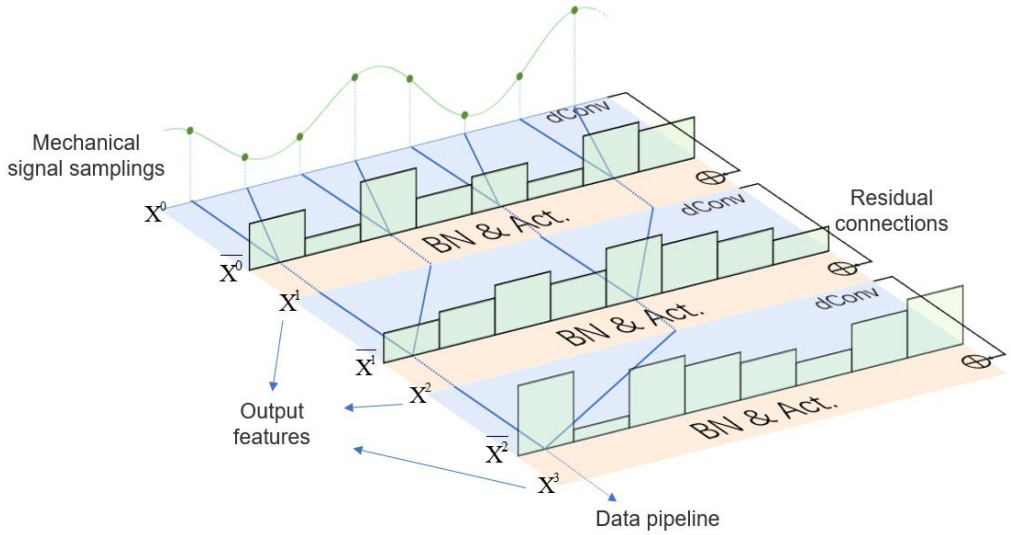

Fig. 3 The dilated-convolution-based feature extractor.

Fig. 3 provides a more intuitive illustration of the feature extractor, which consists of repetitions of blocks comprising dilated convolution (in blue), batch normalization, and nonlinear activation (in orange). The input of the ith block is denoted by X^i, which is also the output of the previous block. The data pipeline visualized by the blue lines can be summarized as:

$$X^{i+1} = Act \cdot BN \cdot dConv \cdot (X^i) + Conv1\times1(X^i) \tag{5}$$

where X^0 is the input for the neural network, namely the mechanical signal sequences. *BN*, *dConv*, and *Conv*1×1 represent batch normalization, dilated convolution, and 1×1 convolution operation.

The advantage of this structure benefits from the capacity of learning translation invariance and the receptive field that grows exponentially with the number of stacked dilated convolutions [21]. By setting the kernel size to 2, the output of the convolutional layer with dilation *d* is such that each feature is a biased weighted sum of two features on the previous layer, with an interval of *d*, visualized by the solid lines in rectangles labeled with "*dConv*" in **Fig. 3**. Let Kr^i, b^i denoting the trainable parameters, namely the kernels and the bias, the dilated convolution operation $\overline{X}^i = dConv_{K^i,b^i}(X^i)$ is defined by:

$$\overline{X}^{i}_{c,n} = \sum_{m=1}^{ch^{i-1}} \sum_{l=0}^{d+1} X^{i}_{m,n-floor(\frac{d}{2})+l} \cdot Kr^{i}_{ch,m,l} + b^{i}_{ch,l}, ch \in [Ch^{i}], n \in [L] \tag{6}$$

where Ch^i refers to the channel size of the i^{th} convolution, a hyper-parameter that must be configured prior to training. Vector Kr^i is of size $Ch^i \times (d+1) \times L$, representing Ch^i kernels, each with dimension $(d+1) \times L$. It should be noted that, for each kernel, all entities except those in the first and last columns are zeros, which model the dilated receptive field. In addition, zero padding is applied so that $X^i_{m,n}=0$ for $n \notin [L]$.

After each convolution operation, the distribution of the output $\overline{X}^i$ is manipulated automatically by batch normalization:

$$X^{i} = \gamma \cdot (\overline{X}^{i} - \mu_B) \cdot (\sigma_B^2 + \varepsilon)^{-1} + \beta \tag{7}$$

where μ_B, σ_B^2 are the mean and variance of the mini-batch obtained during training and γ, β are trainable parameters. This method accelerates convergence significantly by avoiding the unstable behavior of the gradients [22], especially for neural networks as deep as those introduced in this work.

To extract non-linear features, the non-linear activation is applied before outputting in each block:

$$Act(x) = \tanh(x) \otimes \text{sigmoid}(x) \tag{8}$$

where $\otimes$ denotes the element-wise multiplication.

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{9}$$

$$\text{sigmoid}(x) = \frac{1}{1 + e^{-x}} \tag{10}$$

Moreover, the input of each block is propagated directly to the end through a residual connection [23], represented by the second term in Eq.(5) and visualized in **Fig. 3** by the solid arrows on the right. To address the problem of inconsistent channel size, a convolution operator with kernel size 1 (i.e. 1×1 convolution) is applied to the input before adding:

$$Conv1\times1_{\overline{Kr^i},\overline{b^i}}(X^{i}) = \overline{Kr^{i}} X^{i} + \overline{b^{i}} \tag{11}$$

where $\overline{Kr^i} \in R^{Ch^{i-1} \times Ch^i}$ and $\overline{b^i} \in R^{Ch^i}$ are trainable parameters.

The output of the feature extractor S is the stack of half the reduced output of each block and reduction is again performed by 1×1convolutions:

$$S_{n+j} = Conv1\times1_{Kr^i,b^i}(X^{p})_j, n = \sum_{i=1}^{p-1} Ch^{i}, j \in \left[Ch^{i} \right] \tag{12}$$

where $Kr^i \in R^{C^i \times \frac{C^i}{2}}$ and $b^i \in R^{\frac{C^i}{2}}$ are trainable parameters. In comparison to directly using the output of the last block as the feature map, the stacked reduced feature map also includes low-level features, namely the output of earlier blocks, which can be utilized by later layers as well.

2) Global pooling

Since the objective is a constant physical quantity that does not change over time, regardless of the receptive field in which it is observed, we merge the feature map along the time dimension through global average pooling [24]:

$$Y_i^0 = \frac{1}{L}\sum_{j=1}^{L} S_{i,j} \tag{13}$$

This results in a feature vector of size $\frac{1}{2}\sum \mathrm{Ch}^i$, independent of the input length.

3) Multi-layer perceptron

A multi-layer perceptron consists of fully connected layers [25]. Let Y^i denote the input of the i^{th} layer, the fully connected layer computes its biased and activated linear combination:

$$Y^{i+1} = \max(W^i Y^i + b_{fc}^i, 0) \tag{14}$$

where W^i and b^i_{fc} are trainable parameters. Note that, for the final layer, there is no activation, and the output dimension is equal to 256, namely the number of the sampling points.

It is important to note that the number and size of trainable parameters in all layers are not dependent on the length of the input sequence, allowing for the acceptance of input data of varying lengths.

4) Loss function and training

The model is trained to minimize the mean squared error between the ground truth and the prediction, defined as:

$$MSE(Y_{true}, Y_{pre}) = \frac{1}{N}\sum_{i=1}^{N}(Y_{true}^i - Y_{pre}^i)^2 \tag{15}$$

where N is the size of training set. Observing that sub gradients are accessible for all operators introduce above by backward propagation, all trainable parameters will be optimized by sub gradient method [27] iteratively.

The complete method pipeline of the method is summarized in **Fig. 4**.

2.2.2 Baselines

To verify the advantages of our model in accuracy, models of three popular architectures in deep learning were implemented and applied to the problem as baselines: fully connected network (FC), long short-term memory (LSTM), and Convolutional Neural Networks (CNN).

1) FCs are neural networks that consist solely of fully connected layers. Since it only accepts one-dimensional input, the response sequences are normalized and flattened to vectors of length $Ch \times L$ before being fed to the network. FCs naturally have complete receptive fields, but this results in extremely large model sizes.
2) By CNNs, the convolutional layers without dilation are used for feature extraction. To obtain the exponentially growing receptive field as the WaveNet models, max-pooling layers were inserted between convolutions.
3) Long Short Term Memory Networks (LSTM) is a special class of recurrent neural networks, which consists of repeating modules of neural network, each contains multiple layers that are responsible for input, forgiveness, and output, respectively [28].

Note that techniques that improves performance and accelerate training, such as BN, non-linear activation, and dropouts [26], are equally applied to all architectures.

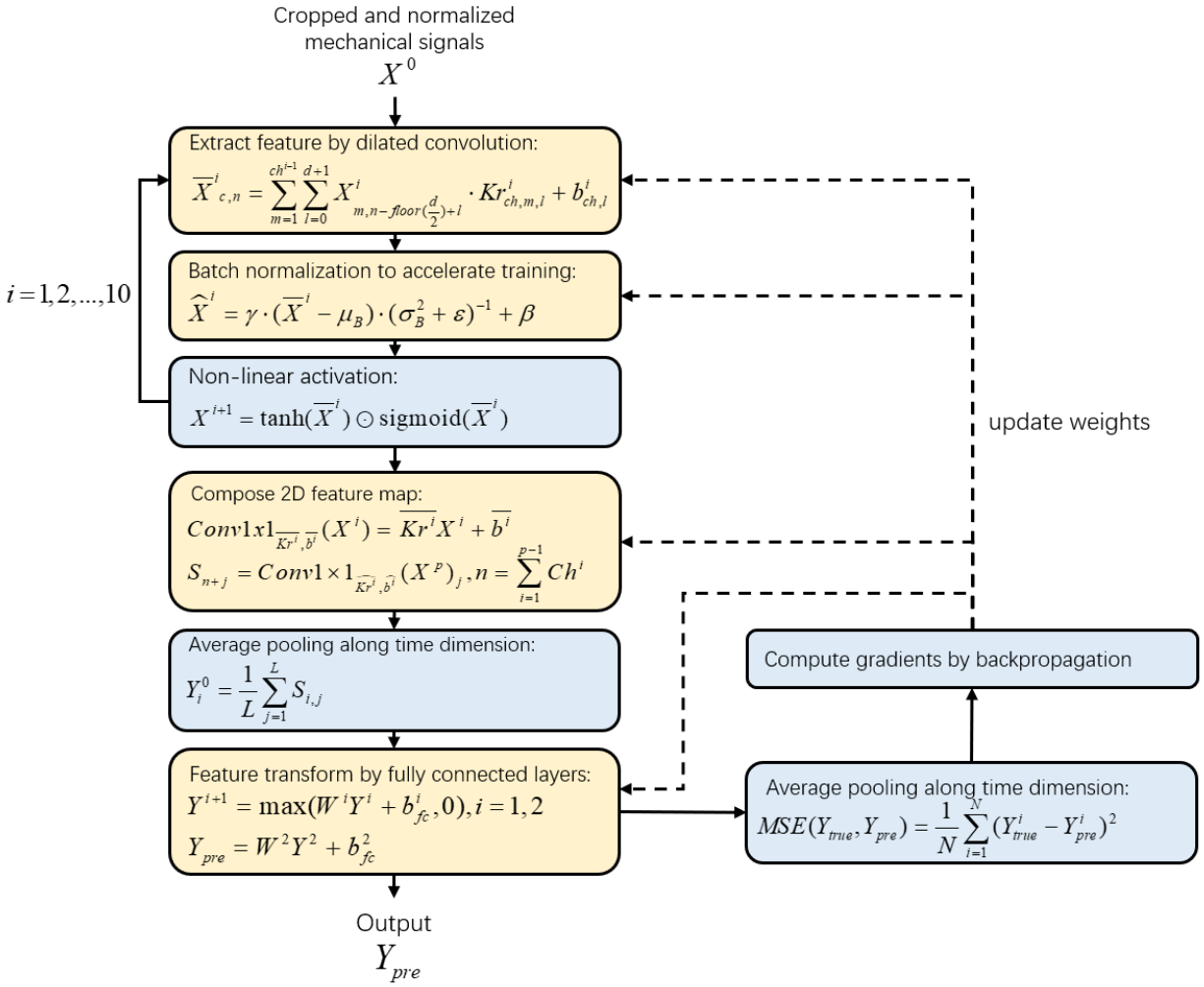

Fig. 4 The complete method pipeline.

3 Numerical simulation

3.1 Multilayer building with Nonlinear Stiffness

A multilayer building with nonlinear stiffness is analyzed to verify the proposed method. The linear structure parameter is used to establish a finite element model based on Ref.[29]. The nonlinear response can be calculated under different nonlinear stiffnesses, as shown in **Fig. 1**.

MLP and CNN are adopted to compare the proposed WaveNet as well as to investigate the advantages of WaveNet. Additionally, the robustness of the proposed method is evaluated by adding different levels of noise to response signals. For the simulation, the details are as follows: 3200 raw samples were generated for training, 320 samples for validation, and 450 samples for testing. Each raw sample contained $f = A_1\sin(20\times t)$, where A_1 was randomly generated from a uniform distribution [10, 20]. Nonlinear stiffness $k_2z^2 + k_3z^3 + k_4z^4$ is randomly generated for $k_2 \in [10^3, 10^4]$, $k_3 \in [10^4, 10^6]$, $k_4 \in [10^5, 10^7]$, which is added to DoF 5 in **Fig. 1**. The raw samples were preprocessed through normalization,

and the number of samples was increased by fivefold through random cropping. Care was taken to avoid overlap between the training, validation, and test sets during the entire process.

3.1.1 Neural network model architectures

Hyper-parameter search was performed for each architecture. **Table 1** illustrates representative models that achieved competitive results within their respective architectures. The focus of the hyper-parameter search was the depth and width of the model, as characterized by the number of layers and hidden units/channels within each layer that is summarized by the number of trainable parameters, respectively. For example, the MLP1 model has 7 fully connected layers and 44.0 million trainable parameters. CNNs and WaveNets in addiction employ convolution and dilated convolution layers, respectively. Compared with MLP, fewer number of trainable parameters are needed for building models of the same width and depth. Note that, other operators such as batch normalization, activations are not described in **Table 1** for simplicity. Further detailed hyper-parameter configurations are available in **Table 4**, **Appendix**.

Table 1 Models architectures of different neural networks.

Architecture	Name	Layer	Trainable parameters
Fully connected neural networks	MLP1	6 fully connected layers	44.9M
	MLP2	7 fully connected layers	48.9M
	MLP3	7 fully connected layers	93.0M
	MLP4	8 fully connected layers	50.0M
Convolutional neural network	CNN1	6 Convolutions 2 fully connected layers	470,052
	CNN2	7 Convolutions 2 fully connected layers	519,588
	CNN3	8 Convolutions 2 fully connected layers	532,068
WaveNet	WaveNet1	10 dilated Convolutions 2 fully connected layers	166,652
	WaveNet2	10 dilated Convolutions 2 fully connected layers	419,536
	WaveNet3	10 dilated Convolutions 2 fully connected layers	1,388,638

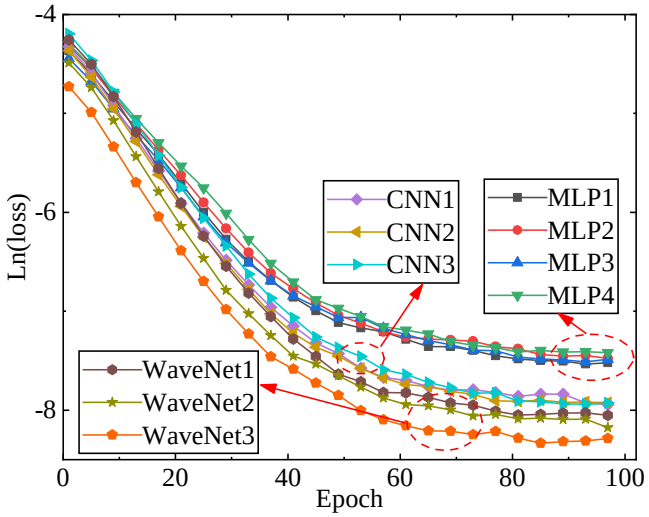

Fig. 5 The Ln(loss) of different training models.

All models were trained using the Adam optimizer until convergence. The batch size for WaveNet models was 32, the initial learning rate was 5e-05, and the total training epoch was 100. The baseline models were trained for 200 epochs with a batch size of 64 and an initial learning rate of 1e-04. The smoothed validation loss curves for the first 100 epochs are shown in **Fig. 5** and indicate that all models converged well.

3.1.2 Identification results and comparisons

In this study, since the training data had a noise addition ratio of 5%, the accuracy of the models can be examined with the same noise addition ratio, as illustrated in the second last row of Table 2. Results showed that the WaveNet models had the lowest error and are therefore the most accurate, followed by the CNN models, which had errors approximately 50% higher than the WaveNet models. The results made by the RNN models were not better than random guess and therefore will not be discussed further in the later content.

Table 2 Mean square errors of different models trained on 5% noise data.

Noise	MLP1	MLP2	MLP3	CNN1	CNN2	CNN3	WaveNet1	WaveNet2	WaveNet3
0%	0.00099	0.00103	0.00124	0.00128	0.00036	0.00026	0.00027	**0.00017**	0.00024
1%	0.00064	0.00058	0.00078	0.00083	0.00034	0.00023	0.00025	**0.00016**	0.00021
2%	0.00042	0.00038	0.00048	0.00054	0.00031	0.00023	0.00024	**0.00016**	0.00019
5%	0.00039	0.00032	0.00036	0.00041	0.00029	0.00023	0.00025	0.00019	0.00016
10%	0.00105	0.00087	0.00104	0.00108	0.00056	0.00049	0.00044	0.00036	**0.00025**

The robustness of the model can be assessed by evaluating its performance under different levels of noise. Note that, it is natural that all models performed worse with less noisy data, as they are trained on data with a 5% noise addition. In comparison, the MLP model displayed weak anti-noise capabilities in the absence of noise and in the presence of high levels of noise (10%), with a prediction error more than 200% higher than that of the 5% noise condition. In contrast, the CNN and WaveNet models did not show a significant increase in error with low levels of noise. For more noisy data, the error is increased by about 100%. For each baseline architecture, we find the largest

model where further increasement in size does not further reduce error. Such models, namely MLP2 and CNN2, will be used to represent its architecture in subsequent tests. In case of WaveNets, the model with the best balance between accuracy and robustness is selected, namely WaveNet2.

Fig. 6 presents four representative samples from the test set that exhibit different characteristics of elastic curves. In the simplest case, shown in Sample (a), where the elastic curve is approximately an odd function, all three architectures provide a good approximation of the curve with only slight differences in accuracy; In Sample (b), which is more asymmetrical, with the elastic force doubling in different directions, the two convolution-based models are still accurate, but the MLP model performs poorly; Sample (c) is a special case where there is only obvious elasticity for positive displacement. All three architectures fit the general trend well, but do not capture the nonlinear increase in elasticity that occurs when displacement increases to a certain extent in the negative direction. It is also observed that the performance of all architectures deteriorates for very small elasticities; In Sample (d), the elastic curve is an approximate even function (i.e., representing an inverse elastic system) with extremely small function values, about 0.05 at maximum after normalization. For this Sample, neither the CNN nor the MLP accurately captured the trend, with a relative error of about 400% for positive displacement. While the WaveNet performs better for positive displacement, it still overestimates the elasticity in the negative range.

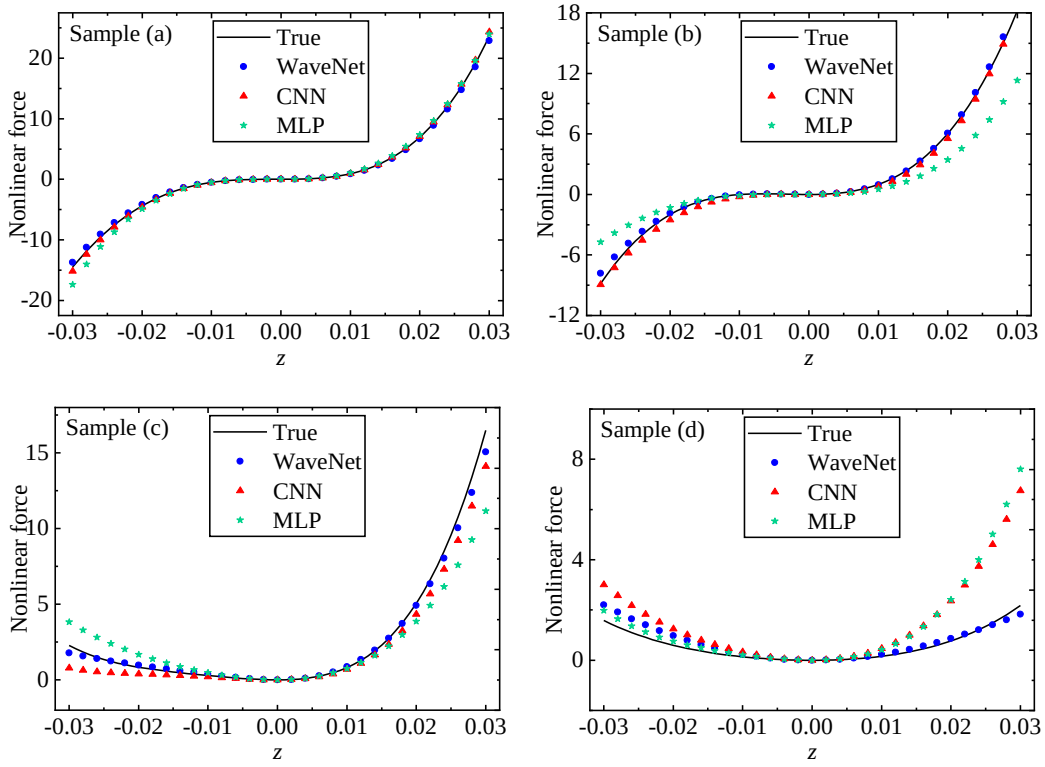

Fig. 6 The nonlinear elastic force curve prediction based on different models.

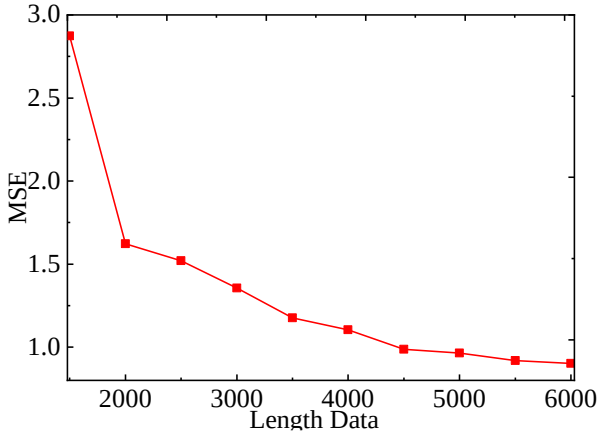

Fig. 7 Test error with respect to different input length.

Another advantage of the WaveNet models is the adaptability to input data of arbitrary length. As shown in **Fig. 7**, the mean squared error decreases while the input length increases. However, the baseline models that only process data of a fixed length, require that excessively long input be cropped, resulting in a waste of data. And data that is too short cannot be processed in any way.

3.2 Multiple-Degree-of-Freedom System with Nonlinear stiffness and damping

To determine the method's general applicability, we applied the same architecture to a problem that reconstructs the mixture model of a friction and elasticity curve in **Fig. 8**. Simultaneous prediction for multiple targets is achieved through hard parameter sharing in multi-task learning [30]. The parameters of the simulation are shown in Table 3. The structure has both cubic nonlinear and Coulomb friction elements.

The dataset size and preprocessing methods are the same as described in 3.1. In each raw sample, the system is excited by a zero-mean Gaussian random force f_6 at DOF 6 only, whose root-mean-value is 30N.Nonlinear stiffness $k_{14}{}^n\times(z_1\text{-}z_4)^3$ is randomly generated for $k_n{}^3\in[10^6, 10^8]$, and nonlinear damping $\alpha_n = \alpha_1\times\text{sign}(\dot{z})$ is randomly generated for $\alpha_1\in[1, 10]$. The equation of motion can be expressed by:

$$\begin{cases} m_1\ddot{z}_1+(c_1+c_2+c_{14})\dot{z}_1-c_2\dot{z}_2-c_{14}\dot{z}_4+(k_1+k_2)z_1-k_2z_2+k_{14}(z_1-z_{14})+k_{14}^n(z_1-z_4)^3=0 \\ m_2\ddot{z}_2+(c_2+c_3+c_{25})\dot{z}_2-c_2\dot{z}_1-c_3\dot{z}_3-c_{25}\dot{z}_5+(k_2+k_3+k_{25})z_2-k_2z_1-k_3z_3-k_{25}z_5=0 \\ m_3\ddot{z}_3+(c_3+c_{30}+c_{36})\dot{z}_3-c_3\dot{z}_2-c_{36}\dot{z}_6+(k_3+k_{30}+k_{36})z_3-k_3z_2-k_{36}z_6=0 \\ m_4\ddot{z}_4+(c_4+c_5+c_{14})\dot{z}_4-c_{14}\dot{z}_1-c_5\dot{z}_5+(k_4+k_5)z_4-k_5z_5-k_{14}(z_1-z_4)-k_{14}^n(z_1-z_4)^3=0 \\ m_5\ddot{z}_5+(c_5+c_6+c_{25})\dot{z}_5-c_{25}\dot{z}_2-c_5\dot{z}_4-c_6\dot{z}_6+(k_5+k_6+k_{25})z_5-k_{25}z_2-k_5z_4-k_6z_6=0 \\ m_6\ddot{z}_6+(c_6+c_{60}+c_{36})\dot{z}_6+\alpha_1 sign(\dot{z}_6)-c_{36}\dot{z}_3-c_6\dot{z}_5+(k_6+k_{60}+k_{36})z_6-k_{36}z_3-k_6z_5=f_6 \end{cases} \tag{16}$$

In the case of WaveNet models, the number of final MLP is increased to match the number of targets. This method is based on the idea that features needed for reconstructing the friction and elasticity curve overlap, meaning that learning one task can benefit the other. In practice, the size of the MLPs may also need to be increased as necessary.

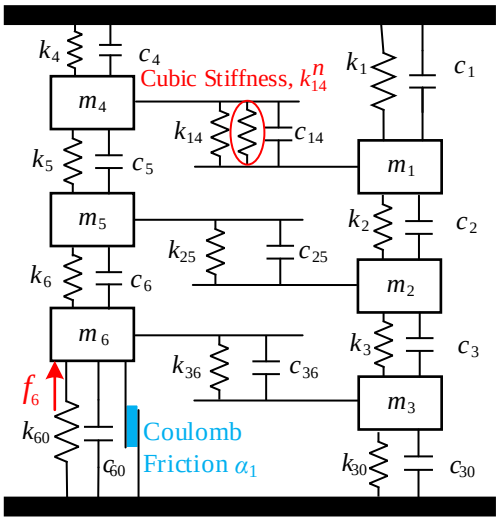

Fig. 8 Multiple-Degree-of-Freedom System with Nonlinear stiffness and damping.

Table 3 System parameters of six degrees of freedom with cubic stiffness and Coulomb friction.

Mass (kg)	Linear stiffness (N/m)	Damping (Ns/m)	Nonlinear stiffness	Nonlinear damping
m_1=m_3=m_5=1	k_1= k_4=6000	c_i=0.2 (i=1,2,3,4,5,6)	k_{14}^n N/m^3	α_1
m_2=m_4=m_6=0.5	k_2= k_3=2000	c_{14}= c_{25}= c_{36}=0.1		
	k_5=4000 k_6=5000	c_{30}= c_{60} =0.05		
	k_{14}= k_{25}= k_{36}=1000			
	k_{30}= k_{60} =1000			

Fig. 9 shows four samples from the test set that exhibit nonlinear stiffness and friction. Results show that: the proposed method based on WaveNet provide a good approximation of the curve with only slight differences in accuracy for nonlinear elastic force and dissipative force curves.

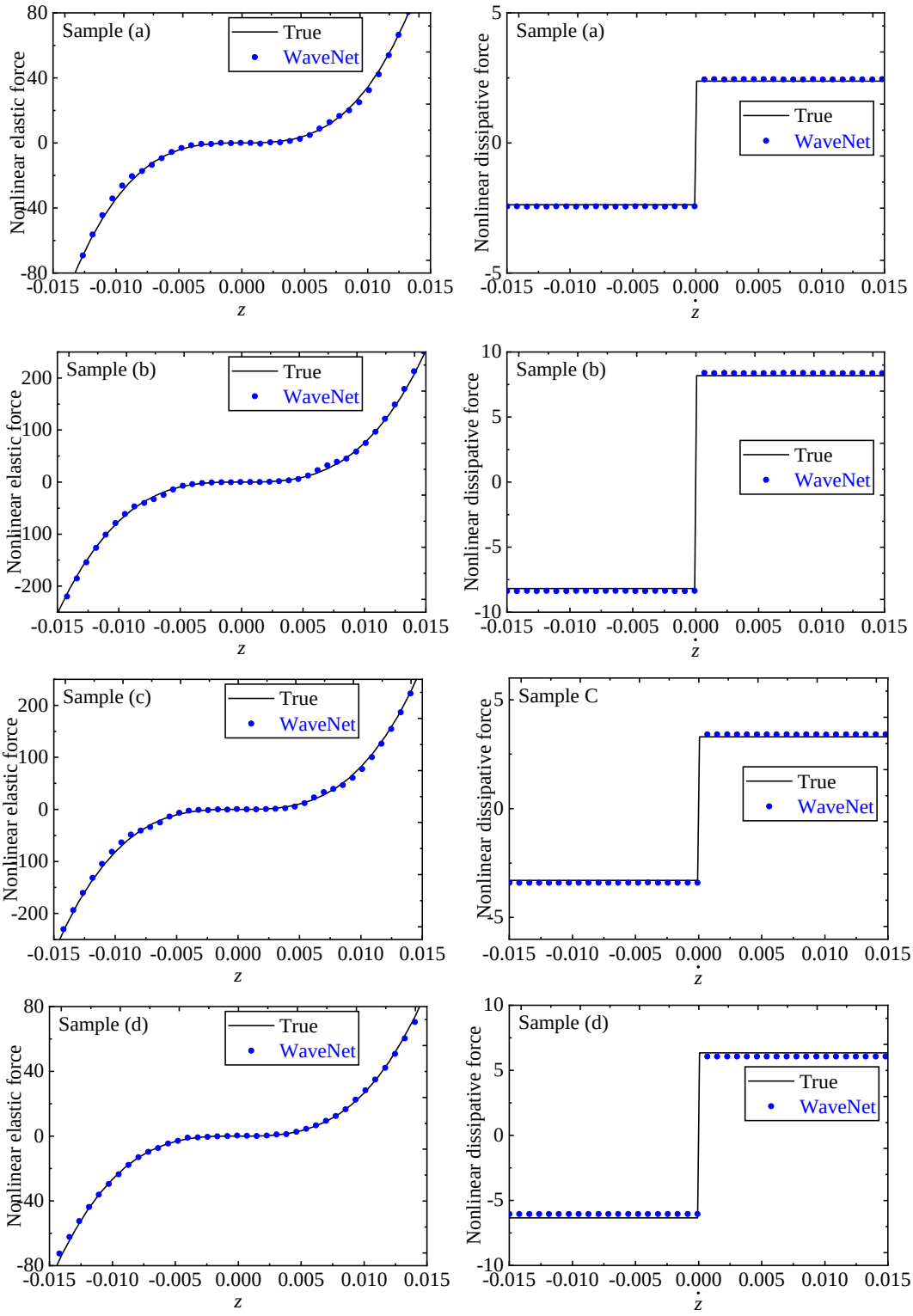

Fig. 9 The nonlinear elastic and dissipative force curve prediction for different samples with 5% noise.

Even in the case of noise, the method still has good accuracy and good robustness. Taking Sample (a) as an example, the real value of the response at 6 node and the reconstructed response signal of the identified parameters are given in **Fig. 10**. Results demonstrated that the estimated nonlinear response curve is in close agreement with the true value curve.

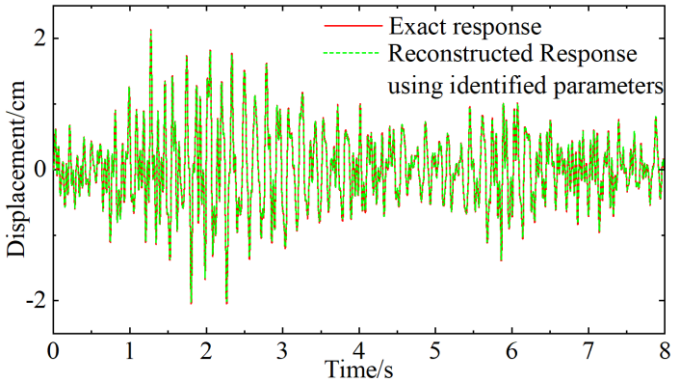

Fig. 10 Comparison of displacement of Dof 6 for sample (a) with 5% noise.

Further, the corresponding absolute error of identified parameter are shown in **Fig. 11**. Four samples are presented. When the noise is 5%, the nonlinear parameter identification error of the system is less than 5%, which verifies the effectiveness of the method.

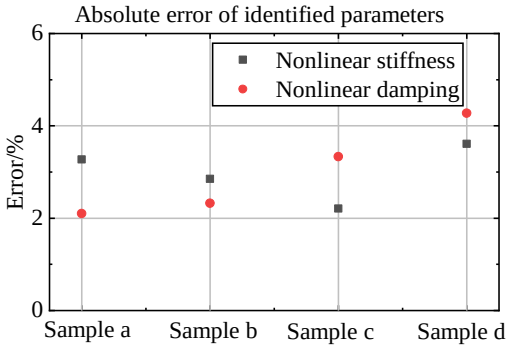

Fig. 11 The absolute error of identified parameters with 5% noise based on WaveNet model.

4 Conclusion

In this paper, we propose a deep learning-based method for system identification to build an inverse model from response signals to system constants. Deep neural networks take the response signal as input and output discrete sampling points of the function curve of the physical constant versus displacement, which can be converted into a continuous function through regression. We compare the proposed WaveNet architecture to several baseline architectures in predicting nonlinear elastic curves and demonstrate its advantages in terms of accuracy, robustness, and model size were demonstrated. The same model successfully predicted the damping and elastic forces of a more complex hybrid system, demonstrating the generalizability of the method. Promising future work includes exploring transfer learning to share the low-level features across different systems different systems and studying the use of an encoder-decoder architecture to learn a latent representation of the system by compressing and reconstructing the mechanical signal.

Acknowledgements

Weixuan Yuan, Rui Zhu and Tao xiang contributed equally to this work. This research work is supported by National Science Foundation for Distinguished Young Scholars (52125209).

Appendix

*(*Ch*, *n*) indicates *n* consecutive blocks with convolutional layers of the same channel size *Ch*

Table 4 Models architectures.

Categories	MLP				CNN			WaveNet		
Name	MLP1	MLP2	MLP3	MLP4	CNN1	CNN2	CNN3	WaveNet1	WaveNet2	WaveNet3
Input	1D flattened data of size (12000)				2D time series of size (2000, 6)			2D time series of size (2000, 6)		
Hidden layers	Hidden layers				Convolution channels			Convolution channels		
	3000	3000	5000	3000	(8, 1)*	(8, 1)	(8, 1)	(8, 2)	(16, 2)	(32, 2)
	2000	2000	3000	2000	(16, 1)	(16, 1)	(16, 1)	(16, 3)	(32, 3)	(64, 3)
	1000	2000	3000	2000	(32, 2)	(32, 2)	(32, 2)	(32, 5)	(64, 5)	(128, 5)
	500	1000	2000	1000	(64, 1)	(64, 1)	(64, 2)			
	400	500	1000	1000	(128, 1)	(128, 2)	(128, 2)			
	300	400	600	500	MLP hidden layers			MLP hidden layers		
		300	400	400	500	500	500	300	500	1000
				300	400	400	400	200	300	500
	1D flattened data of size (256)									
Model size	44.9M	48.9M	93.0M	50.0M	470052	519588	532068	166652	419536	1388468

References

[1] Moore K J , Mojahed A , Bergman L A , et al. “Local Nonlinear Stores Induce Global Dynamical Effects in an Experimental Model Plane,” *AIAA Journal*, Vol. 57, No. 11, 2019, pp: 4953-4965. https://doi.org/10.2514/1.J058311

[2] C. Soize, E. Capiez-Lernout, and R. Ohayon, “Robust updating of uncertain computational models using experimental modal analysis,” *AIAA Journal*, Vol.46, No. 11, 2018, pp: 2955-2965. https://doi.org/10.2514/1.38115

[3] B.H.K. Lee, L. Liu, K.W. Chung, “Airfoil motion in subsonic flow with strong cubic nonlinear restoring forces,” *Journal of Sound and Vibration*, Vol. 281, No. 3, 2005, pp: 699–717. https://doi.org/10.1016/j.jsv.2004.01.034

[4] Zhu, R., Marchesiello, S., Anastasio, D., Jiang, D., and Fei, Q., “Nonlinear system identification of a double-well Duffing oscillator with position-dependent friction,” *Nonlinear Dynamics*, 2022, Vol. 108 pp: 2993-3008. https://10.1007/s11071-022-07346-1

[5] Hammar K, Djamah T, Bettayeb M., “Nonlinear system identification using fractional Hammerstein–Wiener models,”. *Nonlinear Dynamics*, Vol. 98, No. 3, 2019, pp: 2327-2338. https://10.1007/s11071-019-05331-9

[6] Lei Y, Xia D, Erazo K, Nagarajaiah S. “A novel unscented Kalman filter for recursive state-input-system identification of nonlinear systems”. *Mechanical Systems and Signal Processing*, Vol. 127, 2019, pp: 120-135. https://doi.org/10.1016/j.ymssp.2019.03.013

[7] Li M, Liu X, Ding F. “The filtering-based maximum likelihood iterative estimation algorithms for a special class of nonlinear systems with autoregressive moving average noise using the hierarchical identification principle,” *International Journal of Adaptive Control and Signal Processing*, Vol. 33, No. 7, 2019, pp: 1189-1211. https://doi.org/10.1002/acs.3029

[8] Chen H, Chen Y, Yang L. “Intelligent early structural health prognosis with nonlinear system identification for RFID signal analysis,” Computer Communications, Vol. 157, 2020, pp:150-161. https://doi.org/10.1016/j.comcom.2020.04.026

[9] Kerschen, G., Worden K, Vakakis AF, and Golinval JC., “Past, present and future of nonlinear system identification in structural dynamics,” *Mechanical Systems and Signal Processing*, Vol. 20, No. 3, 2006, pp: 505-592. https://doi.org/10.1016/j.ymssp.2005.04.008

[10] Noël, J.P. and G. Kerschen, “Nonlinear system identification in structural dynamics: 10 more years of progress,” *Mechanical Systems and Signal Processing*, Vol. 93, 2017, pp: 2-35. https://doi.org/10.1016/j.ymssp.2016.07.020

[11] Marchesiello, S. and L. Garibaldi, “A time domain approach for identifying nonlinear vibrating structures by subspace methods,” *Mechanical Systems and Signal Processing*, Vol. 22, No. 1, 2008, pp: 81-101. https://doi.org/10.1016/j.ymssp.2007.04.002

[12] Rui Zhu, Dong Jiang, Qingguo Fei, S. Marchesiello, and D. Anastasio. “Identification of nonlinear stiffness and damping parameters using a hybrid approach,” *AIAA Journal*, Vol. 59, No. 11, 2021, pp: 4686-4695. https://doi.org/10.2514/1.J060461

[13] Zhu, R., Fei, Q., Jiang, D., Marchesiello, S., and Anastasio, D., “Bayesian model selection in nonlinear subspace identification,” *AIAA Journal*, Vol. 60, No. 1, 2022, pp: 92-101. https://doi.org/10.2514/1.J060782

[14] Zhu R, Jiang D, Marchesiello S, Anastasio D, Zhang D, Fei Q., "Automatic Nonlinear Subspace Identification Using Clustering Judgment Based on Similarity Filtering," *AIAA Journal*, 2023, pp: 1-9. https://doi.org/10.2514/1.J062816

[15] Wu R T, Jahanshahi M R. "Deep convolutional neural network for structural dynamic response estimation and system identification," *Journal of Engineering Mechanics*, Vol. 145, No. 1, 2019, pp: 04018125. https://doi.org/10.1061/(ASCE)EM.1943-7889.0001556

[16] Yang H, Jiang J, Chen G, Zhao J. "Dynamic load identification based on deep convolution neural network. Mechanical Systems and Signal Processing," Vol. 185, 2023, pp: 109757. https://doi.org/10.1016/j.ymssp.2022.109757

[17] Brunton S L, Proctor J L, Kutz J N. "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the national academy of sciences*, Vol. 113, No. 15, 2016, pp: 3932-3937. https://doi.org/10.1073/pnas.1517384113

[18] Hung C W, Mao W L, Huang H Y. "Modified PSO Algorithm on Recurrent Fuzzy Neural Network for System Identification," *Intelligent Automation & Soft Computing*, Vol. 25, No. 2, 2019, pp: 329-341.

[19] Yu W, Gonzalez J, Li X. "Fast training of deep LSTM networks with guaranteed stability for nonlinear system modeling," *Neurocomputing*, Vol. 422, 2021, pp: 85-94. https://doi.org/10.1016/j.neucom.2020.09.030

[20] Mannarino A, Mantegazza P. "Nonlinear aeroelastic reduced order modeling by recurrent neural networks. Journal of Fluids and Structures," Vol. 48, 2014, pp: 103-121. https://doi.org/10.1016/j.jfluidstructs.2014.02.016

[21] Aaron O, Sander D, Heiga Z, Karen S, Oriol V, Alex G, Nal K, Andrew S, Koray K. "WaveNet: A generative model for raw audio," arXiv preprint arXiv:1609.03499, 2016. https://doi.org/10.48550/arXiv.1609.03499

[22] Santurkar Shibani, Dimitris Tsipras, Andrew Ilyas, Aleksander Madry. "How does batch normalization help optimization?" A*dvances in neural information processing systems* Vol. 31, 2018.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2016, pp: 770-778

[24] Boureau, Y-Lan, Jean Ponce, and Yann LeCun. "A theoretical analysis of feature pooling in visual recognition," *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp: 111-118.

[25] Gardner, Matt W., and S. R. Dorling. "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences," *Atmospheric environment*, Vol. 32, No. 14-15, 1998, pp: 2627-2636. https://doi.org/10.1016/S1352-2310(97)00447-0

[26] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, Vol. 15, No. 1, 2014, pp: 1929-1958.

[27] Boyd, Stephen, Lin Xiao, and Almir Mutapcic. "Subgradient methods," lecture notes of EE392o, Stanford University, Autumn Quarter 2004 (2003): 2004-2005.

[28] Staudemeyer, Ralf C., and Eric Rothstein Morris. "Understanding LSTM--a tutorial into long short-term memory recurrent neural networks," arXiv preprint arXiv:1909.09586 (2019). https://doi.org/10.48550/arXiv.1909.09586

[29] Marchesiello S, Fasana A, Garibaldi L. "Modal contributions and effects of spurious poles in nonlinear subspace identification," *Mechanical Systems and Signal Processing*, Vol. 74, 2016, pp: 111-132. https://doi.org/10.1016/j.ymssp.2015.05.008

[30] Ruder, Sebastian. “An overview of multi-task learning in deep neural networks,” arXiv preprint arXiv:1706.05098 (2017). https://doi.org/10.48550/arXiv.1706.05098