Chapter 15

LSTM – NARX as Emulator for Finite Element Models of Nonlinear Structural Dynamic Systems: From Time-Series Input Vectors to Time-Series Output Vectors

Abdoul Aziz Sandotin Coulibaly, Zhen Hu, and Joel P. Conte

Abstract Performance-based seismic design of structural systems relies on computationally expensive nonlinear time history analyses (NLTHAs) of high-fidelity finite elements (FE) models for predicting structural response to seismic excitation. For risk-based assessment, these FE simulations must be run thousands of times across different realizations of the uncertainty sources, making the computational cost significant and often prohibitive. Even if NLTHAs are reliable and straightforward for assessing civil infrastructure systems, the associated computational cost is too substantial, if not prohibitive, to be disregarded. Consequently, data-driven machine learning surrogate models have gained prominence as fast emulators in replacement of computationally expensive high-fidelity models of engineering, bio-engineering, and biological systems. This paper presents a study of the Long Short-Term Memory (LSTM) networks as global surrogate models for nonlinear inelastic structural dynamic systems, taking a three-dimensional (3D) virtual yet realistic 5-story, 2-by-1 bay reinforced concrete (RC) frame structure subjected to bidirectional horizontal seismic excitation as application example. For high-dimensional, realistic FE structural models with complex nonlinear dynamics, standalone Long Short-Term Memory (LSTM) networks often face challenges in accurately capturing the full range of nonlinear structural behavior. A central approach of our model is a novel approach for selecting earthquake ground motion records with diverse key characteristics to form the input-output pairs for the training set. This method is designed to maximize the accuracy and reliability of predictions from the calibrated surrogate models. This is achieved by combining (i) a convolutional autoencoder (CAE) and (ii) an LTSM network employing the nonlinear autoregressive model with exogenous input (NARX) formulation.

Keywords Nonlinear Inelastic Dynamic Systems · Material nonlinearity · LSTM · Convolutional AutoEncoder · NARX

Introduction

In 2023, a series of record-breaking earthquakes, with intensities ranging from very strong to violent, caused major disruptions in daily life [1–3]. Consequently, there is an urgent operational need to develop risk-based, performance-based seismic design codes to assess and mitigate seismic risks to large-scale civil infrastructure systems and the built environment. Undoubtedly, the use of well-calibrated or even experimentally validated physics/mechanics-based FE models coupled with NLTHAs provides a valuable tool for assessing seismic performance and seismic risk. In risk assessment, even with an accurate and efficient FE analysis method, the computational time remains substantial due to the need for multiple model simulations across varied scenarios. This is compounded by uncertainties arising from (1) uncertainty in the ground motion intensity at the local site, (2) ground motion record-to-record variability, and (3) structural modeling uncertainties, all of which increase the computational demands by orders of magnitude.

To address these demands, data-driven surrogate models serve as fast emulators or meta-models, approximation of the dynamic behavior of structural systems—be it a bridge, or a high-rise building—using either simulated or real-world

Abdoul Aziz Sandotin Coulibaly · Joel P. Conte
Department of Structural Engineering, University of California, San Diego, CA, USA
e-mail: acoulibaly@ucsd.edu; jpconte@ucsd.edu

Zhen Hu
Department of Industrial and Manufacturing Systems Engineering, University of Michigan-Dearborn, Dearborn, MI, USA
e-mail: zhennhu@umich.edu

Matthew Whelan et al. (eds.), *Dynamics of Civil Structures, Vol. 2*, Conference Proceedings
of the Society for Experimental Mechanics Series, https://doi.org/10.13052/97887-438-0147-4_15

recorded data. Recurrent neural networks (RNNs), distinguished by the directional flow of information through their layers, are specifically designed to handle sequential data, such as time series commonly encountered in fields like structural engineering (e.g., input earthquake excitation and structural response time histories) [4]. LSTM networks have emerged as a more sophisticated and powerful variant of RNNs, with LSTM-based surrogate models documented in the literature for modeling nonlinear dynamic systems [5–11]. The predictive capability of these LSTM surrogate models primarily focuses on sequence-to-sequence (Seq-to-Seq) mapping, necessitating that all pairs of input ground motion and corresponding output response time histories have the same length. Furthermore, the current literature on surrogate modeling of dynamic systems is mostly limited to simplistic mathematical models (e.g., spring-mass-dashpot models) or basic planar nonlinear dynamic frame models of structures, which are often subjected to unidirectional earthquake ground motions. Therefore, it is often unrealistic and lacks comprehensiveness. In this paper, we explore the use of LSTM for surrogate modeling of a 3D RC concrete frame structure subjected to bidirectional earthquake excitation. This RC structure is modeled to account for nonlinear material behavior (through distributed plasticity) and nonlinear geometry.

Application Example: 3D Reinforced Concrete (RC) Frame Structure

Our numerical example consists of a 3D, 5-story, 2-by-1 bay RC frame structure subjected to bidirectional horizontal seismic excitations (see Fig. 1) [12]. The structure is designed for a location in downtown Seattle, in the state of Washington, with Site Class D soil conditions, a short-period spectral acceleration $\mathrm{S_{MS}} = 1.37$ g and a one-second spectral acceleration $\mathrm{S_{M1}} = 0.53$ g. The building has plan dimensions of 10 m × 6 m in the longitudinal and transverse directions, respectively, and a floor-to-floor height of 4.0 m, resulting in a total height of 20 m. The dead and live loads and corresponding seismic masses associated with the structure are computed according to the 2012 International Building Code [13].

Fig. 1 shows the overall geometry of the structure with the dimensions and reinforcement details of the beam and columns cross-sections in both longitudinal and transverse directions.

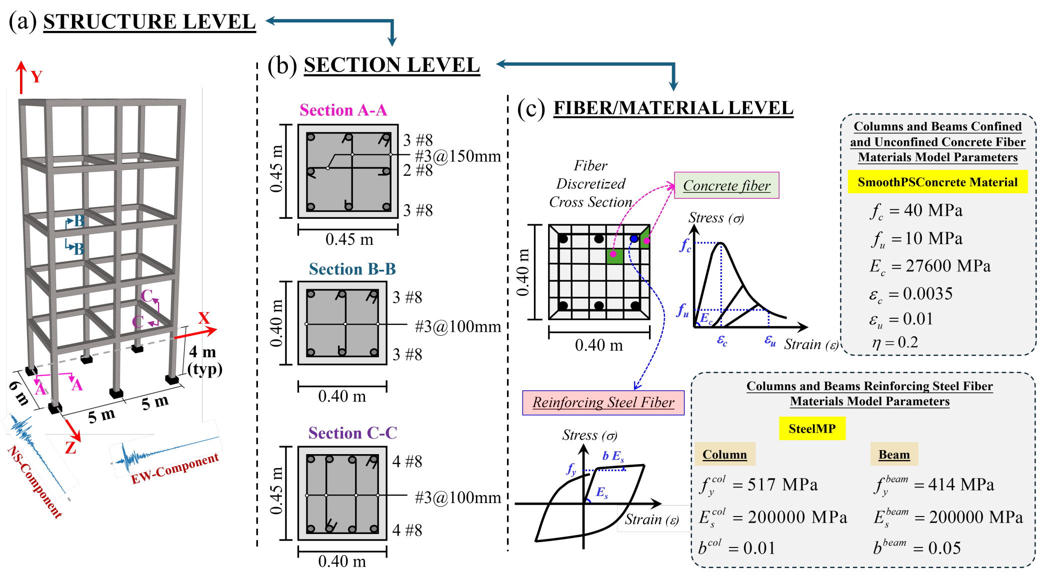

Fig. 1 3D RC frame building: (a) structure level overview, (b) cross-sections of beams and columns, (c) fiber or material level (material model parameters).

The seismic response of the structure is simulated using a mechanics-based nonlinear FE model with displacement-based fiber-section beam-column elements developed in the OpenSees structural analysis framework. The cross-sections of all beams and columns are discretized into longitudinal fibers. At the fiber/ material level, realistic uniaxial material constitutive models are used to represent the steel and concrete materials. Linear elastic shear and torsion models are combined with, but remain uncoupled from, the inelastic axial-flexural behavior at the section level. The effects of nonlinear geometry are accounted for using the corotational formulation for both beams and columns. Other sources of energy dissipation, distinct from the hysteretic energy dissipation due to the nonlinear material behavior, are modeled using mass ($\mathbf{M}$) and tangent-stiffness ($\mathbf{K}_T$) proportional Rayleigh damping, based on the tangent stiffness matrix at the last converged time step of analysis. A critical damping ratio of 2% is assigned to the first and second vibration modes (obtained using the tangent stiffness matrix after the application of the gravity loads) of the structure in the longitudinal direction. The natural periods of these modes are $\mathrm{T}_1 = 1.95$ sec and $\mathrm{T}_2 = 0.62$ sec, respectively.

LSTM: Long Short-Term Memory Networks As Emulators Of Nonlinear Structural Dynamic Systems

LSTM neural networks represent a more advanced and powerful variant of RNNs, capable of overcoming the inherent limitations of conventional RNNs in capturing long-term dependencies, which are constrained to approximately 10 time steps, due to vanishing and exploding gradients encountered during the training process [14]. Unlike vanilla RNNs, LSTM can handle up to 1,000 time steps, a capability enabled by LSTM cells that effectively control error flow through internal states [15].

Assume the LSTM model of the 3D RC nonlinear FE model is trained to establish a relationship between two bidirectional earthquake ground-motion input time histories, denoted as $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$, and the corresponding two output relative displacement time histories at the 5^{th} floor of the structure, $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(2)}$. These two vector pairs are denoted as $\chi = \left\{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}\right\}$ for the inputs and $\psi = \left\{\mathbf{Y}^{(1)}, \mathbf{Y}^{(2)}\right\}$ for the outputs. A full-blown version of the dataset looks like:

$$
\begin{aligned}
\boldsymbol{X}^{(1)} &= \left[\ddot{\mathbf{u}}^{g}_{1,1:C}, \ddot{\mathbf{u}}^{g}_{2,1:C}, \ddot{\mathbf{u}}^{g}_{3,1:C}, \cdots, \ddot{\mathbf{u}}^{g}_{\tau,1:C}, \cdots, \ddot{\mathbf{u}}^{g}_{T_1,1:C}\right]^{(1)} \quad & \mathbf{Y}^{(1)} &= \left[\mathbf{u}^{rel}_{1,1:C'}, \mathbf{u}^{rel}_{2,1:C'}, \mathbf{u}^{rel}_{3,1:C'}, \cdots, \mathbf{u}^{rel}_{\tau,1:C'}, \cdots, \mathbf{u}^{rel}_{T_1,1:C'}\right]^{(1)} \\
\boldsymbol{X}^{(2)} &= \left[\ddot{\mathbf{u}}^{g}_{1,1:C}, \ddot{\mathbf{u}}^{g}_{2,1:C}, \ddot{\mathbf{u}}^{g}_{3,1:C}, \cdots, \ddot{\mathbf{u}}^{g}_{\tau,1:C}, \cdots, \ddot{\mathbf{u}}^{g}_{T_2,1:C}\right]^{(2)} \quad & \mathbf{Y}^{(2)} &= \left[\mathbf{u}^{rel}_{1,1:C'}, \mathbf{u}^{rel}_{2,1:C'}, \mathbf{u}^{rel}_{3,1:C'}, \cdots, \mathbf{u}^{rel}_{\tau,1:C'}, \cdots, \mathbf{u}^{rel}_{T_2,1:C'}\right]^{(2)}
\end{aligned}
\tag{1}
$$

where $\ddot{\mathbf{u}}^{g}_{\tau,\,1\,:\,\mathrm{C}}$ and $\mathbf{u}^{rel}_{\tau,\,1\,:\,\mathrm{C}'}$ are respectively the earthquake ground motion and relative displacement values in both directions at time step τ; T_1 and T_2 denote the total number of time steps in each vector time series; $C = C' = 2$ are the number of channels for the input excitation and output response, respectively. Next, we will delve into the inner workings of the LSTM neural network, followed by outlining the process of constructing our LSTM-NARX emulator using the dataset defined in Eq. (1).

LSTM Cell

The LSTM cell is central to the success of LSTM networks. It functions as a subnetwork with both linear and non-linear connections, utilizing memory states and gate mechanisms to manage and retain information effectively [15]. Fig. 2 shows a graphical representation of an LSTM cell.

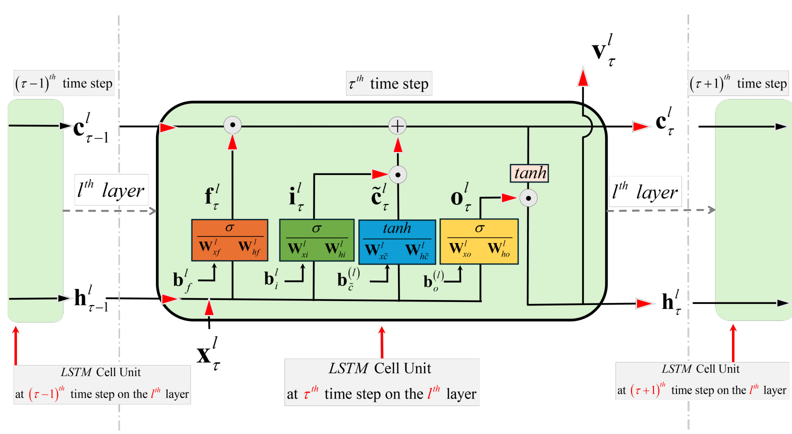

Fig. 2 Graphical representation of an LSTM Cell.

As a subnetwork, an LSTM cell consists of four interacting gates: the input gate $\mathbf{i}^l_\tau$, the forget gate $\mathbf{f}^l_\tau$, the cell state update gate $\mathbf{c}^l_\tau$, and the output gate $\mathbf{o}^l_\tau$ (see Fig. 2). At a specific time step τ and layer l, the four gates selectively retain and update the information from the previous time step $\tau - 1$, regulating the flow of information to the next time step $\tau + 1$ and layer $l+1$. In this context, an LSTM cell receives three inputs, notably: (1) an input vector $\mathbf{x}_\tau^l$, which may represent the bidirectional (C = 2) earthquake ground motion value at time step τ ($\ddot{\mathbf{u}}^g_{\tau,\,1\,:\,\mathrm{C}}$) or/and the response of the structural system at the previous time steps, (2) the hidden state output denoted as $\mathbf{h}_{\tau-1}^l$ and the cell memory state $\mathbf{c}_{\tau-1}^l$ from the previous time step $\tau-1$. Mathematical relationships between the gates and states are given as:

$$
\begin{aligned}
&(a)\ \underbrace{\mathbf{f}_\tau^l=\sigma\left(\mathbf{W}_{hf}^l\mathbf{h}_{\tau-1}^l+\mathbf{W}_{xf}^l\mathbf{x}_\tau^l+\mathbf{b}_f^l\right)}_{forget\ gate}
\quad (c)\ \underbrace{\tilde{\mathbf{c}}_\tau^l=\tanh\left(\mathbf{W}_{h\tilde{c}}^l\mathbf{h}_{\tau-1}^l+\mathbf{W}_{x\tilde{c}}^l\mathbf{x}_\tau^l+\mathbf{b}_{\tilde{c}}^l\right)}_{candidate\ cell\ state}
\quad (b)\ \underbrace{\mathbf{i}_\tau^l=\sigma\left(\mathbf{W}_{hi}^l\mathbf{h}_{\tau-1}^l+\mathbf{W}_{xi}^l\mathbf{x}_\tau^l+\mathbf{b}_i^l\right)}_{input\ gate}\\
&(e)\ \underbrace{\mathbf{o}_\tau^l=\sigma\left(\mathbf{W}_{ho}^l\mathbf{h}_{\tau-1}^l+\mathbf{W}_{xo}^l\mathbf{x}_\tau^l+\mathbf{b}_o^l\right)}_{output\ gate}
\quad (d)\ \underbrace{\mathbf{c}_\tau^l=\mathbf{f}_\tau^l\odot\mathbf{c}_{\tau-1}^l+\mathbf{i}_\tau^l\odot\tilde{\mathbf{c}}_\tau^l}_{cell\ update\ state}
\quad (f)\ \underbrace{\mathbf{h}_\tau^l=\mathbf{v}_\tau^l=\mathbf{o}_\tau^l\odot\tanh\left(\mathbf{c}_\tau^l\right)}_{hidden\ state\ output}
\end{aligned}
\tag{2}
$$

In Eq. (2), the symbol $\odot$ denotes the Hadamard product, also known as the element-wise product, $\sigma(\cdot)$ and $tanh(\cdot)$ represent the sigmoid and hyperbolic tangent activation functions, respectively [15].

An LSTM model can automatically retain or discard stored memory using these gates. The parameters of the LSTM model, denoted as $\boldsymbol{\theta}_{\mathbf{LSTM}}$, include the kernel weight matrices $\left\{\mathbf{W}_{xf}^l,\mathbf{W}_{xi}^l,\mathbf{W}_{x\tilde{c}}^l,\mathbf{W}_{xo}^l\right\}$, recurrent weight matrices $\left\{\mathbf{W}_{hf}^l,\mathbf{W}_{hi}^l,\mathbf{W}_{h\tilde{c}}^l,\mathbf{W}_{ho}^l\right\}$, and the bias vectors $\left\{\mathbf{b}_f^l,\mathbf{b}_i^l,\mathbf{b}_c^l,\mathbf{b}_o^l\right\}$ that are learned across all layers. The LSTM network architecture consists of stacked layers, with each layer increasingly learning abstract representations of the input data through successive transformations. Training the LSTM network involves minimizing the difference between the normalized predicted LSTM network outputs $\hat{\psi}_{\mathrm{NL}}$ and the normalized ground truth values ψ_{NL} for a given set of input-output$(\boldsymbol{\chi}-\boldsymbol{\psi})$data pairs, thereby optimizing all parameters $\boldsymbol{\theta}_{\mathbf{LSTM}}$. Fig. 3 illustrates a generic LSTM architecture, featuring an input layer with unnormalized input, two LSTM layers ($l=\{1\ ,\ 2\}$), and an output layer representing the unnormalized targets.

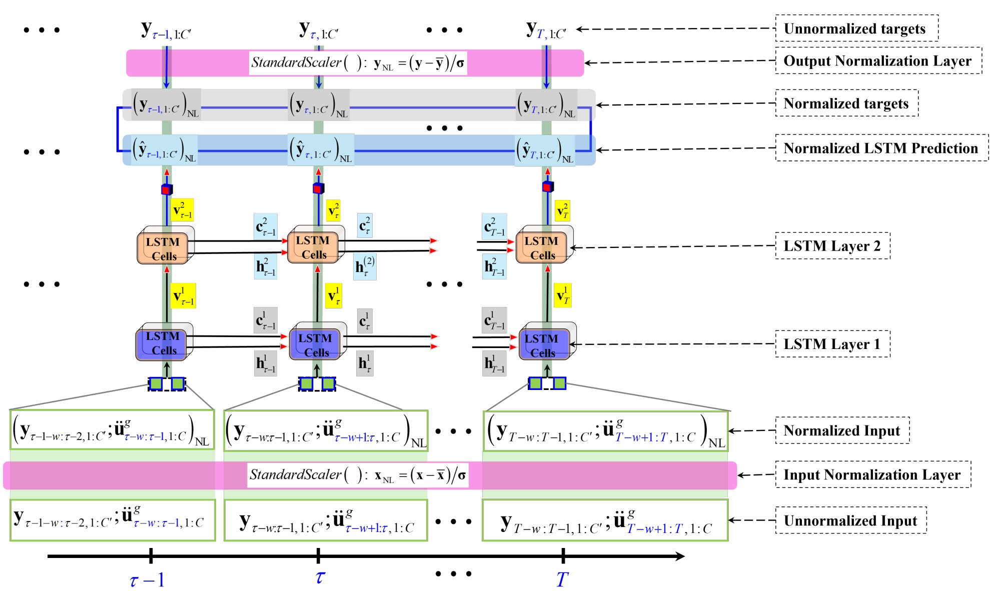

Fig. 3 Typical architecture and dataflow of LSTM Neural Networks.

LSTM-Nonlinear AutoRegressive network with eXogenous inputs Architecture (LSTM-NARX)

When considering a single input-output pair defined by $\mathbf{X}^{(1)}-\mathbf{Y}^{(1)}$, a Nonlinear AutoRegressive eXogenous (NARX) framework in time series modeling regresses the current values of a time series $\mathbf{y}_{t,\,1\,:\,\mathrm{C}'}$ on (1) previous w values of the same time series ($\mathbf{y}_{t-1,\,1\,:\,\mathrm{C}'},\mathbf{y}_{t-2,\,1\,:\,\mathrm{C}'},\mathbf{y}_{t-3,\,1\,:\,\mathrm{C}'},\cdots,\mathbf{y}_{t-w,\,1\,:\,\mathrm{C}'}$) and (2) the current ($\ddot{\mathbf{u}}^g{}_{t,\,1\,:\,\mathrm{C}}$) and past values of the exogenous input time series ($\ddot{\mathbf{u}}^g{}_{t-1,\,1\,:\,\mathrm{C}}\ ,\ddot{\mathbf{u}}^g{}_{t-2,\,1\,:\,\mathrm{C}},\ddot{\mathbf{u}}^g{}_{t-3,\,1\,:\,\mathrm{C}}\ ,\cdots,\ddot{\mathbf{u}}^g_{t-w+1,\,1\,:\,\mathrm{C}}$). The NARX model organizes the input data required for the LSTM architecture as

$$
\underbrace{\mathbf{y}_{t,\,1\,:\,\mathrm{C}'}}_{\substack{current\\ output\ sequence}} \approx \mathbf{F}_{\boldsymbol{\theta}}^{\mathbf{NARX}}\left(\begin{array}{l}\overbrace{\mathbf{y}_{t-1,\,1\,:\,\mathrm{C}'},\mathbf{y}_{t-2,\,1\,:\,\mathrm{C}'},\mathbf{y}_{t-3,\,1\,:\,\mathrm{C}'},\cdots,\mathbf{y}_{t-w,\,1\,:\,\mathrm{C}'}}^{previous\ output\ input\ sequence}, \\ \underbrace{\ddot{\mathbf{u}}^g{}_{t,\,1\,:\,\mathrm{C}}}_{\substack{current\\ EQs\ input\ sequence}}\ ,\underbrace{\ddot{\mathbf{u}}^g{}_{t-1,\,1\,:\,\mathrm{C}}\ ,\ddot{\mathbf{u}}^g{}_{t-2,\,1\,:\,\mathrm{C}},\ddot{\mathbf{u}}^g{}_{t-3,\,1\,:\,\mathrm{C}}\ ,\cdots,\ddot{\mathbf{u}}^g_{t-w+1,\,1\,:\,\mathrm{C}}}_{previous\ EQs\ input\ sequence}\ ;\ \underbrace{\boldsymbol{\theta}_{\mathbf{NARX}}}_{parameters}\end{array}\right)
\tag{3}
$$

where $F_{\theta}^{NARX}(\bullet)$ refers to the LSTM-NARX network parameterized by weights and biases $\boldsymbol{\theta}_{\mathbf{NARX}}$, which need to be learned during the training process. The parameter w enables the model to capture temporal dependencies and patterns in time series data. It is a key parameter in generating, from the input-output vector time histories, the time series sequences dataset for training of the LSTM-NARX models.

Training the LSTM networks involves performing supervised learning by optimizing the network parameters $\boldsymbol{\theta}_{\mathbf{NARX}}$. This is accomplished by minimizing the loss function, as shown in Eq. (4)a, across all training input-output sequences [16].

$$(a) \quad L\left(\boldsymbol{\theta}_{\mathbf{NARX}}\right)^{b} = \frac{1}{N_{SIB}} \cdot \frac{1}{C'} \cdot \sum_{s=1}^{N_{SIB}} \sum_{k=1}^{C'} \left[y_{\mathrm{NL}\,,\,t,\,k,s} - \hat{y}_{\mathrm{NL},\,t,\,k,s}\left(\boldsymbol{\theta}_{\mathbf{NARX}}\right)\right]^{2};$$
$$(b) \quad \boldsymbol{\theta}_{\mathbf{NARX}}{}^{b+1} = \boldsymbol{\theta}_{\mathbf{NARX}}{}^{b} - \eta \frac{\partial L\left(\boldsymbol{\theta}_{\mathbf{NARX}}\right)^{b}}{\partial \boldsymbol{\theta}_{\mathbf{NARX}}{}^{b}} \tag{4}$$

The model parameters $\boldsymbol{\theta}_{\mathbf{NARX}}$ are updated iteratively using a gradient-based optimization algorithm such as stochastic gradient descent (see Eq. (4)b) or the Adam optimizer. Here, η is the learning rate, b represents the batch number, N_{SIB} denotes the number of time series sequences within each batch. These parameters play a pivotal role in ensuring convergence to optimal weights and biases and enhancing predictive accuracy. The optimizer updates $\boldsymbol{\theta}_{\mathbf{NARX}}$ accordingly until the entire dataset has been processed multiple times, with each pass corresponding to an epoch.

Predictive Strategies of LSTM-NARX: Recursive Forecasting

The trained model is now denoted as $F_{\boldsymbol{\theta}^{*}}^{\mathbf{NARX}}(\cdot)$ where $\boldsymbol{\theta}_{\mathbf{NARX}}^{*}$(indicated by the asterisk) denotes the optimal weights and biases. Predictions at a specific time step follow the format of Eq. (3). Similar to traditional time-stepping numerical solvers, the LSTM-NARX model's outputs from the previous time steps (in this instance $\hat{\mathbf{y}}_{t-1,\,1\,:\,\mathrm{C}'}, \hat{\mathbf{y}}_{t-2,\,1\,:\,\mathrm{C}'}, \hat{\mathbf{y}}_{t-3,\,1\,:\,\mathrm{C}'}, \cdots, \hat{\mathbf{y}}_{t-w+1,\,1\,:\,\mathrm{C}'}$) are reintroduced as inputs to predict $\hat{\mathbf{y}}_{t,\,1\,:\,\mathrm{C}'}$ in Eq. (5).

$$\underbrace{\hat{\boldsymbol{y}}_{t,\,1\,:\,\mathrm{C}'}}_{\substack{\textbf{current}\\ \textit{predicted output sequence}}} \approx \underbrace{\mathbf{F}_{\boldsymbol{\theta}*}^{\mathbf{NARX}}}_{\substack{\textit{trained}\\ \mathrm{LSTM-NARX}\\ \textit{model}}} \left(\overbrace{\hat{\boldsymbol{y}}_{t-1,\,1\,:\,\mathrm{C}'}, \hat{\boldsymbol{y}}_{t-2,\,1\,:\,\mathrm{C}'}, \hat{\boldsymbol{y}}_{t-3,\,1\,:\,\mathrm{C}'}, \cdots, \mathbf{y}_{t-w,\,1\,:\,\mathrm{C}'}}^{\textbf{previous}\ \textit{output input sequence}}, \underbrace{\ddot{\mathbf{u}}^{g}{}_{t,\,1\,:\,\mathrm{C}}}_{\substack{\textit{current}\\ \textit{EQs input sequence}}}, \underbrace{\ddot{\mathbf{u}}^{g}{}_{t-1,\,1\,:\,\mathrm{C}}, \ddot{\mathbf{u}}^{g}{}_{t-2,\,1\,:\,\mathrm{C}}, \ddot{\mathbf{u}}^{g}{}_{t-3,\,1\,:\,\mathrm{C}}, \cdots, \ddot{\mathbf{u}}^{g}_{t-w+1,\,1\,:\,\mathrm{C}}}_{\textit{previous EQs input sequence}}; \underbrace{\boldsymbol{\theta}_{\mathbf{NARX}}^{*}}_{\substack{\textit{optimal}\\ \textit{parameters}}} \right) \tag{5}$$

This procedure continues for each subsequent time step within the prediction horizon, employing recursive forecasting to reflect real-world scenarios in which the future structure's responses are unknown. This approach is particularly advantageous for real-time predictions when such data are unavailable.

Selection Of Input-Output Pairs For Training Of The LSTM Surrogate Model

Regardless of the capacity of the LSTM model, training with non-representative earthquake ground motion records will constrain the surrogate model's ability to generalize to unseen events. This section focuses on selecting a diverse set of earthquake records, ranging from low to high intensities, to capture a wide range of structural responses, from quasi-linear to highly nonlinear. This approach aims to improve the training of our LSTM-NARX network.

Engineering Intensity Measures (IM) for Ground Motion Selection in LSTM Training

For our three-dimensional RC frame structure subjected to bidirectional horizontal earthquake ground motion records, specifically along the X and Z axes (see Fig. 1(a)), we combine the spectral ordinates from these two horizontal components into a single value, which is the geometric mean of their respective spectral accelerations: $S_{a-\mathrm{gm}}(\xi = 5\%\,, T_{1-\mathrm{X}}\,, T_{1-\mathrm{Z}})$ or $S_{a-\mathrm{gm}}$ for short. $S_{a-\mathrm{gm}}$ is then used as intensity measure for selecting ground motion records in LSTM training and is defined as

$$S_{a-\mathrm{gm}}(\xi = 5\%\,, T_{1-\mathrm{X}}\,, T_{1-\mathrm{Z}}) = \sqrt{S_{a-\mathrm{X}}(\xi = 5\%\,, T_{1-\mathrm{X}}) \ \times \ S_{a-\mathrm{Z}}(\xi = 5\%\,, T_{1-\mathrm{Z}})} \tag{6}$$

where $T_{1-\text{X}} = 1.95\ \text{sec}$ and $T_{1-\text{Z}} = 1.94\ \text{sec}$ denote the initial fundamental period of the building structure in the X and Z direction, respectively. In Eq. (6), $S_a\,(\xi = 5\%\,,\,T_1)$, S_a for short, is the 5% elastic spectral pseudo-acceleration at the structure's initial fundamental period in a specified direction. S_a, which is structure-dependent and application-specific, is commonly used by structural engineers to assess the severity of earthquake ground motions [17]. For our 3D case, $S_{a-\text{gm}}$ is considered a sufficient intensity measure, as it has traditionally been the preferred choice for structures subjected to two horizontal ground motions [18–20].

A database of 3,462 bidirectional earthquake ground motions was accessible from the PEER Strong Motion Database [21]. A subset of these motions was selected such that $S_{a-\text{gm}}$ falls between the lower and upper bounds of 0.1g and 1.2g. This range ensures that the structure exhibits linear and nonlinear responses, with 1.2g allowing for significant nonlinear inelastic behavior without nearing the point where the iterative time-stepping scheme (to integrate the equations of motion) fails to converge due to the proximity of physical failure. The ranges of [0.1g, 0.3g) , [0.3g, 0.7g), and [0.7g, 1.2g] correspond to quasi-linear, mildly to moderately nonlinear, and highly nonlinear structural responses, respectively. When earthquake ground motions in the high range, typically between 0.6g and 1.2g are underrepresented, randomly selected scaling factors ranging from 1 to 3 are applied to the available unscaled records to ensure sufficient coverage by producing additional ground motions with sufficient intensity to induce significant nonlinear structural response. This process resulted in 366 bidirectional ground motions in the [0.1g, 0.3g) range, 119 in the [0.3g, 0.7g) range, and 34 in the [0.7g, 1.2g] range, totaling 519 records. A Convolutional AutoEncoder (CAE), as described in the following section, is then employed to select a representative subset of these ground motions from each of the three ranges for training the LSTM-NARX model.

CAE for Selection of Earthquake Ground motions in LSTM Training

A CAE-based method is used to select appropriate bidirectional ground-motion records for generating training data for the LSTM-NARX model. Results not shown here indicate that using CAE for ground-motion selection enhances the predictive performance of the surrogate model once trained. Fig. 4 provides an overview of the CAE-based framework for ground motion input selection, with detailed explanations provided below.

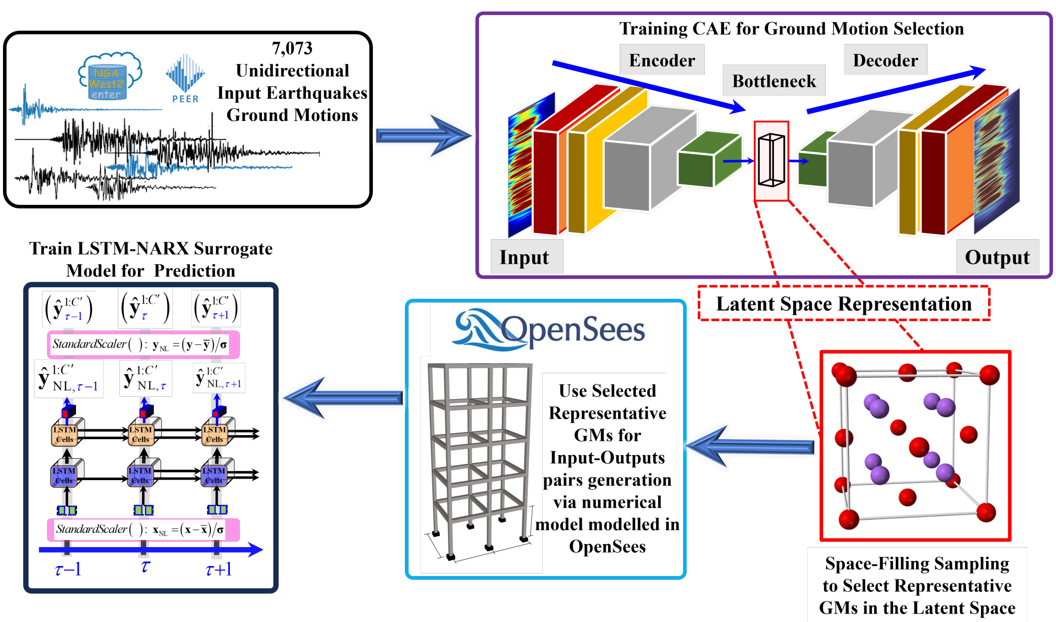

Fig. 4 Overview of the Novel CAE Framework for Ground Motion Selection Using Latent Space for LSTM Surrogate Model Training.

(a) About Convolutional AutoEncoders

Convolutional AutoEncoders (CAE) are unsupervised dimensionality reduction models characterized by convolutional layers, kernel size, and stride sizes adept at extracting salient features from input data [22, 23]. A convolutional layer processes grid-based input by using adaptive filters, which are parametrized 3×3 or 5×5 matrices, to perform element-wise operations. These filters slide across the input, generating feature maps that capture important patterns and spatial hierarchies, while reducing the dimensionality of the input [24]. The convolution layers are organized in three fundamental components: the encoder, the latent space, and the decoder, resembling the structure of a horizontally oriented hourglass (see Fig. 5). The autoencoder is trained to reconstitute the input grid-like data $\mathbf{H}$ by first mapping $\mathbf{H}$ to a lower-dimensional representation $\mathbf{z}$; known as the latent space or bottleneck, via the encoder, and then using a decoder to restore it to its original form grid-like form $\mathbf{H}'$ [25].

The input data for training the CAE consists of spectrograms generated by applying the Short-Term Fourier Transform (STFT) to 7,073 unidirectional time series from ground motion records. This process yields 7,073 2D grid-like matrices, each containing 1 channel of amplitude data, 251 frequency bins, and 32 time intervals. These 7,073 spectrograms, each of size 1×251×32, are then fed into the CAE for training, aiming to minimize the loss function in Eq. (7). A single spectrogram of size 251×32 represents 8,032 features (in Eq. (7), $d = 8,032$) that are passed through the encoder for a compressed representation $\mathbf{z}$.

Learning $\mathbf{z}$ involves minimizing a reconstruction loss in Eq. (7), which quantifies the fidelity of the autoencoder in reproducing the original input data $\mathbf{H}$ from its compressed latent representation $\mathbf{z}$, serving as a pivotal metric for optimizing the model via a search for the optimal weights and biases $\boldsymbol{\theta}_{\mathbf{CAE}}$ during training.

$$L_{CAE}(\boldsymbol{\theta}_{\mathbf{CAE}}) = \frac{1}{N_{SIB}} \sum_{s=1}^{N_{SIB}} \left\| \mathbf{H}_{NL,s} - \mathbf{H}'_{NL,s}(\boldsymbol{\theta}_{\mathbf{CAE}}) \right\|_2^2 = \frac{1}{N_{SIB}} \cdot \frac{1}{d} \sum_{s=1}^{N_{SIB}} \sum_{i=1}^{d} \left[h_{\mathrm{NL}\ ,\ \mathrm{s},\ i} - h'_{\mathrm{NL}\ ,\mathrm{s}\ ,i}\left(\boldsymbol{\theta}_{\mathbf{CAE}}\right) \right]^2 \quad (7)$$

where $\mathbf{H'}_{\mathrm{NL}}$ is the normalized reconstructed output data after passing the normalized input data $\mathbf{H}_{\mathrm{NL}}$ through the encoder; thus $\mathbf{H}'_{\mathrm{NL}} = Dec_{\theta_{CAE}}\left(Enc_{\theta_{CAE}}\left(\mathbf{H}_{\mathrm{NL}}\right)\right)$. In Eq. (7), N_{SIB} is the number of grid-like matrices in a batch, d is the total number of features in $\mathbf{H}_{\mathrm{NL}}$, and $\boldsymbol{\theta}_{\mathbf{CAE}}$ represents the set of CAE model weights and biases to be optimized. Through trial and error, testing the number of latent variables from 6 to 20, a dimension of 10 (i.e., $\mathbf{z}$ is taken as a 10-D vector) yielded the smallest reconstruction error of approximately $L_{CAE}\left(\boldsymbol{\theta}_{\mathbf{CAE}}\right) = 30$.

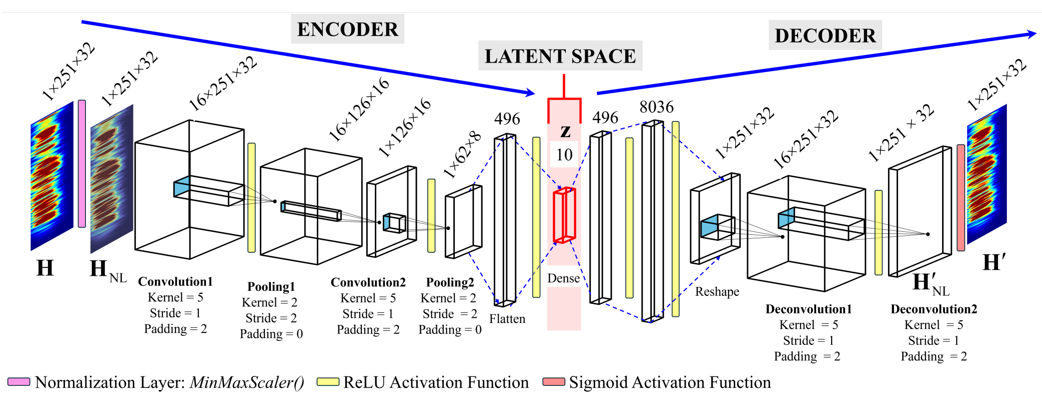

Fig. 5 Convolutional AutoEncoder Architecture.

(b) Selection of Ground Motions in the Latent Space

After training the CAE, we generate uniform-sized spectrogram of 2×251×32, where ”2” corresponds to the North-South and East-West components of the bidirectional EQ ground motions. In the latent space, each spectrogram channel is encoded as a 10-dimensional vector, with both components concatenated into a 20-dimensional representation. This process results in a 519×20 representation of the 519 (unscaled and scaled) earthquake ground motions defined in Section 4.1, followed by ″space-filling″ to select a representative set of 30 bidirectional earthquake ground motions from each of the 3 ranges specified in Section 4.1, totaling 90 earthquake bidirectional ground-motion records for LSTM model training [26, 27]. Fig. 6 shows the distribution of the original 519 bidirectional earthquake ground motions with $S_{a-\mathrm{gm}}$ between 0.1g and 1.2g (red diamonds) and the 90 ground motions selected via the CAE approach using the space-filling algorithm (blue circles).

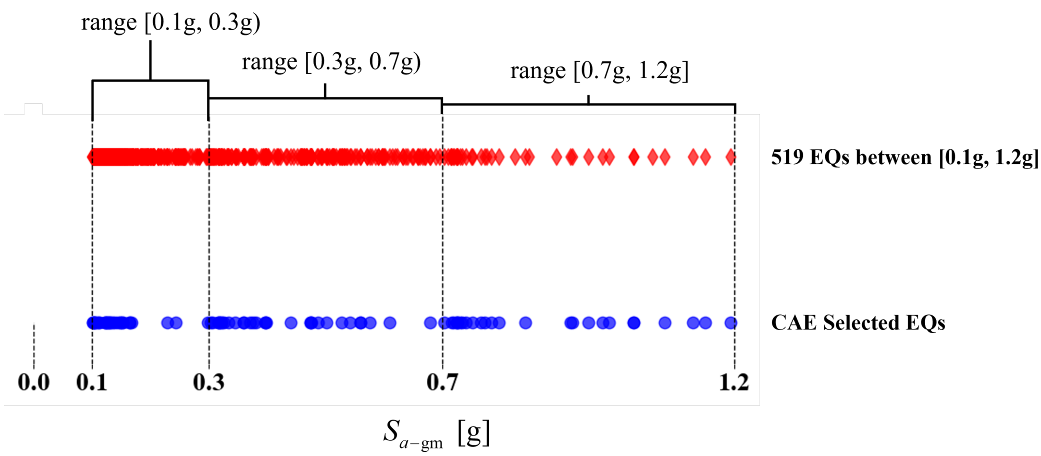

Fig. 6 Distribution of $S_{a-\mathrm{gm}}$ for the ensemble of 519 bidirectional earthquake ground motions defined in Section 4.1 (red diamonds) and after selection by CAE (90 ground motions represented by blue circles).

LSTM Surrogate Models And Results

Training of LSTM-NARX Surrogate Models for 3D Reinforced Concrete Frame

Table 1 summarizes the architecture of the LSTM-NARX neural network developed for the virtual yet realistic 3D reinforced concrete frame structure.

Table 1 LSTM-NARX Model Architecture for the 3D 5-story 2-by-1 bay RC frame structure

Layer Name	Number of units	Activation	Trainable Parameters (θ_{NARX})
LSTM Layer 1	100	elu	44,000
LSTM Layer 2	100	elu	80,400
LSTM Layer 3	100	elu	80,400
Fully Connected	1	linear	1,515
		Total	206,315

Ninety (90) input-output pairs are used to train the univariate LSTM-NARX models. The inputs consist of bidirectional earthquake ground motions, while the outputs represent relative displacements at each of the five floors of the 3D structure in both the X and Z directions. These 90 input-outputs pairs generate 192,060 input-output time series sequence samples using $w = 99$. The training process involves 3,000 epochs (the number of times the model iterates through the 192,060 samples), with a batch size of 1,280. Thus, for each epoch, the model processes 151 batches of 1,280 input-output time series sequence samples. The Adam optimizer is employed, with an initial learning rate of 6×10^{-4}, a final learning rate of 1×10^{-4}, and a weight decay of 1×10^{-5}. The mean squared error is used as the loss function, as defined in Eq. (4)a.

Fig. 7 and Fig. 8 show the LSTM-NARX prediction performance for two new, previously unseen bidirectional ground motion records, referred to as testing earthquakes. Both figures provide a detailed breakdown as follows:

(A) displays the time histories of the bidirectional input ground motion records.

(B) shows the level of nonlinearity at the structural level (total base shear normalized by the structure weight vs roof drift ratio defined as the fifth-floor relative displacement normalized by the total height of the structure), the section level response (bending moment vs curvature plot), and the stress-strain fiber response at the extreme fibers of the sections. All quantities are obtained from nonlinear time history analysis of the FE model.

(C) compares the ground truth relative displacement time histories obtained from the FE model with the LSTM-NARX predicted displacement responses in both the X and Z directions across all floors of the 3D structure, under the bidirectional earthquake excitation shown in (A).

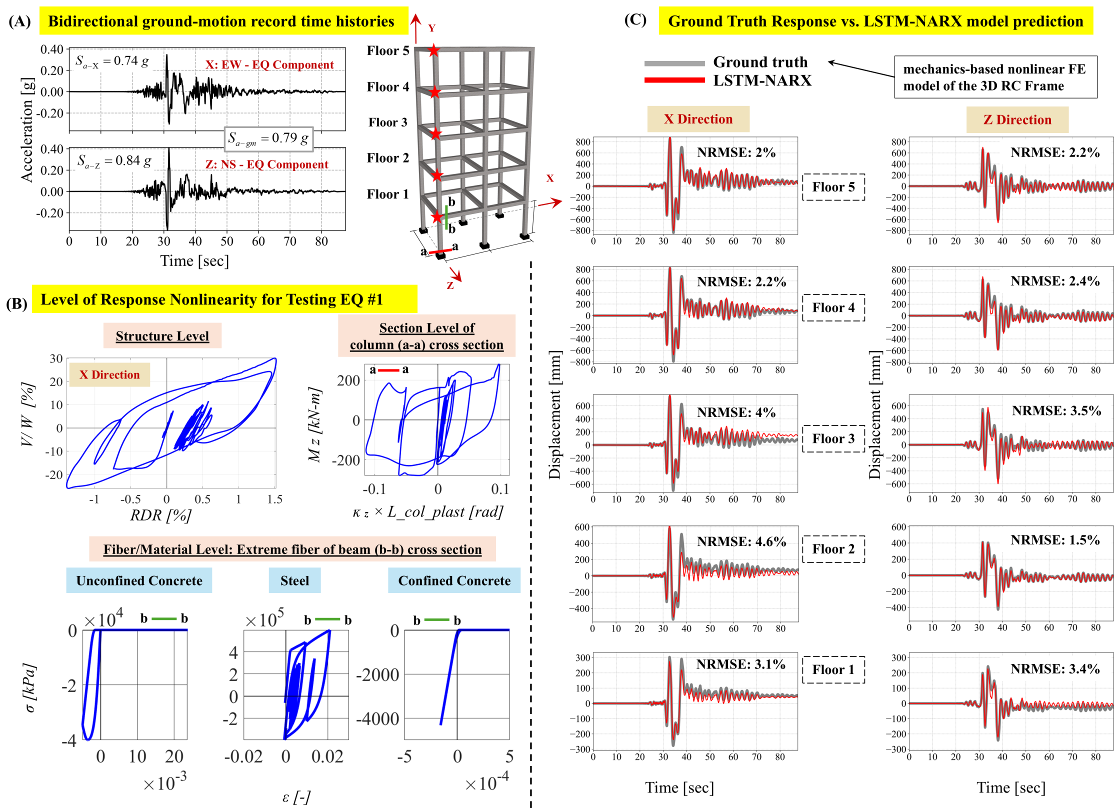

Fig. 7 LSTM-NARX predictions for testing bidirectional earthquake record #1.

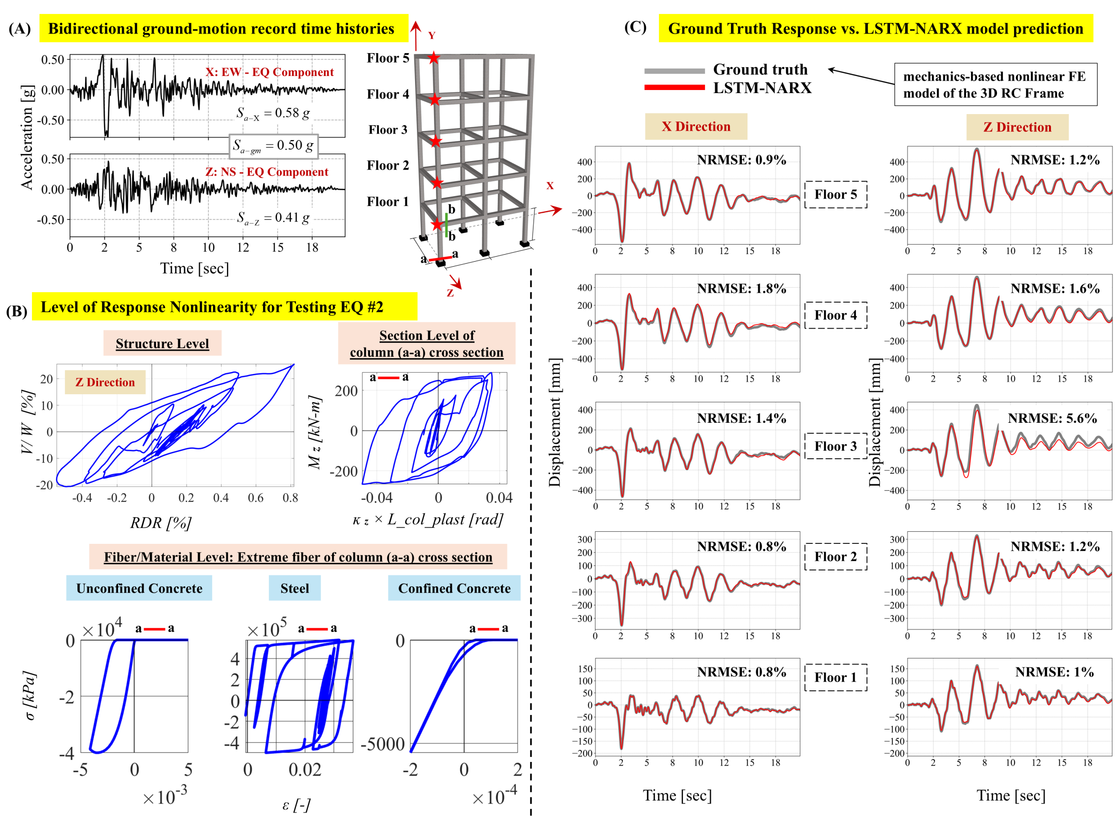

Fig. 8 LSTM-NARX predictions for testing bidirectional earthquake record #2.

Results Discussion

The proposed LSTM-NARX framework provides excellent predictions of the relative displacement response time histories in both the X and Z directions of the 3D reinforced concrete structure. The overall maximum normalized root mean square error (NRMSE) of approximately 5.6% indicates minimal discrepancy between the LSTM-NARX predicted relative displacement time histories and the ground truth values generated using the mechanics-based FE model. This demonstrates strong reliability and predictive capability of the LSTM-NARX model, highlighting the portability and scalability of the framework. The structure exhibits varying levels of nonlinearity when subjected to the testing bidirectional input ground motion records, resulting in moderately nonlinear (Fig. 8) and highly nonlinear (Fig. 7) responses at the structural, section, and material levels. The section-level responses at the base of the column indicate plastic hinge rotations of 0.1 rad and 0.04 rad for the first and second testing examples, respectively. At the fiber/material level at the base of the column, the concrete shows signs of cracking in tension and yielding with some spalling in compression. This condition prompts the steel fibers (reinforcement) to provide post-cracking strength. In general, 3D models better represent real-world structures, enabling a better representation of the structural behavior. During NLTHAs, the nonlinear dynamic response can be substantial, involving significant permanent deformation and pronounced material and geometric nonlinearities. Despite these complexities, the proposed LSTM-NARX model, combined with the CAE ground motion selection process, is able to capture the dynamics of the bidirectional input-output sequences over time via its recurrent connections, thereby preserving the causality inherent in dynamic system behavior modeled across time series.

Long-term time series, such as earthquake ground motions and the corresponding structural responses often display complex variations in period and trend. At the onset of an earthquake, the structure typically remains within the linear elastic range, with its displacement response oscillating around the zero axis. However, once yielding occurs, plastic deformations may result in a permanent shift in the axis of oscillation—commonly referred to as a baseline shift—in the displacement time histories. For instance, the fourth-floor response in the first testing earthquake shows such a baseline shift, with a residual displacement of approximately 100 mm in the X direction. Yet, the LSTM-NARX model, supported by the CAE process, successfully captures this residual drift. The model effectively learns both short- and long-range dependencies within the selected memory length w for output response prediction. The LSTM-NARX architecture enables the model to capture periodic cycles and baseline shifts in long-term trends—80 sec and 18 sec, respectively, for the two testing examples—over extended memory intervals. This improves the model's generalization capability by mitigating overfitting to short-term noise, despite fluctuations in frequency content across different earthquake excitations applied to the surrogate model.

Conclusions

This paper examines the application of the LSTM-NARX (Long Short-Term Memory Nonlinear AutoRegressive model with eXogenous inputs), a type of recurrent neural network (RNN) architecture that combines the strengths of LSTM cells and the NARX approach, to recursively estimate the relative displacement time histories responses of a 3D RC structure in both X and Z direction using external bidirectional input ground motion time histories. The training dataset consists of bidirectional earthquake ground-motion records selected using a convolution autoencoder. These records span a broad range of intensity to generate both linear and nonlinear responses used to train our LSTM-NARX model. After 3,000 epochs, the predicted responses from the LSTM-NARX model, which mimics a traditional time-stepping scheme, closely matched the true responses from unseen bidirectional input motions, with a low relative root-mean-square error. In essence, the LSTM architecture through its recurrent weights mirrors the dynamic response of our application example. The cyclic interplay between its cells and across its layers allows information from previous time steps to shape the current state and output. These recurrent weights, shared across all time steps, allow the LSTM tunable architecture to capture the sequential dynamics accurately over time, resulting in a strong match to the mechanics-based FE model responses.

Acknowledgments Funding for this work was provided by the Eric and Johanna Reissner Chair in the Department of Structural Engineering at UC San Diego. The authors would like to thank Professor Rodrigo Astroza from the Faculty of Engineering and Applied Sciences at the Universidad de los Andes in Santiago, Chile, for sharing the 3D RC nonlinear structural model and his insightful description of the model.

References

1. Wang Z, Zhang W, Taymaz T, He Z, Xu T, Zhang Z. Dynamic rupture process of the 2023 Mw 7.8 Kahramanmaraş earthquake (SE Türkiye): Variable rupture speed and implications for seismic hazard. Geophysical Research Letters. 2023;50:e2023GL104787.

2. Hu J, Liu M, Taymaz T, Ding L, Irmak TS. Characteristics of strong ground motion from the 2023 Mw 7.8 and Mw 7.6 Kahramanmaraş earthquake sequence. Bulletin of Earthquake Engineering. 2024:1–30.
3. Barbot S, Luo H, Wang T, Hamiel Y, Piatibratova O, Javed MT, et al. Slip distribution of the February 6, 2023 Mw 7.8 and Mw 7.6, Kahramanmaraş, Turkey earthquake sequence in the East Anatolian fault zone. Seismica. 2023;2.
4. Salehinejad H, Sankar S, Barfett J, Colak E, Valaee S. Recent advances in recurrent neural networks. arXiv preprint arXiv:180101078. 2017.
5. Zhang R, Chen Z, Chen S, Zheng J, Büyüköztürk O, Sun H. Deep long short-term memory networks for nonlinear structural seismic response prediction. Computers & Structures. 2019;220:55–68.
6. Xu Z, Chen J, Shen J, Xiang M. Recursive long short-term memory network for predicting nonlinear structural seismic response. Engineering Structures. 2022;250:113406.
7. Chen R, Jin X, Laima S, Huang Y, Li H. Intelligent modeling of nonlinear dynamical systems by machine learning. International Journal of Non-Linear Mechanics. 2022;142:103984.
8. Simpson T, Dervilis N, Chatzi E. Machine learning approach to model order reduction of nonlinear systems via autoencoder and LSTM networks. Journal of Engineering Mechanics. 2021;147:04021061.
9. Yu Y, Yao H, Liu Y. Structural dynamics simulation using a novel physics-guided machine learning method. Engineering Applications of Artificial Intelligence. 2020;96:103947.
10. Zhang R, Liu Y, Sun H. Physics-informed multi-LSTM networks for metamodeling of nonlinear structures. Computer Methods in Applied Mechanics and Engineering. 2020;369:113226.
11. Torky AA, Ohno S. Deep learning techniques for predicting nonlinear multi-component seismic responses of structural buildings. Computers & Structures. 2021;252:106570.
12. Astroza R, Ebrahimian H, Li Y, Conte JP. Bayesian nonlinear structural FE model and seismic input identification for damage assessment of civil structures. Mechanical Systems and Signal Processing. 2017;93:661–87.
13. IBC I. International Code Council. International Building Code International Code Council: Washington DC, United States. 2012.
14. Staudemeyer RC, Morris ER. Understanding LSTM–a tutorial into long short-term memory recurrent neural networks. arXiv preprint arXiv:190909586. 2019.
15. Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation. 1997;9:1735–80.
16. Chen G. A gentle tutorial of recurrent neural network with error backpropagation. arXiv preprint arXiv:161002583. 2016.
17. Chopra AK. Dynamics of Structure eBook, Global Edition: Pearson Education; 2015.
18. Baker JW, Cornell CA. Which spectral acceleration are you using? Earthquake Spectra. 2006;22:293–312.
19. Baker J, Bradley B, Stafford P. Seismic hazard and risk analysis: Cambridge University Press; 2021.
20. Stewart JP, Abrahamson NA, Atkinson GM, Baker JW, Boore DM, Bozorgnia Y, et al. Representation of bidirectional ground motions for design spectra in building codes. Earthquake Spectra. 2011;27:927–37.
21. Chiou B, Darragh R, Gregor N, Silva W. NGA project strong-motion database. Earthquake Spectra. 2008;24:23–44.
22. Zhang Y. A better autoencoder for image: Convolutional autoencoder. ICONIP17-DCEC Available online: http://users cecs anu edu au/Tom Gedeon/conf/ABCs2018/paper/ABCs2018_paper_58 pdf (accessed on 23 March 2017)2018.
23. Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. science. 2006;313:504–7.
24. O'shea K, Nash R. An introduction to convolutional neural networks. arXiv preprint arXiv:151108458. 2015.
25. Murphy KP. Machine learning: a probabilistic perspective: MIT press; 2012.
26. Hu Z, Hu C, Mourelatos ZP, Mahadevan S. Model discrepancy quantification in simulation-based design of dynamical systems. Journal of Mechanical Design. 2019;141:011401.
27. Jin R, Chen W, Sudjianto A. On sequential sampling for global metamodeling in engineering design. International design engineering technical conferences and computers and information in engineering conference2002. p. 539–48.