SCIENCE CHINA
Technological Sciences

•Article•

January 2023 Vol.66 No.1: 141–154
https://doi.org/10.1007/s11431-021-2068-8

A computational method for the load spectra of large-scale structures with a data-driven learning algorithm

CHEN XianJia[1], YUAN Zheng[2], LI Qiang[2], SUN ShouGuang[2] & WEI YuJie[1,3*]

[1] *The State Key Laboratory of Nonlinear Mechanics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China;*
[2] *School of Mechanical, Electronic and Control Engineering, Beijing Jiaotong University, Beijing 100044, China;*
[3] *School of Engineering Sciences, University of Chinese Academy of Sciences, Beijing 100049, China*

Received December 9, 2021; accepted April 24, 2022; published online December 26, 2022

For complex engineering systems, such as trains, planes, and offshore oil platforms, load spectra are cornerstone of their safety designs and fault diagnoses. We demonstrate in this study that well-orchestrated machine learning modeling, in combination with limited experimental data, can effectively reproduce the high-fidelity, history-dependent load spectra in critical sites of complex engineering systems, such as high-speed trains. To meet the need for in-service monitoring, we propose a segmentation and randomization strategy for long-duration historical data processing to improve the accuracy of our data-driven model for long-term load-time history prediction. Results showed the existence of an optimal length of subsequence, which is associated with the characteristic dissipation time of the dynamic system. Moreover, the data-driven model exhibits an excellent generalization capability to accurately predict the load spectra for different levels of passenger-dedicated lines. In brief, we pave the way, from data preprocessing, hyperparameter selection, to learning strategy, on how to capture the nonlinear responses of such a dynamic system, which may then provide a unifying framework that could enable the synergy of computation and in-field experiments to save orders of magnitude of expenses for the load spectrum monitoring of complex engineering structures in service and prevent catastrophic fatigue and fracture in those solids.

load spectrum, computational mechanics, deep learning, data-driven modeling, gated recurrent unit neural network

Citation: Chen X J, Yuan Z, Li Q, et al. A computational method for the load spectra of large-scale structures with a data-driven learning algorithm. Sci China Tech Sci, 2023, 66: 141–154, https://doi.org/10.1007/s11431-021-2068-8

1 Introduction

The demand for lighter and stronger structures with better performance in engineering systems, such as cars, trains, and planes, to achieve faster speeds and longer service lives is ever-increasing. The service life of such a system is decisively dependent on the load spectrum at the weakest points of the components subject to a time-varying load history, which is often compiled into a load spectrum composed of information on variable amplitude loads and their cumulative occurrences. Then, fatigue analysis can be conducted for the point prone to fatigue by applying the load spectrum. Although the entire process seems to be straightforward, obtaining reliable load spectra remains to be an immense challenge. Both structural behavior and operating conditions contribute to the load-time sequence and interfere with each other in nonlinear dynamic systems. Either field tests or full-scale computational analyses are inadequate and inefficient to fulfill the desire to obtain precise load spectra, which are prerequisites for both extended service life and an ever-diminishing time from design to production. The rapid development of machine learning techniques, combined with limited real-life load sequence from field tests, may help solve the long-standing problem.

In the conventional scenario, we rely on data for quantitative analysis, verification, and proof of theories, and we use

*Corresponding author (email: yujie_wei@lnm.imech.ac.cn)

computers as a calculator that produces data through programmed mathematical equations, physical laws, and engineering models. As people are now being swamped by data explosion fueled by the rapid increase of electronic devices for data collection and generation, the need for machine learning surges in the hope of a well-developed algorithm to learn the rules underlying a given set of data and even to derive a function that predicts the output values to the desired degree of fidelity [1]. As was summarized by Butler et al. [1], to boost research by employing machine learning techniques, one will likely encounter tremendous challenges in the following scenarios: the massive amount of existing data with reasonable fidelity; the complex patterns of data that are difficult to decipher without new scientific laws; and the existing resources and tools for experiments, simulations, or theories are insufficient or cost too much to piece a comprehensive understanding together. As a consequence, machine learning and data analytics have been developed and employed across diverse scientific disciplines, including engineering [2–8], materials [9–12], biomedicine [13], clinical diagnosis [14,15], chemicals [16], quantum mechanics [17–20], seismology [21,22], geosciences [23], and genomics [24].

The advantages of machine learning seem to fit well with fault diagnosis in complex engineering systems [25,26]. In practice, structural reliability and life cycle assessment in complex engineering systems are contingent on our deep understanding of the load environment to which they are subjected. Traditionally, researchers and engineers relied on experiments to obtain the load spectra of the weakest link of an engineering system. Examples of such cover a broad field of applications, from structural stress of aircraft [27], sea wave, wind, and even seismic loads to offshore wind turbines [28], the vibration of a long-span sea-crossing bridge [29], to stress on the bogie frame of a high-speed train (HST) [30].

As the load spectrum of vulnerable sites in those complex systems is crucial, accurate and adequate data from which the spectrum can be obtained are desired, and this step is always challenging. Although the kinematics and kinetics of each piece of a dynamic system may be reasonably well formulated with classic conservation laws and equilibrium equations, the collective behavior of those components in contact, which is dissipative and nonlinear, is difficult to handle. However, the loading history of weak zones with precise positioning needs to be determined, and their lifetime needs to be predicted based on limited data. In the face of such constraints, engineers soon learn that "more is different" [31], even at the macroscopic spatial scale. Such complex systems involve massive combinatorial time, space, and nonlinear processes, which conventional computational mechanics procedures, particularly the finite element method (FEM), either cannot solve or can solve at a high cost.

Unprecedented data acquisition resources and tools enable us to amass vast amounts of data by monitoring the mechanical behavior of aircraft, space shuttles, and other large structures [32,33]. In combination with the lowered entry barrier of machine learning techniques, many engineering-oriented researchers have exploited machine learning to tap into those vast amounts of data for underlying patterns and scientific laws. A physics-informed deep learning framework was developed to extract the velocity and pressure fields directly from the flow visualizations by encoding the Navier-Stokes equations into the neural networks [34]. Notably, both recurrent neural network (RNN) and temporal convolutional neural network (TCNN), which are used as black-box tools, exhibit considerable potentials and promises in the reconstruction of dynamic phenomena by extracting deep features from data [35–38]. Recent works revealed the outstanding capability of sequence learning in predicting path-dependent plasticity-constitutive laws of general materials in the context of computational plasticity [39,40]. Even the complex rheology and topology of fractures in the crust layer of the solid earth are considered [41].

At present, research on the load spectrum of HSTs in operation mainly focuses on the standardized compilation methods [30,42] and the reasonable characterizations of the load spectra for critical components, such as bogie frames. Ma et al. [43] used the diffusion-based kernel density estimation method to obtain the load spectrum distributions of train bogie frames and achieve reasonable extrapolation. Yuan et al. [44] proposed a segmented load spectrum model to accurately describe the load spectra of the bogie frame in an HST and used the inflection stress in the model to characterize the quality of lines. Machine learning approaches may also provide an alternative to the analysis and prediction of load spectra. Klemenc and Fajdiga [45] utilized the localized basis function neural network to predict the load spectrum by mapping the relationship between the factors of operating conditions and the corresponding load spectra in the frequency domain. Yang et al. [46] adopted the Markov chain Monte Carlo-peak over threshold model to realize the time-domain extrapolation of tractor drive shaft loads in stationary operating conditions. Moreover, Zou et al. [47] proposed a new fault diagnosis method for the bearing of traction motor in HSTs based on discrete wavelet transform and improved deep belief network by identifying the correlation between fault features and fault types. In other engineering fields, RNN exhibits considerable potentials for the prediction of time-dependent factors, such as indoor seismic response prediction of nonlinear structures [35], state-of-charge estimation of Li-ion batteries [48], short-term electric power load forecasting [49], and Internet-of-things traffic prediction [50].

In reality, all of these load spectra can be considered as history-related continuous stochastic processes. To the best of our knowledge, solid examples that machining learning

could be employed to handle the load spectra of real large-scale engineering systems and reduce the expenses of repeated experiments are still lacking. As was pointed out by Jordan and Mitchell [51], fault diagnosis in complex systems is considered as one of the three major industrial fields concerned with data-intensive issues where the effect of machine learning could significantly emerge. In this study, we aim to explore a unifying framework for a combination of machine learning modeling with limited experimental data to realize high-fidelity predictions of history-dependent load spectra of vulnerable sites in many engineering systems in operation, such as an HST. To meet the needs for in-service monitoring, we propose a segmentation and randomization strategy for long-duration historical data processing to improve the accuracy of our data-driven model for long-term load-time history prediction and reveal the optimal length of subsequence, which is associated with the characteristic dissipation time of the dynamic system. In brief, we pave the way, from data preprocessing, hyperparameter selection, to learning strategy, on how to capture the nonlinear responses of such a dynamic system, which may then provide a unifying framework that could enable the synergy of computation and in-field experiments to save orders of magnitude of expenses for the load spectrum monitoring of complex engineering structures in service and to prevent catastrophic fatigue and fracture in those solids.

2 Problem formulation

In most dynamic systems, such as an HST, their components are subjected to loads from different sources, such as road conditions, driver-induced manoeuvers, occasional track change, tunnel entrance and exit, vehicle crossing, and transient aerodynamics. Ideally, the exact contribution of those distinct sources to the load of a particular site needs to be determined for fatigue analysis. In practice, we encounter difficulties in measuring the respective amounts of the contributions of those load sources. Only the collective reaction of all loads at certain points of the structure is available. The stress at a critical point in the structure, which is the resultant of the reactions, is either inferred from large-scale computations or directly obtained from in-field experiments for a limited number of points and can only be possible within a short time. Hence, the central question that we aim to explore here is "Can we employ a limited number of reactions on a component under service conditions to derive the stress at the critical points of the component for a long duration without conducting large-scale computations and obtain the precise stress spectra in those points?" With an HST as the model case, we describe in detail how supervised machine learning may be orchestrated to answer the aforementioned question.

2.1 Problem statement

Many components and regions of an HST may be vulnerable to fatigue and failure; hence, they should be monitored during operation and regularly diagnosed to prevent catastrophe. We focus on the bogie frame of an HST, as is shown in Figure 1(a), which is a critical load-bearing component located between the wheelsets and the car body. To generate sufficient data for supervised learning, we conduct long-term field tests that cover approximately 1000000 km of operating mileage on the Beijing–Guangzhou and Beijing–Taiyuan passenger-dedicated lines (PDLs; as is depicted in Figure 1 (b)) on the CRH-380A electric multiple units (EMU) platform.

As is shown in Figure 1(c), the time sequence of the lateral and vertical loads acting on the bogie frame under real loading conditions, i.e., F_1 to F_8 in Figure 1(a) (I), could lead to stress at the vulnerable point for fatigue identified in Figure 1(a) (II), which is the welded joint of the transverse beam and the side frame of the bogie frame. As was validated by our finite element (FE) calculations, the maximum principal stress at the vulnerable spot is perpendicular to the weld toe. Through well-designed strain gauge assembly, we effectively capture the time history of the maximum principal stress at the investigated point, as is illustrated in Figure 1(c). Figure 1(c) shows the real dynamic characteristics of the bogie frame under investigation. In the conventional computational mechanics approach, we rely on the FEM to simulate the stress-time sequence shown in Figure 1(c), with forces $F_1, \cdots, F_8$ as inputs. We demonstrate the bottleneck of such traditional computational methods in predicting the load spectra of materials in a complex engineering system in Appendix A in the Supporting Information. For full-scale modeling of a three-car marshaling HST with high spatial and temporal resolutions, the use of conventional approaches becomes computationally prohibitive (as shown in Table S1) and physically impossible. Thus, data-driven machine learning may become a potential option to solve such problems.

2.2 Integrated computational mechanics with machine learning

2.2.1 Learning objective

We adopt the data-driven learning approach to model the dynamic and nonlinear characteristics of the concerned bogie frame system because the governing physical laws cannot be explicitly formulated due to their complexity. Two challenges, i.e., (a) the history-dependent dynamics and (b) sufficiently large duration prediction for practical usage, need to be overcome for the successful application of deep learning in predicting the load spectrum.

The learning procedures are as follows: (1) Our objective

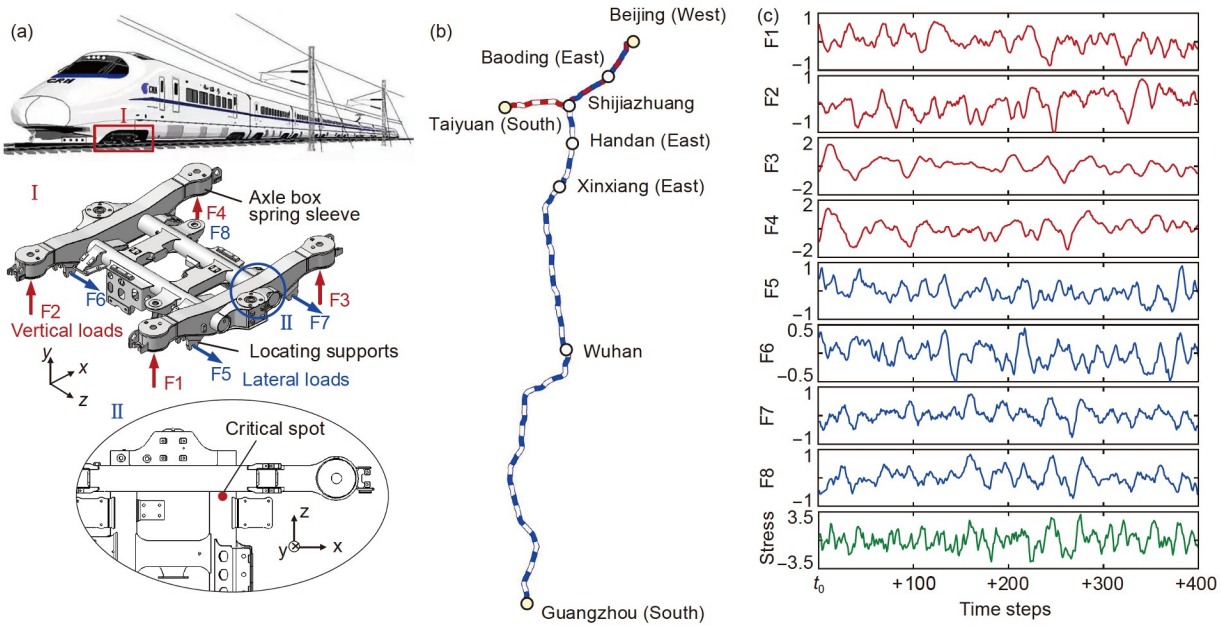

Figure 1 Conventional computational mechanics challenge—fault diagnosis of a typical complex system. (a) Hierarchical structure of a CRH-380A electric multiple units (EMU) platform, where we focus on the bogie frame, the most critical part for safety. The vulnerable location is at the junction of the transverse beam and side frame. (b) On-track tests on the Beijing–Guangzhou and Beijing–Taiyuan passenger-dedicated lines (PDLs) for more than one year with an operating mileage of approximately 1000000 km. For convenience, the station name is hereinafter referred to as its city name. (c) Load (kN)-time histories acting on the bogie frame and the corresponding stress (MPa)-time history at the critical point, where each time step is 0.002 s.

is to obtain the load spectrum of a bogie frame of an HST. (2) The inputs are the time histories of four vertical and four lateral forces acting on the bogie frame (F_1 to F_8; as is shown in Figure 1(a) (I)). (3) Our desired output is the stress-time history at the critical point identified in Figure 1(a) (II). (4) We utilize the learning algorithm to model the dynamic system by capturing the intricate relationship between the inputs and the outputs. (5) The force-time sequences from in-service monitoring are used as the inputs of the trained model to predict the stress-time histories and thus the load spectrum.

2.2.2 Physics-based model of a complex nonlinear dynamic system

In general, the physics-based equilibrium equation of a nonlinear dynamic system with multiple degrees of freedom can be formulated as follows [52]:

$$\mathbf{M}\ddot{\mathbf{u}}_t + \mathbf{C}\dot{\mathbf{u}}_t + \mathbf{K}\mathbf{u}_t = \mathbf{F}_t, \tag{1}$$

where $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$ denote the mass, damping, and stiffness, respectively; $\mathbf{u}_t$, $\dot{\mathbf{u}}_t$, and $\ddot{\mathbf{u}}_t$ denote the displacement, velocity, and acceleration of the system at time t, respectively; and $\mathbf{F}_i$ denotes the external load. With the known responses of the system, we compute the field quantities, such as strain and stress, for safety analysis. Then, we use $\mathbf{R}_t$ to denote the responses of the system (e.g., displacements, strains, and stresses), and it can be expressed as follows:

$$\mathbf{R}_t = f_R(\mathbf{F}_t, \mathbf{M}, \mathbf{C}, \mathbf{K}, \mathbf{R}_{t-1}). \tag{2}$$

Eq. (2) indicates that $\mathbf{R}_t$ depends not only on the systemic property and current load $\mathbf{F}_t$ but also on the historic response $\mathbf{R}_{t-1}$. The strong nonlinearity and hysteresis of interactions between connecting components render the damping and stiffness matrices infeasible in an explicit formula. For instance, the stiffness and damping characteristics of air springs connecting a car body and a bogie frame of an HST heavily rely on the displacement, velocity, amplitude, and frequency of the external excitations and the responses of the structures to those loadings. The output force of a damper in a suspension usually lags behind the displacement and velocity transmission. As is depicted in Figure 1(c), the physical correlation between external loads and structural responses governed by eq. (2) is intricate, and full-scale modeling of high spatial and temporal resolutions is computationally prohibitive and physically impossible.

2.2.3 Gated recurrent unit neural network for load spectrum prediction

To capture the history dependence, we adopt an RNN to handle sequential data. The RNN calculates the output $\mathbf{o}_t(\mathbf{x}_t, \mathbf{h}_t)$ by taking advantage of the history-dependent hidden state $\mathbf{h}_t(\mathbf{x}_t, \mathbf{h}_{t-1})$. To avoid the "vanishing/exploding gradients" [53] for long sequence learning using RNN, two variations of RNN, i.e., long short-term memory (LSTM) [54] and gated recurrent unit (GRU) [55] neural networks, were developed. Multiple data gates that control the transmission of information flow in hidden states and outputs are

introduced. Therefore, the augmented sequential data learning in GRU neural network is adopted to capture the history-dependent responses of the dynamic systems of our concern.

The standard architecture of a GRU neural network includes an input layer, multiple hidden layers composed of the GRU and fully connected (FC) layers, and an output layer, as is illustrated in Figure 2(a) and (b), where the network has 2 GRU layers and 1 FC layer as an example. The input layer is

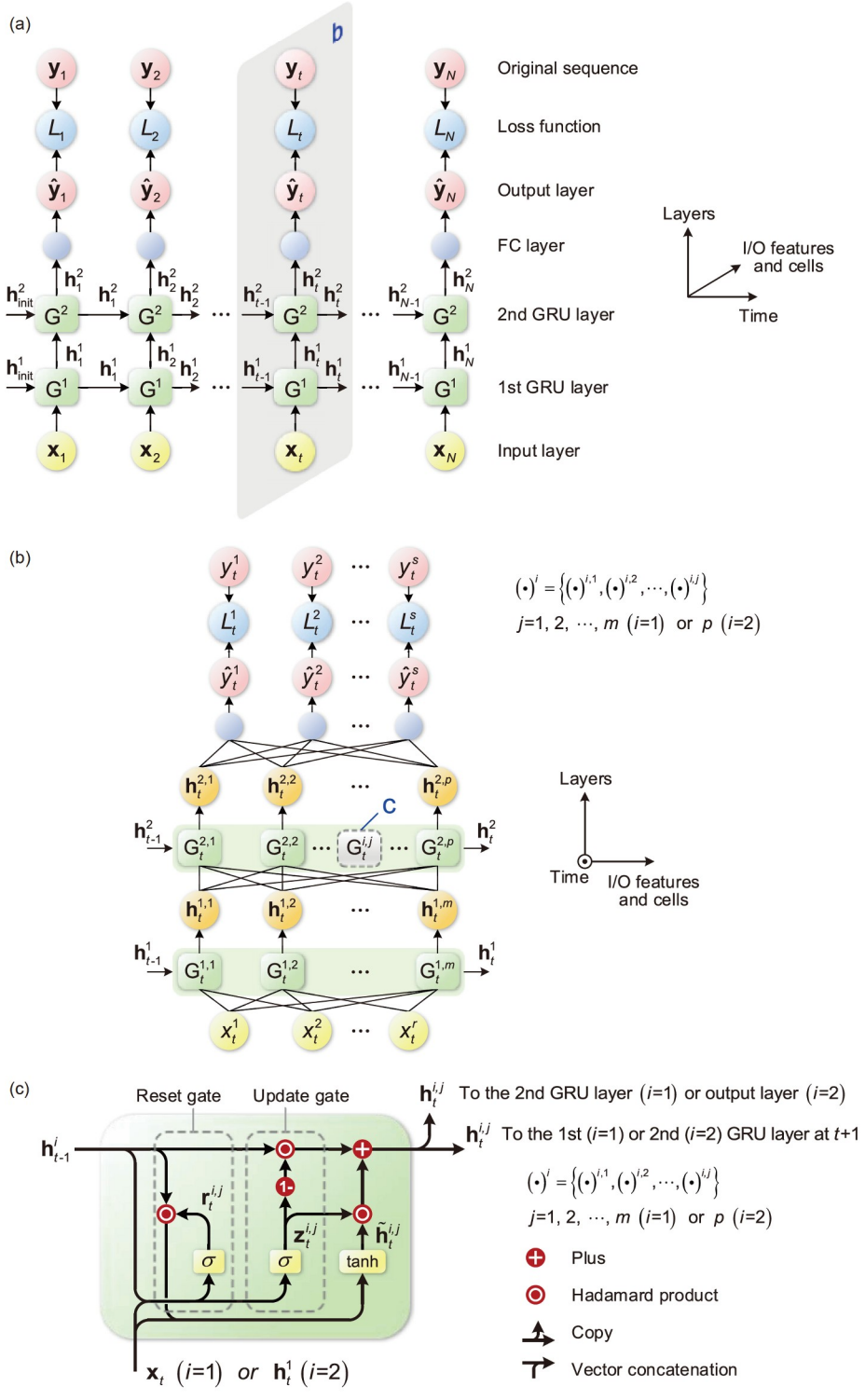

Figure 2 Schematic diagram of a gated recurrent unit (GRU) neural network used for load spectrum prediction. (a) Architecture of a GRU neural network with 2 GRU layers and 1 fully connected (FC) layer across the temporal space from 1 to N. (b) GRU neural network at time t, where the dimensions of the input and output features are r and s, respectively, and the first and second GRU layers have m and p GRU cells, respectively. Notably, all trainable parameters in (b) are shared across the entire temporal space. (c) Inner structure of the j-th GRU cell in the i-th GRU layer.

connected to the input sequence $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_t, \cdots, \mathbf{x}_N\}$, $t = 1, 2, \cdots, N$, and $\mathbf{x}_t = \left[x_t^1, x_t^2, \cdots, x_t^r\right]^{\mathrm{T}}$ is the input with r-dimensional features. Every GRU layer contains a suite of GRU cells, and each of them, whose inner structure is illustrated in Figure 2(c), has an independent set of weights and biases, which are shared across the entire temporal space within the layer.

For a network with a total of L GRU layers, the information flow in the l-th GRU layer at time t is formulated as follows:

$$\mathbf{r}_t^l = \sigma\left(\mathbf{W}_{hr}^l \cdot \mathbf{h}_{t-1}^l + \mathbf{W}_{xr}^l \cdot \mathbf{x}_t^l + \mathbf{b}_r^l\right), \tag{3}$$

$$\mathbf{z}_t^l = \sigma\left(\mathbf{W}_{hz}^l \cdot \mathbf{h}_{t-1}^l + \mathbf{W}_{xz}^l \cdot \mathbf{x}_t^l + \mathbf{b}_z^l\right), \tag{4}$$

$$\tilde{\mathbf{h}}_t^l = \tanh\left(\mathbf{W}_{h\tilde{h}}^l \cdot \left(\mathbf{r}_t^l \odot \mathbf{h}_{t-1}^l\right) + \mathbf{W}_{x\tilde{h}}^l \cdot \mathbf{x}_t^l + \mathbf{b}_{\tilde{h}}^l\right), \tag{5}$$

$$\mathbf{o}_t^l = \mathbf{h}_t^l = \left(1 - \mathbf{z}_t^l\right) \odot \mathbf{h}_{t-1}^l + \mathbf{z}_t^l \odot \tilde{\mathbf{h}}_t^l, \tag{6}$$

where $\mathbf{W}_{\alpha\beta}^l$ (here α={h,x} and $\beta = \{r, z, \tilde{h}\}$) denotes the weight matrix, $\mathbf{b}_\beta^l$ denotes the corresponding bias vector, $\sigma(\cdot)$ denotes the sigmoid function, $\tanh(\cdot)$ denotes the hyperbolic tangent function, and $\odot$ denotes the Hadamard product. The reset gate $\mathbf{r}_t^l$ controls how much information from the previous hidden state $\mathbf{h}_{t-1}^l$ can be transmitted to the current candidate hidden state $\tilde{\mathbf{h}}_t^l$. Meanwhile, the update gate $\mathbf{z}_t^l$ regulates the proportion of information from the previous hidden state $\mathbf{h}_{t-1}^l$ and the current candidate hidden state $\tilde{\mathbf{h}}_t^l$ transmitted to the output $\mathbf{o}_t^l$, which is also used as the new hidden state $\mathbf{h}_t^l$. The input of the l-th GRU layer $\mathbf{x}_t^l$ is the output of the $(l-1)$-th GRU layer $\mathbf{h}_t^{l-1}$. In particular, the input of the first GRU layer is the input layer of the entire network $\mathbf{x}_t$.

Thus, considering the information flow in a GRU network, the hidden state of the L-th layer at time t, $\mathbf{h}_t^L$, could be formulated as follows:

$$\mathbf{h}_t^L = f_{\mathrm{GRU}}\left(\mathbf{x}_t, \boldsymbol{\theta}, \mathbf{h}_{t-1}^L\right), \tag{7}$$

where $\boldsymbol{\theta}$ denotes all weights and bias parameters involved in the GRU layers. The output layer of the network $\hat{\mathbf{y}} = \{\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, \cdots, \hat{\mathbf{y}}_t, \cdots, \hat{\mathbf{y}}_N\}$, $t = 1, 2, \cdots, N$, is connected to the last GRU layer through the FC layers, where $\hat{\mathbf{y}}_t = \left[\hat{y}_t^1, \hat{y}_t^2, \cdots, \hat{y}_t^s\right]^{\mathrm{T}}$ is the predicted output with s-dimensional features. When there is only 1 FC layer, the output of the network at time t is expressed as follows:

$$\hat{\mathbf{y}}_t = g\left(\mathbf{W}_o \cdot \mathbf{o}_t^L + \mathbf{b}_o\right) = g\left(\mathbf{W}_o \cdot \mathbf{h}_t^L + \mathbf{b}_o\right), \tag{8}$$

where $\mathbf{W}_o$ and $\mathbf{b}_o$ denote the output weight matrix and output bias vector, respectively, and $g(\cdot)$ denotes the output transform function. The sigmoid function is adopted as the output transform function in our model.

Notably, the relationship between $\mathbf{h}_t^L$ and $\mathbf{h}_{t-1}^L$ expressed in eq. (7) and the dynamic response formulated in eq. (2) is similar, which enables the GRU neural network to describe similar history-dependent problems. The reset and update gates in a GRU neural network make it possible to characterize the dissipative property of a target system by controlling the extent and duration of information correlation during training. These two mechanisms, along with the nonlinear activation functions in the hidden and output layers of the GRU neural network, provide sufficient interpretability for the model [56] to capture the history-dependent, nonlinear, and dissipative behavior of a complex dynamic system.

3 Learning framework

3.1 Learning algorithm

To learn the intricate correlation between external loads and structural responses from experimental data in the context of sequence learning, we develop appropriate methods and strategies for test data processing, input (output) format handling, hyperparameter choice, and predictive performance assessment. The learning framework for load spectrum prediction of a dynamic engineering system is shown in Figure 3, which includes (1) raw data testing and processing to generate an effective database including inputs and corresponding outputs (Figure 3(a)); (2) segmentation and randomization of long input/output sequences to improve the trainability and predictive performance of the model, as shown in Figure 3(b); (3) hyperparameter optimization to construct a history-dependent neural network with the best performance, as is shown in Figure 3(c); (4) model training to explore the intricate correlation between load inputs and response outputs and error analysis to assess the performance of the learned model in load spectrum prediction. In practice, the enumeration strategy is adopted to optimize hyperparameters, such as the number of GRU layers, the cells in each layer, and the length of subsequence, to achieve a learned model with the best predictive accuracy and training efficiency.

3.2 Segmentation and randomization for long-duration sequences

In general, the total number of time steps of load and response histories for an engineering system in service is tens of millions or even much more. For instance, the force (input) and stress (output) histories of the bogie frame from a one-way operation on the Beijing–Guangzhou PDL under

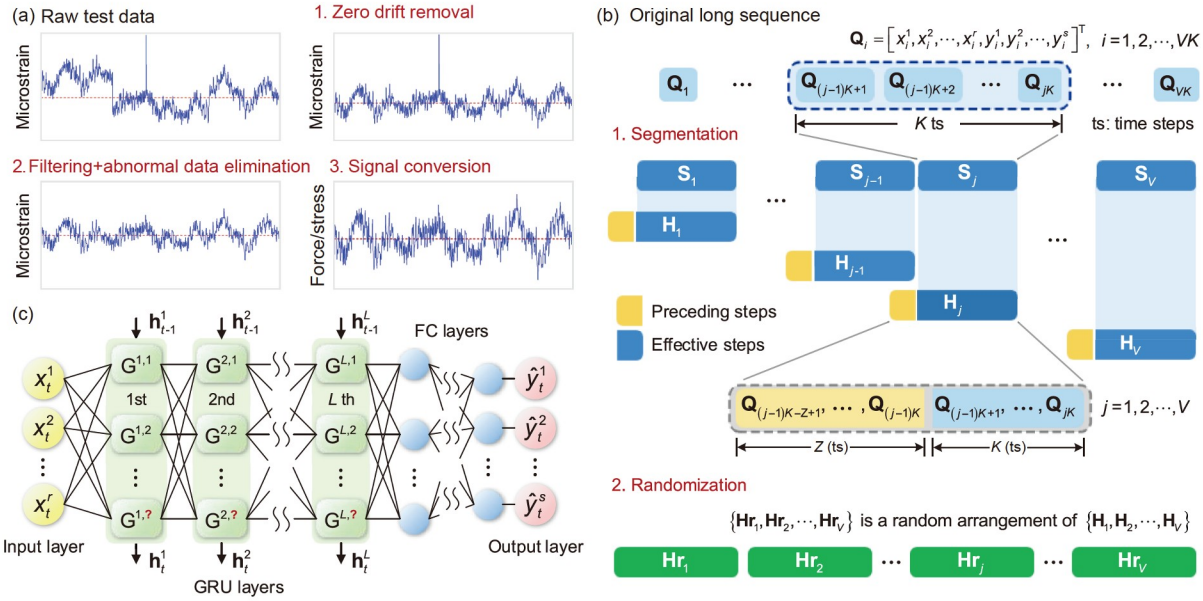

Figure 3 Machine learning procedures for load spectrum prediction. (a) Raw data processing, including (1) zero drift removal, (2) filtering and abnormal data elimination, and (3) conversion of strain signals into force and stress histories. (b) Segmentation and randomization of long input/output sequences: The original sequence is divided into V shuffled subsequences with K effective time steps and z preceding steps to eliminate the possible influence of the initial conditions. (c) GRU neural network construction and hyperparameter optimization, with subsequence length, GRU layers, and cells in each GRU layer as the hyperparameters to be optimized.

investigation last more than 15000000 time steps. Hence, we develop a segmentation and randomization method to pre-process such a long sequence to improve the predictability of our learned model for data from a sufficiently long duration.

As is illustrated in Figure 3(b), the original long sequence with a length of VK time steps can be expressed as a combination of input and output sequences in the column direction, as follows:

$$\mathbf{Q} = \{\mathbf{Q}_1, \mathbf{Q}_2, \cdots, \mathbf{Q}_i, \cdots, \mathbf{Q}_{VK}\}^{\mathrm{T}} \in \mathbb{R}^{VK\times(r+s)}, \tag{9}$$

where $\mathbf{Q}_i = \left[x_i^1, x_i^2, \cdots, x_i^r, y_i^1, y_i^2, \cdots, y_i^s\right]^{\mathrm{T}}$, $i = 1, 2, \cdots, VK$. We segment the original sequence $\mathbf{Q}$ into subsequences $\{\mathbf{S}_1, \mathbf{S}_2, \cdots, \mathbf{S}_j, \cdots, \mathbf{S}_V\}$, and each of them has K effective time steps, which can be expressed as follows:

$$\mathbf{S}_j = \left\{\mathbf{Q}_{(j-1)K+1}, \mathbf{Q}_{(j-1)K+2}, \cdots, \mathbf{Q}_{jK}\right\}^{\mathrm{T}} \in \mathbb{R}^{K\times(r+s)} \quad (j = 1, 2, \cdots, V). \tag{10}$$

To eliminate the impact of the initial conditions on the outputs at effective time steps, each subsequence is extended Z time steps ahead, which can be expressed as follows:

$$\mathbf{H}_j = \left\{\mathbf{Q}_{(j-1)K-Z+1}, \cdots, \mathbf{Q}_{(j-1)K}, \mathbf{Q}_{(j-1)K+1}, \cdots, \mathbf{Q}_{jK}\right\}^{\mathrm{T}} \in \mathbb{R}^{(K+Z)\times(r+s)} (j = 1, 2, \cdots, V). \tag{11}$$

Notably, during the training process, only the predictive errors at the effective time steps are used for gradient descent to update the parameters of the model.

Finally, we shuffle the V ordered subsequences to achieve the random segment $\{\mathbf{Hr}_1, \mathbf{Hr}_2, \cdots, \mathbf{Hr}_j, \cdots, \mathbf{Hr}_V\}$, which is a random arrangement of $\{\mathbf{H}_1, \mathbf{H}_2, \cdots, \mathbf{H}_j, \cdots, \mathbf{H}_V\}$, and $j = 1, 2, \cdots, V$.

To improve the convergence capability during model training, we divide all V subsequences into equal length batches. For simplicity, we let n_b=V/L_b, with L_b being the batch size (number of subsequences in a batch); hence, n_b is the total number of batches. In each epoch of the training phase, we iterate from the first batch to the n_b-th batch to calculate the gradient descent and update the parameters of the learning model. Therefore, the batch size L_b is another crucial hyperparameter that may affect the predictability of the learning model.

3.3 Predictability evaluation metric

The weights and biases of the neural network model are identified through supervised learning by minimizing the difference between the model predictions and the desired model output. Here, we use the normalized root mean square error (NRMSE) and the probability density function (PDF) of its distribution as the criteria to evaluate the predictive performance of the trained model. To better illustrate the definition of the error evaluation metrics, we assume that we have only one feature for output, i.e., s=1. After the desired and predicted output sequences are normalized to $\mathbf{y}' = [y'_1, y'_2, \cdots, y'_N]$ and $\widehat{\mathbf{y}}' = [\widehat{y}'_1, \widehat{y}'_2, \cdots, \widehat{y}'_N]$, respec-

tively, between 0 and 1 with the same criteria, the NRMSE metric is expressed as follows:

$$\mathrm{NRMSE}=\frac{\sqrt{\sum_{i=1}^{N}\left(\widehat{y'}_i-y'_i\right)^2/N}}{\sum_{i=1}^{N}\left|y'_i\right|/N}. \tag{12}$$

To further explore the predictive accuracy of our model over the entire time evolution, the PDF of the normalized error distribution, i.e., π_i, is expressed as follows:

$$\pi_i=\mathrm{PDF}\left\{\frac{\widehat{y'}_i-y'_i}{\sum_{i=1}^{N}\left|y'_i\right|/N}\right\}(i=1,2,\cdots,N). \tag{13}$$

The neural network models are constructed in the Python environment via the Keras library 2.3.0 [57]. In particular, the input and output histories of the database are normalized to the ranges of [−5,5] and [0, 1], respectively, to enhance the training performance. The training process is performed utilizing the Adam optimization algorithm [58] to minimize the NRMSE between the original outputs and the predictions on the validation set. All training, validation, and prediction processes are conducted on a standard PC with a Core i7-8700 central processing unit (CPU) and 16 G memory.

4 Predictability of test data

4.1 Field test

After the successful implementation and data preparation, we illustrate the capability of our machine learning algorithms. As is shown in Figure 1(a) (I), the external forces acting on the bogie frame of a motor car, including four vertical and four lateral forces, are used as inputs. The corresponding maximum principal tensile stress at the critical point accounting for fatigue failure [42], as is shown in Figure 1(a) (II), is the output of the dynamic system. We note that different types of loads, such as stress, strain, force, moment, and acceleration, may be measured at a certain point; however, all of them should be converted into stress for fatigue analysis.

On-track tests were performed on the same CRH-380A EMU platform under full operating conditions. All force and stress signals are collected through the Somat eDAQ data acquisition system of Hottinger Baldwin Messtechnik (HBM) Company. To ensure the authenticity of the acquired data, we set the sampling frequency to 500 Hz, which is approximately ten times higher than the potentially activated natural frequencies of the point. Raw data from on-track tests are converted into force and stress histories using the nCode 9.0 software of HBM Company [59]. A more detailed description of the on-track test system and the raw data preprocessing is provided in Appendix B in the Supporting Information.

4.2 Predictions with test data

First, to illustrate the benefit of the segmentation and randomization preprocessing, we use the force and corresponding stress histories of the bogie frame of a motor car measured between Beijing and Wuhan of the Beijing–Guangzhou PDL to generate a database with a length of 8567000 time steps, where the first 4000000 time steps are used for training while the rest are used for validation. Here, a deep learning model consisting of 2 GRU layers with 100 cells in each layer and 1 FC layer is employed for data training, and the effective length of each segmented subsequence is set to 1000 while the preceding time steps are 100. In our investigation, the batch size L_b used for model training is constant, i.e., L_b=40. Notably, both training and validation sets are segmented, but only the training set is randomized. We illustrate in Figure 4 the influence of segmentation and randomization of long-duration sequences on the predictability of the data-driven model for the database from on-track tests. As is shown in Figure 4, the segmented and randomized preprocessed database results in lower NRMSEs of predictions on the training and validation sets compared with both original and merely segmented preprocessed ones. Consequently, the segmentation of long-duration sequences could lead to a significant improvement in the predictive accuracy of the data-driven model and, combined with randomization preprocessing, could further enhance the predictability and trainability of the model.

For predictions with test data, the entire database generated from the Beijing–Guangzhou and Beijing–Taiyuan PDLs is divided into four subsets. Notably, the Beijing–Guangzhou PDL, given its long distance, occupies three of the four subsets, namely, Beijing–Xinxiang, Xinxiang–Wuhan, and

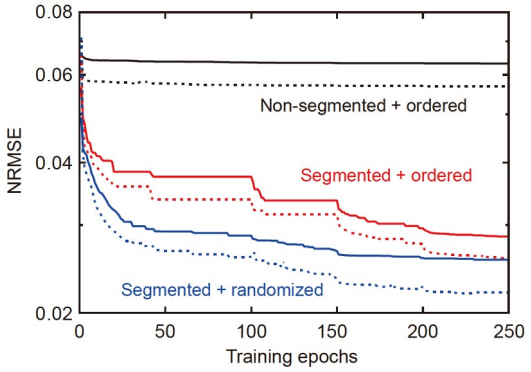

Figure 4 Effect of segmentation and randomization preprocessing on the performance of the data-driven model (the dashed lines denote the training sets and the solid lines denote the validation sets). Here, only the update of model parameters that leads to a smaller normalized root mean square error (NRMSE) of predictions on the validation set is recorded at each epoch.

Wuhan–Guangzhou. Among the four subsets, Beijing–Xinxiang and Xinxiang–Wuhan are used as "seen datasets" and labeled as the training and validation sets, respectively, while Wuhan–Guangzhou and Beijing–Taiyuan are used as "unseen datasets" for prediction and labeled as prediction sets 1 and 2, respectively, as is described in detail in Appendix C and Figure S3(a)–(c). All subsets are segmented, but only the training set (i.e., Beijing–Xinxiang) is randomized to improve trainability and predictability.

Here, the effective length of a subsequence is 1000 time steps, which leads to high predictive accuracy (Appendix D and Table S2). The deep learning model is composed of 2 GRU layers with 100 cells in each layer. Our exploration reveals that 250 epochs of training with the Beijing–Xinxiang dataset lead to a minimum NRMSE of predictability when applying the model to the Xinxiang–Wuhan validation set. In Appendix D, Figures S4(a), (b), and S5, we provide the details of the hyperparameter analyses of the network.

We obtain the predicted stress histories by feeding the force history from different datasets into the trained model. The NRMSEs of the predictions using the training set, validation set, prediction set 1, and prediction set 2 are 0.0218, 0.0255, 0.0331, and 0.0342, respectively. In Figure 5(a)–(d), we show snippets of the original output from the tests and the predicted output from the trained model of the four datasets, and the corresponding loci of the snippets are labeled in the inset at the top. Both the amplitudes and phases of the predicted stress histories match the original ones. In particular, the trained model accurately captures nearly all peaks and valleys in the stress histories for both seen and unseen datasets.

The overall accuracy of the trained model across the entire temporal space is evaluated by analyzing the normalized error distribution of predictions, as is illustrated in Figure 6(a). Notably, the errors of the four datasets are approximately ±0.05, with confidence levels of 97.11%, 84.83%, 90.36%, and 86.36%. The predictability of the "unseen datasets" is better than that of the validation set from the "seen datasets". More importantly, for two operating speed levels on two PDLs, i.e., 300 km/h (maximum speed) for Beijing–Guangzhou and 200 km/h (maximum speed) for Shijiazhuang–Taiyuan, the same learning model demonstrated excellent predictability, which is pivotal for its generalization to other PDLs and operating speed levels.

To further explore the predictability of our data-driven learning algorithm from a statistical perspective, we compare the stress spectra (stress distribution using the rainflow counting method [60]) predicted by the model with those obtained through experiments. As is shown in Figure 6(b)–(e), despite the slight deviations from the "unseen datasets" when the stress amplitudes are small, the predicted stress spectra from all datasets are consistent with the test ones. With the known stress amplitudes and frequencies of a load spectrum, we calculate the damage D in the crucial site of the bogie frame based on the elementary Miner's linear cumulative damage law, as follows [61]:

$$D = \sum_{i=1}^{N_s} \frac{f_i(\sigma_i)^{m_s}}{C}, \tag{14}$$

where N_s denotes the total number of stress steps, $\{\sigma_1, \cdots, \sigma_i, \cdots, \sigma_{N_s}\}$ denote the stress amplitudes after classification, f_i denotes the frequency of the i-th step, and m_s denotes the slope of the S-N curve. Here, we take welded steel joint as the model case. The welded joint can endure approximately 2000000 cycles under a stress amplitude of

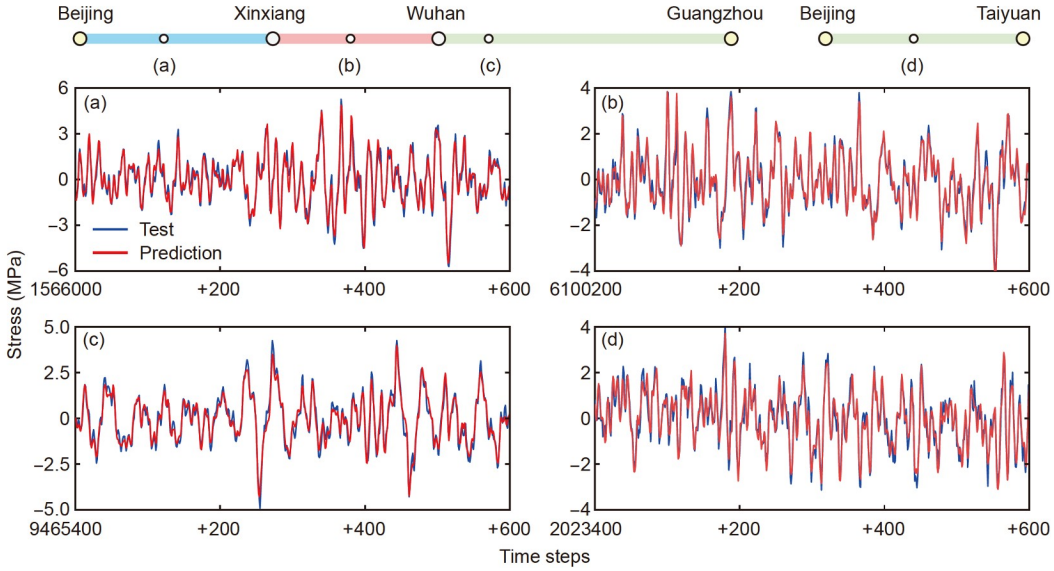

Figure 5 Predictability of the data-driven model. Comparisons of snippets of stress histories from tests and those predicted with the trained model, arbitrarily extracted from the (a) training set, (b) validation set, (c) prediction set 1, and (d) prediction set 2.

70 MPa and has a reliability of 99%, with m_s=3.5 and C=5.74×10^{12}. We show in Table 1 the comparison of damages calculated from the measured stress spectra and the predicted ones obtained by our data-driven model on the four datasets. The learning model could predict the damages with an acceptable accuracy on all four datasets, and the predictability of the "unseen datasets" is again better than that of the validation set from the "seen datasets".

Notably, the stress levels on prediction set 1 are generally higher than those on the training set because of the more complicated mountainous route conditions from Wuhan to Guangzhou of the Beijing–Guangzhou PDL. Despite the slower operating speed, the stress levels in the latter half of prediction set 2 increase slightly because the tracks are switched from ballastless track to ballasted track. These features are demonstrated in the stress spectra shown in Figure 6(b)–(e). As was discussed previously, our data-driven model performs better on both "unseen datasets" than the validation set from the "seen datasets" in terms of damage prediction and the proportion of stress history predictions with a normalized error between −0.05 and 0.05. The model has good extrapolation capability for harsh line conditions with consequentially higher load levels. Other extrinsic factors, such as weather conditions and loading capacity, which are difficult to obtain from the current in-field experiments, do not alter the accuracy of the prediction as long as the loading transmission path remains the same.

In terms of computation cost, the entire training process in this work could be completed in tens of hours on a standard PC with a Core i7-8700H CPU and 16 G memory, while predictions of long-duration load spectrum using the trained model only take a few minutes. The computational complexity of the training process is positively related to the number of samples on the training set and the length of the subsequence. In contrast to the conventional finite-element-based calculations shown in Table S1 of Appendix A, the current computation cost is indeed negligible. Hence, we believe that the present learning method, combined with limited experimental data, may provide an efficient method to reproduce precise load spectra, i.e., both load-time history and load amplitude-frequency statistics, for fatigue analysis of engineering structures.

In general, forces acting on the bogie frame may be conveniently monitored. The force-time sequences are the input of our trained model to predict the stress-time sequences. Notably, the time sequences of all forces transmitted to a component need not be obtained to achieve high-fidelity mapping of the stress-time histories of vulnerable sites, which is of significant advantage in engineering practice and outperforms traditional computational methods that require a full description of boundary conditions for a well-posed boundary value problem. With the predicted stress-time histories, we can extract a stress spectrum and proceed with fatigue analysis.

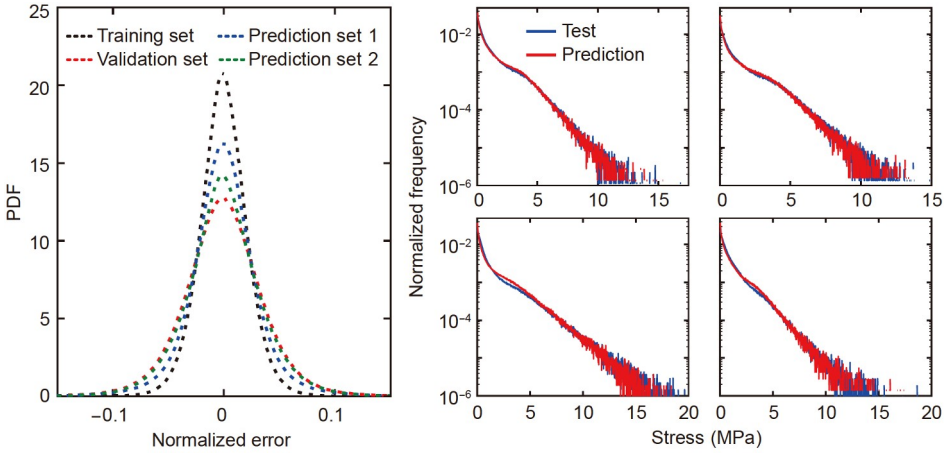

Figure 6 Overall predictability of the data-driven model. (a) Normalized error distributions of the four predictions. Stress spectra of the four datasets corresponding to the (b) training set, (c) validation set, (d) prediction set 1, and (e) prediction set 2 (the blue lines denote the test data and red lines denote the predictions).

Table 1 Damages in the crucial site of the bogie frame calculated from the measured stress spectra and the predicted ones obtained by the learning model, along with their relative errors

Dataset	Training set	Validation set	Prediction set 1	Prediction set 2
Test	4.696×10^{-7}	4.451×10^{-7}	1.725×10^{-6}	4.920×10^{-7}
Prediction	4.583×10^{-7}	4.145×10^{-7}	1.680×10^{-6}	4.707×10^{-7}
Relative error	−2.42%	−6.87%	−2.61%	−4.33%

5 Segmentation strategy for stress learning from dissipative systems

We demonstrate in previous sections the success in learning the load spectrum of highly dissipative structures in a nonlinear system. Now, we aim to examine the efficiency and effectiveness of the computational method for two considerations, i.e., (1) the length of the redundant data employed in our learning algorithm (Figure 3(b)) to eliminate the possible influence of nonlinear dissipation and (2) the length of the segmentation used to update the model to achieve better predictability at a low computation cost.

We show in Figure 7(a) snippets of the original stresses from the field test (Beijing–Baoding, top right inset) and the predicted stresses from the learning model trained in Sect. 4. For an arbitrary independent subsequence, the predicted stress at the beginning does not match well with the experimental test (the shaded regions in Figure 7(a)) because of the lack of inherited information from previous time steps in the system with a viscous response. Therefore, redundant data segmentation is used in our learning algorithm to eliminate the influence of preceding time steps. The length of this redundancy can be determined by examining the predictability of the learning algorithm. Figure 7(a) and (b) show that the characteristic dissipation time scale in the current dynamic system is approximately 60 steps, which corresponds to 0.12 s.

To determine the "optimal" length of subsequences, we employ a sufficiently long force and stress histories from Beijing to Baoding of the Beijing–Guangzhou PDL to produce a database that covers 400000 time steps, in which the first 300000 time steps are used for training while the rest is used for validation. Based on the numerical experiments and hyperparameter analyses, we use a learning model consisting of 2 GRU layers with 40 cells in each layer for data training. We adopt segmented subsequences of different time steps, i.e., from 20 to 5000, to train the learning model with the same architecture. By exploring the training efficiency and prediction accuracy of the trained model, we can deduce the relationship between the subsequence length and the characteristic dissipation time of the system. The redundancy is now set to 100 steps, which is greater than the minimum requirement of 60 time steps in our previous investigation.

Ideally, the longer the subsequence length is, the better the prediction. In practice, a low efficiency can be observed for a long subsequence length. We show in Figure 7(c) the tradeoff between predictability and computation time for segmented subsequences with different subsequence lengths, from which an "optimal" length is obtained. As is shown in Figure 7(c), the increase of the subsequence length could initially (before 200 time steps) lead to a considerable improvement in both predictability and training efficiency of the model, which is due to the reducing weight of "redundancy" in computation. A further increase of subsequence length leads to a slow improvement in predictability but a high computation cost, due to a deeper time sequence in the learning algorithm, which corresponds to a greater N in Figure 2(a): The computational time using the backpropagation through time algorithm is proportional to $O(N)$. The competition between these two factors leads to the "optimal" sub-

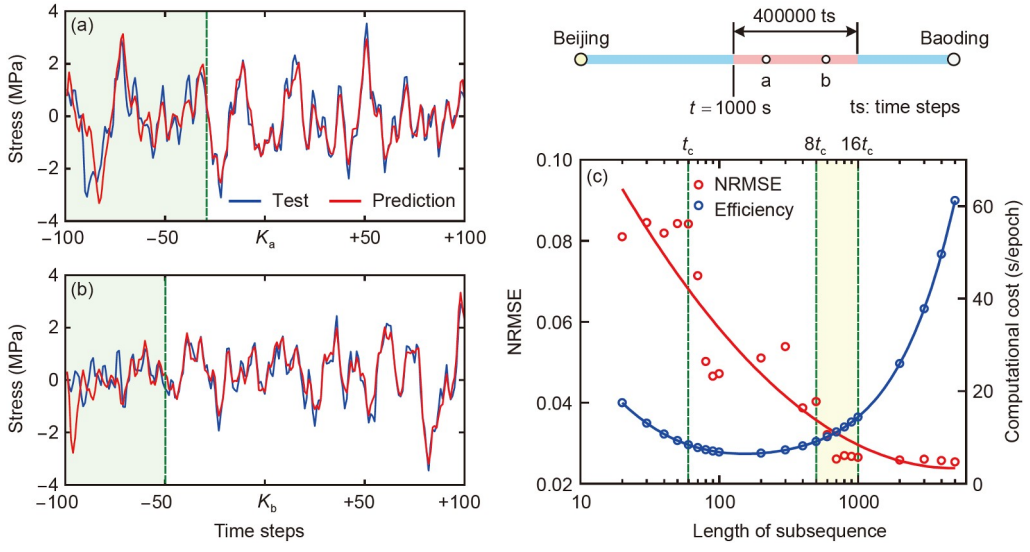

Figure 7 Correlation between the length of the segmented subsequence and the characteristic dissipation time of the target system. ((a) and (b)) Snippets of stress histories from the tests and those predicted with the trained model, respectively. Here, negative abscissas denote the preceding time steps, while the positive abscissas denote the effective steps in the segmented subsequences. Moreover, K_a and K_b are the starting effective steps for the two subsequences. Then, the characteristic dissipation time of the system t_c is deduced from the significant difference in the phase and amplitude between the field test data and the predictions. (c) Tug of war between predictability and computation time of the learning algorithm leads to the "optimal" subsequence length, i.e., approximately 8–16 times the characteristic dissipation time.

sequence length, i.e., approximately 8–16 times the characteristic dissipation time (the shaded region in Figure 7(c)) of the target system.

6 Conclusions

Although load spectra are indispensable and ubiquitously employed for the fault diagnosis and life assessment of engineering systems, precise spectra composed of an adequate amount of data are difficult to obtain. Traditional computational mechanics requires discretizing regions considerably smaller than the characteristic size of defects and possible deformation concentration regions, leading to the curse of complexity. By building an intelligent model to learn from limited experimental data and observations, the deep learning method has the potential to circumvent such challenges by ensuring reliable mapping from forces to stress fields, where forces are easy to get while stress fields are difficult to obtain. Successful applications of this idea to complex systems are still scarce. Thus, standardized benchmarks and metrics for neural network pruning are necessary [62]. With this practice, we consider a well-known complex system—an HST, to learn the time history of stress at a certain point of the bogie frame, starting from the time histories of four vertical and four lateral forces transmitted from the wheelsets and acting on the bogie frame. Such a relationship between acting force and stress would involve intensive computation endeavors as the eight forces are just a small portion of the boundary conditions, and stresses at any point are outcomes of collective motion and deformation of bogie frames, carriages, transmission and connecting components, and aerodynamics. A routine normally leads to computation disaster. We demonstrate that the learning algorithm could effectively predict the load spectrum of a complex dynamic system at a low computation cost. Notably, the trained model is more accurate than traditional simulation techniques because the training process is directly driven by the test data without any model simplification and mechanical assumption in the context of sequence learning.

Generally, the success of machine learning is dependent on the quantity and quality of available data. Our data-driven model, fed with a small amount of data for supervised learning tasks, exhibits high fidelity in predicting the subsequent load spectrum with sufficiently large duration and excellent generalization capability under different working conditions of the dynamic system, such as operating speed levels and routes. This result is pivotal in fields where experimental data are scarce or at a high cost. In terms of quality, the model exhibits excellent predictability, computability and stability [63] for the database containing approximately 1000000 km of service, which includes multiple operating routes and different speed levels from the same CRH-380A EMU platform.

As iterative solutions of physical equations for high resolutions at spatial and temporal scales are not needed, the deep learning strategy provides a promising alternative to the current load spectrum calculation scheme for general dynamic systems by directly building a bridge between history-dependent load inputs and response outputs through data learning. The refined segmentation and randomization strategy for long-duration historical data and historical learning in current algorithms enable machines to "learn" by using smaller datasets to realize a load spectrum of high fidelity in long-duration service—a fundamental development that would eliminate the need to access immense datasets.

Although we use a particular engineering system to demonstrate the effectiveness of the algorithm, the unifying framework that enables the synergy to save orders of magnitude of time in monitoring the load spectra of complex engineering structures is fundamental and universal and can serve as a benchmark and be employed to model the time-dependent mechanical behavior of many other dynamic systems where fatigue fracture is of paramount concern.

This work was supported by the Basic Science Center of the National Natural Science Foundation of China for "Multiscale Problems in Nonlinear Mechanics" (Grant No. 11988102), the National Key Research and Development Program of China (Grant Nos. 2017YFB0202800 and 2016YFB1200602), the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDB22020200), and the Science Challenge Project (Grant No. TZ2018002).

Supporting Information

The supporting information is available online at tech.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

1 Butler K T, Davies D W, Cartwright H, et al. Machine learning for molecular and materials science. Nature, 2018, 559: 547–555
2 Rovinelli A, Sangid M D, Proudhon H, et al. Predicting the 3D fatigue crack growth rate of small cracks using multimodal data via Bayesian networks: *In-situ* experiments and crystal plasticity simulations. J Mech Phys Solids, 2018, 115: 208–229
3 Tyulyukovskiy E, Huber N. Neural networks for tip correction of spherical indentation curves from bulk metals and thin metal films. J Mech Phys Solids, 2007, 55: 391–418
4 Shi Z, Tsymbalov E, Dao M, et al. Deep elastic strain engineering of bandgap through machine learning. Proc Natl Acad Sci USA, 2019, 116: 4117–4122
5 Lu L, Dao M, Kumar P, et al. Extraction of mechanical properties of materials through deep learning from instrumented indentation. Proc Natl Acad Sci USA, 2020, 117: 7052–7062
6 Wen J, Zou Q, Wei Y. Physics-driven machine learning model on temperature and time-dependent deformation in lithium metal and its finite element implementation. J Mech Phys Solids, 2021, 153: 104481
7 Xiong J, Zhang T Y, Shi S Q. Machine learning of mechanical properties of steels. Sci China Tech Sci, 2020, 63: 1247–1255
8 Cai S Z, Mao Z P, Wang Z C, et al. Physics-informed neural networks

(PINNs) for fluid mechanics: A review. Acta Mech Sinica PRC, 2021, 37: 1–12
9 Raccuglia P, Elbert K C, Adler P D F, et al. Machine-learning-assisted materials discovery using failed experiments. Nature, 2016, 533: 73–76
10 Jha D, Choudhary K, Tavazza F, et al. Enhancing materials property prediction by leveraging computational and experimental data using deep transfer learning. Nat Commun, 2019, 10: 5316
11 Schmidt J, Marques M R G, Botti S, et al. Recent advances and applications of machine learning in solid-state materials science. npj Comput Mater, 2019, 5: 83
12 Sun W, Zheng Y, Yang K, et al. Machine learning-assisted molecular design and efficiency prediction for high-performance organic photovoltaic materials. Sci Adv, 2019, 5: eaay4275
13 Senior A W, Evans R, Jumper J, et al. Improved protein structure prediction using potentials from deep learning. Nature, 2020, 577: 706–710
14 Agius R, Brieghel C, Andersen M A, et al. Machine learning can identify newly diagnosed patients with CLL at high risk of infection. Nat Commun, 2020, 11: 363
15 Jiao W, Atwal G, Polak P, et al. A deep learning system accurately classifies primary and metastatic cancers using passenger mutation patterns. Nat Commun, 2020, 11: 728
16 Segler M H S, Preuss M, Waller M P. Planning chemical syntheses with deep neural networks and symbolic AI. Nature, 2018, 555: 604–610
17 Biamonte J, Wittek P, Pancotti N, et al. Quantum machine learning. Nature, 2017, 549: 195–202
18 Rocchetto A, Aaronson S, Severini S, et al. Experimental learning of quantum states. Sci Adv, 2019, 5: eaau1946
19 Zhang L, Lin D Y, Wang H, et al. Active learning of uniformly accurate interatomic potentials for materials simulation. Phys Rev Mater, 2019, 3: 023804
20 Zhang Y, Wang H, Chen W, et al. DP-GEN: A concurrent learning platform for the generation of reliable deep learning based potential energy models. Comput Phys Commun, 2020, 253: 107206
21 DeVries P M R, Thompson T B, Meade B J. Enabling large-scale viscoelastic calculations via neural network acceleration. Geophys Res Lett, 2017, 44: 2662–2669
22 Holtzman B K, Paté A, Paisley J, et al. Machine learning reveals cyclic changes in seismic source spectra in Geysers geothermal field. Sci Adv, 2018, 4: eaao2929
23 Reichstein M, Camps-Valls G, Stevens B, et al. Deep learning and process understanding for data-driven Earth system science. Nature, 2019, 566: 195–204
24 Kopp W, Monti R, Tamburrini A, et al. Deep learning for genomics using Janggu. Nat Commun, 2020, 11: 3488
25 Lei Y, Yang B, Jiang X, et al. Applications of machine learning to machine fault diagnosis: A review and roadmap. Mech Syst Signal Process, 2020, 138: 106587
26 Wang R X, Gao X, Gao J M, et al. An artificial immune and incremental learning inspired novel framework for performance pattern identification of complex electromechanical systems. Sci China Tech Sci, 2020, 63: 1–13
27 Martins T, Infante V, Sousa L, et al. Numerical and experimental study of aircraft structural health. Int J Fatigue, 2020, 132: 105348
28 Zuo H, Bi K, Hao H, et al. Influence of earthquake ground motion modelling on the dynamic responses of offshore wind turbines. Soil Dyn Earthq Eng, 2019, 121: 151–167
29 Zhou Y, Sun L. Effects of high winds on a long-span sea-crossing bridge based on structural health monitoring. J Wind Eng Indust Aero Dyn, 2018, 174: 260–268
30 Chen D, Xiao Q, Mou M, et al. Study on establishment of standardized load spectrum on bogie frames of high-speed trains. Acta Mech Sin, 2019, 35: 812–827
31 Anderson P W. More is different: Broken symmetry and the nature of the hierarchical structure of science. Science, 1972, 177: 393–396
32 Diamanti K, Soutis C. Structural health monitoring techniques for aircraft composite structures. Prog Aerosp Sci, 2010, 46: 342–352
33 Fujino Y, Siringoringo D M, Ikeda Y, et al. Research and implementations of structural monitoring for bridges and buildings in Japan. Engineering, 2019, 5: 1093–1119
34 Raissi M, Yazdani A, Karniadakis G E. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. Science, 2020, 367: 1026–1030
35 Zhang R, Chen Z, Chen S, et al. Deep long short-term memory networks for nonlinear structural seismic response prediction. Comput Struct, 2019, 220: 55–68
36 Zhang W J, Qin J, Mei F, et al. Short-term power load forecasting using integrated methods based on long short-term memory. Sci China Tech Sci, 2020, 63: 614–624
37 Jiang Y H, Yu Y F, Huang J Q, et al. Li-ion battery temperature estimation based on recurrent neural networks. Sci China Tech Sci, 2021, 64: 1335–1344
38 Wu P, Sun J, Chang X, et al. Data-driven reduced order model with temporal convolutional neural network. Comput Methods Appl Mech Eng, 2020, 360: 112766
39 Mozaffar M, Bostanabad R, Chen W, et al. Deep learning predicts path-dependent plasticity. Proc Natl Acad Sci USA, 2019, 116: 26414–26420
40 Gorji M B, Mozaffar M, Heidenreich J N, et al. On the potential of recurrent neural networks for modeling path dependent plasticity. J Mech Phys Solids, 2020, 143: 103972
41 Bergen K J, Johnson P A, de Hoop M V, et al. Machine learning for data-driven discovery in solid Earth geoscience. Science, 2019, 363: aau0323
42 Zhang S G. Study on testing and establishment method for the load spectrum of bogie frame for high-speed trains. Sci China Ser E, 2008, 51: 2142–2151
43 Ma S, Sun S, Wang B, et al. Estimating load spectra probability distributions of train bogie frames by the diffusion-based kernel density method. Int J Fatigue, 2020, 132: 105352
44 Yuan Z, Chen X, Ma L, et al. A segmented load spectrum model for high-speed trains and its inflection stress as an indicator for line quality. Int J Fatigue, 2021, 148: 106221
45 Klemenc J, Fajdiga M. Prediction of loading spectra under diverse operating conditions by a localised basis function neural network. Int J Fatigue, 2005, 27: 555–568
46 Yang Z, Song Z, Zhao X, et al. Time-domain extrapolation method for tractor drive shaft loads in stationary operating conditions. Bio Syst Eng, 2021, 210: 143–155
47 Zou Y, Zhang Y, Mao H. Fault diagnosis on the bearing of traction motor in high-speed trains based on deep learning. Alex Eng J, 2021, 60: 1209–1219
48 Chen J, Feng X, Jiang L, et al. State of charge estimation of lithium-ion battery using denoising autoencoder and gated recurrent unit recurrent neural network. Energy, 2021, 227: 120451
49 Veeramsetty V, Reddy K R, Santhosh M, et al. Short-term electric power load forecasting using random forest and gated recurrent unit. Electr Eng, 2022, 104: 307–329
50 Patil S A, Raj L A, Singh B K. Prediction of IoT traffic using the gated recurrent unit neural network- (GRU-NN-) based predictive model. Secur Commun Netw, 2021, 2021: 1425732
51 Jordan M I, Mitchell T M. Machine learning: Trends, perspectives, and prospects. Science, 2015, 349: 255–260
52 Clough R W, Penzien J. Dynamics of Structures. 3rd ed. Berkeley: Computers & Structures Inc., 2003
53 Hochreiter S. Untersuchungen zu dynamischen neuronalen Netzen (in German). Dissertation for Master's Degree. Munich: Technische Universität München, 1991
54 Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput, 1997, 9: 1735–1780
55 Cho K, Van Merrienboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine

translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha: Association for Computational Linguistics, 2014. 1724–1734
56 Murdoch W J, Singh C, Kumbier K, et al. Definitions, methods, and applications in interpretable machine learning. Proc Natl Acad Sci USA, 2019, 116: 22071–22080
57 Chollet F. Keras Version: 2.3.0, 2019. https://keras.io. (accessed on 2020-06-28)
58 Kingma D P, Ba J. Adam: A method for stochastic optimization. arXiv: 1412.6980
59 nCode 9.0. Hottinger Baldwin Messtechnik Company, Darmstadt, Germany, 2013
60 Amzallag C, Gerey J, Robert J, et al. Standardization of the rainflow counting method for fatigue analysis. Int J Fatigue, 1994, 16: 287–293
61 Miner M A. Cumulative damage in fatigue. J Appl Mech, 1945, 12: A159–A164
62 Blalock D, Ortiz J J G, Frankle J, et al. What is the state of neural network pruning? arXiv: 2003.03033
63 Yu B, Kumbier K. Veridical data science. Proc Natl Acad Sci USA, 2020, 117: 3920–3929