Computational Mechanics (2019) 64:365–379
https://doi.org/10.1007/s00466-019-01728-w

ORIGINAL PAPER

Derivation of heterogeneous material laws via data-driven principal component expansions

Hang Yang[1] · Xu Guo[1,2] · Shan Tang[1,2] · Wing Kam Liu[3]

Received: 9 February 2019 / Accepted: 3 May 2019 / Published online: 22 May 2019

Abstract
A new data-driven method that generalizes experimentally measured and/or computational generated data sets under different loading paths to build three dimensional nonlinear elastic material law with objectivity under arbitrary loadings using neural networks is proposed. The proposed approach is first demonstrated by exploiting the concept of representative volume element (RVE) in the principal strain and stress spaces to numerically generate the data. A computational data-training algorithm on the generalization of these principal space data to three dimensional objective isotropic material laws subjected to arbitrary deformation is given. To validate these data-driven derived material laws, large deformation and buckling analysis of nonlinear elastic solids with reference material models and engineering structure with microstructure are performed. Numerical experiments show that only seven sets of data under different stress loading paths on RVEs are required to reach reasonable accuracy. The requirements for constitutive law such as objectivity are preserved approximately. The consistent tangent modulus is also derived. The proposed approach also shows a great potential to obtain the material law between different scales in the multiscale analysis by pure data.

Keywords Computational data-driven · Artificial neural network · 3D objective material laws · Principal strain and stress space · Engineering structure with microstructure

1 Introduction

Material laws describe the mechanical response between stress and stretch. They play a key role to solving the boundary-value problems in mechanics. The material law can determine if the solved boundary-value problem can match or explain experimental observations and its application in complex loading modes or extreme environment. The prevailing way in the past is to calibrate the material law with experimentally observed data. In this paper, we replace this phenomenological component of the boundary-value problem by a data-driven model.

Recently, Kirchdoerfer and Ortiz [1–3], Conti et al. [4], Leygue et al. [5], Chinesta et al. [6] have proposed a new paradigm to bypass the empirical fitting of the material law and formulate the calculation directly from experimental and/or computational data for elastic material and viscoelastic material from quasi-static loading to dynamics loadings. A strategy to minimize the discrepancy between experimental data and predicted response through the optimization is proposed under the constraint in the phase space of both stress and stretch. For multiple-dimensional problems with the rotation of the materials, the convergence is usually slow because the rotation of the material plays an important role. Another issue is that many materials have microstructures spanning several orders in magnitude. For example, the two dimensional materials such as graphene nowadays are often mixed into soft materials such as silicone rubber to increase both toughness and strength. The data obtained from the uniaxial tension and/or compression are not enough to capture the material behaviors in general stress states.

✉ Xu Guo
guoxu@dlut.edu.cn

✉ Shan Tang
shantang@dlut.edu.cn

[1] State Key Laboratory of Structural Analysis for Industrial Equipment, Department of Engineering Mechanics, Dalian University of Technology, Dalian 116023, People's Republic of China

[2] International Research Center for Computational Mechanics, Dalian University of Technology, Dalian 116023, People's Republic of China

[3] Department of Mechanical Engineering, Northwestern University, Evanston, IL 60208, USA

Another approach parallel to those of Ortiz and collaborators is to use the supervised learning to train material laws. This approach can be dated back to the 1990s and categorized into two types. One is that the material model is primary known but the parameters involved in the model needs to be identified. This parameter identification can be carried out by solving an optimization or constrained optimization problem to minimize the objective function, which is usually defined as a metric to measure the discrepancy between the benchmark (usually experimental data) and predictions (usually numerical simulations) [7–9]. Machine learning can be used to accelerate the identification process and is also widely employed. We just name a few in our view. Al-Haik et al. [10] developed a model based on an artificial neural network (ANN) to predict the stress relaxation of the polymer matrix composites. Zopf and Kaliske [11] coupled the neural network with microsphere model, which can take into the microstructure of polymer chains into account. The pure elastic response and inelastic material behavior are obtained via Recurrent Neural Network (RNN). For the other type, nothing is known about the material model in prior. Ghaboussi and Sidarta [12] first employed the artificial feed-forward neural network to train their experimental data for the material model. The stress increment is trained with the input of strain increment and the state variables in the previous steps by Nested Artificial Neural Networks (NANN). Their work is limited to the small deformation regime. The training for nonlinearly elastic–plastic material is also proposed and the tangent modulus at the each time step is derived [13]. The authors claim that the derived tangent modulus is independent of the specific material response. However, in finite deformation, objectivity of the material laws and loading–unloading of the material are not considered in their training. The complexity of the microstructured solids inspires the data-driven model to link information across multiple scales via offline training [14–16]. The training approach incorporating the microstructural data and direct numerical simulations (DNS) with the representative volume element (RVE) has proposed, aiming at unifying data-driven framework for designing and modeling of materials and structures. A self-consistent clustering analysis (SCA) method is also proposed to reduce computational costs and avoid the curse of dimensionality in the offline training [17–22]. SCA is an efficient tool for concurrent analysis on materials with multiscale structures. Wang and Sun [23] has extended the ANN training strategy from the single-physics solid mechanics problem to hydro-mechanical coupling problem of geological materials. These data-driven approachs can accelerate the process for the engineering design. Recently, it is further developed for real-time topology optimization [24].

In practice, it is hard to generate the stress-stretch data under arbitrary deformation modes/paths to construct the material laws. However, it is relatively easy to obtain the experimental or numerically generated data of materials under the principal deformation modes (uniaxial tension, biaxial loading or triaxial loading etc.). The experiments in the principal space are consistent with the spirit of the principal component analysis (PCA) in data analysis. PCA is a technique which is widely used to convert a set of possibly correlated data into a set of linearly uncorrelated data, called principal component [25]. Further, the generated data is usually trained by neural network as a black box. Then building a material law is a pure process from data to data. It is questionable if the requirements for material law established in continuum mechanics such as objectivity are preserved.

Although it is possible to generate the data through physical experiments in principal space, we will demonstrate data generation by numerical experiments with the help of the representative volume element (RVE) approach. The principal stretch/stress is imposed on a representative volume element (RVE) and the principal stress/stretch response is generated. In this paper, we resort to the established mechanics theory to build the material law for finite deformation nonlinear isotropic elastic materials based on the data generated by RVE. The organization of the paper is as follows. In Sect. 2, two widely used models for hyperelastic material is employed to generate the data in principal stretch-stress through RVE: neo-Hookean model [26] and Arruda–Boyce model [27] based on the homogenization theory. We describe in Sect. 3 the mechanics-theory-based supervised machine learning techniques to train the material law. The details on training procedure, the selection of machine learning method and related techniques, are given in "Appendix B". Section 4 gives the computational results of the trained material laws to simulate the structure in three-dimensional geometry under different loading modes with and without microstructure. Finally a short conclusion is drawn in Sect. 5.

2 Data generation

Heterogeneous material usually consists of different phases. RVE is often used to build the material law for these heterogeneous materials. That is, RVE is used to define the relationship between the imposed homogeneous deformation gradient $\bar{\mathbf{F}}$ and the homogenous second Piola-Kirchhoff (PK) stress $\bar{\mathbf{S}}$. The data generated by RVE computation can be trained for material law. Due to the composition of different phases, a constitutive response of each phase should be first given for RVE computations. The readers who are familiar with data generation with RVE can skip this section.

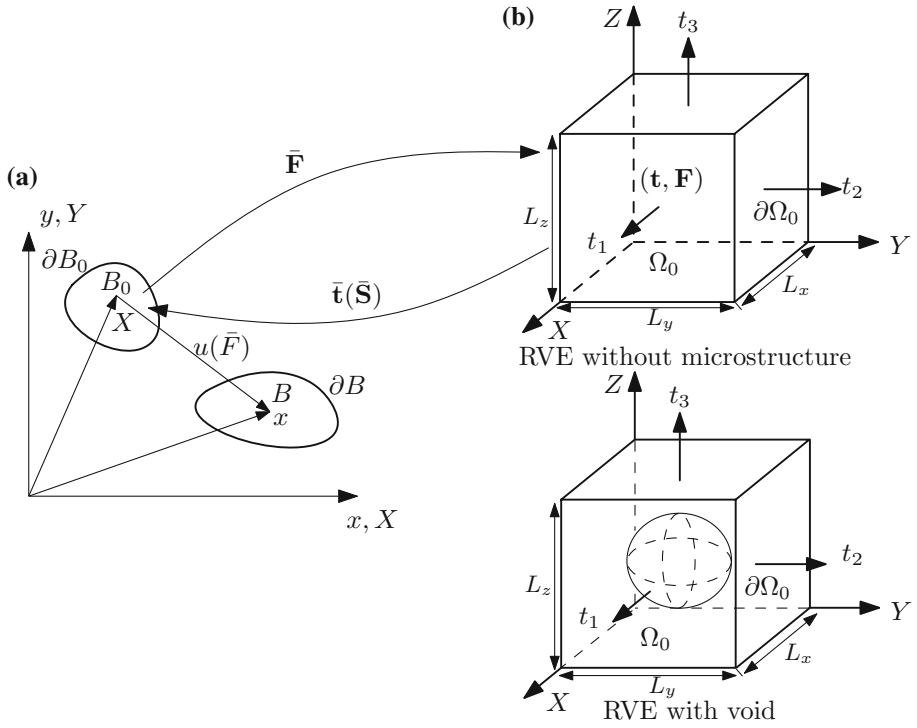

Fig. 1 **a** A material point X with/without microstructure in the deformable body moves from initial configuration to current configuration. A RVE (Representative Volume Element) can be thought as being attached at each material point at the macroscale. **b** Two RVEs without microstructure and with void to generate the stress-strain data in principal spaces to be trained for material laws

2.1 Homogenization of RVE in principal space

A RVE can be thought as being attached at each material point (X, Y, Z) at the macroscale (Fig. 1a) in the reference configuration. Let Ω_0 be the region occupied by a RVE, consisting of single or multiple phases, in an unstressed reference configuration with bounding surface $\partial\Omega_0$ (Fig. 1b). The RVE is associated with a Cartesian coordinate system with orthogonal frame $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$, which are base vectors in Cartesian coordinates in x, y and z direction respectively. Only widely used cuboid RVE is considered in this work with lengthes of L_x, L_y and L_z in x, y and z direction respectively. The detailed derivation of homogenization on the RVE is given in "Appendix A".

In terms of the displacement vector $\boldsymbol{u}$, the boundary condition on the RVE can be rewritten as

$$\boldsymbol{u} = (\lambda_1 - 1)\, X\mathbf{e}_1 + (\lambda_2 - 1)\, Y\mathbf{e}_2 + (\lambda_3 - 1)\, Z\mathbf{e}_3 \tag{1}$$

It can be seen that $(\lambda_1, \lambda_2, \lambda_3)$ are the principal stretches which can be computed based on the imposed displacement on RVE. Correspondingly, the homogenized second PK stress can be computed by

$$\bar{\mathbf{S}} = S_1\mathbf{e}_1 \otimes \mathbf{e}_1 + S_2\mathbf{e}_2 \otimes \mathbf{e}_2 + S_3\mathbf{e}_3 \otimes \mathbf{e}_3 \tag{2}$$

with

$$S_1 = \frac{1}{\lambda_1 L_y L_z} \int_0^{L_y} \int_0^{L_z} t_1 dY dZ, \tag{3}$$

$$S_2 = \frac{1}{\lambda_2 L_x L_z} \int_0^{L_x} \int_0^{L_z} t_2 dX dZ, \tag{4}$$

$$S_3 = \frac{1}{\lambda_3 L_x L_y} \int_0^{L_x} \int_0^{L_y} t_3 dX dY \tag{5}$$

Here t_1, t_2 and t_3 are traction forces on the outer boundary of the RVE and can be numerically computed from the nodal reaction force. It has shown by many studies that the proportional ratio of Cauchy stresses can be realized even under the displacement controlled loading [28–31].

2.2 Data of stress-strain in principal space by RVE

Based on the homogenization theory, the data of stress-strain in principal space can be generated. Two RVEs are used in the present work (Fig. 1b). One is without any microstructure and the other with a void at the center. The one without the microstructure is easy for us to compare with the existing models to verify the accuracy and effectiveness of the proposed method. The material law for a RVE with the void is not known in prior. It can help to further test the predicability of the proposed method.

The reference material models of neo-Hookean model for nonlinear elastic material with $\mu = 2$, $D_m = 0.1$ and Arruda–Boyce model with parameters $\mu = 2$, $D_m = 0.1$ and $\lambda_m = 7$ are adopted in this work. The elastic deformation energy of neo-Hookean model and Arruda–Boyce model are

Table 1 14 and 7 sets of loading paths with constant ratios between Cauchy stress components to generate the data of principal components of the second PK stress and principal stretch components

Load path ID	$\bar{\Sigma}_1$	$\bar{\Sigma}_2$	$\bar{\Sigma}_3$	$\{a_1, a_2, a_3\}^{[\alpha,\beta]}$	$\{S_1, S_2, S_3\}^{[\alpha,\beta]}$
1*	$-0.43R$	$-0.50R$	$-0.75R$	Computed	Computed
2	$0.00R$	$-0.50R$	$-0.87R$		
3*	$0.00R$	$0.00R$	$-1.00R$		
4	$0.43R$	$-0.50R$	$-0.75R$		
5*	$0.50R$	$-0.43R$	$-0.75R$		
6	$0.50R$	$0.00R$	$-0.87R$		
7*	$0.50R$	$0.43R$	$-0.75R$		
8	$0.75R$	$-0.43R$	$-0.50R$		
9	$0.75R$	$0.43R$	$-0.50R$		
10*	$0.75R$	$0.50R$	$-0.43R$		
11	$0.75R$	$0.50R$	$0.43R$		
12*	$0.87R$	$0.00R$	$-0.50R$		
13	$0.87R$	$0.50R$	$0.00R$		
14*	$1.00R$	$0.00R$	$0.00R$		

The number of loading path with * indicates the 7 data sets among the 14 data sets. The loading path is defined by Cauchy stress for convenience only, and it is also possible to describe the loading path using other stress or strain measures

$$W = \frac{\mu}{2}(\bar{I}_1 - 3) + \frac{1}{D_m}(J-1)^2$$

and

$$W = \mu\left\{\frac{1}{2}(\bar{I}_1 - 3) + \frac{1}{20\lambda_m^2}(\bar{I}_1^2 - 9) + \frac{11}{1050\lambda_m^4}(\bar{I}_1^3 - 27) + \frac{19}{7000\lambda_m^6}(\bar{I}_1^4 - 81) + \frac{519}{673750\lambda_m^8}(\bar{I}_1^5 - 243)\right\} + \frac{1}{D_m}\left(\frac{J^2-1}{2} - \ln J\right)$$

respectively where $\bar{I}_1 = J^{-\frac{2}{3}}(\lambda_1^2 + \lambda_2^2 + \lambda_3^2)$, $J = \det(\mathbf{F})$ and $\mathbf{F}$ is the deformation gradient. The stress can be derived from the deformation energy based on the classical continuum mechanics [32].

Different proportional loading paths are designed to generate the data by RVE (see Table 1). The loading path can be defined by the controlling parameter R and the ratios $\bar{\Sigma}_i/R$, $(i = 1 \cdots 3)$. Here $\bar{\Sigma}_i$ are the principal Cauchy stress (16). History data of principal components of the second PK stress (S_1, S_2, S_3) with the principal stretch components (a_1, a_2, a_3) on each loading path can be generated by increasing the controlling parameter R from 0 to 1.5μ. Where $a_i = \lambda_i^2$ $(i = 1 \cdots 3)$ and μ is the shear modulus. Each loading history is divided into N_s time steps evenly and N_L loading paths in the principal stress space. N_s is set to 500 and N_L is set to 7 or 14, see Table 1. The data of $\{a_1, a_2, a_3\}^{[\alpha,\beta]}$ and $\{S_1, S_2, S_3\}^{[\alpha,\beta]}$ can be generated through Eqs. (1) and (3) respectively where α represents the loading path while β represents the loading step. It should be commented here that the loading path should be evenly distributed in the stress space as far as possible. The 14 loading cases in the Table 1 may not be the optimal one, but it can train an isotropic material models effectively according to our numerical experiences.

Figure 2 visualizes how the data (a_1, a_2, a_3) and (S_1, S_2, S_3) for loading path ID 3, 7 and 14 in Table 1 at different time steps can be obtained through RVE computation according to Eqs. (1) and (3). Figure 3 shows the generated data of (a_1, a_2, a_3) and (S_1, S_2, S_3) versus the controlling loading parameter R for loading path ID 3, 7, and 14 respectively.

3 Data training and on-line computation

3.1 Data training in principal space by neural networks

Through RVE computation, the data of $\{a_1, a_2, a_3\}^{[\alpha,\beta]}$ and $\{S_1, S_2, S_3\}^{[\alpha,\beta]}$ in the principal space can be generated. Here α represents the loading path $(\alpha = 1 \cdots N_L)$ and β the time step of loading history $(\beta = 1 \cdots N_s)$. The data of $\{a_1, a_2, a_3\}^{[\alpha,\beta]}$ and $\{S_1, S_2, S_3\}^{[\alpha,\beta]}$ can be stored in three-dimensional arrays $a[\alpha, \beta, k]$ and $S[\alpha, \beta, k]$. The dimension of both arrays is $N_L \times N_s \times 3$.

These data will be used to train a data-driven model between $\{a_1, a_2, a_3\}$ and $\{S_1, S_2, S_3\}$ by neural network. It should be commented here that the generated data on each loading path is highly correlated, but data on different load paths is uncorrelated. Therefore, the data from 14 loading paths are evenly distributed in the stress space, which

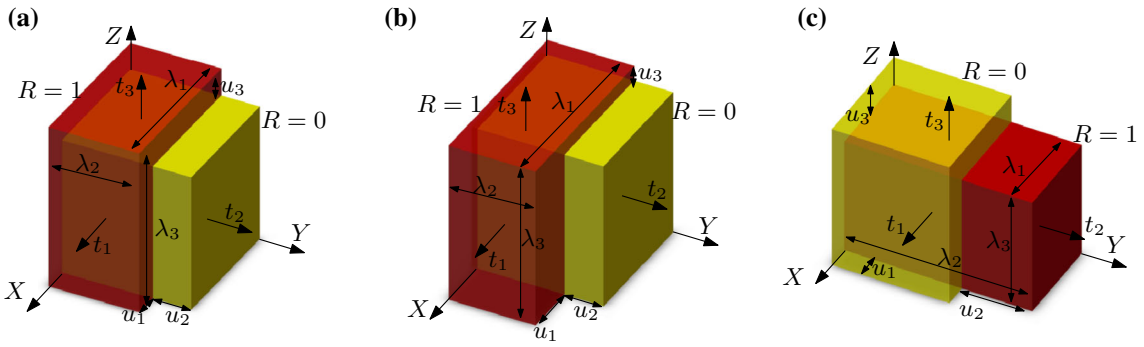

Fig. 2 The visualization of data generation of $\{a_1, a_2, a_3\}$ and $\{S_1, S_2, S_3\}$ through RVE for **a** loading path ID 3; **b** loading path ID 7; **c** loading path ID 14 in Table 1. Yellow color represents the original configuration of RVE and red color represents the configuration of RVE at the time step when the controlling loading parameter $R = 1$. (Color figure online)

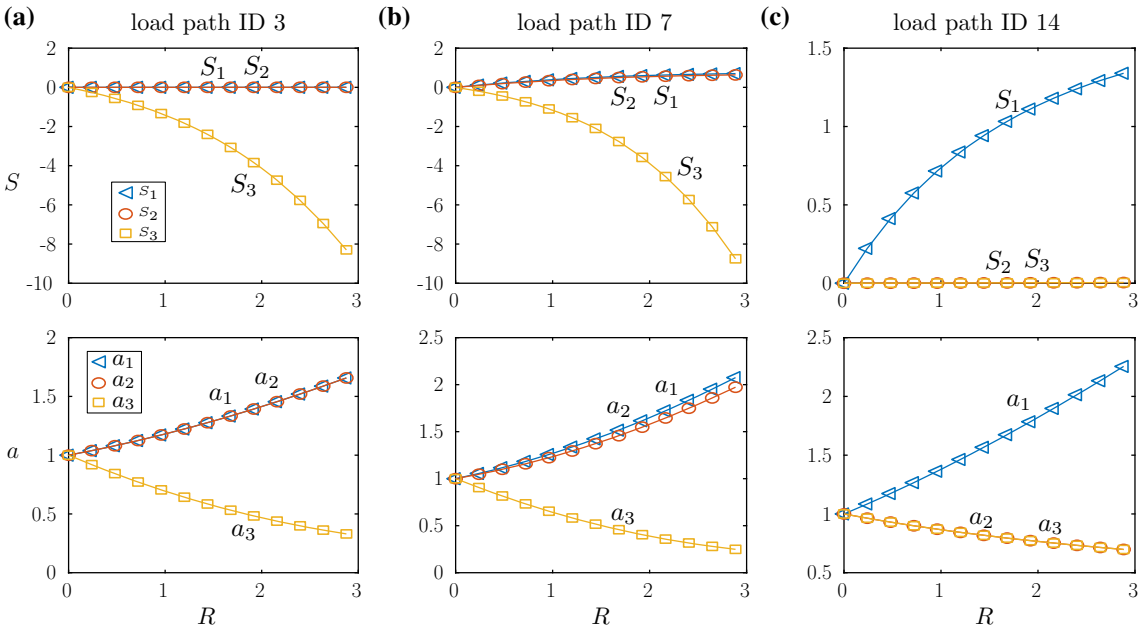

Fig. 3 The generated data of $\{a_1, a_2, a_3\}$ and $\{S_1, S_2, S_3\}$ versus the controlling loading parameter of through RVE for **a** loading path ID 3; **b** loading path ID 7; **c** loading path ID 14 in Table 1. 500 time steps are adopted for data sampling. For the showing purpose, only sampled data at a few time steps are shown

can meet the i.i.d. assumption for Artificial Neural Network (ANN) training conditions.

Various types of neural networks are proposed in the past years. We adopt a standard multi-layer ANN to train the data, which is shown in Fig. 4. This neural network includes an input layer (c^1), three hidden layers (c^2, c^3 and c^4) and output layer (c^5). The input layer, the three hidden layer and output layer have N_{input}, N_{hidden} and N_{output} neurons. For neural network shown in Fig. 4. N_{input}, N_{hidden} and N_{output} are 3, 6 and 3 respectively. W_{ij}^{n+1} are the weights for the link between ith neuron on layer c^n and jth neuron on layer c^{n+1}. b_j^{n+1} are the biases on jth neuron on layer c^{n+1}. Here, the superscript n represents the layer number ($n = 1 \cdots N$). N is the total number of the layers of ANN (the input layer is excluded in this definition and $N = 4$ in the present work).

Let $S^N_{\alpha,\beta,k}$ ($k = 1 \cdots 3$) denote the output principal stress of the ANN with input principal strain $a[\alpha, \beta, k]$. If ANN has three layers (one input layer c^1, one hidden layer c^2 and one output layer c^5), the principal stress predicted by ANN can be written:

$$S^{N=2}_{\alpha,\beta,k} = \tanh\left(a[\alpha, \beta, m]W^2_{mi} + b^2_i\right) W^5_{ik} + b^5_k \tag{6}$$

in which $m = 1 \cdots N_{input}$, $i = 1 \cdots N_{hidden}$ and $k = 1 \cdots N_{output}$ and the summation of dummy index should be carried out. tanh is the hyperbolic tangent function. And so

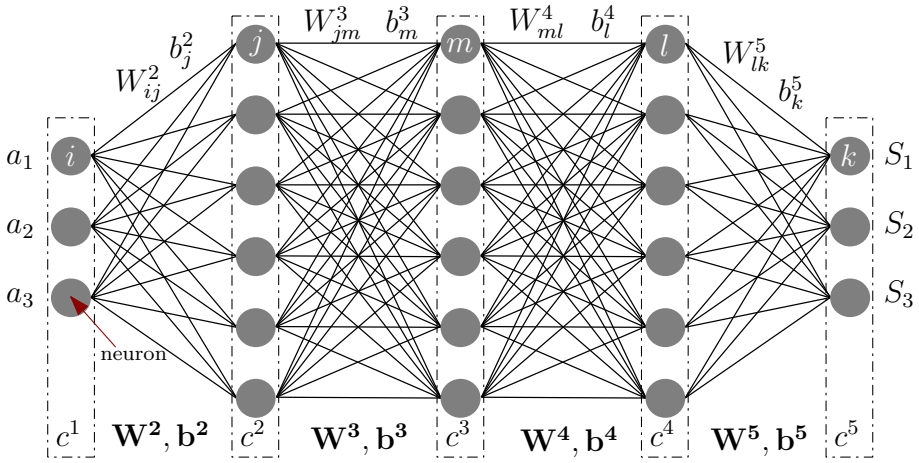

Fig. 4 Artificial neural network to train the stress-stretch data in principal space with 3 inputs, 3 hidden layers with the same number of neurons and 3 outputs. The weights $\mathbf{W^n}$ and biases $\mathbf{b^n}$ and its components W^n_{ij} and b^n_j $(n = 2 \cdots 5)$ are marked. The principal stress can be obtained by the input arbitrary principal stretch after training

on, for ANN with five layers $(c^1 \cdots c^5)$ in the present work, the principal stress predicted by ANN can be written:

$$S^{N=4}_{\alpha,\beta,k} = \tanh\left(\tanh\left(\tanh\left(a[\alpha,\beta,m]W^2_{mo} + b^2_o\right)W^3_{oi} + b^3_i\right)W^4_{ip} + b^4_p\right)W^5_{pk} + b^5_k \tag{7}$$

where $m = 1 \cdots N_{input}$, $o, i, p = 1 \cdots N_{hidden}$ and $k = 1 \cdots N_{output}$ and the summation of dummy index also should be carried out.

The training of the data tries to minimize the distance between the predicted points $S^{N=4}_{\alpha,\beta,k}$ and the generated data points $S[\alpha, \beta, k]$ on all the loading paths and history, which can be written as

$$\underset{W^2_{ij},b^2_j,\cdots,W^5_{ij},b^5_j}{\operatorname{argmin}} \sum_{k=1}^{3}\sum_{\alpha=1}^{N_L}\sum_{\beta=1}^{N_s}\left(S^{N=4}_{\alpha,\beta,k} - S[\alpha,\beta,k]\right)^2 \tag{8}$$

by optimizing the weights W^n_{ij} and biases b^n_j $(n = 2 \cdots 5)$.

The whole algorithm for training the material law is shown in Table 2. The Neural Fitting Toolbox (nftool) of MATLAB is used for training. The derivation details about the general ANN are given in "Appendix B".

3.2 On-line computation with the trained material law

At any material point, the right Cauchy-Green tensor can be expressed in the principal space:

$$\bar{\mathbf{C}} = a_i \mathbf{N}_i \otimes \mathbf{N}_i \tag{9}$$

Here a_i $(i = 1 \cdots 3)$ are the eigenvalues of $\bar{\mathbf{C}}$ and $\mathbf{N}_i$ $(i = 1 \cdots 3)$ are the eigenvector and the second PK stress can be described in the similar way:

$$\bar{\mathbf{S}} = S_i \mathbf{N}_i \otimes \mathbf{N}_i \tag{10}$$

Table 2 The algorithm for training the material law (Offline learning)

Algorithm 1: Offline Learning

1. Database by RVE: $\{a_1, a_2, a_3; S_1, S_2, S_3\}$

2. Extend data-set through a symmetric operation.

$\{a_1, a_2, a_3; S_1, S_2, S_3\}$ $\{a_2, a_3, a_1; S_2, S_3, S_1\}$
$\{a_3, a_1, a_2; S_3, S_1, S_2\}$
$\{a_2, a_1, a_3; S_2, S_1, S_3\}$ $\{a_1, a_3, a_2; S_1, S_3, S_2\}$
$\{a_3, a_2, a_1; S_3, S_2, S_1\}$

3. Train the data through the ANN.

$$\underset{W^2_{ij},b^2_j,\cdots,W^5_{ij},b^5_j}{\operatorname{argmin}} \sum_{k=1}^{3}\sum_{\alpha=1}^{N_L}\sum_{\beta=1}^{N_s}\left(S^{N=4}_{\alpha,\beta,k} - S[\alpha,\beta,k]\right)^2$$

weights W^n_{ij} and biases b^n_i $(n = 2 \cdots 5)$ are determined by solving the optimization problem with the database generated by the RVE

where S_i $(i = 1 \cdots 3)$ are the eigenvalues. Defining the second order tensor $\mathbf{A}_i = \mathbf{N}_i \otimes \mathbf{N}_i$ with no summation on i, the tangent modulus is given by the following:

$$\begin{aligned}\mathbf{C}^M = 2\frac{\partial \bar{\mathbf{S}}}{\partial \bar{\mathbf{C}}} &= \sum_{i=1}^{3}\sum_{j=1}^{3} 2\frac{\partial S_i}{\partial a_j}\frac{\partial a_j}{\partial \bar{\mathbf{C}}}\mathbf{N}_i \otimes \mathbf{N}_i + \sum_{i=1}^{3} 2S_i \frac{\partial \mathbf{N}_i \otimes \mathbf{N}_i}{\partial \bar{\mathbf{C}}} \\ &= \sum_{i=1}^{3}\sum_{j=1}^{3} 2\frac{\partial S_i}{\partial a_j}\mathbf{A}_i \otimes \mathbf{A}_j \\ &\quad + 2\sum_{i\neq j, i\neq k}^{3} S_i\left(\frac{\mathbf{A}_i\mathbf{A}_j^T + \mathbf{A}_j\mathbf{A}_i^T}{a_i - a_j} + \frac{\mathbf{A}_i\mathbf{A}_k^T + \mathbf{A}_k\mathbf{A}_i^T}{a_i - a_k}\right)\end{aligned} \tag{11}$$

where $\otimes$ is dyadic symbol for vectors. The derivation of $\left(\frac{\partial \mathbf{N}_i \otimes \mathbf{N}_i}{\partial \bar{\mathbf{C}}}\right)$ can refer to Rosati and Valoroso [33] and Tang et al. [34]. When a_i approaches a_j, it looks that Eq. (11) leads to singularity. Then Eq. (11) should be computed in terms of limitation. The tangent modulus is composed of two terms, one corresponding to the derivatives of the principal stress

with respect to principal stretch; the other corresponding to the spin of the principal axes.

After the data training by ANN, $(S_m, m = 1 \cdots 3)$ are given by the implicit function:

$$[S_1, S_2, S_3] = f(a_1, a_2, a_3; \mathbf{W}, \mathbf{b}) \tag{12}$$

which are used to update the stress in on-line computations (ref. Eq. 23). Here it should be noted that (a_1, a_2, a_3) are arbitrary for on-line computation and **(W,b)** are known by the ANN training of the data. In practice, the switch of the stretch components a_i and a_j may not lead to the switch of the stress components S_i and S_j computed by neural network (the detailed discussion is given in "Appendix B"). Then permutation of (a_i, a_j, a_k) is carried out to compute the stress (S_i, S_j, S_k). Equation (12) is computed six times by permutation of (a_1, a_2, a_3).

$$\begin{aligned}
\left[S_1^1, S_2^1, S_3^1\right] &= f(a_1, a_2, a_3; \mathbf{W}, \mathbf{b}) \\
\left[S_2^2, S_3^2, S_1^2\right] &= f(a_2, a_3, a_1; \mathbf{W}, \mathbf{b}) \\
\left[S_3^3, S_1^3, S_2^3\right] &= f(a_3, a_1, a_2; \mathbf{W}, \mathbf{b}) \\
\left[S_1^4, S_3^4, S_2^4\right] &= f(a_1, a_3, a_2; \mathbf{W}, \mathbf{b}) \\
\left[S_3^5, S_2^5, S_1^5\right] &= f(a_3, a_2, a_1; \mathbf{W}, \mathbf{b}) \\
\left[S_2^6, S_1^6, S_3^6\right] &= f(a_2, a_1, a_3; \mathbf{W}, \mathbf{b})
\end{aligned}$$

Then the results from the computation of six times are summed up and average to get the final S_i and $\frac{\partial S_i}{\partial a_j}$ (ref. Eq. 24):

$$\begin{aligned}
S_i &= \frac{1}{6}\sum_{m=1}^{6} S_i^m \\
\frac{\partial S_i}{\partial a_j} &= \frac{1}{6}\sum_{m=1}^{6}\left(\frac{\partial S_i}{\partial a_j}\right)^m
\end{aligned} \tag{13}$$

Then $\mathbf{S}$ and $\mathbf{C^M}$ can be computed based on the above results. The whole algorithm for applying the trained material law to simulate the deformation of structures is shown in Table 3.

Remarks i. *In the above, we only show how to derive tangent modulus* $\mathbf{C^M}$. *According to Belytschko et al.* [32], *pushing forward of* $\mathbf{C^M}$ *can obtain tangent modulus based on Truesdell rate:*

$$C^T_{ijkl} = \frac{1}{J} C^M_{mnpq} F_{im} F_{jn} F_{kp} F_{lq}$$

The tangent modulus based on Jaumann rate can be obtained through further transformation:

$$C^J_{ijkl} = C^T_{ijkl} + \frac{1}{2}\left(\delta_{ik}\sigma_{jl} + \delta_{ij}\sigma_{jk} + \delta_{jk}\sigma_{il} + \delta_{jl}\sigma_{ik}\right) - \delta_{ij}\sigma_{kl}$$

ii. *The proposed approach can greatly reduce the computational cost for data training. In the previous works such as* Hashash et al. [13], *the history of all the components of stretch are used in the training*. Wang and Sun [23] *train their model with the history of the principal strain and history of the incremental rotation. The introduction of incremental rotation is used to resolve the issue for the objectivity of the material laws. It can be seen that our derivation is still within the classical framework. Only mathematical form in the continuum mechanics theory is replaced by the trained data. It can preserve the objectivity of material law approximately. This will be discussed next in the section of numerical examples.*

Table 3 The algorithm for applying the trained material law to simulate the deformation of structures (Online computation)

Algorithm 2: Online FEM Computation

1. Deformation gradient $\bar{\mathbf{F}}$ is given.

2. Compute $\bar{\mathbf{C}} = \bar{\mathbf{F}} \cdot \bar{\mathbf{F}}^T$ and its eigenvalues $\{a_1, a_2, a_3\}$ and eigen vectors $\{\mathbf{N_1}, \mathbf{N_2}, \mathbf{N_3}\}$.

3. Call the ANN six times. The order of input variables is switched for each call.

$$\begin{aligned}
\left[S_1^1, S_2^1, S_3^1\right] &= f(a_1, a_2, a_3; \mathbf{W}, \mathbf{b}) \\
\left[S_2^2, S_3^2, S_1^2\right] &= f(a_2, a_3, a_1; \mathbf{W}, \mathbf{b}) \\
\left[S_3^3, S_1^3, S_2^3\right] &= f(a_3, a_1, a_2; \mathbf{W}, \mathbf{b}) \\
\left[S_1^4, S_3^4, S_2^4\right] &= f(a_1, a_3, a_2; \mathbf{W}, \mathbf{b}) \\
\left[S_3^5, S_2^5, S_1^5\right] &= f(a_3, a_2, a_1; \mathbf{W}, \mathbf{b}) \\
\left[S_2^6, S_1^6, S_3^6\right] &= f(a_2, a_1, a_3; \mathbf{W}, \mathbf{b})
\end{aligned}$$

4. The outputs of ANN is used to compute the principal stress by averaged six outputs.

$$S_i = \frac{1}{6}\sum_{m=1}^{6} S_i^m$$

5. The partial derivatives of ANN is used to compute tangent modulus (Eq. 11).

$\frac{\partial S_i}{\partial a_j} = \frac{1}{6}\sum_{m=1}^{6}\left(\frac{\partial S_i}{\partial a_j}\right)^m$ (more details are given in the "Appendix")

6. Solve equilibrium equation.

4 Numerical examples

The numerical algorithm for both data training and online finite element computation is shown in Tables 2 and 3. The predictions by the ANN trained model will compare with those by the reference neo-Hookean or Arruda–Boyce model. In all our examples, we use the consistent unit of measurement. The unit of length is mm; force is N; bending moment is N · mm; stress, pressure and modulus are MPa.

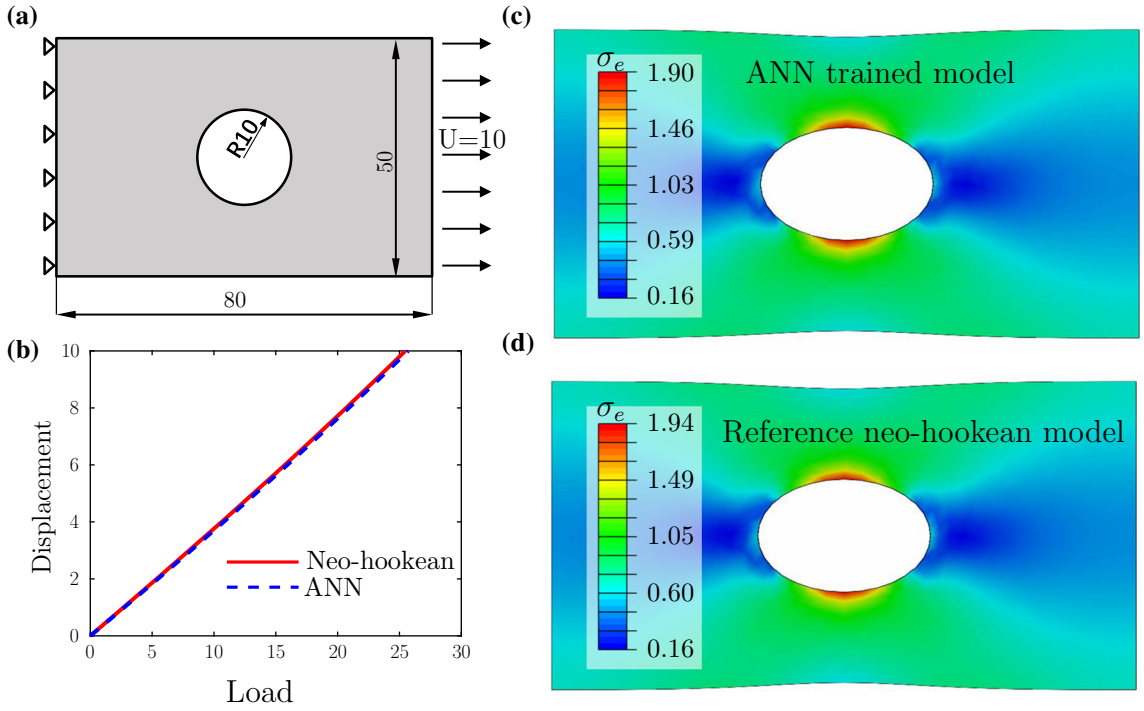

Fig. 5 The finite element analysis on a rectangular plate with a circular hole. **a** The geometric model and boundary conditions of the voided plate. **b** The load-displacement curves for the voided plate, predicted by the ANN trained model and the neo-Hookean model. The contour plots of effective stress by **c** the ANN trained model and **d** the reference neo-Hookean model under the same levels of imposed displacement 10. The FEM model includes 2348 nodes and 1093 elements

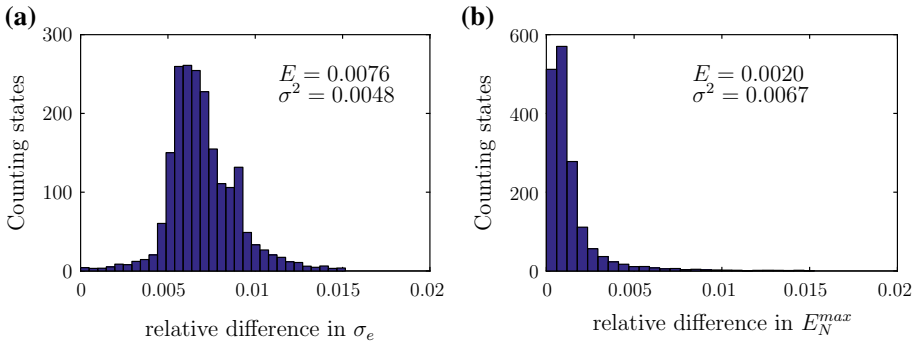

Fig. 6 Difference comparison between the results obtained from the ANN trained model and reference neo-Hookean model. Frequency histogram of the mechanical states versus the relative difference between two models on the effective stress and the maximum logarithmic strain in (**a**) and (**b**) respectively. The mean (E) and variance (σ^2) of the difference are marked in the figure

4.1 The material law by RVE without microstructure

We first show the results for the material law trained by RVE without microstructure. Because the RVE is without any microstructure, therefor the material response of RVE should be the same as the material model chosen for RVE analysis.

A rectangular plate with a circular hole of radius 10 at the center under the tensile loading is investigated first under plane stress conditions. The geometric setup is shown in Fig. 5a. The mesh is refined around the hole. A displacement of 10 is applied on the right edge and the left edge is fixed in the x direction. Both the neo-Hookean model and the ANN trained model are employed (The ANN model is trained by the data generated with 14 loading paths and 500 time steps of load). Figure 5b shows the load-displacement curves for both models. A very good agreement between the two models' prediction is observed. Figure 5c, d plots the contour of effective stress predicted by the two models. It is hard to distinguish the difference obtained by the two models. Figure 6a, b shows the statistical data to compare the perdition by both models. Figure 6a counts the frequency of the states versus the relative difference between two models on the effective stress. The average relative difference for two models is around 0.76%. Figure 6b counts the frequency of the state versus the relative difference between two mod-

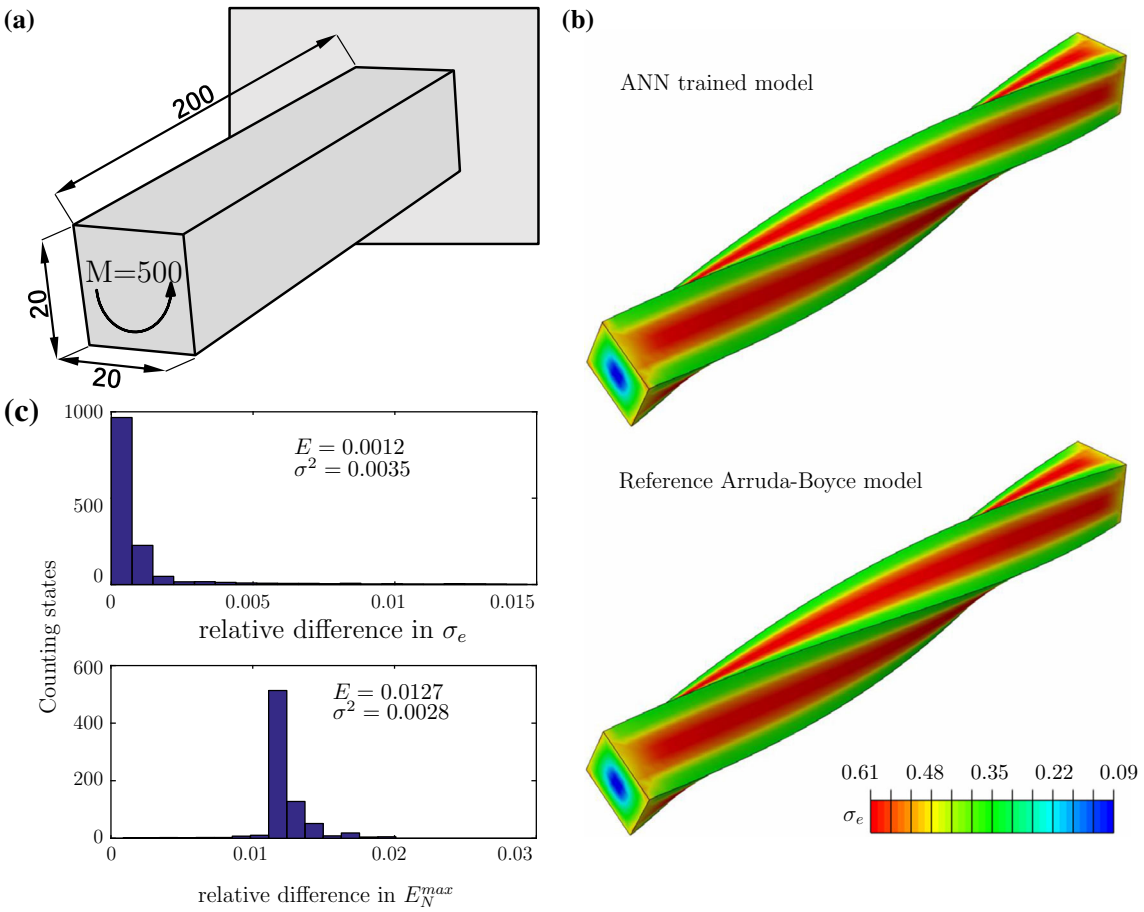

Fig. 7 The finite element analysis on a cuboid bar under the imposed torsion. **a** The geometric model and boundary conditions. **b** The contour plots of effective stress for the ANN trained model and the reference Arruda–Boyce model. **c** Frequency histogram of the mechanical states versus the relative difference between two models on the effective stress and the maximum logarithmic principal strain. The mean (E) and variance (σ^2) of the difference are marked in the figure. The FEM model includes 4352 nodes and 3283 elements

els on the maximum logarithmic principal strain $E_N^{\max}$. The average relative difference for two models is around 0.20%.

We then use the ANN trained model to predict the mechanical response for a three-dimensional problem where a torque ($M = 500$) is imposed on cuboid beam on the one end of surface and the beam is fixed on the other end. The torque is applied at a reference point, and the reference point is coupled with the end surface. The geometric setup is given in Fig. 7a. Figure 7b plots the contour of effective stress for the ANN trained model and the reference Arruda–Boyce model. Same as the two dimensional problem, there is a tiny difference between two models. This tinty difference can be identified through the statistical analysis on the relative difference of effective stress and the maximum logarithmic principal strain. The average error for effective stress and maximum logarithmic strain is around 0.12% and 1.27%. It should be emphasized that the large rotation of body exists in this example. The agreement between the ANN trained model and the reference Arruda–Boyce implies the objectivity of the ANN trained model.

A more complicated example is studied. The combined force ($F = 4.5$) and torsion ($M = 500$) is imposed on cuboid beam on the one end of surface and fixed on the other end. The geometrical setup is shown in Fig. 8a. A through-hole with radius 5 at the center of the specimen is introduced. The material law is trained by ANN using 7 or 14 data-sets with the neo-Hookean model. Figure 8c plots the contour of effective stress of the reference neo-Hookean model and the ANN trained model with 7 or 14 data-sets. It can be seen from Fig. 8b there is still no distinct difference between the ANN trained model and the reference neo-Hookean model even under the combined complicated loading conditions (see the statistical data in Fig. 9). It should be noted that the ANN trained model with 7 data-sets is almost the same as that trained by 14 data-sets. It looks that the material law trained by 7 data-sets can cover the full range of stress-states. This can greatly reduce the offline training costs.

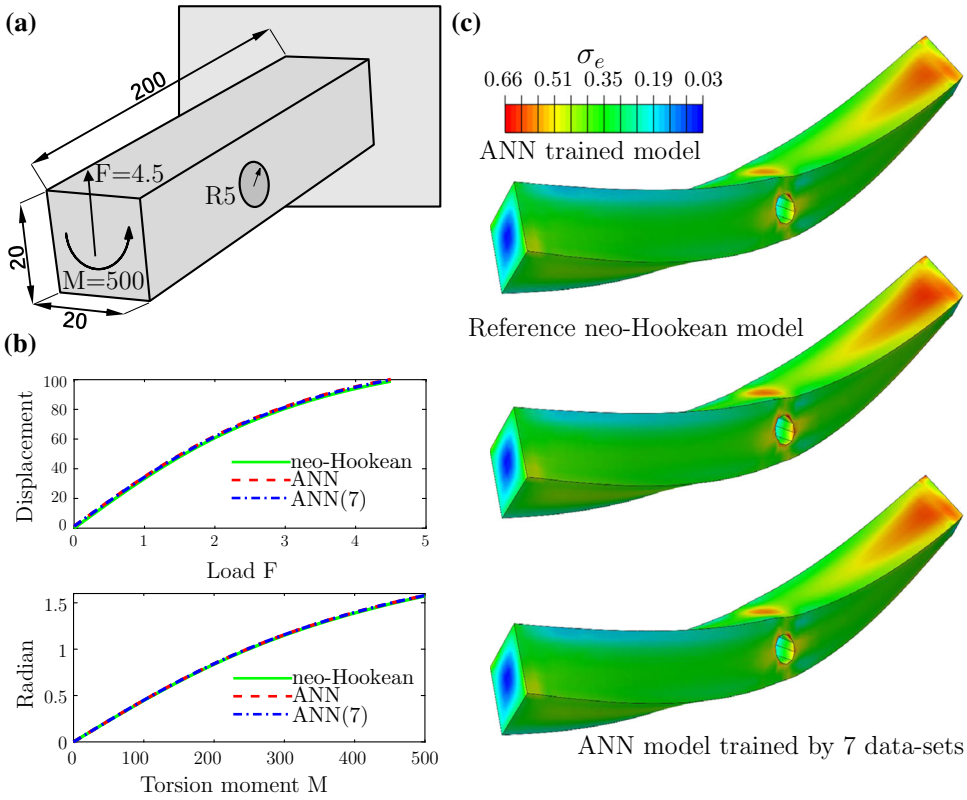

Fig. 8 The finite element analysis on a cuboid bar with voids under the combined torsion and bending. **a** The geometric model and boundary conditions. **b** The load-displacement curves predicted by the ANN trained model and the neo-Hookean model. **c** The contour plots of effective stress predicted by the ANN trained model (14 and 7 data-sets) and Arruda–Boyce model. The FEM model includes with 4880 nodes and 3711 elements

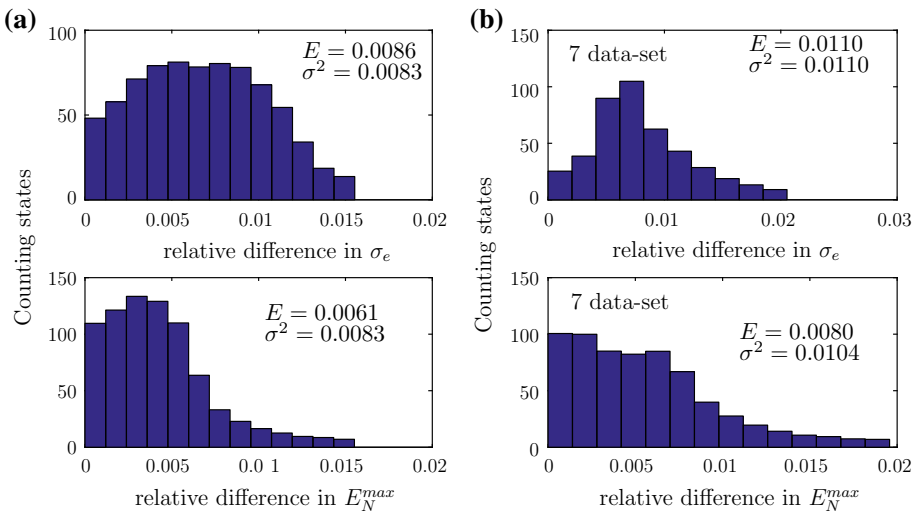

Fig. 9 Frequency histogram of the mechanical states versus the relative difference on the effective stress and the maximum logarithmic strain with the ANN trained model by **a** 14 data-sets and **b** 7 data-sets. The mean (E) and variance (σ^2) of the difference are marked in the figure

Buckling of soft solids recently attracts a lot of research attention [34–37]. Buckling of soft solids created many new opportunities to design the materials with complex microstructures to realize specific functions. Here we show an example to analyze the buckling with microstructure with the ANN trained model. The geometrical setup of the problem is shown in Fig. 10a. The specimen is cuboid with a through-hole with radius 5 at the center. The cuboid is under compressive loadings. As predicted by the stability theory Timoshenko [38], the cuboid loses stability when the applied compressive loading is beyond threshold of the critical load. Buckling analysis is used to predict the critical load and mode (morphology). To measure the difference between the two modes (morphology), a L2-norm is defined though u_i^j which is the displacement of node i at the degree of freedom j:

$$Norm = \sqrt{\sum_{i=1}^{N_{node}} \sum_{j=1}^{3} (u_i^j)^2}$$

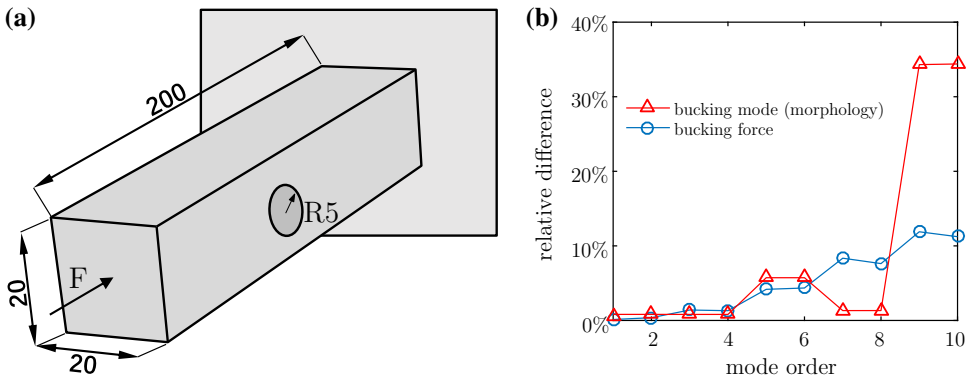

Fig. 10 Finite element buckling analysis on a cuboid beam with holes. **a** The geometry and boundary conditions. **b** The order of buckling mode versus relative difference in terms of buckling force and buckling mode (morphology) predicted by the reference neo-Hookean model and the ANN trained model. The FEM model includes 4880 nodes and 3711 elements

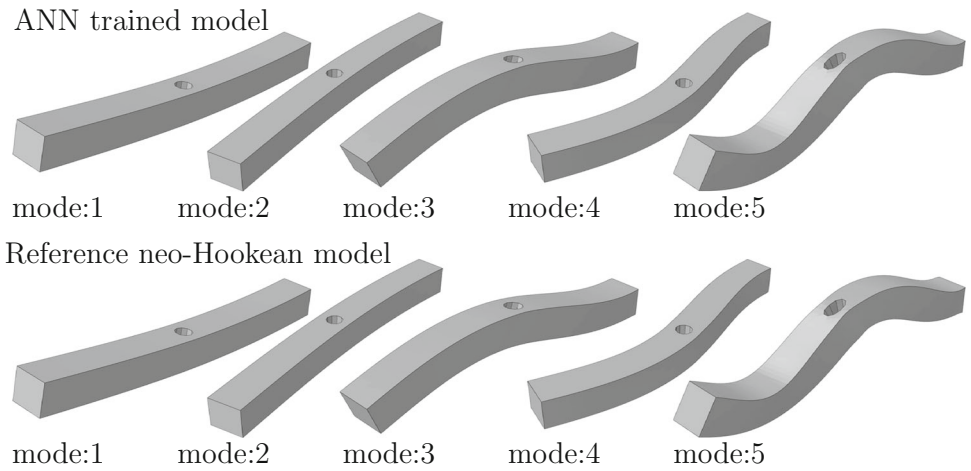

Fig. 11 The buckling modes (morphology) of the cuboid beam through buckling analysis predicted by the neo-Hookean model and the ANN trained model. The first 5 modes are shown for comparison

where N_{node} is the total number of nodes in finite element model. The relative difference predicted by the ANN trained model and the reference neo-Hookean model for different buckling modes is shown in Fig. 10b. The relative difference of the first four modes is less than 1%, which increases with mode number. However, high-order modes are rarely considered in engineering applications. The first to fifth buckling modes are shown in Fig. 11. The agreement between the ANN trained model and the reference neo-Hookean model is shown clearly.

4.2 The material law by RVE with void

In the previous examples, we show the capability of the proposed method based on the material law trained by the RVE without the microstructure. We can forecast that the material model trained by the RVE without microstructure should almost the same as the adopted neo-Hookean model or Arruda–Boyce model. However, we do not know the material model for a void in the neo-Hookean solid in prior. Then we will use the ANN trained material law based on the data generated by the RVE with void to show the predicative capability.

Let us consider a three-dimensional problem that a cuboid beam which one end surface is imposed a coupling constraint with a reference point and the other end is fixed. A force ($F = 0.3$) is imposed on the reference point. The geometric setup is given in Fig. 12a. The beam contains 40 spherical voids with the same size of void in the RVE, which are evenly distributed. We will use direct numerical simulation (DNS) with neo-Hookean model to solve the problem first. Then the spherical voids are smeared out inside the beam and the material law is described by the ANN trained model. The DNS involves 50,702 nodes and 35,621 elements to resolve all the voids in the beam (Fig. 12b). The FEM mesh smearing out the voids only has 40 elements shown in Fig. 12c.

Figure 13 plots the deformed configuration of DNS and the ANN trained model in the same coordinates at the final step of the imposed loading. The displacement scale is 1. It can be seen clearly that the the deformed shape predicted by the DNS is also the same as that predicted by the FEM model with the ANN trained isotropic material law even under the large bending deformation. Figure 14 shows the force versus displacement curve of the reference point for both DNS and the FEM model with the ANN trained material law. The results predicted by both models are almost the same. The largest difference is less then 1%. We also compare the resid-

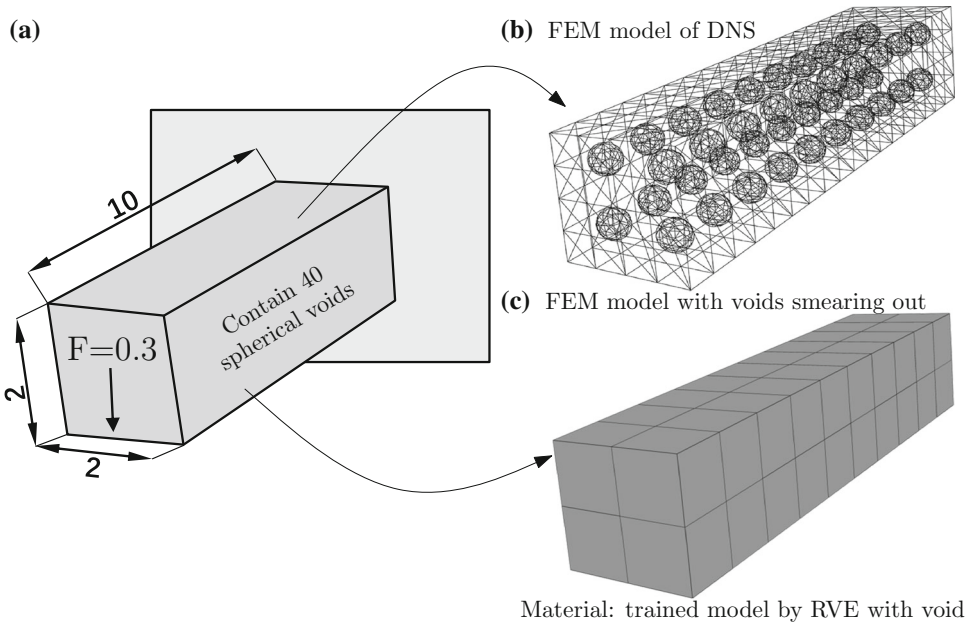

Fig. 12 The finite element analysis on a cuboid beam under the imposed force. The beam contains 40 spherical voids, which are evenly distributed. **a** The geometric model and boundary conditions. **b** The FEM model for direct numerical simulations (DNS) with 50,702 nodes and 35,621 elements. **c** The FEM model with the voids smeared out (the material is described by the ANN trained material law). It involves 40 elements

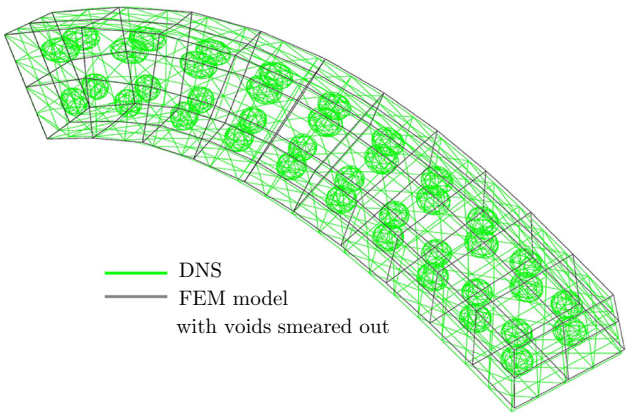

Fig. 13 The deformed configuration predicted by the DNS and the FEM model with voids smeared out in which the ANN trained material law is used to describe the material behavior

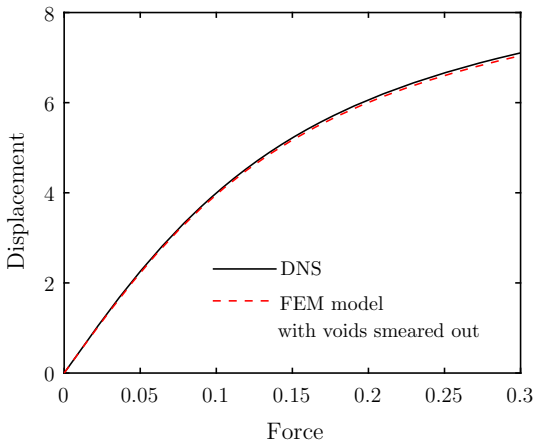

Fig. 14 The force-displacement curve predicted by the DNS and the FEM model with voids smeared out in which the ANN trained material law is employed to describe the material behavior

ual norms for both models during the iteration process for time step 1, shown in Table 4. It is observed that the classical neo-Hookean model converges faster than the ANN trained model as expected. However, the ANN trained model also can converge at the same accuracy with two more iterations. Because of the consistent tangent modulus is derived, the second order convergence with the ANN trained material law is implied. Due to the less elements used by the ANN trained model, the ANN trained model shows some advantages. It should be commented that this example with a single centered void is slightly orthotropic. However, isotropic assumption is widely used to approximate slightly anisotropic behavior. Because of the multiple voids and their even distribution in the beam, it makes the porous material nearly isotropic. This is the reason why the trained model for isotropic materials can compare with DNS very well. The spirit of the proposed method can be extended to consider the anisotropic nonlinear elastic solids but further work should be carried out.

Finally, the mesh information for all the above examples is summarized in Fig. 15.

5 Conclusions

With advent of big data science and machine learning, it is possible to obtain the material law through a data-driven approach. In this work, we have presented an efficient data-driven computational framework to build material law for nonlinear isotropic elastic materials based on the principal

Table 4 Comparison of residual norm for the ANN trained model and the reference neo-Hookean model at time step 1 (time increment 5% strain) for the voided beam problem shown in Fig. 12

Iteration	1	2	3	4	5	6
Reference neo-Hookean model	0.59	4.43E−04	9.17E−07	1.44E−09	–	–
ANN trained model	0.40	3.86E−03	5.42E−04	1.14E−04	2.36E−05	4.90E−06

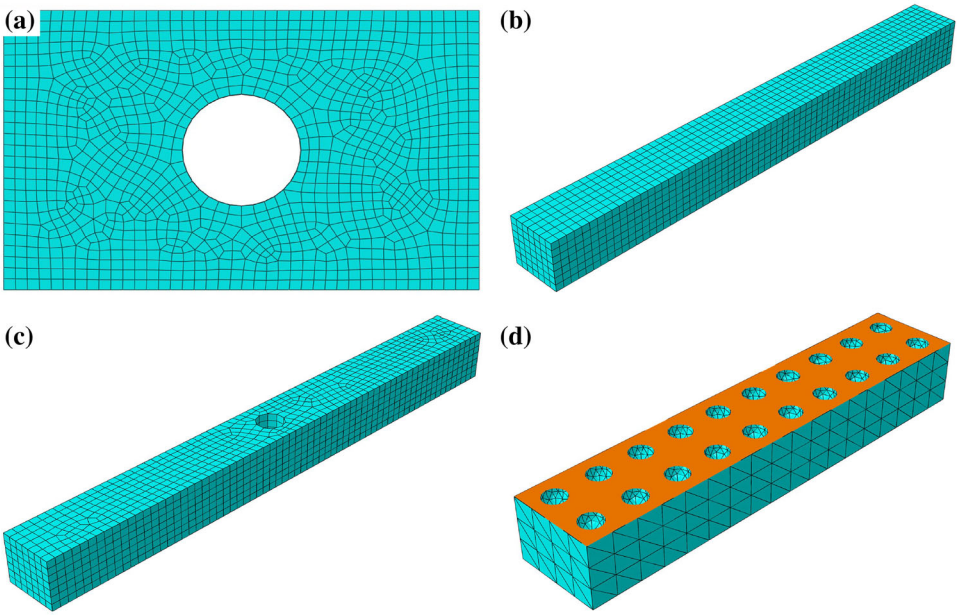

Fig. 15 The summarized mesh information for all the examples shown in the present paper. Corresponding to **a** Fig. 5; **b** Fig. 7; **c** Fig. 8 and Fig. 10. **d** Cross-section view corresponding to Fig. 12

component expansion. Carefully designed RVE based on principal stretches is used to generate the stress-stretch data, which greatly reduces the required training cost for material law and obtained a high quality model. Our framework can satisfy the requirements for material law such as objectivity approximately. With the derived consistent tangent modulus based on the data in principal space, the second-order convergence capability is implied. The proposed approach can be used under the multiscale computational homogenization framework naturally. It can provide a way to obtain the material law involved at different scales effectively by pure data.

Acknowledgements S. T. appreciates the support from NSF of China (Project No. 11872139). X. G. Thanks the support from NSF of China (11732004, 11821202), and Program for Changjiang Scholars, Innovative Research Team in University (PCSIRT).

Appendix A: Homogenization in principal space

At the material point inside the RVE, the deformation gradient tensor $\boldsymbol{F}$ $(= \partial \boldsymbol{x}/\partial \boldsymbol{X})$ and the first Piola-Kirchhoff stress tensor $\boldsymbol{P}$ constitute one basic work-conjugate pair. Homogenization of the micro-variables $\boldsymbol{F}$ and $\boldsymbol{P}$ at finite strains yields the macro-variables $\bar{\boldsymbol{F}}$ and $\bar{\boldsymbol{P}}$. They are governed by surface data of their microscopic fields [39].

$$\bar{\boldsymbol{F}} = \frac{1}{V_0}\int_{\partial\Omega_0} \boldsymbol{x} \otimes \boldsymbol{N}\, dS, \tag{14}$$

$$\bar{\boldsymbol{P}} = \frac{1}{V_0}\int_{\partial\Omega_0} \boldsymbol{t} \otimes \boldsymbol{X}\, dS \tag{15}$$

where $\boldsymbol{x}$ is the current position, V_0 is the volume of the body Ω_0 at the original configuration, $\boldsymbol{N}$ the outward normal vector to surface $\partial\Omega_0$ and $\boldsymbol{t} = \boldsymbol{P} \cdot \boldsymbol{N}$ the traction vector. The homogenized Cauchy stress can also be expressed:

$$\bar{\boldsymbol{\Sigma}} = \frac{1}{V}\int_{\partial\Omega_0} \boldsymbol{t} \otimes \boldsymbol{x}\, dS \tag{16}$$

where V is the volume of the body Ω_0 at the current configuration. The homogenized secondary PK stress $\bar{\mathbf{S}}$ is defined by

$$\bar{\boldsymbol{S}} = \bar{\boldsymbol{F}}^{-1} \cdot \bar{\boldsymbol{P}} \tag{17}$$

We only consider the principal components on RVE. The boundary conditions on the RVE are given by

$$u_1 = 0, \quad t_2 = 0 \quad t_3 = 0 \quad \text{at } x = 0, \tag{18}$$
$$u_2 = 0, \quad t_1 = 0 \quad t_3 = 0 \quad \text{at } y = 0, \tag{19}$$
$$u_3 = 0, \quad t_1 = 0 \quad t_2 = 0 \quad \text{at } z = 0, \tag{20}$$

where t_1, t_2 and t_3 are the components of the traction vector, $\boldsymbol{t} = t_1\mathbf{e}_1 + t_2\mathbf{e}_2 + t_3\mathbf{e}_3$. At the boundary of the RVE, the homogenous deformation is applied

$$\boldsymbol{x} = \bar{\boldsymbol{F}} \cdot \boldsymbol{X} \tag{21}$$

where

$$\bar{\boldsymbol{F}} = \lambda_1\mathbf{e}_1 \otimes \mathbf{e}_1 + \lambda_2\mathbf{e}_2 \otimes \mathbf{e}_2 + \lambda_3\mathbf{e}_3 \otimes \mathbf{e}_3 \tag{22}$$

in which λ_1, λ_2, λ_3 are the principal stretches in the x, y and z directions respectively. With the displacement on the outer boundary of RVE (Eq. 1), substituting (Eq. 22) into (Eq. 17) and invoking (Eq. 15) yields the stress expressed by the traction force on the outer boundary.

Appendix B: Data training by ANN in principal space

For general ANN, a functional transformation is defined

$$\Phi\left(c_i^n, W_{ij}^{n+1}, b_j^{n+1}\right) = \tanh\left(c_i^n W_{ij}^{n+1} + b_j^{n+1}\right)$$

on nth layer (superscript n represents the layer number), which can map the data unit on ith neuron (c_i^n) from the nth layer to $n+1$th. W_{ij}^{n+1} are the weights for the link between ith neuron on layer c^n and jth neuron on layer c^{n+1}. b_j^{n+1} are the biases on jth neuron on layer c^{n+1} ($\mathbf{W^{n+1}}$, $\mathbf{b^{n+1}}$ in vector form).

In terms of stress-strain relationship in principal space, the training of data tries to minimize Eq. (8), which starts at the input layer with the initial value:

$$[c_1^1, c_2^1, c_3^1] = [a_1, a_2, a_3]$$

The mapping from nth layer to $n+1$th ($n = 1 \cdots N$) layer takes:

$$c_j^{n+1} = \Phi\left(c_i^n, W_{ij}^{n+1}, b_j^{n+1}\right)$$

The output at the final layer is the principal stress:

$$S_j = c_i^N W_{ij}^{N+1} + b_j^{N+1} \tag{23}$$

It can be seen from Eq. (11) that $\frac{\partial S_i}{\partial a_j}$ needs to be computed for tangent modulus. It can be obtained through a recursion formula

$$\frac{\partial S_i}{\partial a_j} = W_{ik}^{N+1} \frac{\partial c_k^N}{\partial a_j} \tag{24}$$

where

$$\frac{\partial c_k^n}{\partial a_j} = B_{ki}^{n-1} W_{im}^n \frac{\partial c_m^{n-1}}{\partial a_j}$$

can be obtained from above layer. Here

$$B_{ki}^n = \left(1 - \tanh^2\left(c_m^n W_{mk}^{n+1} + b_k^{n+1}\right)\right)\delta_{ki} \tag{25}$$

no summation on k and δ_{ki} is Kronecker delta. The process will return and end at the first input layer:

$$\frac{\partial c_i^1}{\partial a_j} = \delta_{ij}.$$

Even with the data in the principal space, data training using ANN still faces some difficulties (demonstrated in Fig. 16). In Fig. 16a, with the input stretch (a_1, a_2, a_3), we can obtain the output stress $(S_1^\diamond, S_2^\diamond, S_3^\diamond)$. When we switch a_2 and

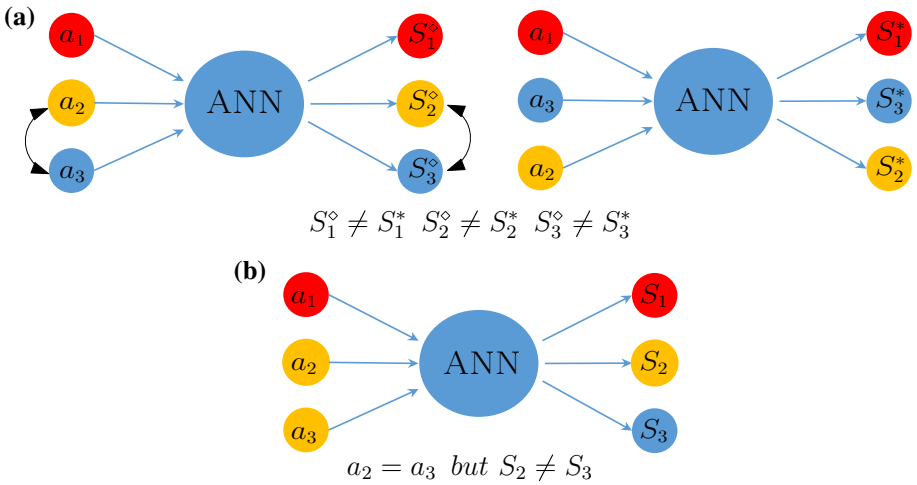

Fig. 16 Two issues on artificial neural network to train material law for the materials. **a** The order of the inputs is switched, the output stress can be completely different. **b** Although the two inputs (stretch) are exactly the same, the output (stress) can be different

a_3 with input (a_1, a_3, a_2), we cannot get the symmetric result $(S_1^\diamond, S_3^\diamond, S_2^\diamond)$. $(S_1^* \neq S_1^\diamond, S_2^* \neq S_2^\diamond, S_3^* \neq S_3^\diamond)$. In Fig. 16b, with the input (a_1, a_2, a_3) with $a_2 = a_3$, the output data does not have $S_2 \neq S_3$. This is the reason why the permutation is carried out in the data training and take the average to get (S_1, S_2, S_3). Then the coefficient $1/6$ is introduced in Eq. (13). It also implies that the data training with the full component of stress and stretch is more difficult, demonstrating the advantage of the present method in principal space.

References

1. Kirchdoerfer T, Ortiz M (2016) Data driven computational mechanics. Comput Methods Appl Mech Eng 304:81–101
2. Kirchdoerfer T, Ortiz M (2017) Data-driven computing in dynamics. Int J Numer Methods Eng 113(11):1697–1710
3. Kirchdoerfer T, Ortiz M (2017) Data Driven Computing with noisy material data sets. Comput Methods Appl Mech Eng 326:622–641
4. Conti S, Mller S, Ortiz M (2017) Data driven problems in elasticity. Arch Ration Mech Anal 229(1):79–123
5. Leygue A, Coret M, Rthor J, Stainier L, Verron E (2018) Data based derivation of material response. Comput Methods Appl Mech Eng 331:184–196
6. Chinesta F, Ladeveze P, Ibanez R, Aguado JV, Abisset-Chavanne E, Cueto E (2017) Data-driven computational plasticity. Proc Eng 207:209–214
7. Nguyen LTK, Keip M-A (2018) A data-driven approach to nonlinear elasticity. Comput Struct 194:97–115
8. Rthor J, Muhibullah, Elguedj T, Coret M, Chaudet P, Combescure A (2013) Robust identification of elasto-plastic constitutive law parameters from digital images using 3D kinematics. Int J Solids Struct 50(1):73–85
9. Rthor J (2010) A fully integrated noise robust strategy for the identification of constitutive laws from digital images. Int J Numer Methods Eng 84(6):631–660
10. Al-Haik M, Hussaini M, Garmestani H (2006) Prediction of nonlinear viscoelastic behavior of polymeric composites using an artificial neural network. Int J Plast 22(7):1367–1392
11. Zopf C, Kaliske M (2017) Numerical characterisation of uncured elastomers by a neural network based approach. Comput Struct 182:504–525
12. Ghaboussi J, Sidarta DE (1998) New nested adaptive neural networks (NANN) for constitutive modeling. Comput Geotech 22(1):29–52
13. Hashash YMA, Jung S, Ghaboussi J (2004) Numerical implementation of a neural network based material model in finite element analysis. Int J Numer Methods Eng 59(7):989–1005
14. Clement A, Soize C, Yvonnet J (2012) Computational nonlinear stochastic homogenization using a nonconcurrent multiscale approach for hyperelastic heterogeneous microstructures analysis. Int J Numer Methods Eng 91(8):799–824
15. Yvonnet J, Monteiro E, He QC (2012) Computational homogenization method and reduced database model for hyperelastic heterogeneous structures. Int J Multiscale Comput Eng 11(3):201–225
16. Le BA, Yvonnet J, He QC (2016) Computational homogenization of nonlinear elastic materials using neural networks. Int J Numer Methods Eng 104(12):1061–1084
17. Liu Z, Bessa MA, Liu WK (2016) Self-consistent clustering analysis: an efficient multi-scale scheme for inelastic heterogeneous materials. Comput Methods Appl Mech Eng 306:319–341
18. Bessa MA, Bostanabad R, Liu Z, Hu A, Apley DW, Brinson C, Chen W, Liu WK (2017) A framework for data-driven analysis of materials under uncertainty: countering the curse of dimensionality. Comput Methods Appl Mech Eng 320:633–667
19. Liu Z, Fleming M, Liu WK (2018) Microstructural material database for self-consistent clustering analysis of elastoplastic strain softening materials. Comput Methods Appl Mech Eng 330:547–577
20. Kafka OL, Cheng Y, Shakoor M, Liu Z, Wagner GJ, Liu WK (2018) Data-driven mechanistic modeling of influence of microstructure on high-cycle fatigue life of nickel titanium. JOM 70:1154–1158
21. Tang S, Lei Z, Liu WK (2018) From virtual clustering analysis to self-consistent clustering analysis: a mathematical study. Comput Mech 62(6):1443–1460
22. Shakoor M, Kafka OL, Liu WK (2018) Data science for finite strain mechanical science of ductile materials. Comput Mech. https://doi.org/10.1007/s00466-018-1655-9
23. Wang K, Sun WC (2018) A multiscale multi-permeability poroplasticity model linked by recursive homogenizations and deep learning. Comput Methods Appl Mech Eng 334:337–380
24. Lei X, Liu C, Du Z, Zhang W, Guo X (2019) Machine learning-driven real-time topology optimization under moving morphable component-based framework. J Appl Mech 86(1):011004
25. Bishop CM, Nasrabadi NM (2006) Pattern recognition and machine learning. Academic Press, Cambridge
26. Mooney M (1940) A theory of large elastic deformation. J Appl Phys 11(9):582–592
27. Arruda EM, Boyce MC (1993) A 3-dimensional constitutive model for the large stretch behavior of rubber elastic materials. J Mech Phys Solids 41(2):389–412
28. Tang S, Guo TF, Cheng L (2008) Rate effects on toughness in elastic nonlinear viscous solids. J Mech Phys Solids 56(3):974–992
29. Gao B, Li Y, Guo TF, Guo X, Tang S (2018) Void nucleation in alloys with lamella particles under biaxial loadings. Extreme Mech Lett 22:42–50
30. Wong WH, Guo TF (2015) On the energetics of tensile and shear void coalescences. J Mech Phys Solids 82:259–286
31. Liu ZG, Wong WH, Guo TF (2016) Void behaviors from low to high triaxialities: transition from void collapse to void coalescence. Int J Plast 84:183–202
32. Belytschko T, Liu WK, Moran B (2014) Nonlinear finite elements for continua and structures. Wiley, Hoboken
33. Rosati L, Valoroso N (2004) A return map algorithm for general isotropic elasto-visco plastic materials in principal space. Int J Numer Methods Eng 60(2):461–498
34. Tang S, Yang Y, Peng XH, Liu WK, Huang XX, Elkhodary K (2015) A semi-numerical algorithm for instability of compressible multilayered structures. Comput Mech 56(1):63–75
35. Tang S, Li Y, Yang Y, Guo Z (2015) The effect of mechanical-driven volumetric change on instability patterns of bilayered soft solids. Soft Matter 11(40):7911–7919
36. Li Z, Zhou Z, Li Y, Tang S (2017) Effect of cyclic loading on surface instability of silicone rubber under compression. Polymers 9(4):148
37. Zhou Z, Li Y, Wong W, Guo T, Tang S, Luo J (2017) Transition of surface-interface creasing in bilayer hydrogels. Soft Matter 13(35):6011–6020
38. Timoshenko SP (1936) Theory of elastic stability. McGraw-Hill Book Co, New York
39. Hill R, Rice JR (1972) Constitutive analysis of elastic-plastic crystals at arbitrary strain. J Mech Phys Solids 20(6):401–413