

Determination of Material Parameters in Constitutive Models using Adaptive Neural Network Machine Learning

Jikun Wang^a, Bangguo Zhu^b, Chung-Yuen Hui^{b,c,*}, Alan T. Zehnder^{b,*}

^a*Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853, USA*

^b*Sibley School of Mechanical and Aerospace Engineering, Field of Theoretical and Applied Mechanics, Cornell University, Ithaca, New York, 14853, USA*

^c*Global Station for Soft Matter, GI-CoRE, Hokkaido University, Sapporo 001-0021, Japan*

Abstract

A challenge in the constitutive modeling of time dependent, non-linear solids is the identification of potentially large numbers of material parameters.. Here we present an efficient method to determine these parameters using a machine learning algorithm based on Singular Value Decomposition (SVD) and an adaptive neural network (NN). SVD compresses training data and provides outputs for the NN. The trained network rapidly computes the material responses for a large set of material parameters. These responses are then compared with experimental data to determine the optimal parameters. We test our algorithm by performing uniaxial cyclic and relaxation tests on three hydrogels with very different time dependent behaviors: a chemically and physically crosslinked polyampholyte (c-PA) gel; a pure physically crosslinked PA gel, and a Poly(vinylalcohol) (PVA) hydrogel with both chemical and physical crosslinks.

Keywords: parameters fitting, singular value decomposition, Neural network, PVA hydrogel, PA hydrogel, viscoelastic model

1. Introduction

Soft polymeric materials such as elastomers and gels often exhibit complex time dependent mechanical behavior [1, 2]. These time dependent behaviors can be controlled by crosslinking the network by a combination of chemical and physical crosslinks, as well as controlling the density of crosslinks [3]. Physical crosslinks can be formed using dynamic bonds such as dynamic covalent bonds, hydrogen bonds, ionic bonds, metal-ligand coordination, host-guest interactions, hydrophobic interactions, and $\pi - \pi$ stacking [4, 5, 6, 7, 8, 9, 10]. Experiments by different research groups have shown that these dynamic networks can break and reform resulting in self-healing solids [11, 12, 13, 14, 15, 16, 17, 18]. More importantly, viscoelasticity associated with bond breaking and reforming can significantly increase energy dissipation which results in enhance fracture toughness [19]. In comparison to networks crosslinked by physical bonds, those consisting of chemical bonds generally

*Corresponding author

Email addresses: ch45@cornell.edu (Chung-Yuen Hui), atz2@cornell.edu (Alan T. Zehnder)

exhibit weaker time-dependent behavior. The time-dependent load-transfer between chemically crosslinked and physically crosslinked networks remain a topic of active investigation [20].

Theory plays an important role in the development of constitutive models. In the past two-
15 decades, many rate-dependent constitutive models have been developed to study the time-dependent behavior of soft materials made of dynamic networks as well as the flow of solvent in polymer gels [21, 22]. The complex micromechanics governing mechanical behavior requires many material parameters to construct a constitutive model with good predictive capability. For example, Mao et al [23]. has developed a constitutive model for a hydrothermally activated malleable covalent net-
20 work polymer. This model has 24 material parameters. Lu et al. [24] developed a 12 parameter constitutive model which incorporates viscoelasticity and Mullins effect to study the time dependent mechanical responses of four soft materials. Mao et al. [25] developed a large deformation viscoelasticity model to study the behavior of a double network hydrogel consisting of a covalently crosslinked polyacrylamide and an ionically crosslinked alginate network. This model has 18 material
25 parameters. Chester [21] has developed a 10 parameters constitutive model to study the coupling of poroelasticity and viscoelasticity in polymer gels. Tang et al. [26] have developed a micro-stress based viscoelastic model to describe the viscoelasticity of polymers. This model has 9 material parameters.

In physically based constitutive models, some material parameters can be determined based on
30 their physical significance, while others are typically calibrated manually or through an optimization process. For example, Crespo-Cuevas et al. [27] use transient network theory to develop a physical based model to quantify the poro-viscoelasto-plasticity of a Agarose gel. To estimate the material parameters in their model, they compare experimental creep and stress relaxation data with finite element simulations. They start the finite element simulation with some initial guess of the material
35 parameters and then update the input values iteratively using a minimization algorithm in MATLAB.

For complex constitutive models with a large number of parameters, fitting can be extremely challenging since computation can be time consuming even with an efficient optimization scheme. In this work, we highlight an alternative method that combines singular value decomposition (SVD) and neural networks (NN) to address this challenge. In the past few years, there has been intensive
40 interest in using machine learning techniques to solve solid mechanics problems [28, 29, 30]. One approach is to train regression trees or neural networks to quickly solve a specific type of boundary value problems in solid mechanics. For example, Liu et al. [31] have shown that machine learning can rapidly determine the fracture toughness of a pre-notched pentagonal cross-section microcantilever loaded at its end. The machine learning solutions for different cantilever geometries were found to
45 agree well with finite element results.

A different approach is to use machine learning to construct constitutive models; that is, to provide mechanical responses without the use of analytic constitutive equations. For example, Wang

et al. [32] used temporal convolutional network to simulate the ultra-long-history-dependent stress-strain relationship of reinforced concrete. Saharuddin et al. [33] used artificial neural network and extreme learning machine to simulate the mechanical behavior of a magnetorheological elastomer. Masi et al. [34] proposed a thermodynamics-based artificial neural network and applied their model to study the behavior of elasto-plastic solids.

Although there are a few papers that discuss parameter determination using machine learning [35, 36, 37], we have not found any published work using machine learning to determine material parameters in complex constitutive models. As we will demonstrate later, machine learning provides an efficient means to determine these material parameters. Our method has certain advantages over traditional nonlinear fitting methods such as the Levenberg-Marquardt algorithm [38, 39, 40]. This method also preserves the physical meaning of analytic constitutive models. Our method is also adaptive, the NN can adapt to a wide variety of constitutive models by automatically updating the size of the neural network and training set size.

The basic idea is to train a NN to determine the optimal material parameters given the constitutive model and stress histories from several different types of loading histories. For each type of load history, e.g., a relaxation test, our training set consists of many stress histories associated with the loading history. Each stress history corresponds to a different parameter set. Note that these stress histories are obtained from the constitutive model with given $\lambda(t)$ and $\vec{x}$, so the experimental stress history curve is not required at this stage. These calculated stress histories are stored as rows in a stress history matrix. We apply SVD to this stress matrix and use the dominant principal components as outputs of the neural network (input to NN is the material parameters). Here we note that SVD has been employed for many years to analyze and reduce the size of training or experimental data [30, 41]. It has been successfully utilized by researchers in many fields, such as the analysis of spectroscopic data [42], of dynamics simulations [43], the genome-wide expression data processing [44], and stress-strain curves compression [30]. SVD can also serve as a noise filter and a small subset of singular values and vectors can effectively represent the experimental data [45]. We then use the trained network to generate a very large number of stress histories. We determine the optimal parameter set by finding the stress history that best fits the experimental data.

We test our adaptive algorithm on three different soft hydrogels with very different time dependent behavior. Specifically, the constitutive model of these gels has 4, 9 and 13 material parameters respectively. For each gel, we perform four uniaxial tension experiments (i.e. four loading histories). Model curves generated from these loading histories are provided to train the NN.

The plan of this paper is as follows: The method is given in section 1. This section focuses on SVD and the input and output layer of the NN. Section 2 summarizes the constitutive model of the three gels we use for validation of our algorithm. Section 3 focuses on the implementation of the adaptive NN. Section 4 presents the main results. Here we compare the prediction of the constitutive

models with experiments after using the adaptive NN to obtain the optimal material parameters.

85 Summary is presented in the last section.

2. Parameter space, Singular value Decomposition, Input, and Output for Neural Networks

Constitutive models of complex materials usually contain many material parameters (say q), many of which cannot be determined directly from experiments. In the following, material parameters are denoted by a q vector $\vec{x} = (x_1, \dots, x_q)^\top \in R^q$. The parameter space Ω is a subset of R^q which contains the parameter vectors $\vec{x}$. Let us consider a uniaxial test with a given stretch history. For any parameter vector, say $\vec{x}_i$, we can compute the stress history of this test by integrating the constitutive model. The computed stress history can be stored as a row vector $\vec{\sigma}_i^\top$, where t is the current time and the subscript i indicates that the stress vector is associated with $\vec{x}_i$. The length of the stress vector m typically increases with the observation time in the experiment. In principle, one can generate a very large number of stress histories (say $n = 10^8$), each corresponding to a different, random parameter vector. If the constitutive model captures the correct physics, some of these 10^8 stress vectors should be very close to the observed stress vector $\vec{\sigma}_{exp}^\top$ in experiment. The difficulty with this approach is the large amount of time required to generate a data set consisting of 10^8 stress histories. For example, if it takes on the average of one second to compute one stress history, it will take about 27,000 hours to compute 10^8 histories. Here we note that the calculation of stress history can be numerically challenging, depending on the constitutive model. Often, it requires solving a set of differential and/or integral equations.

The basic idea is to train a NN to integrate the constitutive model. As we shall see, this reduces the computation time dramatically. The training set Ω_T is a small subset of Ω . Let n denote the number of parameter vectors in Ω_T . For a given stretch history, we numerically integrate the constitutive model and store the stress history as a row vector $\vec{\sigma}_j^\top$ in a n by m matrix S , which we call the stress matrix. The SVD theorem states that

$$S = \sum_{i=1}^r e_i \vec{u}_i \vec{v}_i^\top \quad (1)$$

where r is the rank of S , and $e_1 > e_2 > \dots > e_r > 0$ are the singular values of S , $B_{row} \equiv \{\vec{v}_i^\top, 1 \leq i \leq r\}$ is an orthonormal basis of the row space of S , and $B_{col} = \{\vec{u}_i, 1 \leq i \leq r\}$ is an orthonormal basis for the column space of S . These bases and singular values can be determined by finding eigenvalues and eigenvectors of the symmetric m by m matrix $S^\top S$. Eq. (1) states that the j^{th} row of S , $\vec{\sigma}_j^\top$, is given by

$$\vec{\sigma}_j^\top = \sum_{i=1}^r e_i u_{ij} \vec{v}_i^\top \quad j = 1, \dots, n \quad (2)$$

where u_{ij} is the j^{th} component of the column vector $\vec{u}_i$. Since B_{row} is an orthonormal basis of the row space, $p_{ij} = e_i u_{ij} = \vec{v}_i^\top \cdot \vec{\sigma}_j^\top$ (no sum on i , $\cdot$ is the usual dot product of two row vectors in R^m) is the i^{th} component of $\sigma_j^\top$ with respect to B_{row} . These components p_{ij} are called the principal components of $\sigma_j^\top$. Because the stress vectors are obtained using the same constitutive model, one expects that all but a few of the singular values (say $r^* \ll r$) will be very small (relative to e_1) [30, 46]. If we neglect these, we should get a good approximation for any of the stress vectors, i.e.,

$$\vec{\sigma}_j^\top \approx \sum_{i=1}^{r^*} e_i u_{ij} \vec{v}_i^\top \quad j = 1, \dots, n \quad (3)$$

Since $\vec{v}_i^\top$ is fixed for a given experiment, any row vector or stress history can be represented by r^* principal components $p_{ij} = e_i u_{ij} = \vec{v}_i^\top \cdot \vec{\sigma}_j^\top$, $i = 1, \dots, r^*$. For example, for the PA gels studied in this work, $r^* = 4$. In the following, we denote the vector containing these r^* components as the principal vector $\vec{p}_j$.

Once the NN is trained, we input the parameters in a parameter set Ω_p which is much larger than the training set Ω_T . We then use the NN to compute the stress histories of all vectors in Ω_p . By comparing these stress histories with experiments, we will be able to select the parameter vectors that best fit the data. Details of this procedure will be given below.

3. Three nonlinear viscoelastic solids: Summary of theories and experiments

Our algorithm is applied to study three hydrogel systems subjected to large deformation. The first two systems are highly nonlinear viscoelastic polyampholyte (PA) hydrogels [7]. These gels are synthesized by radical copolymerization of oppositely charged ionic monomers around zero-net charge composition at very high concentration. The first is a physical gel with only dynamic bonds formed by p-styrenesulphonate (NASS) and 3-(methacryloylamino) propyl-trimethylammonium chloride (MPTC). This pure physical (p-PA) gel phase separates into water rich and water poor domains with different dynamic bond strengths. The second gel has the same dynamic bonds but is lightly chemically crosslinked by N, N' -methylenebisacrylamide (MBAA). This chemical crosslinked gel (c-PA) does not phase separate and has much lower toughness than the p-PA gel [47].

The third system is a dual crosslinked Poly(vinylalcohol) (PVA) hydrogel with PVA chains chemically crosslinked by glutaraldehyde and physically crosslinked by borate ions [48, 49]. We have previously developed a three-dimensional (3D) constitutive model which combines the finite strain elasticity of elastomers with the kinetics of bond breaking and reattachment [16, 50] and demonstrated that our model can accurately capture the behavior of both uniaxial tension and shear (cyclic, small strain torsion) tests with complex loading histories at different temperatures [51].

The synthesis of these gels and experimental procedure for tension tests have been reported in

130 [48, 49] and will not be repeated here. Here we report the results of four tension tests for each gel. These test results are used to determine the parameters in the constitutive model.

3.1. Constitutive models

In the following we summarize the constitutive model governing a uniaxial tension test where the stretch ratio $\lambda(t)$ in the loading direction is prescribed for the three gels used in this work. The multi-axial version of these constitutive models were developed and fully explained in our previous works [22, 52]. We start with the c-PA gel (intermediate complexity with $q = 9$), this is followed by the p-PA gel ($q = 13$) and ends with the PVA gel ($q = 4$). In the following, stretching commence at $t = 0$, before this time the sample is in its virgin unstressed state.

The nominal stress for the c-PA gel is [52]

$$\begin{aligned} \sigma(t) = & \omega_{chem} \times 2 \frac{dW_0}{dI_1} \bigg|_{I_1(t)} + 2\rho \frac{dW_0}{dI_1} \bigg|_{I_1(t)} \phi_B(\tau = 0, t, H^{0 \rightarrow t}) (\lambda(t) - \lambda^{-2}(t)) \\ & + \int_0^t \left[\chi(\tau) \phi_B(\tau, t, H^{\tau \rightarrow t}) 2 \frac{dW_0}{dI_1} \bigg|_{I_1=H^{\tau \rightarrow t}} \left[\frac{\lambda(t)}{\lambda^2(\tau)} - \frac{\lambda(\tau)}{\lambda^2(t)} \right] \right] d\tau \end{aligned} \quad (4a)$$

where ω_{chem} is the molar fraction of chemical bonds, W_0 is the strain energy density function for the network, I_1 is the trace of the deformation gradient tensor evaluated at t , ρ is the molar fraction of connected physical cross-links at $t = 0$ and

$$\phi_B(\tau, t, H^{\tau \rightarrow t}) = \left[1 + \frac{\alpha_B - 1}{t_B} \int_{\tau}^t f(H^{\tau \rightarrow t}) ds \right]^{\frac{1}{1-\alpha_B}} \quad (4b)$$

is the survivability function, it is the fraction of physical cross-links that survive from the time of their formation τ to the current time t . In Eq. (4b), t_B is the characteristic breaking time of physical crosslinks, $\alpha_B \in (1, 2)$ is a material parameter that controls the rate of decay of the survivability function and f is a function which measures the dependence of breaking rate on the stretch experienced by a physical bond from its formation at time τ to the current time $t > \tau$. $H^{\tau \rightarrow t}$ is the trace of the deformation tensor at t using the configuration at τ as the reference state. In uniaxial tension, it is

$$H^{\tau \rightarrow t} = \frac{\lambda^2(t)}{\lambda^2(\tau)} + \frac{2\lambda(\tau)}{\lambda(t)} \quad (4c)$$

Finally, $\chi(\tau)$ in Eq. (4a) is the fraction of broken physical bonds divided by the characteristic healing time t_H . This healing rate $\chi(\tau)$ is history dependent and is given by the solution of the nonlinear Volterra integral equation

$$1 - \omega_{chem} = \chi(t)t_H + \int_{-\infty}^t \phi_B(\tau, t, H^{\tau \rightarrow t}) \chi(\tau) d\tau \quad (4d)$$

Here we note that ρ is related to the other parameters by $\rho = \frac{(1-\omega_{chem})t_B}{(2-\alpha_B)t_H+t_B}$; W_0 is given by a 3 term Yeoh's model, i.e.,

$$W_0(I_1) = \sum_{i=1}^3 c_i (I_1 - 3)^i \quad (4e)$$

where c_i are material constants and the breaking function f is

$$f(I_1) = \left(1 + \frac{I_1 - 3}{I_c - 3}\right)^m \quad (4f)$$

where $I_c > 3$ and $m > 0$ are material constants. I_c represents an effective stretch ratio where breaking accelerates, and m measures the severity of accelerated bond breaking. Eq. (4) implies that this model has 9 unknown material parameters $\{c_1, c_2, c_3, \alpha_B, t_B, t_H, m, I_c, \omega_{chem}\}$.

The physical meaning of Eq. (4a) is as follows: the first term is the stress carried by the chemically crosslinked network, the second term is the stress carried by the physical cross-links that remain connected from $t = 0$ to the current time t , the third integral is the stress carried by the healed chains in the physical network which survives to the current time.

The constitutive model for the p-PA gel is similar, except it has no chemical network. However, due to phase separation, there are two physical networks, each with their survivability and breaking functions. The material parameters associated with these networks have the same physical meaning as the physical network in c-PA gel and are labelled by the subscript i ($i = 1, 2$). For example, characteristic breaking times and healing times are denoted by t_{Bi} , t_{Hi} respectively. The nominal stress $\sigma(t)$ is related to the stretch history $\lambda(t)$ by:

$$\begin{aligned} \sigma(t) = & \left[\sum_{i=1}^2 \rho_i \left[1 + \frac{\alpha_{Bi} - 1}{t_{Bi}} f_i(I_i(t)) t \right]^{\frac{2-\alpha_{Bi}}{1-\alpha_{Bi}}} \right] 2 \frac{dW_0}{dI_1} \Big|_{I_1(t)} (\lambda(t) - \lambda^{-2}(t)) \\ & + \sum_{i=1}^2 \int_0^t \chi_i(\tau) \phi_{Bi}(\tau, t, H^{\tau \rightarrow t}) 2 \frac{dW_0}{dI_1} \Big|_{I_1=H^{\tau \rightarrow t}} \left[\frac{\lambda(t)}{\lambda^2(\tau)} - \frac{\lambda(\tau)}{\lambda^2(t)} \right] d\tau \end{aligned} \quad (5a)$$

where $\rho_i = \frac{\omega_i t_{Bi}}{(2-\alpha_{Bi})t_{Hi} + t_{Bi}}$. Similar to Eq. (4d), the healing rate $\chi_i(t)$ for each network is obtained by solving the integral equation

$$\omega_i - \rho_i [\phi_{Bi}(0, t)]^{2-\alpha_{Bi}} = \chi_i(t) t_{Hi} + \int_0^t \phi_{Bi}(\tau, t) \chi_i(\tau) d\tau \quad (5b)$$

The survivability functions have the same form as Eq. (4b) except that the breaking function Eq. (4f) is replaced by

$$f_i(I_1) = \exp \left\{ \left(1 + \frac{I_1 - 3}{I_c - 3} \right)^{m_i} - 1 \right\} \quad (5c)$$

There is a total of 13 independent material parameters in this model, i.e., a parameter vector $\vec{x}$ has components $\{c_1, c_2, c_3, \alpha_{B1}, t_{B1}, t_{H1}, m_1, \alpha_{B2}, t_{B2}, t_{H2}, m_2, I_c\}$.

Lastly, we summarize the model for the PVA gel. The model is a simplified version of c-PA gel where the chain breaking kinetics is independent of stretching, thus $f = 1$ and Eq. (4b) reduces to

$$\phi_B(\tau, t) = \left[1 + \frac{\alpha_B - 1}{t_B} (t - \tau) \right]^{\frac{1}{1-\alpha_B}} \quad (6a)$$

Since chain breaking kinetics is independent of stretch, the system can reach steady state where chain breaking rate equals chain healing rate, that is, there is a steady state solution $\chi = \chi_{ss} =$

$(2 - \alpha_B)t_B^{-1}\rho$ to the integral equation Eq. (4d). Assuming steady state, the integral equation is no longer needed and $\chi(\tau)$ in Eq. (4a) is replaced by χ_{ss} . Finally, for the stretch ratios considered here, strain hardening can be neglected, so $c_2 = c_3 = 0$ in (4e). In previous works, we use $c_1 = \mu/2$ where μ is the small strain shear modulus of the network. Thus, Eq. (4a) reduces to

$$\sigma(t) = \mu[\omega_{chem} + \rho\phi_B(t)] (\lambda(t) - \lambda^{-2}(t)) + \mu\chi_{ss} \int_0^t \left[\phi_B(t - \tau) \left[\frac{\lambda(t)}{\lambda^2(\tau)} - \frac{\lambda(\tau)}{\lambda^2(t)} \right] \right] d\tau \quad (6b)$$

In this model, there are four independent parameters $\{\mu\omega_{chem}, \mu\chi_{ss}, \alpha_B, t_B\}$. Note ρ is not an independent parameter since $\chi_{ss} = (2 - \alpha_B)t_B^{-1}\rho$ and $\rho = \frac{(1 - \omega_{chem})t_B}{(2 - \alpha_B)t_H + t_B}$.

150 To be consistent with our previous work related to PVA constitutive model [50, 46], we use the four parameters $\{\mu\rho, \mu\bar{\gamma}_\infty, \alpha_B, t_B\}$ for model fitting, with $\bar{\gamma}_\infty = \frac{1 - \rho}{t_H + t_B/(2 - \alpha_B)}$.

3.2. Experiments in this study

In this work, we choose these three different materials and their corresponding constitutive models, from simple to complex, to test our algorithm.

155 Uniaxial tension tests are conducted on rectangular specimens of dimension 30 mm (length) x 10 mm (width) x 2mm (thickness). Three cyclic tests and one relaxation test are carried out to determine the parameters for different hydrogels. For PVA gels, we stretch the sample to a stretch ratio of 1.3 with stretch rate 0.003/s (EXP A), 0.01/s (EXP B) and 0.03/s (EXP C) respectively and then unload them to their original lengths (stretch ratio = 1) at the same rates. Then, we
160 stretch the sample to a stretch ratio of 1.3 with stretch rate 0.5/s and then hold for 200 seconds (EXP D). For PA gel, we stretch the sample to a stretch ratio of 2 with stretch rate 0.001/s (EXP 1), 0.01/s (EXP 2) and 0.1/s (EXP 3) respectively and then unload them to their original lengths (stretch ratio = 1) at the same rates. Then we stretch the sample to a stretch ratio of 2 with stretch rate 0.5/s and then hold for 300 seconds (EXP 4). To prevent those gels from drying or swelling, all
165 tests for PVA gels are done in mineral oil, and all tests for PA gels are done in deionized water.

4. Neural network (NN): Implementation

4.1. NN for constitutive models

There are various types of NN [53, 54]. Here we choose the deep neural network (DNN) for its simplicity. A DNN always contains three types of layers: input layer, hidden layer, and output layer.
170 For our parameter fitting problem, the input layer is the parameters of the constitutive model so the number of input nodes equals q ; the output layer is the first r^* principal components, so the number of output nodes is r^* .

There can be many hidden layers. More layers or more nodes in each hidden layer means greater ability to simulate complex models but usually requires larger training sets to reach the best prediction result. In Fig. 1, we show an example of two hidden layers. Each node in the first hidden

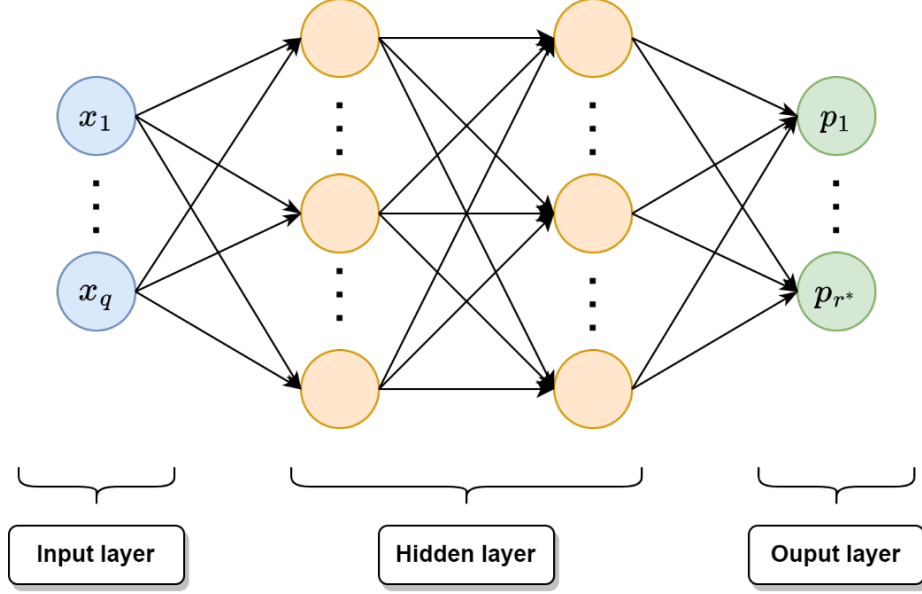


Figure 1: Schematics of DNN. The number of nodes in the input layer equals the number of modal parameters, q , and the number of nodes in the output layer is the number of singular values needed to approximate the stress history, r^* . The outputs are the principal components associated with a particular stress history. Only two hidden layers are shown but more can be used.

layer (say node k) takes as input the q input values of a parameter vector $\vec{x}$ and computes its output $o_{1k}(\vec{x})$ as:

$$o_{1k}(\vec{x}) = \ln(1 + \exp(\vec{w}_{1k} \cdot \vec{x} + b_{1k})) \quad (7a)$$

where $\ln(1 + \exp())$ is the Softplus activation function. Here the components of $\vec{w}_{1k}$ are the weights of the k^{th} node in the first hidden layer, $\cdot$ is the dot product of two vectors and b_{1k} is the bias.

Similarly, the nodes of the 2^{nd} hidden layer will take as input the outputs of the first hidden layer $\vec{o}_1 = (o_{11}, o_{12}, \dots, o_{1h_1})^\top$ (h_1 denotes the total number of nodes in the first hidden layer), and so on (using different weights and biases). For example, for the second hidden layer,

$$o_{2k}(\vec{o}_1) = \ln(1 + \exp(\vec{w}_{2k} \cdot \vec{o}_1 + b_{2k})) \quad (7b)$$

Finally, the output is given by

$$p_i = \vec{w}_{3i} \cdot \vec{o}_2 + b_{3i}, \quad i = 1, 2, \dots, r^* \quad (7c)$$

175 where $\vec{w}_{3i}$ and b_{3i} are the weights (in our case the principal components) and bias of the last hidden layer. Here $\vec{o}_2$ is the output vector of hidden layer 2.

We define the training error by

$$err_{train} = \frac{1}{|\Omega_T|} \sum_{\vec{x}_j \in \Omega_T} \frac{\|\vec{p}_j - \vec{p}_j^{SVD}\|}{\|\vec{p}_j^{SVD}\|} \quad (8)$$

where $\vec{p}_j$ is the output principal vector of the NN associated with $\vec{x}_j$ in the training set and $\vec{p}_j^{SVD}$ is the computed principal vector obtain using SVD. Ω_T is the training set, so $|\Omega_T|$ is the number of elements in the training set. Training means we tune all the weights and biases in the neural network so err_{train} is less than an assigned small number. The tuning process is done using gradient descent method. Details are given in SI.

4.2. Verification and Adaptive Neural network

In the discussion above we show an example of two hidden layers. However, different constitutive models require different number of hidden layers and training set sizes to achieve the best prediction. For example, a constitutive model with fewer material parameters usually requires fewer hidden layers and a smaller training set, whereas a model with more parameters requires more hidden layers and a larger training set. Our goal is to design an accurate and efficient algorithm to determine material parameters for any constitutive model. In the following, we design an adaptive network to update the neural network and training set automatically. The flow chart of our algorithm is given in Fig. 2.

In our NN, the input and output layers are fixed. For the hidden layers, we always start with two hidden layers with the 32 nodes, denoted by $\{32, 32\}$ (It means there are two hidden layers, with 32 nodes in the first hidden layer and 32 nodes in the second hidden layer. If we have $\{32, 64\}$, it means there are two hidden layers, with 32 nodes in the first hidden layer and 64 nodes in the second hidden layer), and 1000 sets of parameters in the training set. We then train this neural network until the training error err_{train} approaches an approximate constant minimum value. Our strategy is as follows: if err_{train} is smaller than the designated value (e. g., 3%), the program goes to the validation part; if err_{train} is larger than the designated value, the number of hidden layers will be increased by one and the new NN is re-trained with the same training set. The number of nodes in the new hidden layer is 64. We continue this process of adding new hidden layers if the training error is larger than the designated value. The number of nodes in these hidden layers are given in equation Eq. (9) below:

$$\frac{\{32, 32\}}{two\ layers} \rightarrow \frac{\{32, 64, 32\}}{three\ layers} \rightarrow \frac{\{32, 64, 64, 32\}}{four\ layers} \rightarrow \frac{\{32, 64, 128, 64, 32\}}{five\ layers} \rightarrow \dots \dots \quad (9)$$

We continue this procedure until is less than the designated value. The strategy is summarized by the flow chart in Fig. 2.

The structure of the neural network and the size of the training set are key factors controlling the accuracy of the prediction. How to choose and optimize the neural network for specific problems is still an open question. In this work, the neural network and the training set update logistics are based on a simple heuristic - build incrementally for more complex systems. Here the number of layers, nodes, and training sets are doubled by a factor of two at each increment. These choices are

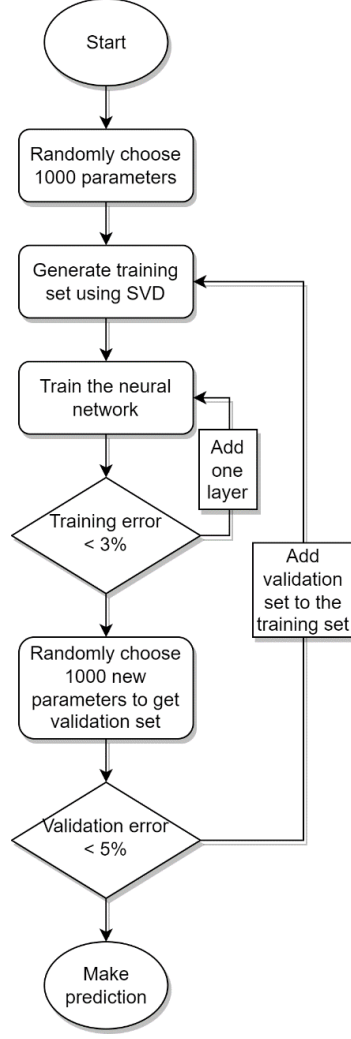


Figure 2: Workflow of adaptive neural network. This framework automatically increases the number of layers and training size according to training error and validation error

efficient enough for the problems in this work, and we believe they can be applied to a wide range of constitutive models.

A small training error means the neural network can predict parameter vectors it has already seen before, not necessarily new vectors. So, the next step is to validate that the trained neural network can still give a small error for parameters that lie outside the training set. This step is called validation. For this part, we uniformly randomly choose 1000 sets of new parameters, Ω_V , from the parameter space Ω and determine their principal vectors using our constitutive model and SVD ($\vec{p}_j^{SVD}$) and neural networks ($\vec{p}_j$) respectively. The validation error is defined in the same way as the training error, i.e.,

$$err_{validation} = \frac{1}{|\Omega_V|} \sum_{\vec{x}_j \in \Omega_V} \frac{\|\vec{p}_j - \vec{p}_j^{SVD}\|}{\|\vec{p}_j^{SVD}\|} \quad (10)$$

200 If the validation error is smaller than a pre-assigned value, we stop training and validation and go to the prediction phase; if the validation error is large, which means the NN makes bad predictions, we will add these new parameter vectors and their principal components (computed using constitutive model) to our training set and return to training.

When training and validation finally ends, the trained NN makes very good prediction of principal
 205 components for any parameter vector in the parameter space. An example is shown in Fig. 3 for a c-PA gel subjected to cyclic loading (strain rate is 0.01/s for loading and unloading, and maximum stretch is 2). After training, we randomly selected 1000 parameters from Ω and then determine their principal components by calculating the c-PA model. We compare these principal components with the predicted principal components using the trained NN. From Fig. 3, the predicted components
 210 are practically identical to the components obtained by integrating the constitutive model, which demonstrates the effectiveness of our NN.

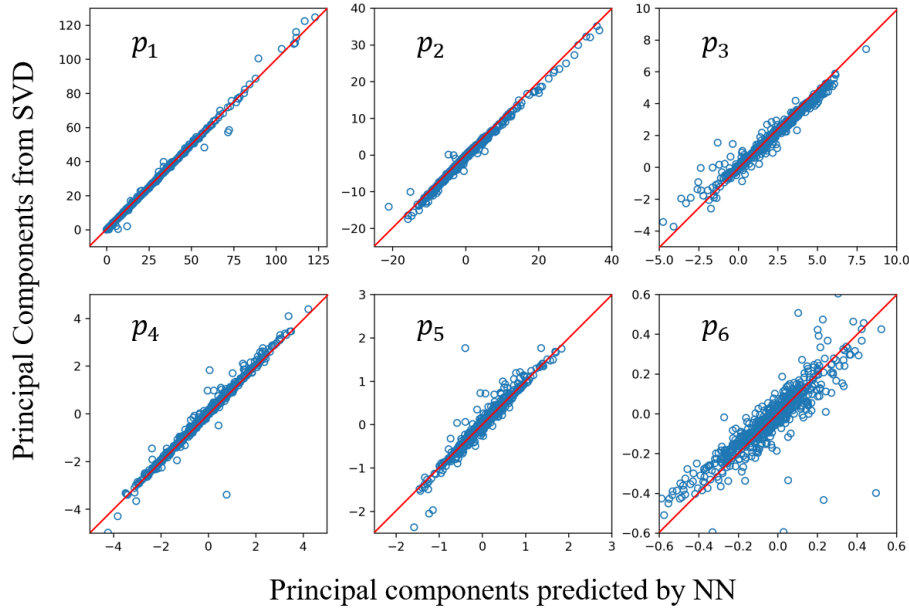


Figure 3: Calculated principal components using constitutive model for c-PA gel versus predicted principal components using NN for cyclic loading. The straight line with slope 1 is used to guide the eye.

Here we briefly explain the mathematical strategy of our update logic. When the training error is large, it means the neural network underfits the data due to simplicity, so we need to increase the complexity of the neural network. When the validation error is large, the neural network does
 215 not sample enough points in the parameter space to make a good prediction, so it is necessary to increase the size of the training set.

4.3. Fitting Experiments

After training and validation, the neural network can predict the principal components of any parameters in the parameter space Ω precisely. Here we utilize our neural network to determine material parameters quickly from experiments.

Let us suppose we carried out a uniaxial experiment with some loading history. We first take the stress history from this experiment to form a vector and then calculate its projection on the basis $\vec{v}_i^\top$ to determine its principal components. Then, we uniformly randomly select a large number of new parameter vectors in a subset Ω_P of the parameter space Ω and predict their principal vectors using the trained neural network. If Ω_P is large enough, there should exist some parameter vectors which fit the experimental data well. To ensure good fitting, we choose $|\Omega_P| = 1000|\Omega_T|$, so the size of the prediction set is larger than 1 million. It is very time-consuming to calculate the constitutive model 1 million times (time is roughly 1s per calculation). However, as we shall see below, it is very fast for the neural network to predict principal components for 1 million parameter vectors. The optimal set of parameters should have principal components closest to the experimental principal components, as defined by Eq. (11a) below.

$$\vec{x}_{opt} = \arg \min_{\vec{x}_j \in \Omega_P} \frac{\|\vec{p}_j - \vec{p}_{exp}\|}{\|\vec{p}_{exp}\|} \quad (11a)$$

Often, one can obtain a set of parameters which fits one experiment extremely well. A more stringent test of a constitutive model is to find a set of parameters which works for a wide variety of loading histories. This situation is more complicated since the basis $\vec{v}_i^\top$ and principal components are different for different loading histories. Here we adopt one of the most straightforward methods, that is, we train different neural networks independently for different loading histories, and then make predictions for different loading histories, using the same prediction set. The optimal parameter vector $\vec{x}_{opt}$ is the one with the smallest average error among all experiments, i.e.,

$$\vec{x}_{opt} = \arg \min_{\vec{x}_j \in \Omega_P} \frac{1}{L} \sum_{l=1}^L \frac{\|\vec{p}_j^{(l)} - \vec{p}_{exp}^{(l)}\|}{\|\vec{p}_{exp}^{(l)}\|} \quad (11b)$$

where L means there are L experiments in total, and l means the l^{th} loading history.

5. Results and Discussion

For each material, we first choose 1000 sets of parameters from their parameter space, calculate the stress matrix and then get the singular values to determine how many singular values we need to approximate an experimental curve. All the results are summarized in figure 4. For all the models and loading histories, the singular value becomes negligible after the 4^{th} term, which means $r^* = 4$ is enough in this work.

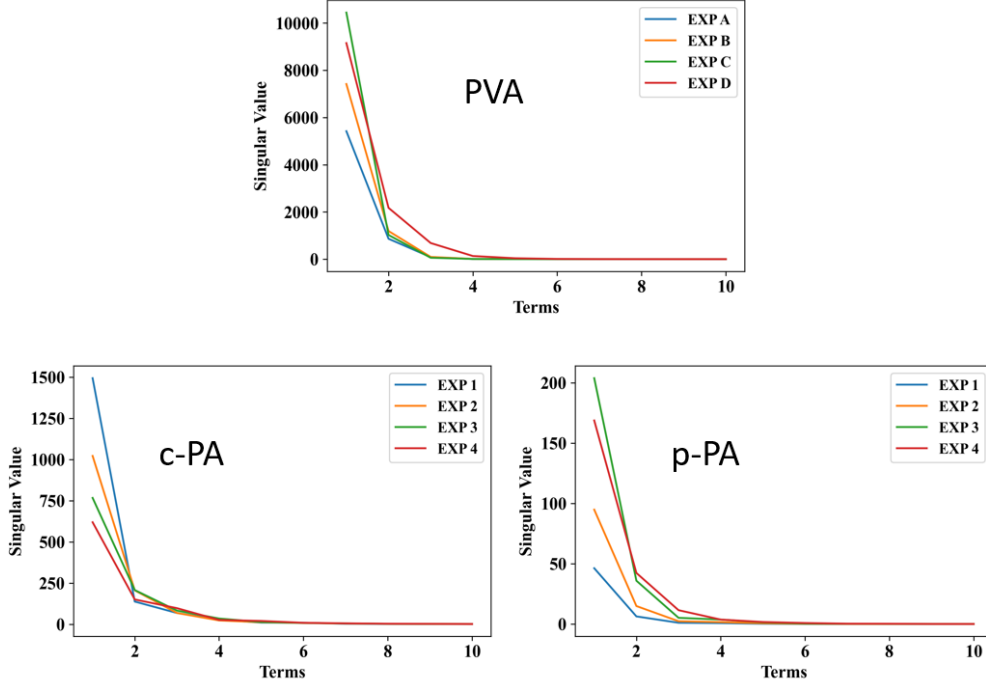


Figure 4: Singular values for different models and loading histories. These results show that 4 singular values are sufficient to well approximate all our experiments.

5.1. PVA model fitting

For PVA gel, the training results are shown in Table 1. The PVA constitutive model has 4 parameters, so it is relatively simple to fit. A neural network with two hidden layers and 1000 training size is good enough to predict the principal components extremely well. In our algorithm, $|\omega_P| = 1000|\Omega_T| = 1 \times 10^6$. This is a large number of parameters, and the last row in Table 1 shows that a brute force calculation will cost 23.4 hours in total to calculate the PVA constitutive model 2 million times for all the four experiments. However, the neural network needs less than 2 seconds to predict the principal components for all parameters in Ω_P . Even if we include the training time, less than 5 minutes is needed to perfectly fit all the experimental curves. The best parameters are

$$(\mu\rho, \alpha_B, t_B \cdot \mu\bar{\gamma}_\infty) = (4.319kPa, 1.648, 0.2206s, 63.85kPa) \quad (12)$$

The fitting plots using Eq. (6b) are shown in Fig. 5

Table 1: Training results for PVA constitutive model

	EXP A	EXP B	EXP C	EXP D
Hidden layers	{32, 32}	{32, 32}	{32, 32}	{32, 32}
Training size	1000	1000	1000	1000
Training error	0.3%	0.4%	0.2%	0.3%
Validation error	3.4%	4.5%	2.8%	2.8%
Training Time	72s	72s	45s	110s
Prediction time	0.34s	0.34s	0.32s	0.34s
Compared to model	4.6h	4.6h	4.6h	9.6h

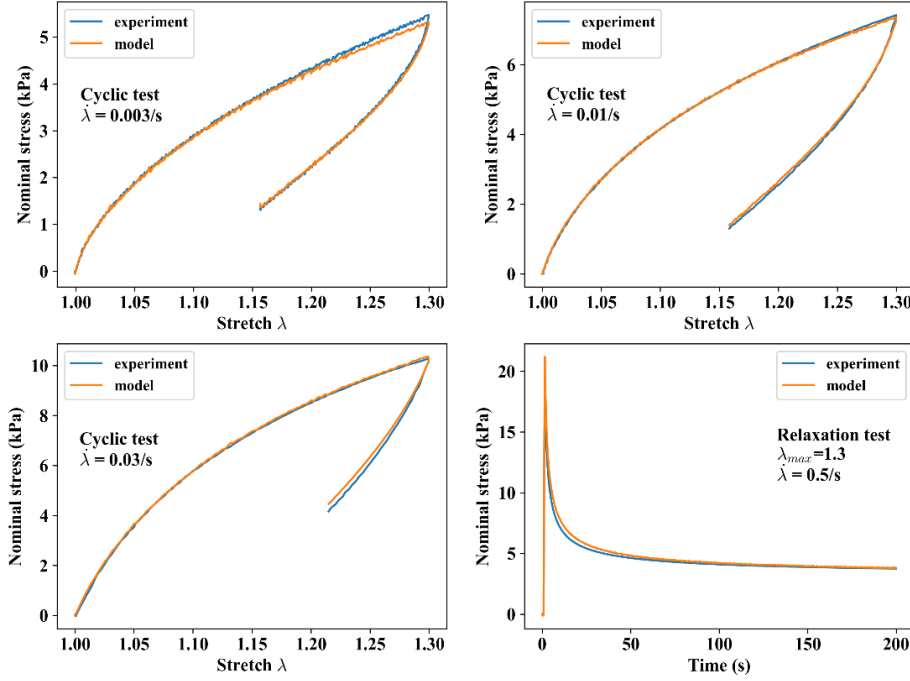


Figure 5: Comparison of prediction using best fitted parameters (Eq. (12)) from NN (orange) and experiment data (blue) for a PVA gel sample subjected to four loading histories.

5.2. c-PA model fitting

For c-PA gel, the training results are shown in Table 2. The c-PA model is more complex than the PVA model since it has 9 parameters. The adaptive neural network captures this complexity, and a good prediction can be achieved with three hidden layers and about 3000 training size. The c-PA model is much more complex as it involves solving an integral equation for each parameter vector input. For $|\Omega_P| = 1000|\Omega_T| = 3 \times 10^6$, roughly 210 hours in total are needed to determine the principal components for all the four loading histories without the use of neural network (last

Table 2: Training results for c-PA constitutive model

	EXP 1	EXP 2	EXP 3	EXP 4
Hidden layers	{32, 64, 32}	{32, 64, 32}	{32, 64, 32}	{32, 64, 32}
Training size	3000	3000	3000	3000
Training error	1.3%	1.5%	1.3%	1.2%
Validation error	4.2%	4.4%	3.2%	4.1%
Training Time	477s	487s	452s	592s
Prediction time	1.03s	0.98s	1.02s	0.99s
Compared to model	42h	42h	42h	84h

row of Table 2).

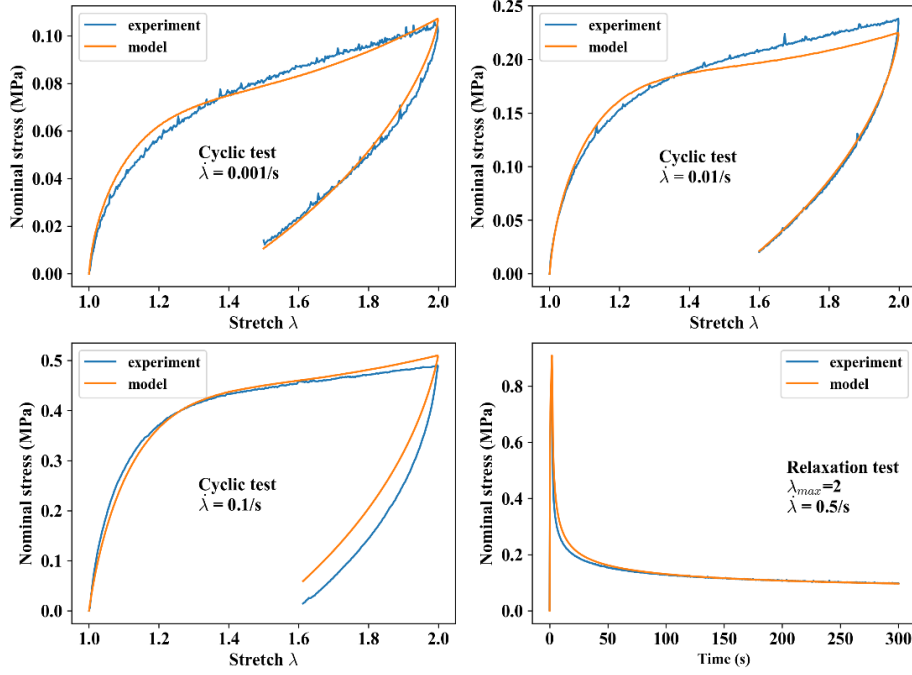


Figure 6: Comparison of prediction using best fitted parameters (Eq. (13)) from NN (orange) and experiment data (blue) for a c-PA gel sample subjected to four loading histories.

The best parameters for c-PA model are

$$\begin{aligned}
 c_1 &= 2.298 \text{ MPa} & c_2/c_1 &= 0.2827 & c_3/c_1 &= 0.0011 & \lambda_c &= 1.2302 \\
 \omega_{chem} &= 0.0023 & \alpha_B &= 1.7030 & t_B &= 0.0984 \text{ s} & m &= 1.2823 & t_H &= 0.5638 \text{ s}
 \end{aligned} \tag{13}$$

The comparison between the prediction of constitutive model using the parameters obtained by NN and experiments is shown in Fig. 6

Table 3: Training results for p-PA constitutive model

	EXP 1	EXP 2	EXP 3	EXP 4
Hidden layers	{32, 64, 128, 64, 32}	{32, 64, 128, 64, 32}	{32, 64, 64, 32}	{32, 64, 64, 32}
Training size	7000	7000	6000	5000
Training error	2.8%	2.3%	2.7%	2.4%
Validation error	3.9%	3.3%	3.8%	4.9%
Training Time	1851s	1551s	1226s	1208s
Prediction time	5.62s	3.96s	3.03s	4.12s
Compared to model	128h	128h	110h	240h

240 5.3. p-PA model fitting

For p-PA gel, the training results are shown in Table 3. The p-PA model has 13 parameters, so it is very complicated and difficult to fit. To get good prediction results, the adaptive neural network chooses five hidden layers and a training set size of 6000. The p-PA model requires solving two integral equations so it takes even more time to integrate the model than the c-PA model. For
245 $|\Omega_P| = 1000|\Omega_T| = 6 \times 10^6$, about 600 hours in total are needed without the help of neural network (last row of table 3), but only less than 2 hours of training and 20 seconds of prediction are needed using our adaptive neural network approach.

The best parameters for p-PA model are

$$\begin{aligned}
c_1 &= 3.459MPa & c_2/c_1 &= 0.4223 & c_3/c_1 &= 0.0506 & \lambda_c &= 1.1166 \\
\omega_1 &= 0.5611 & \alpha_{B1} &= 1.7868 & t_{B1} &= 0.0032s & m_1 &= 0.3940 & t_{H1} &= 0.3850s \\
\omega_2 &= 0.4389 & \alpha_{B2} &= 1.7268 & t_{B2} &= 0.0566s & m_2 &= 0.4722 & t_{H2} &= 0.3542s
\end{aligned} \tag{14}$$

The comparison between the prediction of constitutive model using the parameters (Eq. (14)) obtained by NN and experiments is shown in Fig. 7.

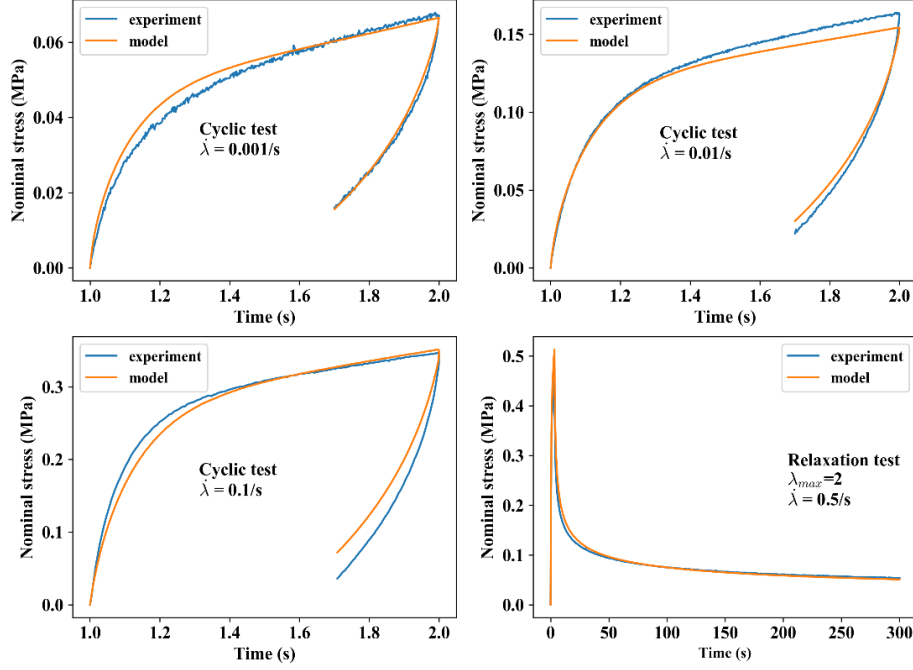


Figure 7: Comparison of prediction using best fitted parameters from NN (orange) and experiment data (blue) for a c-PA gel sample subjected to four loading histories.

6. Summary

A machine learning algorithm based on singular value decomposition and deep neural network are used in this paper to build metamodels for constitutive models, without relying on experimental data. Principal components of stress histories in the training set are extracted using SVD. Deep neural network is trained to simulate the relationship between model parameters and principal components of stress histories. After training, the neural network can predict the stress histories not in the training set very well. This allows us to generate a large number of stress histories based on different parameter sets to fit experiments precisely and quickly. An important feature of our algorithm is its ability to adapt to a wide variety of constitutive models. This method not only aids in parameter fitting but also advances the understanding and analysis of constitutive models. The trained neural network can be stored and distributed, allowing users to quickly obtain optimal parameters without the need for repeated optimization processes. This is in contrast to traditional optimization algorithms, which require new optimization processes for different materials and test batches. Our adaptive neural network method updates the size of the neural network and training size automatically. Three different time dependent constitutive models, from simple to complex, are used as examples to validate this adaptive neural network framework. Both the training results and the fitting results show that our method can automatically capture the complexity of the constitutive model and find one precise and efficient neural network. No extra operation is needed for neural

network building, training, validation, and prediction to accommodate this method for different constitutive models. With the number of parameters in the examples ranging from 4 to 13, our approach in this work shows universality and scalability, thus can be applied to almost any parametric models. In other words, it is not necessary to restrict to rate dependent material models. One of the advantages of our method is that for each type of experiment, one can find a large number of parameter sets that fit the experimental data well. This means that there is a high probability that there are many parameter sets which works well for all experiments (4 in our case). If such a parameter set does not exist, one would conclude that the constitutive model is flawed. The one limitation of our method is that we need different neural networks for different loading histories since the basis is not included in the neural network. Future work may consider possible solutions like including loading conditions in the training set or using some recurrent neural networks to study the time series.

Acknowledgement

This work is supported by the National Science Foundation, under Grant No. CMMI-1903308.

References

- [1] D. T. Chen, Q. Wen, P. A. Janmey, J. C. Crocker, A. G. Yodh, Rheology of soft materials, *Annu. Rev. Condens. Matter Phys.* 1 (1) (2010) 301–322.
- [2] M. Oyen, Mechanical characterisation of hydrogel materials, *International Materials Reviews* 59 (1) (2014) 44–59.
- [3] Z. Zhang, Q. Chen, R. H. Colby, Dynamics of associative polymers, *Soft Matter* 14 (16) (2018) 2961–2977.
- [4] J. Liu, C. S. Y. Tan, Z. Yu, Y. Lan, C. Abell, O. A. Scherman, Biomimetic supramolecular polymer networks exhibiting both toughness and self-recovery, *Advanced Materials* 29 (10) (2017) 1604951.
- [5] Y.-X. Lu, Z. Guan, Olefin metathesis for effective polymer healing via dynamic exchange of strong carbon–carbon double bonds, *Journal of the American Chemical Society* 134 (34) (2012) 14226–14231.
- [6] K. Mayumi, J. Guo, T. Narita, C. Y. Hui, C. Creton, Fracture of dual crosslink gels with permanent and transient crosslinks, *Extreme Mechanics Letters* 6 (2016) 52–59.
- [7] T. L. Sun, T. Kurokawa, S. Kuroda, A. B. Ihsan, T. Akasaki, K. Sato, M. A. Haque, T. Nakajima, J. P. Gong, Physical hydrogels composed of polyampholytes demonstrate high toughness and viscoelasticity, *Nature materials* 12 (10) (2013) 932–937.

- [8] C. Wang, N. Liu, R. Allen, J. B.-H. Tok, Y. Wu, F. Zhang, Y. Chen, Z. Bao, A rapid and efficient self-healing thermo-reversible elastomer crosslinked with graphene oxide, *Advanced materials* 25 (40) (2013) 5785–5790.
- [9] C. Wang, H. Wu, Z. Chen, M. T. McDowell, Y. Cui, Z. Bao, Self-healing chemistry enables the stable operation of silicon microparticle anodes for high-energy lithium-ion batteries, *Nature chemistry* 5 (12) (2013) 1042–1048.
- [10] K. Yu, A. Xin, Q. Wang, Mechanics of self-healing polymer networks crosslinked by dynamic bonds, *Journal of the Mechanics and Physics of Solids* 121 (2018) 409–431.
- [11] P. Cordier, F. Tournilhac, C. Soulié-Ziakovic, L. Leibler, Self-healing and thermoreversible rubber from supramolecular assembly, *Nature* 451 (7181) (2008) 977–980.
- [12] U. Gulyuz, O. Okay, Self-healing poly (acrylic acid) hydrogels with shape memory behavior of high mechanical strength, *Macromolecules* 47 (19) (2014) 6889–6899.
- [13] A. B. Ihsan, T. L. Sun, T. Kurokawa, S. N. Karobi, T. Nakajima, T. Nonoyama, C. K. Roy, F. Luo, J. P. Gong, Self-healing behaviors of tough polyampholyte hydrogels, *Macromolecules* 49 (11) (2016) 4245–4252.
- [14] K. Imato, M. Nishihara, T. Kanehara, Y. Amamoto, A. Takahara, H. Otsuka, Self-healing of chemical gels cross-linked by diarylbibenzofuranone-based trigger-free dynamic covalent bonds at room temperature, *Angewandte Chemie* 124 (5) (2012) 1164–1168.
- [15] F. R. Kersey, D. M. Loveless, S. L. Craig, A hybrid polymer gel with controlled rates of cross-link rupture and self-repair, *Journal of the royal society interface* 4 (13) (2007) 373–380.
- [16] R. Long, K. Mayumi, C. Creton, T. Narita, C.-Y. Hui, Time dependent behavior of a dual cross-link self-healing gel: Theory and experiments, *Macromolecules* 47 (20) (2014) 7243–7250.
- [17] F. Luo, T. L. Sun, T. Nakajima, T. Kurokawa, Y. Zhao, K. Sato, A. B. Ihsan, X. Li, H. Guo, J. P. Gong, Oppositely charged polyelectrolytes form tough, self-healing, and rebuildable hydrogels, *Advanced materials* 27 (17) (2015) 2722–2727.
- [18] A. Phadke, C. Zhang, B. Arman, C.-C. Hsu, R. A. Mashelkar, A. K. Lele, M. J. Tauber, G. Arya, S. Varghese, Rapid self-healing hydrogels, *Proceedings of the National Academy of Sciences* 109 (12) (2012) 4383–4388.
- [19] J.-Y. Sun, X. Zhao, W. R. Illeperuma, O. Chaudhuri, K. H. Oh, D. J. Mooney, J. J. Vlassak, Z. Suo, Highly stretchable and tough hydrogels, *Nature* 489 (7414) (2012) 133–136.

- [20] J. Wang, K. Cui, B. Zhu, J. P. Gong, C.-Y. Hui, A. T. Zehnder, Load transfer between permanent and dynamic networks due to stress gradients in nonlinear viscoelastic hydrogels, *Extreme Mechanics Letters* 58 (2023) 101928.
- [21] S. A. Chester, A constitutive model for coupled fluid permeation and large viscoelastic deformation in polymeric gels, *Soft Matter* 8 (31) (2012) 8223–8233.
- [22] S. P. Venkata, K. Cui, J. Guo, A. T. Zehnder, J. P. Gong, C.-Y. Hui, Constitutive modeling of bond breaking and healing kinetics of physical polyampholyte (pa) gel, *Extreme Mechanics Letters* 43 (2021) 101184.
- [23] Y. Mao, F. Chen, S. Hou, H. J. Qi, K. Yu, A viscoelastic model for hydrothermally activated malleable covalent network polymer and its application in shape memory analysis, *Journal of the Mechanics and Physics of Solids* 127 (2019) 239–265.
- [24] T. Lu, J. Wang, R. Yang, T. Wang, A constitutive model for soft materials incorporating viscoelasticity and mullins effect, *Journal of Applied Mechanics* 84 (2) (2017) 021010.
- [25] Y. Mao, S. Lin, X. Zhao, L. Anand, A large deformation viscoelastic model for double-network hydrogels, *Journal of the Mechanics and Physics of Solids* 100 (2017) 103–130.
- [26] S. Tang, M. S. Greene, W. K. Liu, Two-scale mechanism-based theory of nonlinear viscoelasticity, *Journal of the Mechanics and Physics of Solids* 60 (2) (2012) 199–226.
- [27] V. Crespo-Cuevas, V. L. Ferguson, F. Vernerey, Poroviscoelasto-plasticity of agarose-based hydrogels, *Soft Matter* (2023).
- [28] J. N. Fuhg, C. Böhm, N. Bouklas, A. Fau, P. Wriggers, M. Marino, Model-data-driven constitutive responses: application to a multiscale computational framework, *International Journal of Engineering Science* 167 (2021) 103522.
- [29] S. Saha, Z. Gan, L. Cheng, J. Gao, O. L. Kafka, X. Xie, H. Li, M. Tajdari, H. A. Kim, W. K. Liu, Hierarchical deep learning neural network (hidenn): An artificial intelligence (ai) framework for computational science and engineering, *Computer Methods in Applied Mechanics and Engineering* 373 (2021) 113452.
- [30] C. Yang, Y. Kim, S. Ryu, G. X. Gu, Prediction of composite microstructure stress-strain curves using convolutional neural networks, *Materials & Design* 189 (2020) 108509.
- [31] X. Liu, C. E. Athanasiou, N. P. Padture, B. W. Sheldon, H. Gao, A machine learning approach to fracture mechanics problems, *Acta Materialia* 190 (2020) 105–112.
- [32] J.-J. Wang, C. Wang, J.-S. Fan, Y. Mo, A deep learning framework for constitutive modeling based on temporal convolutional network, *Journal of Computational Physics* 449 (2022) 110784.

- [33] K. D. Saharuddin, M. H. M. Ariff, I. Bahiuddin, S. A. Mazlan, S. A. A. Aziz, N. Nazmi, A. Y. A. Fatah, K. Mohmad, Constitutive models for predicting field-dependent viscoelastic behavior of magnetorheological elastomer using machine learning, *Smart Materials and Structures* 29 (8) (2020) 087001.
- [34] F. Masi, I. Stefanou, P. Vannucci, V. Maffi-Berthier, Thermodynamics-based artificial neural networks for constitutive modeling, *Journal of the Mechanics and Physics of Solids* 147 (2021) 104277.
- [35] M. Liu, L. Liang, W. Sun, Estimation of in vivo constitutive parameters of the aortic wall using a machine learning approach, *Computer methods in applied mechanics and engineering* 347 (2019) 201–217.
- [36] R. Schulte, C. Karca, R. Ostwald, A. Menzel, Machine learning-assisted parameter identification for constitutive models based on concatenated loading path sequences, *European Journal of Mechanics-A/Solids* 98 (2023) 104854.
- [37] S. Zheng, Z. Liu, The machine learning embedded method of parameters determination in the constitutive models and potential applications for hydrogels, *International Journal of Applied Mechanics* 13 (01) (2021) 2150001.
- [38] K. Levenberg, A method for the solution of certain non-linear problems in least squares, *Quarterly of applied mathematics* 2 (2) (1944) 164–168.
- [39] D. W. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, *Journal of the society for Industrial and Applied Mathematics* 11 (2) (1963) 431–441.
- [40] J. J. Moré, The levenberg-marquardt algorithm: implementation and theory, in: *Numerical Analysis: Proceedings of the Biennial Conference Held at Dundee, June 28–July 1, 1977*, Springer, 2006, pp. 105–116.
- [41] E. Henry, J. Hofrichter, [8] singular value decomposition: Application to analysis of experimental data, in: *Methods in enzymology*, Vol. 210, Elsevier, 1992, pp. 129–192.
- [42] E. R. Henry, The use of matrix methods in the modeling of spectroscopic data sets, *Biophysical journal* 72 (2) (1997) 652–673.
- [43] P. Doruker, A. R. Atilgan, I. Bahar, Dynamics of proteins predicted by molecular dynamics simulations and analytical approaches: Application to α -amylase inhibitor, *Proteins: Structure, Function, and Bioinformatics* 40 (3) (2000) 512–524.

- [44] O. Alter, P. O. Brown, D. Botstein, Singular value decomposition for genome-wide expression data processing and modeling, *Proceedings of the National Academy of Sciences* 97 (18) (2000) 10101–10106.
- 395 [45] M. Schmidt, S. Rajagopal, Z. Ren, K. Moffat, Application of singular value decomposition to the analysis of time-resolved macromolecular x-ray data, *Biophysical journal* 84 (3) (2003) 2112–2129.
- [46] J. Wang, T. Li, F. Cui, C.-Y. Hui, J. Yeo, A. T. Zehnder, Metamodeling of constitutive model using gaussian process machine learning, *Journal of the Mechanics and Physics of Solids* 154 (2021) 104532.
- 400 [47] K. Cui, Y. N. Ye, T. L. Sun, C. Yu, X. Li, T. Kurokawa, J. P. Gong, Phase separation behavior in tough and self-healing polyampholyte hydrogels, *Macromolecules* 53 (13) (2020) 5116–5126.
- [48] K. Mayumi, A. Marcellan, G. Ducouret, C. Creton, T. Narita, Stress–strain relationship of highly stretchable dual cross-link gels: separability of strain and time effect, *ACS Macro Letters* 2 (12) (2013) 1065–1068.
- 405 [49] T. Narita, K. Mayumi, G. Ducouret, P. Hebraud, Viscoelastic properties of poly (vinyl alcohol) hydrogels having permanent and transient cross-links studied by microrheology, classical rheometry, and dynamic light scattering, *Macromolecules* 46 (10) (2013) 4174–4183.
- [50] J. Guo, R. Long, K. Mayumi, C.-Y. Hui, Mechanics of a dual cross-link gel with dynamic bonds: Steady state kinetics and large deformation effects, *Macromolecules* 49 (9) (2016) 3497–3507.
- 410 [51] M. Liu, J. Guo, C.-Y. Hui, C. Creton, T. Narita, A. Zehnder, Time-temperature equivalence in a pva dual cross-link self-healing hydrogel, *Journal of Rheology* 62 (4) (2018) 991–1000.
- [52] S. P. Venkata, K. Cui, J. Guo, A. T. Zehnder, J. P. Gong, C.-Y. Hui, Constitutive modeling of strain-dependent bond breaking and healing kinetics of chemical polyampholyte (pa) gel, *Soft Matter* 17 (15) (2021) 4161–4169.
- 415 [53] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, H. Arshad, State-of-the-art in artificial neural network applications: A survey, *Heliyon* 4 (11) (2018) e00938.
- [54] R. Abou Khamis, A. Matrawy, Evaluation of adversarial training on different types of neural networks in deep learning-based idss, in: *2020 international symposium on networks, computers and communications (ISNCC)*, IEEE, 2020, pp. 1–6.
- 420