Metamodeling of Constitutive Model Using Gaussian Process Machine Learning

Jikun Wang[a,1], Tianjiao Li[a,1], Fan Cui[a], Chung-Yuen Hui[b,*], Jingjie Yeo[a,*], Alan T. Zehnder[b,*]

[a] *Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853, USA*
[b] *Field of Theoretical and Applied Mechanics, Cornell University, Ithaca, NY 14853, USA*

Abstract

A method based on Singular Value Decomposition (SVD) and Gaussian process machine learning is proposed to build a metamodel of a material that exhibits time dependent and nonlinear behavior. To test this method, we apply it to determine the material parameters of a nonlinear viscoelastic (poly(vinylalcohol)) hydrogel (PVA). Using the metamodel, we are able to rapidly generate the stress histories for a large set of data points spanning a wide range of material parameters without solving the constitutive model of the PVA gel explicitly. To determine the material parameters, we compare the stress histories predicted by the metamodel with the observed stress histories from laboratory experiments consisting of uniaxial tension cyclic and relaxation tests. The efficiency of the metamodel allows us to determine the material parameters of the constitutive model governing the time-dependent behavior of the PVA gel in a short time. The proposed method shows that there exist many sets of material parameters that can faithfully reproduce the experimental data. Further, our method reveals important relationships between the material parameters in the constitutive model. Although the focus is on the PVA gel system, the method can be easily transferred to build a metamodel for any material model.

Keywords: parameters fitting, singular value decomposition, Gaussian process, PVA hydrogel, viscoelastic model

1. Introduction

A fundamental problem in the mechanics of solids is to develop constitutive models which accurately predict the mechanical response of materials subjected to applied loads. The simplest constitutive model is an ideal elastic solid, which is a reasonable approximation for traditional engineering materials. The elastic solid represents an extreme case of material behavior where there is a unique relation between stress and strain. Many materials, for example, soft solids such as rubber and gels, are viscoelastic or viscoplastic [1, 2, 3]. These materials, besides being rate-sensitive, may

*Corresponding author
Email addresses: ch45@cornell.edu (Chung-Yuen Hui), jingjieyeo@cornell.edu (Jingjie Yeo), atz2@cornell.edu (Alan T. Zehnder)
[1] Contributed equally

sustain very large deformation before failure [2, 4, 5]. The mechanical behavior of rate dependent solids remains a major challenge to material physicists and mechanicians [6, 7, 3, 8].

Experiments on material behavior are time consuming and are often limited by the amount of material available. Further, it is impossible to cover the entire spectrum of loading configurations and histories by experiments. Thus, the development of constitutive models must be complemented by theoretical modeling. Many constitutive models contain many material parameters that cannot be directly determined by experimental data [3, 6]. In addition, validating the theoretical model may be hindered by the limited amount of experimental data available.

An outstanding and important question is how to determine the material parameters for a constitutive model given a finite amount of experimental data. First, there can be many sets of material parameters that can fit a given set of experimental data. Second, the parameters that fit a set of test data may not fit a larger set of data which may or may not contain the original set. The latter case occurs when new data is available, for example, from another research team, or when a different type of test is conducted, say torsion instead of tension. The process of fitting experimental data to theory can be extremely tedious and time-consuming. When the number of material parameters is large, poor fitting of data can occur even if the model captures the correct physics. To our best knowledge, it is still a great challenge to rapidly determine the parameters for complex viscoelastic constitutive models.

There is no doubt that physical insight and intuition are extremely important in the development and validation of material models; nevertheless, recent works show that in a variety of modeling techniques, physical insights can be complemented by advances in machine learning algorithms, see for examples [9, 10, 11, 12, 13, 14, 15, 16, 17]. Specifically, the tools used in the present work, singular value decomposition (SVD) and Gaussian process, have been used by different groups of researchers to predict material behavior or to find crucial parameters for controlling system behavior. For example, Guo et al. used SVD and Gaussian process to predict the solution of parameterized ordinary differential equations [11, 12]; Frankel et al. used Gaussian process to predict the stress response of hyper-elastic solids without assuming a specific constitutive model [14]; Yang et al. used SVD to decompose several stress-strain curves from FEM and then used a convolutional neural network to predict the stress-strain curves of composites beyond the elastic limit [15]; Zheng et al. developed logistic regression-like and neural network-like algorithms to determine the fitting parameters of a constitutive model of hydrogels [16]. The purpose of our current work is to develop a systematic method that uses modern statistics and data science to determine material parameters in complex material systems, especially rate dependent solids. Our method combines SVD and machine learning tools, specifically, Gaussian process. Although our formulation is general, we demonstrate its usage and validate our algorithm by applying it to study the mechanical behavior of a nonlinear viscoelastic hydrogel.

In a series of papers, we and our coworkers have conducted experiments on a hydrogel cross-linked by physical and chemical bonds [6, 7]. Specifically, this gel is composed of chains of poly(vinyl alcohol) (PVA) chemically crosslinked by glutaraldehyde and physically crosslinked by Borax molecules [18, 19]. This PVA gel is highly viscoelastic and can support strains of up to $400 - 500\%$ before failure. The remarkable viscoelasticity of this gel is due to breaking and reforming of physical cross-links. In previous works, we have developed a 3D finite strain viscoelastic model which correctly predicts the time-dependent stress response in uniaxial tension tests and torsion tests subjected to complex strain histories and large deformation, such as loading followed by relaxation or unloading at different strain rates or cyclic loading over a range of frequencies [7]. Therefore, this gel system provides an excellent platform to test the feasibility of using machine learning as a tool to determine material parameters.

The plan of this paper is as follows: In section 2 we define the constitutive model and the training data set. In section 3 we review the tools, SVD and Gaussian process and how to use them to construct metamodels. Subsequently, we explain how to use these tools to find optimal parameters for experimental data in section 4. In section 5, we use the PVA constitutive model as an example to demonstrate how to implement this method. Finally, we show the fitting results and remark on how the Gaussian process machine learning can help us understand the model, and discuss how this method can potentially be improved.

2. Constitutive model

A general form of a nonlinear viscoelastic model can be written as

$$\boldsymbol{\sigma}(\vec{x}, t) = \boldsymbol{f}\left(\vec{x},\, \boldsymbol{F}(t'),\, -\infty < t' \leq t\right) \tag{1}$$

where $\boldsymbol{\sigma}(\vec{x}, t)$ is the nominal stress tensor at time t and $\boldsymbol{F}(t)$ is the deformation gradient tensor at time t; $\boldsymbol{f}$ is a nonlinear functional stating that the stress at time t depends on the entire history of deformation up to the current time t. In Eq. (1), $\vec{x} = (x_1, x_2, \cdots, x_\beta)^\top$ is a vector with β components. Each component represents a material parameter of the constitutive model. In the following, we assume $\vec{x} \in \Omega$ which is an open connected set in $\mathbb{R}^\beta$. We shall call $\vec{x}$ a parameter vector. The nonlinear function $\boldsymbol{f}$ is usually given by theory. We shall call Ω the parameter space. The material parameters are often determined empirically by fitting theory to experiments [6, 7, 3, 20].

A simple way to study the constitutive behavior of materials is to perform a uniaxial tension test. In such a test, there is only one non-trivial stress $\sigma(t)$ and Eq. (1) can be written as

$$\sigma(\vec{x}, t) = f\left(\vec{x},\, \lambda(t'),\, -\infty < t' \leq t\right) \tag{2}$$

$\sigma(\vec{x}, t)$ is the nominal tensile stress and $\lambda(t)$ is the stretch ratio in the loading direction at time t. A common approach to determining $\vec{x}$ is to fit the constitutive model to experimental data. For a

given stretch history, the stress history can be calculated using Eq. (2) for any $\vec{x}$, say $\vec{x} = \vec{x}_1$. This stress history is compared with the experimental data. If the difference between them is unfavorably large, then a different $\vec{x}$, say $\vec{x} = \vec{x}_2$, is chosen and the process is repeated until the fit is acceptable. Here we note that the calculation of the stress history can be numerically challenging, depending on the formulation of the constitutive model. Such calculation often involves solving a set of differential or integral equations.

To overcome these complexities, we propose an alternative and potentially superior approach. The basic idea is as follows: the stress history in an unaxial test can be approximated as a vector $\vec{\sigma}_{exp}$ in some finite dimensional vector space $\mathbb{R}^\zeta$ where ζ is a large number and each entry of $\vec{\sigma}_{exp}$ represents the stress at a particular time point. For each parameter vector, say $\vec{x} \in \Omega$, we can compute the stress history of this test using the constitutive model. In principle, we can generate a very large number of stress histories (say 10^8). This procedure would provide 10^8 vectors $\vec{\sigma}(\vec{x})$ in $\mathbb{R}^\zeta$. If the constitutive model has the correct physics, a small number of these vectors (could be hundreds) should be very close to $\vec{\sigma}_{exp}$. In this way, we can determine many parameter vectors that fit the experiments well. 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 1 sec to compute one stress history, it would take about 27,000 hours to compute 10^8 histories. The goal of this work is to use modern statistical methods to resolve this difficulty.

3. Metamodeling for parametric constitutive models

3.1. Generating stress matrix

As part of the effort to construct the training data set, we construct the stress matrix first. We sample ω different parameter vectors $\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_\omega$ from the parameter space Ω, where ω depends on the number of parameters in the constitutive model. Larger values of ω may be chosen for constitutive models with larger number of parameters. For our PVA model, which has four parameters ($\beta = 4$), see Section 5.2, we use $\omega = 10^3$ (details of how to select $\vec{x}_j$ are given below). ω parameter vectors are in the matrix X that has dimensions of $\omega \times \beta$, i.e.,

$$X = [\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_\omega]^\top, \tag{3}$$

which is also called the matrix of the training inputs. The superscript $\top$ denotes matrix transpose.

For each $\vec{x}_j$, $1 \le j \le \omega$, we compute the stress history $\vec{\sigma}^\top(\vec{x}_j)$ as a function of time t for *the same stretch history* using the constitutive model Eq. (2). This stress history is stored as the j^{th} row vector

$$\vec{\sigma}^\top(\vec{x}_j) = (\sigma(\vec{x}_j, t = \Delta t), \sigma(\vec{x}_j, t = 2\Delta t), \cdots, \sigma(\vec{x}_j, t = \zeta \Delta t)) \tag{4}$$

in a $\omega \times \zeta$ matrix A, which is shown in Fig. 1. The schematic for generating the stress matrix A is illustrated in Fig. 2a. The numerical calculations begin at $t = \Delta t$ and Δt is a small time increment.

	Δt	$2\Delta t$	$\cdots$	$\zeta\Delta t$
$\vec{x}_1$	$\sigma(\vec{x}_1, \Delta t)$	$\sigma(\vec{x}_1, 2\Delta t)$		$\sigma(\vec{x}_1, \zeta\Delta t)$
$\vdots$				
$\vec{x}_j$	$\sigma(\vec{x}_j, \Delta t)$	$\sigma(\vec{x}_j, 2\Delta t)$		$\sigma(\vec{x}_j, \zeta\Delta t)$
$\vdots$				
$\vec{x}_\omega$	$\sigma(\vec{x}_\omega, \Delta t)$	$\sigma(\vec{x}_\omega, 2\Delta t)$		$\sigma(\vec{x}_\omega, \zeta\Delta t)$

Figure 1: Stress matrix A. The j^{th} row of A stores the stress history $\vec{\sigma}^\top(\vec{x}_j)$.

In general, A is a fairly large matrix, e.g. $\omega \sim 10^3$, and $\zeta\Delta t \sim 10^1$ to 10^3 with $\Delta t = 0.1s$, then $\zeta \sim 10^2$ to 10^4 and the matrix has up to 10 million elements. The information stored in the stress matrix A reflects how material parameters control stress histories. However, since the stress histories in this matrix is constructed according to the constitutive model, they should be correlated. Hence one anticipates that the data in this matrix can be compressed and the information in this matrix can be extracted using SVD, which is described below.

3.2. Singular value decomposition (SVD)

Suppose the rank of the stress matrix A is r. The compact SVD of the rank-r $\omega \times \zeta$ matrix A can be factorized into

$$A = U\Sigma V^\top \tag{5a}$$

where

$$U = [\vec{u}_1, \vec{u}_2, \cdots, \vec{u}_r],\ V = [\vec{v}_1, \vec{v}_2, \cdots, \vec{v}_r] \text{ and } U^\top U = V^\top V = I_{r\times r}, \tag{5b}$$

$$\Sigma = \text{diag}(s_1^2, s_2^2, \cdots, s_r^2) \text{ for singular values } s_1^2 \geq s_2^2 \geq, \cdots, \geq s_r^2 > 0. \tag{5c}$$

The columns of V, a $\zeta \times r$ matrix, denoted by $\vec{v}_k \in \mathbb{R}^\zeta,\ 1 \leq k \leq r$, form an *orthonormal basis* of the row space of A. The matrix U, a $\omega \times r$ matrix, also has rank r; the non-zero columns of U, denoted by $\vec{u}_k,\ 1 \leq k \leq r$ are orthonormal vectors in $\mathbb{R}^\omega$ and form a basis for the column space of A.

For many matrices of real-world data, the diagonal entries in Σ decay rapidly [15]. That is, there exists a positive integer ξ such that

$$\sum_{i=1}^{\xi} s_i^2 \gg \sum_{i=\xi+1}^{r} s_i^2, \quad \xi \ll r \tag{6}$$

This means that although $\vec{\sigma}(\vec{x}_j)$ reside in a r-dimensional space, their projections on most directions are small. Thus, the matrix A can be approximated by a smaller rank-ξ matrix, A_ξ, i.e.,

$$A \approx A_\xi = U_\xi \Sigma_\xi V_\xi^\top = s_1^2 \vec{u}_1 \vec{v}_1^\top + s_2^2 \vec{u}_2 \vec{v}_2^\top + \cdots + s_\xi^2 \vec{u}_\xi \vec{v}_\xi^\top \tag{7}$$

where $U_\xi = [\vec{u}_1, \cdots, \vec{u}_\xi]$, $\Sigma_\xi = \text{diag}(s_1^2, \cdots, s_\xi^2)$, $V_\xi^\top = [\vec{v}_1, \cdots, \vec{v}_\xi]^\top$, are $\omega \times \xi$, $\xi \times \xi$ and $\xi \times \zeta$ matrices respectively.

Eq. (7) shows that A can be written as a sum of projection matrices $\vec{u}_k \vec{v}_k^\top$, $k = 1, \cdots, \xi$. Eqs. (5a) and (7) imply that the j^{th} row of A can be approximated as

$$\vec{\sigma}^\top(\vec{x}_j) \approx s_1^2 u_{1j} \vec{v}_1^\top + s_2^2 u_{2j} \vec{v}_2^\top + \cdots + s_\xi^2 u_{\xi j} \vec{v}_\xi^\top = \vec{\psi}^\top(\vec{x}_j) V_\xi^\top, \quad 1 \le j \le \omega \tag{8}$$

where u_{kj} denotes the j^{th} component of the vector $\vec{u}_k$ and $\vec{\psi}(\vec{x}_j)$ is a row vector of length ξ defined by

$$\vec{\psi}(\vec{x}_j) = \left(s_1^2 u_{1j}, s_2^2 u_{2j}, \cdots, s_\xi^2 u_{\xi j}\right)^\top \tag{9}$$

Since the rows of $V_\xi^\top$ or $B = \left\{\vec{v}_k^\top, 1 \le k \le \xi\right\}$ is an approximate orthonormal basis for the row space of A, the entries of $\vec{\psi}(\vec{x}_j)$ represent the coordinates of stress history $\vec{\sigma}^\top(\vec{x}_j)$ with respect to B. The generation of $\vec{v}_k^\top$, $1 \le k \le \xi$ and $\vec{\psi}(\vec{x}_j)$, $1 \le j \le \omega$ is shown in Fig. 2a. Here we state a key assumption:

Assumption 1. *If one keeps sampling from a fixed probability distribution on Ω, and calculate and store the stress histories as rows in A (i.e. the number of rows of A, ω, increases accordingly), $V_\xi^\top$ or $B = \left\{\vec{v}_k^\top, 1 \le k \le \xi\right\}$ will converge.*

Assumption 1 states that, for a *given loading history*, all stress histories generated by the constitutive model are linear combinations of the *same* basis. This is a reasonable assumption because there must exist some relation between stress histories since they are generated by the same constitutive model. We can therefore use the same orthonormal basis B in the row space to compute $\vec{\sigma}^\top(\vec{x}_j)$ where $\vec{x}_j$ *can be any parameters in* Ω. This means we can find a good approximation of $\vec{\sigma}^\top(\vec{x}_j)$ in Ω *without using the constitutive model if we can find* $\vec{\psi}(\vec{x}_j)$, the projection of $\vec{\sigma}^\top(\vec{x}_j)$ on B. This allows us to dramatically increase the data set (in our case from 1000 stress histories to 1 to 10 million stress histories) without doing additional computations.

The next problem to be addressed is determining $\vec{\psi}(\vec{x}_j^*)$ if the parameter vector $\vec{x}_j^*$ is not in the training set. We will solve this problem using Gaussian Process machine learning (GPML).

3.3. Gaussian process machine learning (GPML): estimating $\vec{\psi}(\vec{x}_j^*)$

In the following, $\vec{x}^*$ denotes *a parameter vector that is not in the training inputs but belongs to* Ω. This notation separates two types of stress histories: $\vec{\sigma}^\top(\vec{x}_j)$ denotes a stress history calculated using the constitutive model with parameter vector $\vec{x}_j$ and $\vec{\psi}(\vec{x}_j)$ is the projection of $\vec{\sigma}^\top(\vec{x}_j)$ on B;

$\vec{\sigma}^{\top}_{GP}(\vec{x}^*_j)$ is a stress history estimated using a trained Gaussian process model and $\vec{\psi}_{GP}(\vec{x}^*_j)$ is the projection of $\vec{\sigma}^{\top}_{GP}(\vec{x}^*_j)$ on B. In the following, we review the basics of Gaussian process.

A collection of random variables $\{\, y(\vec{x}_j) \mid \vec{x}_j \in \Omega \,\}$ is said to be drawn from a Gaussian process $\mathcal{GP}$ with a prior of zero mean and covariance function (also known as the kernel function) $k_{x_j x_{j'}} = k(\vec{x}_j, \vec{x}_{j'})$. In the present work, $\vec{x}_j$ and $\vec{x}_{j'}$ are *any two parameter vectors* in the parameter space Ω. We assume, for any finite set of input vectors $\vec{x}_1, \cdots, \vec{x}_\omega \in \Omega$, that the associated finite set of scalar outputs (dependent variable) $\{y(\vec{x}_1), \cdots, y(\vec{x}_\omega)\}$ have a joint Gaussian distribution:

$$\vec{y} = \begin{pmatrix} y(\vec{x}_1) \\ \vdots \\ y(\vec{x}_\omega) \end{pmatrix} \sim \mathcal{N}(0, [K_{XX}]), \quad [K_{XX}] \equiv \begin{bmatrix} k_{x_1 x_1} & \cdots & k_{x_1 x_\omega} \\ \vdots & \ldots & \vdots \\ k_{x_\omega x_1} & \cdots & k_{x_\omega x_\omega} \end{bmatrix} \tag{10}$$

where $[K_{XX}]$ is the covariance matrix which is semi-positive definite and symmetric, and $y \in \mathbb{R}^\omega$. Two popular covariance functions are RBF kernel and Matérn kernel that have kernel hyperparameters [21]. Our choice of the covariance function is discussed in Section 3.5.

Recall that when a single random variable y is distributed normally with mean 0 and variance σ, the standard notation is $y = \mathcal{N}(0, \sigma)$ or, equivalently, the probability density function p is

$$p(y) = \frac{1}{2\pi\sigma} e^{-\frac{y^2}{2\sigma^2}} \tag{11}$$

Consistent with our notation, we identify Ω as the parameter space of our constitutive model and, correspondingly, an input vector $\vec{x}_j \in \Omega$ is a parameter vector specifying a unique location in Ω. The physical meaning of $\vec{y}$ in Eq. (10) will be given in the next section. The theory of Gaussian process addresses the following question: what is the probability distribution of $\vec{y}^* = (y(\vec{x}^*_1), \cdots, y(\vec{x}^*_\eta))^\top \in \mathbb{R}^\eta$ for inputs $\vec{x}^*_j \in \Omega,\ 1 \le j \le \eta$, given that $\vec{y} = (y(\vec{x}_1), \cdots, y(\vec{x}_\omega))^\top$ has occurred? It is important to note that the number of predictions need not be the same as the number of observations, i.e., η need not to be equal to ω. In this work, $\eta = 10^6$ and $\omega = 10^3$. These η parameter vectors that are previously not in the training inputs are called the prediction inputs, and they are stored in a $\eta \times \beta$ matrix X^*, i.e.,

$$X^* = [\vec{x}^*_1, \vec{x}^*_2, \cdots, \vec{x}^*_\eta]^\top, \tag{12}$$

which is called the matrix of the prediction inputs. The theory of Gaussian process states that the distribution over any set of inputs belonging to Ω must have a joint multivariate Gaussian distribution, i.e.,

$$\begin{bmatrix} \vec{y} \\ \vec{y}^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} [K_{XX}] + \delta^2[I] & [K_{XX^*}] \\ [K_{X^*X}] & [K_{X^*X^*}] \end{bmatrix}\right) \tag{13}$$

In Eq. (13), δ^2 represents the noise level and can be treated mathematically as a regularizer (e.g. from experiments or numerical calculation), and $[I]$ is the identity matrix. Here $[K_{XX}]$, $[K_{XX^*}]$, $[K_{X^*X}]$ and $[K_{X^*X^*}]$ are $\omega \times \omega$, $\omega \times \eta$, $\eta \times \omega$ and $\eta \times \eta$ matrices, respectively. Given $\vec{y}$, the Gaussian

process regression predicts that

$$\overline{\vec{y}^*} = [K_{X^*X}] \left[[K_{XX}] + \delta^2 [I] \right]^{-1} \vec{y} \tag{14a}$$

where $\overline{\vec{y}^*}$ is the mean value of the random variable, $\vec{y}^*$; it represents the best estimation of the output value for inputs $\vec{x}_j^* \in \Omega$, $1 \leq j \leq \eta$ when we already know the values of the training set $\{\, y(\vec{x}_j) \mid \vec{x}_j \in \Omega,\, 1 \leq j \leq \omega \,\}$. The variance of $\vec{y}^*$ is given by:

$$var(\vec{y}^*) = \text{diag}\left([K_{X^*X^*}] - [K_{X^*X}] \left[[K_{XX}] + \delta^2 [I] \right]^{-1} [K_{XX^*}] \right) \tag{14b}$$

Eq. (14b) measures the variablity of $\vec{y}^*$ and provides useful information on the reliability of the estimate given by Eq. (14a). The next step is to identify the random variables $y(\vec{x}_j)$ and $y(\vec{x}_j^*)$ in the Gaussian process, $\mathcal{GP}$, with physical quantities in our problem.

3.4. Metamodeling and predicting

Next, we determine stress histories from parameter vectors that are not in the training inputs without using the constitutive model. In other words, we use SVD and Gaussian process to build a *"metamodel of constitutive behavior"* based on the following assumption:

Assumption 2. *Each component in $\vec{\psi}(\vec{x}_j)$ and $\vec{\psi}(\vec{x}_j^*)$ conforms to an independent Gaussian process on Ω.*

Recall $\vec{\psi}(\vec{x}_j)$ and $\vec{\psi}(\vec{x}_j^*)$ both contain ξ components, where ξ is the number of dominant singular values of A. Each component of these coordinate vectors $\vec{\psi}(\vec{x}_j)$ and $\vec{\psi}(\vec{x}_j^*)$ will now be identified as the random variable $y(\vec{x}_j)$ or $y(\vec{x}_j^*)$ in a $\mathcal{GP}$. We further assume these $\mathcal{GP}$s are independent. We have ξ independent $\mathcal{GP}$s. Specifically, we choose

$$\psi_k(\vec{x}_j) = k^{th} \text{ component of } \vec{\psi}(\vec{x}_j), \quad k = 1, 2, \cdots, \xi \tag{15a}$$

$$\psi_k(\vec{x}_j^*) = k^{th} \text{ component of } \vec{\psi}(\vec{x}_j^*), \quad k = 1, 2, \cdots, \xi \tag{15b}$$

For the rest of this paper, $\psi_k(\vec{x}_j)$ or $\psi_k(\vec{x}_j^*)$ denote the k^{th} component of $\vec{\psi}(\vec{x}_j)$ or $\vec{\psi}(\vec{x}_j^*)$. The vectors $\vec{y}_k$, $k = 1, 2, \cdots, \xi$, which are the vectors of the training outputs in the k^{th} Gaussian process model $\mathcal{GP}_k$, correspond to $\vec{y}$ in Eq. (14a):

$$\vec{y}_k \equiv \begin{pmatrix} \psi_k(\vec{x}_1) \\ \vdots \\ \psi_k(\vec{x}_\omega) \end{pmatrix}, \quad k = 1, 2, \cdots, \xi \tag{16a}$$

To predict $\vec{y}_k^*(\vec{x}_j^*)$, we train a total of ξ independent $\mathcal{GP}$s, namely $\mathcal{GP}_k$, $k = 1, 2 \cdots, \xi$, using the training set that comprises the matrix of the training inputs X and the vector of the training outputs $\vec{y}_k$ for $k = 1, 2 \cdots, \xi$. The training process is discussed in detail in Section 3.5. Using Eq. (14a) we obtain

$$\overline{\vec{y}_k^*} = \begin{pmatrix} \overline{\psi_k}(\vec{x}_1^*) \\ \vdots \\ \overline{\psi_k}(\vec{x}_\eta^*) \end{pmatrix} = [K_{X^*X}]_k \left[[K_{XX}]_k + \delta^2 [I] \right]^{-1} \vec{y}_k, \quad k = 1, 2, \cdots, \xi \tag{16b}$$

where $\overline{y}_k^*(\vec{x}_j^*)$ is the *predicted mean value of* k^{th} *component of* $\vec{\psi}(\vec{x}_j^*)$. The subscript k in $[K_{XX}]_k$ and $[K_{X^*X}]_k$ indicates that the elements of these matrices are dependent on $\vec{y}_k$. From Eq. (16b) we obtain $\vec{\psi}_{GP}(\vec{x}_j^*)$ which is an approximation of $\vec{\psi}(\vec{x}_j^*)$, i.e.,

$$\vec{\psi}_{GP}(\vec{x}_j^*) = \left(\overline{\psi_1}(\vec{x}_j^*), \overline{\psi_2}(\vec{x}_j^*), \cdots, \overline{\psi_\xi}(\vec{x}_j^*) \right)^\top \approx \vec{\psi}(\vec{x}_j^*) \tag{17}$$

With the estimated value of $\psi(\vec{x}_j^*)$, the stress history corresponding to $\vec{x}_j^*$ is

$$\vec{\sigma}^\top(\vec{x}_j^*) \approx \vec{\psi}^\top(\vec{x}_j^*) V_\xi^\top \approx \vec{\psi}_{GP}^\top(\vec{x}_j^*) V_\xi^\top \tag{18}$$

In summary, the stress history can be predicted from matrix-vector multiplications without solving for the stress using the constitutive model, as shown in Fig. 2(c).

We now have a better understanding why SVD is first applied to the stress matrix. After SVD, each stress history can be represented by ξ quantities. To build metamodels for all those quantities, a total of ξ independent $\mathcal{GP}$ should be trained, namely $\mathcal{GP}_k$, $k = 1, 2 \cdots, \xi$. Note, in the original stress history matrix, each stress history is defined by ζ quantities, i.e., the number of columns of matrix A. As a result, to build metamodels for the constitutive model, a total of ζ independent $\mathcal{GP}$ are needed. Unfortunately, ζ is usually 10 to 1000 times larger than ξ, so applying Gaussian process directly to the matrix is extremely uneconomical.

3.5. Anisotropic RBF kernel and training

To determine $\psi_{GP}(\vec{x}_j^*)$, we need to specify and evaluate the covariance matrices in Eq. (16b). There are many forms of covariance functions [21]. In this work we use the anisotropic RBF kernel which defines the elements of the covariance matrix $[K_{XX}]$ in Eq. (14a) as

$$k_{x_j x_{j'}} \equiv k(\vec{x}_j, \vec{x}_{j'}) = \sigma_f^2 \exp\left(-\frac{1}{2} \sum_{d=1}^{\beta} (x_{jd} - x_{j'd})^2 / \ell_d^2 \right), \quad 1 \le d \le \beta \tag{19}$$

where the subscript d in x_{jd} denotes the d^{th} component of the parameter vector $\vec{x}_j \in \mathbb{R}^\beta$. ℓ_d is the *characteristic length* specific to x_{jd} while σ_f is the *signal variance*. This class of covariance functions naturally takes care of the coordinate-dependent scaling of the parameter vector by specifying a unique ℓ_d for the d^{th} coordinate of the parameter vector for $1 \le d \le \beta$. This property is especially

suitable in our case where the parameters in a constitutive model usually have different units and scales. In the literature, the anisotropic RBF kernel has been successfully applied to material modeling [22, 23].

Let the set $\boldsymbol{\theta} = \{\ell_1^2, \ell_2^2, \cdots, \ell_\beta^2, \sigma_f^2\}$ denote the parameters in the covariance function. These components are called the hyperparameters of the machine learning model $\mathcal{GP}$. The characteristic lengths ℓ_d measure the distances in the parameter space when $y(\vec{x}_j)$ become uncorrelated [21]. ℓ_d characterizes the dependence of $y(\vec{x}_j)$ on the d^{th} component of $\vec{x}_j$, i.e., x_{jd}. A larger ℓ_d means that $y(\vec{x}_j)$ is smoother in x_{jd}. If ℓ_d is very large compared with the scale of x_{jd}, then $y(\vec{x}_j)$ is approximately linear in x_{jd}. If ℓ_d is very small compared to the scale of x_{jd}, then a small change of x_{jd} can lead to a dramatic change of $y(\vec{x}_j)$. The signal variance controls the range of the functions drawn from a $\mathcal{GP}$. Roughly, σ_f represents the absolute scale of the training outputs.

To find the optimal hyperparameters, we use Assumption 2 which states that $\{y_k(\vec{x}_j), \cdots, y_k(\vec{x}_\omega)\}$ have a joint multivariate Gaussian distribution whose covariance matrix $[K_{XX}]$ depends on the training inputs collected in X and $\boldsymbol{\theta}$, as indicated by Eq. (10) and Eq. (19). Thus, the probability density function of $\vec{y}_k$ given X and $\boldsymbol{\theta}$ is [21]

$$p(\,\vec{y}_k \mid X, \boldsymbol{\theta}\,) = \frac{\exp\left(-\frac{1}{2}\vec{y}_k^{\top}\left[[K_{XX}] + \delta^2[I]\right]^{-1}\vec{y}_k\right)}{\sqrt{(2\pi)^\omega\,|[K_{XX}] + \delta^2[I]|}} \tag{20}$$

where $X = [\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_\omega]^{\top}$. We remark that during training, the hyperparameters in $\boldsymbol{\theta}$ are variables and each element in $[K_{XX}]$ is a function of $\boldsymbol{\theta}$ (dark green box in Fig. 2b). In a nutshell, the training process is as follows: $k = 1, \cdots, \xi$, choose $\boldsymbol{\theta} = \boldsymbol{\theta}_k$ which made the observation of $\vec{y}_k$ most probable (dark boxes in Fig. 2b), i.e.,

$$\boldsymbol{\theta}_k = \arg\max_{\boldsymbol{\theta}}\; p(\,\vec{y}_k \mid X, \boldsymbol{\theta}\,) = \arg\max_{\boldsymbol{\theta}}\; \log p(\,\vec{y}_k \mid X, \boldsymbol{\theta}\,) \equiv \arg\max_{\boldsymbol{\theta}}\; \mathcal{L}_k(\boldsymbol{\theta}), \quad k = 1, 2, \cdots \xi \tag{21}$$

where $\arg\max_{\boldsymbol{\theta}}\; p(\,\vec{y}_k \mid X, \boldsymbol{\theta}\,)$ denotes the $\boldsymbol{\theta}$ that maximizes $p(\,\vec{y}_k \mid X, \boldsymbol{\theta}\,)$. The second equality in Eq. (21) is because the logarithmic function monotonically increases. $\mathcal{L}_k(\boldsymbol{\theta})$ denotes the log marginal likelihood of the observation of $\vec{y}_k$, which is

$$\mathcal{L}_k(\boldsymbol{\theta}) \equiv \log p(\,\vec{y}_k \mid X, \boldsymbol{\theta}\,) = -\frac{1}{2}\vec{y}_k^{\top}\left[[K_{XX}] + \delta^2[I]\right]^{-1}\vec{y}_k - \frac{1}{2}\log\left|[K_{XX}] + \delta^2[I]\right| - \frac{\omega}{2}\log 2\pi \tag{22}$$

Numerical methods such as gradient based techniques are used to solve Eq. (21) for the optimal hyperparameters. In this work the `sklearn.gaussian_process.GaussianProcessRegressor` module implemented in Python [24] is used to determine $\overline{\vec{y}^*}$ while the `L-BFGS-B` algorithm [25, 26] is used to maximize the log marginal likelihood function.

We use the set $\boldsymbol{\theta}_k = \{\ell_{k1}^2, \ell_{k2}^2, \cdots, \ell_{k\beta}^2, \sigma_{fk}^2\}$ to specify the optimized hyperparameters in the covariance function of the k^{th} Gaussian process machine learning model $\mathcal{GP}_k$ (light green boxes in Fig. 2b). They uniquely define $\mathcal{GP}_k$ by Eq. (19) and Eq. (10). Thus, the covariance matrices $[K_{XX}]_k$ and $[K_{X^*X}]_k$ in Eq. (16b) are $[K_{XX}]$ and $[K_{X^*X}]$ evaluated at $\boldsymbol{\theta}_k$. That is,

$$[K_{XX}]_k = [K_{XX}]|_{\boldsymbol{\theta}_k} \quad \text{and} \quad [K_{X^*X}]_k = [K_{X^*X}]|_{\boldsymbol{\theta}_k} \tag{23}$$

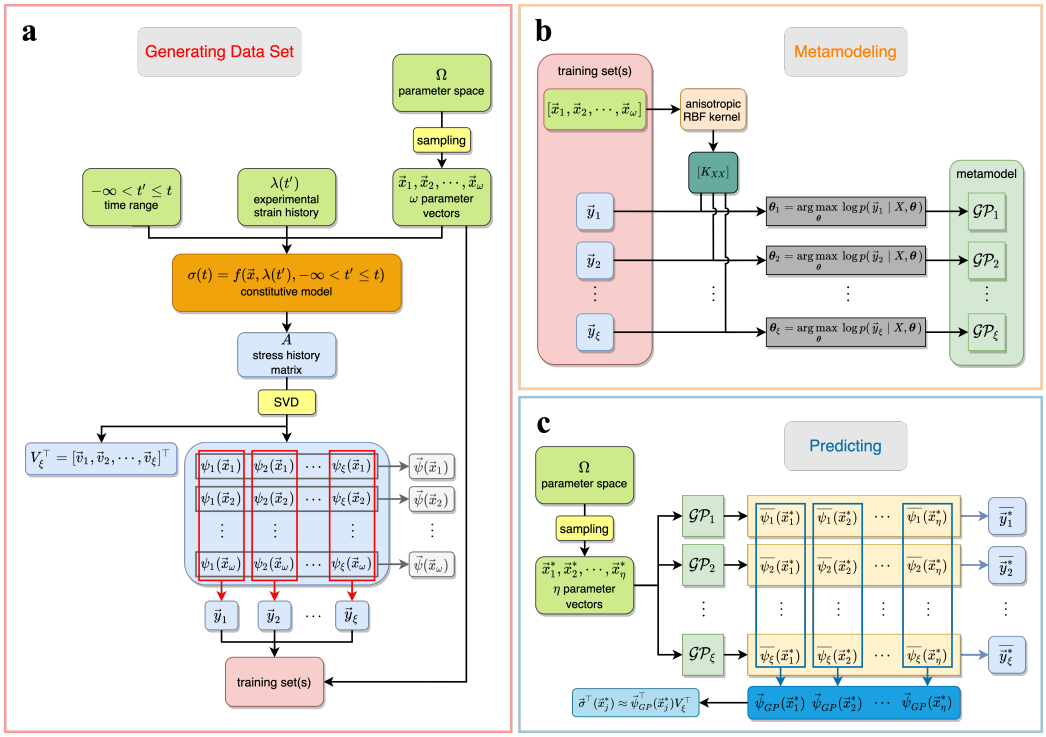

Figure 2: The overall flowchart for (a) generating the data set, (b) metamodeling of the constitutive behaviour, and (c) predicting the stress history $\vec{\sigma}^\top(\vec{x}_j^*)$. The structure of the stress matrix A in (a) is shown in Fig. 1. Matrix $V_\xi^\top$ in (a) and (c) is the transpose of the V matrix in SVD but only with the first ξ columns. Each of the the dark boxes in (b) represents an individual training process that is based on hyperparameter optimization by maximizing the log marginal likelihood.

4. Optimal parameter vector and ranking of parameter vector

Suppose an uniaxial experiment was carried out that had the same strain history as in A. The stress history in this experiment, stored as a vector, is projected onto B to determine a row vector $\vec{\psi}_{exp}$ of length ξ. We define the optimal parameters $\vec{x}_{opt}$ for this experimental data as

$$\vec{x}_{opt} = \arg\min_{\vec{x}^*} \left\| \vec{\psi}(\vec{x}^*) - \vec{\psi}_{exp} \right\|^2, \quad \vec{x}^* \in \Omega \tag{24a}$$

where $\arg\min_{\vec{x}^*} \left\| \vec{\psi}(\vec{x}^*) - \vec{\psi}_{exp} \right\|^2$ means the $\vec{x}^*$ that minimizes $\left\| \vec{\psi}(\vec{x}^*) - \vec{\psi}_{exp} \right\|^2$. Theoretically, it is extremely difficult to evaluate Eq. (24a), because there may be many local minimum in Ω. A practical way of determining $\vec{x}_{opt}$ is to randomly choose q points in Ω and compute

$$\vec{x}_{opt} \approx \arg\min_{\vec{x}_j^*} \left\| \vec{\psi}_{GP}(\vec{x}_j^*) - \vec{\psi}_{exp} \right\|^2, \quad 1 \le j \le \eta \tag{24b}$$

What if experiments are performed with N different strain histories (e.g. a relaxation test and tension tests with different stretch rates)? This situation is more complicated since we have N

different matrices $A^{(i)},\ 1 \leq i \leq N$ and the basis set

$$B^{(i)} = \left\{ \left(\vec{v}_k^{(i)} \right)^\top, \quad 1 \leq k \leq \xi^{(i)} \right\} \tag{25}$$

of the row space of $A^{(i)}$ is specific to $A^{(i)}$. *A superscript* (i) *denotes the quantity's association with the* i^{th} *stress history.* For this case we use the average prediction error to find the material parameters that can fit all the experiments well and define the optimal material parameter vector by $\vec{x}_{opt}$

$$\vec{x}_{opt} \approx \arg\min_{\vec{x}_j} \frac{1}{N} \sum_{i=1}^{N} \frac{\left\| \vec{\psi}_{GP}^{(i)}(\vec{x}_j^*) - \vec{\psi}_{exp}^{(i)} \right\|^2}{\left\| \vec{\psi}_{exp}^{(i)} \right\|^2}, \quad 1 \leq j \leq \eta \tag{26}$$

where $\vec{\psi}_{GP}^{(i)}(\vec{x}_j^*)$ denotes the predicted projections of the stress history of parameter vector $\vec{x}_j^*$ on the basis $B^{(i)}$, as defined in Eq. (17), and $\vec{\psi}_{exp}^{(i)}$ denotes the projections of the experimental stress history on $B^{(i)}$.

We remark that there can be many material vectors $\vec{x}_j^*$ that almost minimize the error defined in Eq. (26). It is important to note that these vectors need not be close to $\vec{x}_{opt}$. There can be different sets of material parameters that can equally fit experiments. Therefore, we rank the material parameter vectors obtained using GP. The rank of a parameter vector $\vec{x}_j^*$ is determined by its associated objective quantity in Eq. (26): the optimal parameter vector defined by Eq. (26) is ranked 1^{st}; the closer this quantity is to the minimum, the lower the rank.

5. Example: PVA hydrogel

In this section we test our theory by applying it to determine the material parameters for the PVA gel system. We start by briefly describing the experiments and the constitutive model.

5.1. Experiments: choosing strain history

We synthesized sheets of PVA gel in our laboratory and performed four different type of uniaxial tension tests ($N = 4$) to provide the experimental data for fitting. Since the pertinent details on chemistry and experimental methods were reported extensively in previous works [6, 7, 20], we summarized the experimental procedures in the supporting information (SI). The four experiments are illustrated in Fig. 3. The first three experiments consist of cyclic loading where the sample is uniaxially stretched at three different constant stretch rates until a maximum stretch ratio of 1.3. The sample is then unloaded at the same magnitude of stretch rate. The fourth loading history is from a stress relaxation test. Here the sample is rapidly stretched at a rate of $0.5/s$ to $\lambda = 1.3$, then held at this stretch for the rest of the test. We denote the four test types as EXP 1 (Cyclic test, $0.003/s$), EXP 2 (Cyclic test $0.01/s$), EXP 3 (Cyclic test $0.03/s$) and EXP 4 (Relaxation test, $0.5/s$).

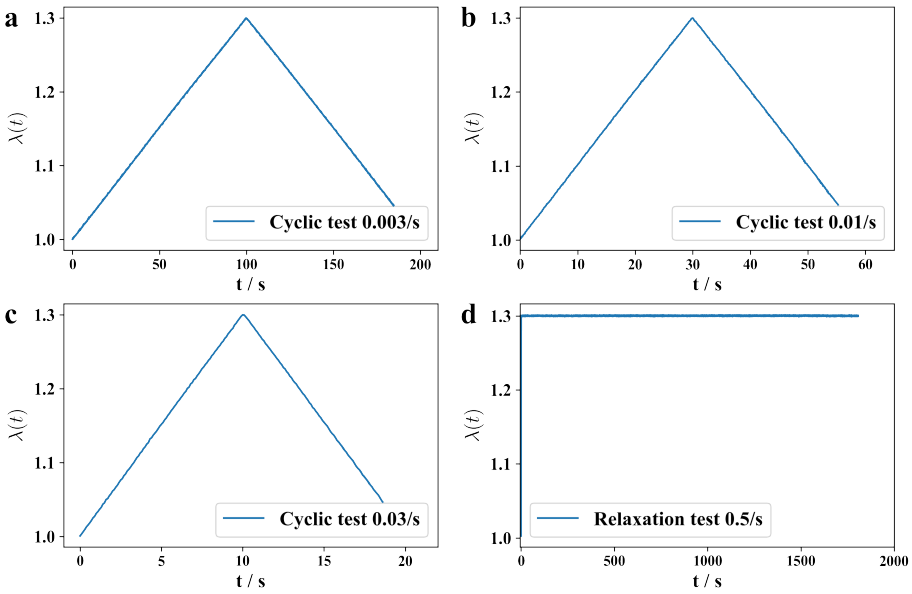

Figure 3: Stretch versus time for four loading histories. (a) EXP 1 (Cyclic test, 0.003/s), (b) EXP 2 (Cyclic test 0.01/s), (c) EXP 3 (Cyclic test 0.03/s), and (d) EXP 4 (Relaxation test, 0.5/s).

5.2. PVA constitutive model

In our previous publications [7, 27, 28], we have developed a 3D constitutive model which combines the finite strain elasticity of elastomers with the kinetics of bond breaking and reattachment. We have also demonstrated that our model accurately predicts uniaxial tension and torsional tests with complex loading histories. The constitutive model for the PVA gel is completely determined by four material parameters $\mu\rho,\ \alpha_B,\ t_B$ and $\mu\bar{\gamma}_\infty$. Hence a parameter vector is specified by a four components vector $\vec{x} = (\mu\rho,\ \alpha_B,\ t_B,\ \mu\bar{\gamma}_\infty)\ (\beta = 4)$. According to our constitutive model, in a uniaxial tension test where the stretch ratio $\lambda(t)$ is prescribed, the nominal stress $\sigma(\vec{x}, t)$ corresponding to the parameter set $\vec{x} = (\mu\rho,\ \alpha_B,\ t_B,\ \mu\bar{\gamma}_\infty) \in \Omega \subset \mathbb{R}^4$ is

$$\begin{aligned}\sigma(\vec{x},t) = &\left[\mu\rho + \mu\bar{\gamma}_\infty \frac{t_B}{2-\alpha_B}\left(1+(\alpha_B-1)\frac{t}{t_B}\right)^{\frac{2-\alpha_B}{1-\alpha_B}}\right]\left[\lambda(t) - \frac{1}{\lambda^2(t)}\right] \\ &+ \mu_j\bar{\gamma}_\infty \int_0^t \phi_B\left(\frac{t-\tau}{t_B}\right)\left[\frac{\lambda(t)}{\lambda^2(\tau)} - \frac{\lambda(\tau)}{\lambda^2(t)}\right] d\tau\end{aligned} \tag{27a}$$

where

$$\phi_B\left(\frac{t}{t_B}\right) = \left(1+(\alpha_B-1)-\frac{t}{t_B}\right)^{\frac{1}{1-\alpha_B}} \tag{27b}$$

The units of parameters are $\mu\rho$(kPa), $\alpha_B(1)$, t_B(s), and $\mu\bar{\gamma}_\infty$(kPa), and the unit of $\sigma(\vec{x}, t)$ is kPa. These units are used in this paper and will not be indicated explicitly. For any parameter vector $\vec{x}_j = \left(\mu_j\rho_j,\ \alpha_{Bj},\ t_{Bj},\ \mu_j\bar{\gamma}_{\infty j}\right) \in \Omega$, we define the stress history as

$$\sigma_j(t) \equiv \sigma(\vec{x}_j, t) \tag{28}$$

5.3. Training and prediction inputs

Next, we use SVD and Gaussian process to build metamodels for the four loading histories ($N = 4$) in Fig. 3. As discussed before, to construct the metamodel, we need to sample parameter vectors for SVD and Gaussian process model training, i.e. $\vec{x}_j,\ 1 \leq j \leq \omega$, and parameters for prediction, i.e. $\vec{x}_j^*,\ 1 \leq j \leq \eta$. Here we limit $\vec{x}_j$ and $\vec{x}_j^*$ in the parameter space Ω defined by

$$\Omega = \left\{ (\mu\rho, \alpha_B, t_B, \mu\bar{\gamma}_\infty) \;\middle|\; \begin{array}{c} 0 \leq \mu\rho \leq 15 \\ 1.30 \leq \alpha_B \leq 1.90 \\ 0 \leq t_B \leq 1.5 \\ 0 \leq \mu\bar{\gamma}_\infty \leq 100 \end{array} \right\} \tag{29}$$

The parameter space Ω is chosen based on data in our previous works [6, 7].

For $\vec{x}_j$, we use Latin Hypercube Sampling (LHS) to randomly pick 1000 points in Ω ($\omega = 10^3$). The main feature of LHS is that, in contrast to simple random sampling, it simultaneously stratifies on all dimensions of the parameter space Ω by partitioning each dimensional distribution into many intervals of equal probability, and selects one sample from each interval, resulting in an efficient and effective sampling scheme for many computer simulations [29, 30, 31]. We also explore the possibilities of sampling $\vec{x}_j$ in an active-learning manner as shown in Fig. S3, and we find that the LHS sampling strategy is simpler and more effective for the presented work. For $\vec{x}_j^*$, we randomly pick 1 million points in Ω ($\eta = 10^6$) using uniform random sampling since we want to cover as much of Ω as possible.

Next, we follow the methods in Section 3 (Fig. 2) to construct metamodels of the PVA constitutive model for the four uniaxial tensile tests mentioned above. Then we collect experimental data with the same tensile strain histories as shown in Fig. 3.

6. Results and Discussion

6.1. The metamodels of PVA constitutive model for four uniaxial tensile tests

We choose $\omega = 1000$ and $\eta = 10^6$. Recall ω is the number of stress histories in the training set, that is, the stress histories calculated directly using the constitutive model, and η is the number of stress histories generated by the Gaussian process models. We first conduct SVD on the stress matrix $A^{(i)},\ 1 \leq i \leq 4$, where each matrix has ω rows generated by solving our constitutive model, as shown in Fig. 1. Fig. 4 shows the SVD of $A^{(i)}$. Note that the singular values of $A^{(i)}$ decay rapidly to zero for $k > 3$ in all four strain histories. This shows that $\xi = 3$ is sufficient for accurately estimating the stresses for all four strain histories.

As described earlier, the hyperparameters can provide useful information about our constitutive model. The optimized hyperparameters are listed in Table 1. Table 1 shows that the stress history is most sensitive to α_B, less sensitive to t_B and least sensitive to $\mu\rho$ and $\mu\bar{\gamma}_\infty$. Specifically, in all

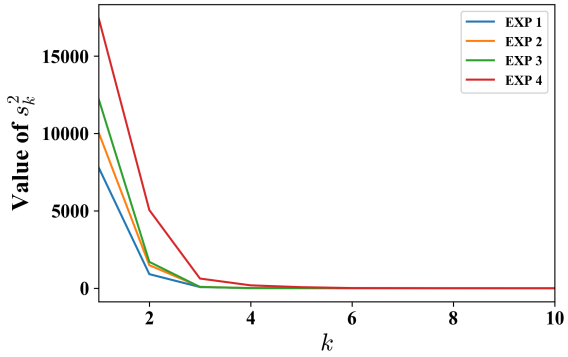

Figure 4: Singular values of four strain histories. The singular values for each strain history are presented by different color lines. s_k^2 values for $k > 10$ are not shown because they are too small compared with the first 4 singular values

Table 1: Summary of optimized hyperparameters. Since $\xi = 3$ and $N = 4$, we have a total 12 different sets of optimized hyperparameters. To simplify notation, we omit the subscript k for each component. ℓ_1, ℓ_2, ℓ_3 and ℓ_4 are the characteristic lengths of $\mu\rho$, α_B, t_B and $\mu\overline{\gamma}_\infty$ respectively, as defined in Eq. (19)

	Train	σ_f	ℓ_1 †	ℓ_2	ℓ_3	ℓ_4
EXP 1	$\vec{y}_1^{(1)}$	196	1.00×10^4	0.185	2.94	393
	$\vec{y}_2^{(1)}$	197	1.00×10^4	0.178	2.75	416
	$\vec{y}_3^{(1)}$	188	9.92×10^3	0.170	2.66	477
EXP 2	$\vec{y}_1^{(2)}$	164	1.00×10^4	0.199	2.66	381
	$\vec{y}_2^{(2)}$	148	1.00×10^4	0.189	2.54	408
	$\vec{y}_3^{(2)}$	144	9.98×10^3	0.194	1.85	491
EXP 3	$\vec{y}_1^{(3)}$	133	1.00×10^4	0.213	2.39	367
	$\vec{y}_2^{(3)}$	123	1.00×10^4	0.199	2.23	428
	$\vec{y}_3^{(3)}$	74	7.52×10^3	0.224	1.07	461
EXP 4	$\vec{y}_1^{(4)}$	145	1.00×10^4	0.213	2.36	377
	$\vec{y}_2^{(4)}$	266	1.00×10^4	0.174	2.91	437
	$\vec{y}_3^{(4)}$	238	9.66×10^3	0.170	2.21	468

† the upper limit of ℓ_1 in the optimization is set to be 1.00×10^4.

experiments, the characteristic length ℓ_1 is very large compared to the scale of $\mu\rho$ ($0 \leq \mu\rho \leq 15$), this is consistent with Eq. (27a) which indicates that the stress history depends almost linearly on $\mu\rho$. On the other hand, ℓ_2 is very small compared to the scale of α_B ($1.3 \leq \alpha_B \leq 1.9$), consistent with the fact that the stress history is very sensitive to α_B. Finally, ℓ_3 is very close to the scale of t_B ($0 \leq t_B \leq 1.5$) and ℓ_4 is 2 to 6 times larger than the scale of $\mu\overline{\gamma}_\infty$ ($0 \leq \mu\overline{\gamma}_\infty \leq 100$). This is again consistent with Eq. (27a) which shows that stress changes smoothly (but not linearly) with

t_B and $\mu\overline{\gamma}_\infty$.

To justify our choice of ω and η, we randomly choose 1000 random parameter vectors from all of $\vec{x}_j^*$. We then compute the $\vec{\psi}(\vec{x}_j^*)$ using Eq. (27a) for both the cyclic and relaxation tests and compare them to the predicted $\vec{\psi}_{GP}(\vec{x}_j^*)$ using the Gaussian process models. For each test and each component of $\vec{\psi}(\vec{x}_j^*)$, the prediction of our constitutive model vs the prediction of the Guassian process models is shown in Fig. 5. In all cases, the predicted value is practically the same as the computed value. This means the predicted $\vec{\psi}_{GP}(\vec{x}_j^*)$ by the models is highly accurate and we can determine stress histories outside the training set without integrating the constitutive model.

Other than comparing the projections, we can also compare the stress histories calculated by the PVA constitutive model in Eq. (27a) and the stress histories predicted by metamodels for a random set of parameters in Ω, as shown in Fig. 6. The curves predicted by the metamodels overlap almost perfectly with the curves calculated using the constitutive model. This further demonstrates the accuracy of our metamodels.

6.2. Comparison with Experiments

We then assess whether the material parameters determined by our method can accurately fit experimental data. Fig. 7 compares the experimental stress-strain curves predicted using Eq. (27a) with the rank 1^{st} vector given by Eq. (30) for the 4 different experiments in Fig. 7. The agreement between experiments and theory is excellent. These results further demonstrate that our PVA constitutive model correctly captures the mechanical behavior of PVA gels. In addition, it shows that our machine learning algorithm is a powerful tool for determining material parameters in constitutive models. The optimal set of parameters (rank 1^{st}) is

$$(\mu\rho,\ \alpha_B,\ t_B,\ \mu\overline{\gamma}_\infty) = (4.744, 1.579, 0.841, 13.48) \tag{30}$$

where the units are noted in section 5.2.

For comparison, we use the parameter vector that ranks 500^{th}. This set of parameters is

$$(\mu\rho,\ \alpha_B,\ t_B,\ \mu\overline{\gamma}_\infty) = (4.819, 1.651, 0.169, 82.72) \tag{31}$$

There are significant differences between parameters t_B, $\mu\overline{\gamma}_\infty$ in Eq. (30) and Eq. (31). Despite these differences, the prediction based on Eq. (31) still agrees well with the experimental data, as shown in Fig. 8. Indeed, the top-ranked 500 parameter vectors all produce stress histories that fit the experiments well.

6.3. Distribution of possible parameters

As mentioned above, the prediction of the constitutive model using the top-ranked 500 parameter vectors can fit all four experiments very well, despite considerable variations amongst them. This suggests there must be some relation between parameters. Here we make 2D projections of $\vec{x}_j^*$ onto

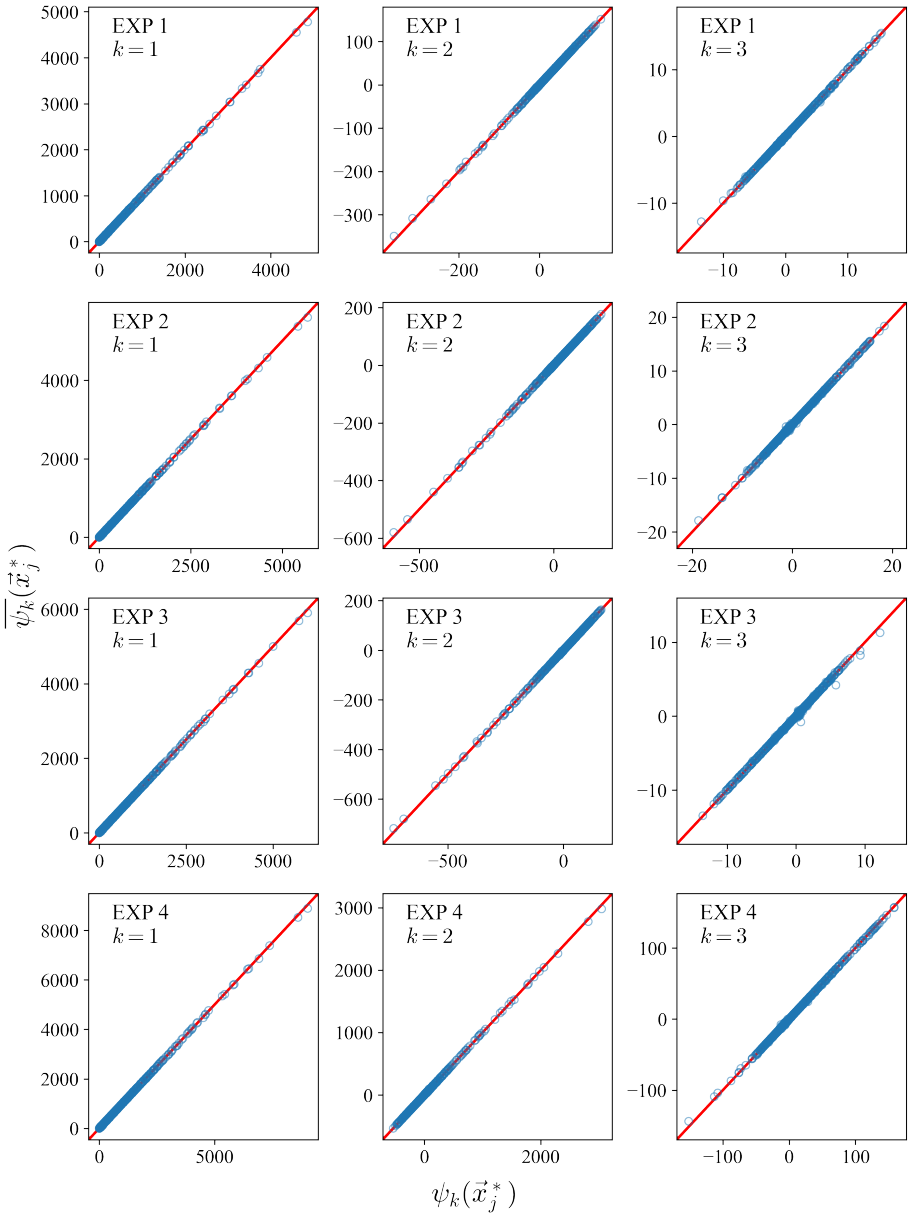

Figure 5: Predicted projection $\overline{\psi_k}(\vec{x}_j^*)$ using GP versus projections $\psi_k(\vec{x}_j^*)$ obtained using constitutive model for four different types of experiments.

two-dimensional subspaces of Ω and plot them in Fig. 9. To be consistent with Eq. (26), the fitting error of a parameter vector $\vec{x}_j^*$ is defined by

$$err(\vec{x}_j^*) = \frac{1}{N}\sum_{i=1}^{N} \frac{\left\| \vec{\psi}_{GP}^{(i)}(\vec{x}_j^*) - \vec{\psi}_{exp}^{(i)} \right\|^2}{\left\| \vec{\psi}_{exp}^{(i)} \right\|^2}, \quad 1 \le j \le \eta \tag{32}$$

Fig. 9 shows that these 500 parameter vectors vary a great deal despite the fact that $err(\vec{x}_j^*)$ are less than 0.065, with most below 0.05. We let $\mathcal{X}$ denote the set of these 500 parameter vectors. The

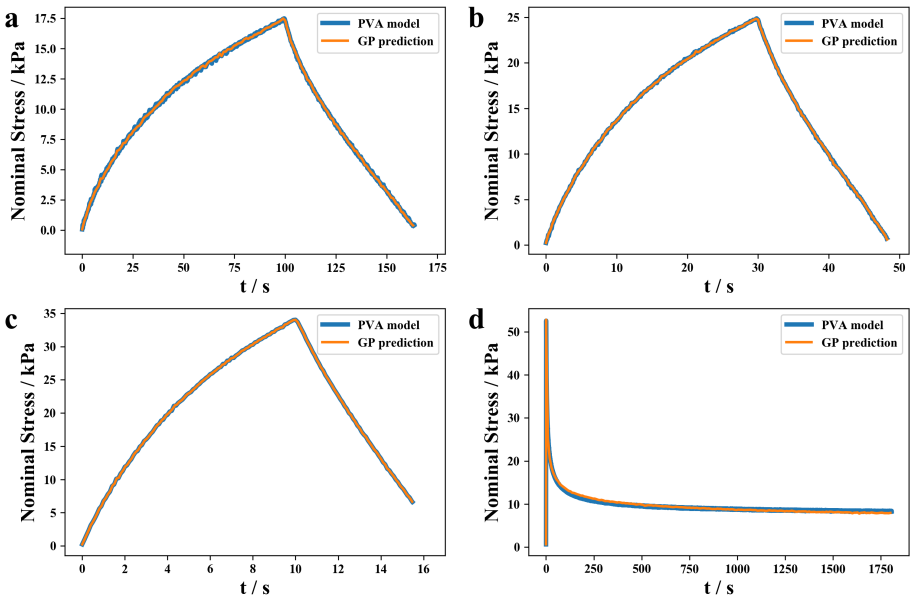

Figure 6: Predicted stress history versus stress history obtained by solving Eqs. (27a) and (27b) using a random parameter vector. Each history uses the same parameter vector.

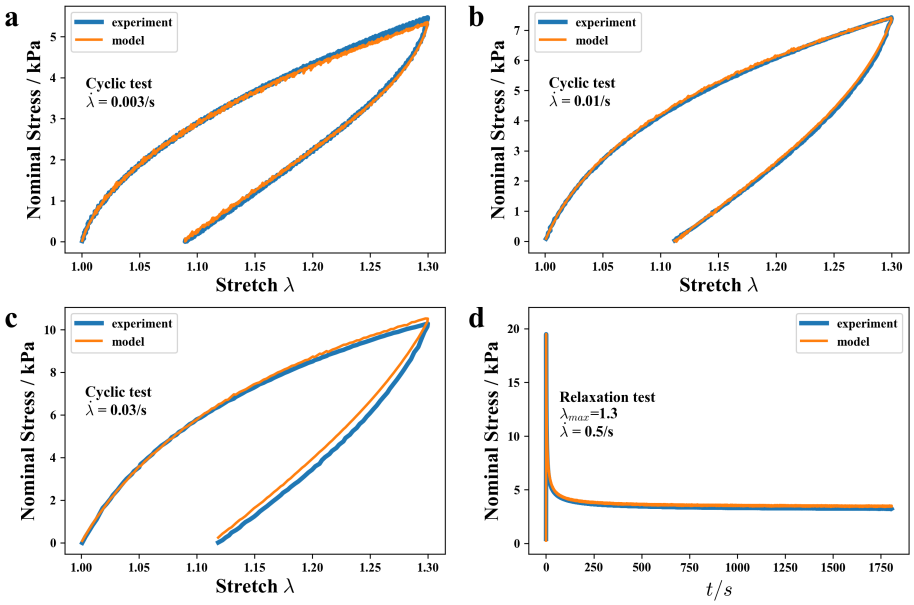

Figure 7: Comparison of model prediction with experiments. Model prediction is based the rank 1^{st} parameter vector given by Eq. (30).

inescapable conclusion is that there are many sets of material parameters that fit these experiments. More importantly, these plots show that there are strong relationships between $\mu\rho$ and α_B, as well as t_B and $\mu\overline{\gamma}_\infty$.

To understand this dependence, we use a result from our previous work [7] which shows that

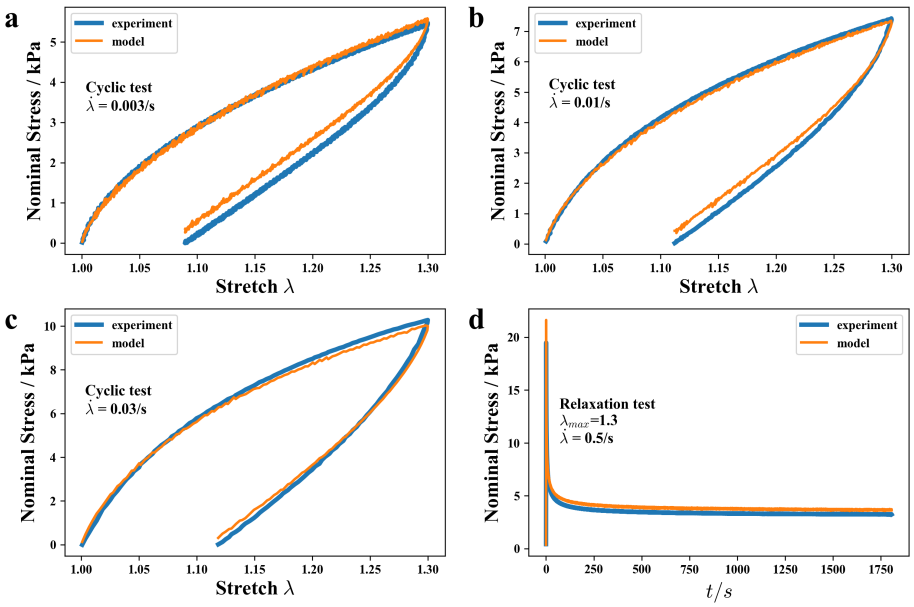

Figure 8: Comparison of model prediction with experiments. Model prediction is based the rank 500^{th} parameter vector given by Eq. (31).

when strains are small, that is, when $\lambda(t) = 1 + \epsilon(t) \sim 1$, our nonlinear viscoelastic model becomes linear, resulting in

$$\sigma(\vec{x}, t) = \int_{-\infty}^{t} Y\left(\frac{t-\tau}{t_B}\right) \frac{d\epsilon(\tau)}{d\tau} d\tau \tag{33a}$$

where Y is the relaxation function in tension and is given by

$$Y(t) \equiv \frac{3\mu\overline{\gamma}_\infty t_B}{2-\alpha_B}\left(1 + (\alpha_B - 1)\frac{t}{t_B}\right)^{\frac{2-\alpha_B}{1-\alpha_B}} + 3\mu\rho \tag{33b}$$

Eqs. (33a) and (33b) states that $\sigma(\vec{x}, t)$ depends linearly on $\mu\overline{\gamma}_\infty t_B$. If the constitutive law is correct, then $\mu\overline{\gamma}_\infty t_B$ must be a material constant C for a specific experiment, therefore $\mu\overline{\gamma}_\infty = C/t_B$. Also, Eqs. (33a) and (33b) shows that $\sigma(t)$ increases with $\mu\rho$ and α_B. Therefore, to produce the same stress history, a smaller α_B must be chosen if a larger value of $\mu\rho$ is chosen. Fig. 9 shows that these relationships are consistent with machine learning prediction.

6.4. Insight for experiment design and model analysis

The question to be addressed in this section is to what extent the set of good parameters $\mathcal{X}$ can be narrowed down if more experiments are incorporated. To answer this question, we design four additional loading histories (EXP 5 – 8) which we could do in our lab, as shown in Fig. 10. Then we build metamodels for these newly designed loading histories according to Eq. (1) through Eq. (23). With these metamodels, we can predict the stress histories for these loading histories for any parameter vector.

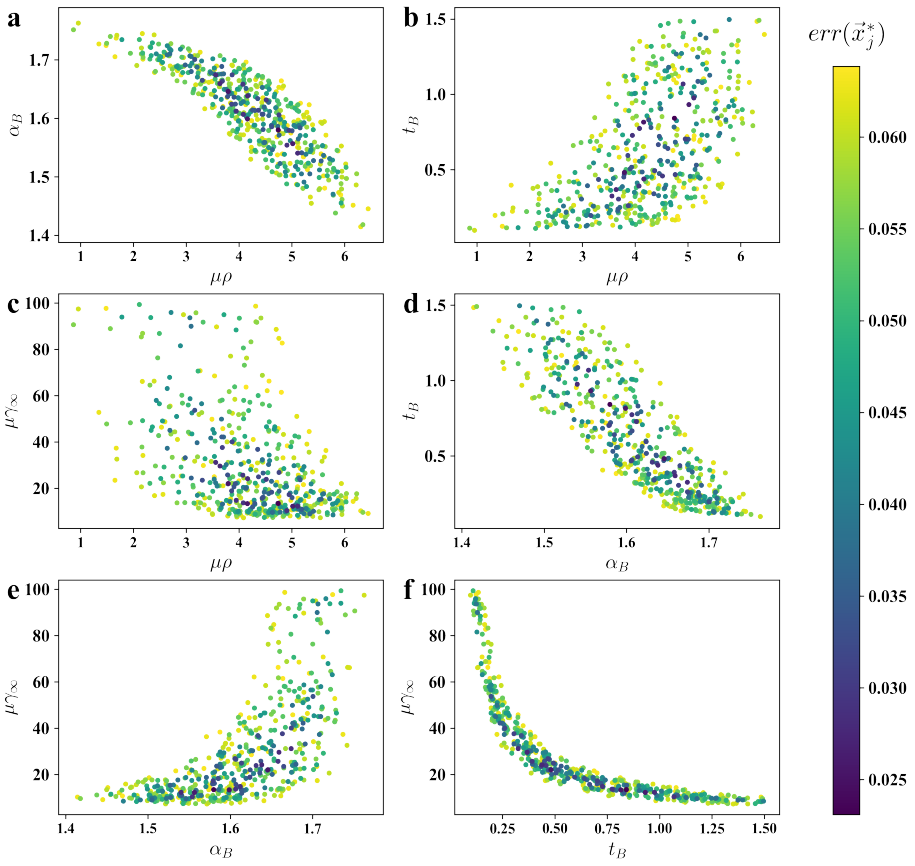

Figure 9: Distribution of the top 500 parameter vectors in parameter space. (a) shows that $\mu\rho$ and α_B are strongly correlated. (f) shows that t_B and $\mu\gamma_\infty$ are strongly correlated. The points are colored according to their corresponding errors defined in Eq. (32).

It is important to specify that we do not plan to do all the newly designed experiments (i = 5, 6, 7, 8). Instead, with the help of the metamodels, we first analyze whether these new experiments (i = 5, 6, 7, 8) could reduce the number of good parameter vectors. When plugging all $\vec{x}_j^* \in \mathcal{X}$ into one of the newly built metamodels, the metamodel returns a set of stress histories, which may or may not deviate significantly from each other. Naturally, one expect an experiment worth doing to show significantly diverse stress history curves on $\mathcal{X}$. To quantify this diversification, we use the dispersion of vectors $\vec{\psi}_{GP}^{(i)}(\vec{x}_j^*)$ in experiment i. Specifically, we use the generalization of normalized standard deviation of $\vec{\psi}_{GP}^{(i)}(\vec{x}_j^*), \vec{x}_j^* \in \mathcal{X}$ for each of the experiments, $S(i, \mathcal{X})$, $1 \leq i \leq 8$, to capture such dispersion:

$$S(i,\mathcal{X}) = \sqrt{\frac{\frac{1}{|\mathcal{X}|}\sum_{\vec{x}_j^* \in \mathcal{X}} \left\| \vec{\psi}_{GP}^{(i)}(\vec{x}_j^*) - \overline{\vec{\psi}}_{\mathcal{X}} \right\|^2}{\left\| \overline{\vec{\psi}}_{\mathcal{X}} \right\|^2}} \tag{34a}$$

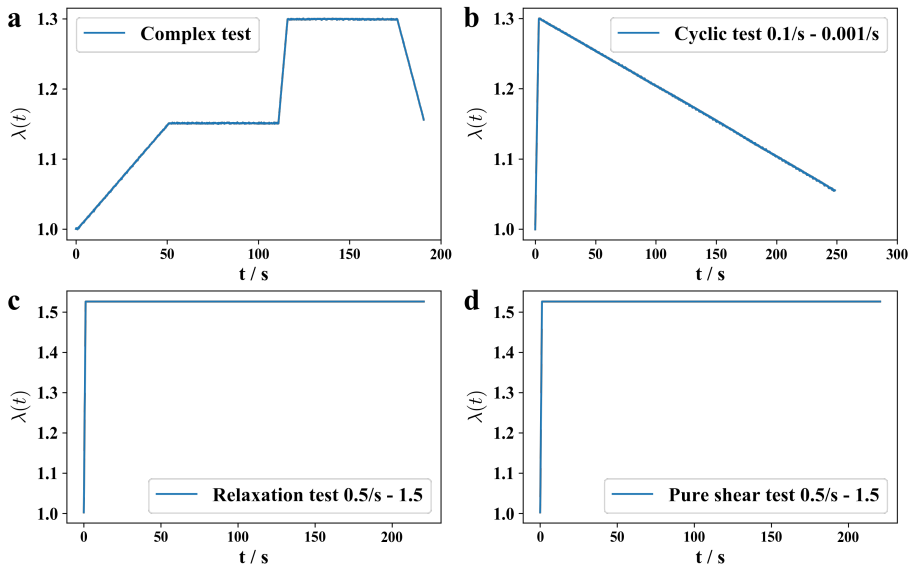

Figure 10: Stretch versus time for four new loading histories. (a) EXP 5 (Complex loading), (b) EXP 6 (Cyclic test, loading - 0.1/s, unloading - 0.001/s), (c) EXP 7 (Relaxation test, 0.5/s, 1.5 stretch), and (d) EXP 8 (Pure shear relaxation test, 0.5/s, 1.5 stretch).

where

$$\overline{\vec{\psi}}_{\mathcal{X}} = \frac{1}{|\mathcal{X}|} \sum_{\vec{x}_j^* \in \mathcal{X}} \vec{\psi}_{GP}^{(i)}(\vec{x}_j^*) \tag{34b}$$

The values of $S(i, \mathcal{X})$ for all experiments are shown in Fig. 11. We find that these values are all small. Additionally, EXP 5 – 8 has even smaller $S(i, \mathcal{X})$ than EXP 4. This means that even if we conduct these new experiments, they will offer very limited help to narrow down the set of good parameter vectors. To further test our hypothesis, we use EXP 1 – 6 to fit our model, and the results are shown in Fig. S5. It turns out that the number of good parameter vectors remains technically the same. This result is consistent with our fluctuation analysis above.

Therefore, adding more experiments do not necessarily effectively reduce the number of feasible parameter sets. The key is to find the right experiment. This is an interesting and important topic which we will investigate in the future.

6.5. Computational efficiency

We have successfully built a metamodel, which predicts the stress history for parameter vectors, to determine the material parameters in a viscoelastic model of a PVA hydrogel. Why do we not simply compute the stress histories for each parameter vector using the constitutive model? *Time is the most important factor.* Table 2 shows that Eq. (27) takes 2.7 to 41.0 ms to calculate the stress history for cyclic loading (EXP 1-3) per set of parameters (using Eqs. (27a) and (27b)), while it takes the metamodel no more than 0.1 ms. Furthermore, the time needed to compute the stress history is

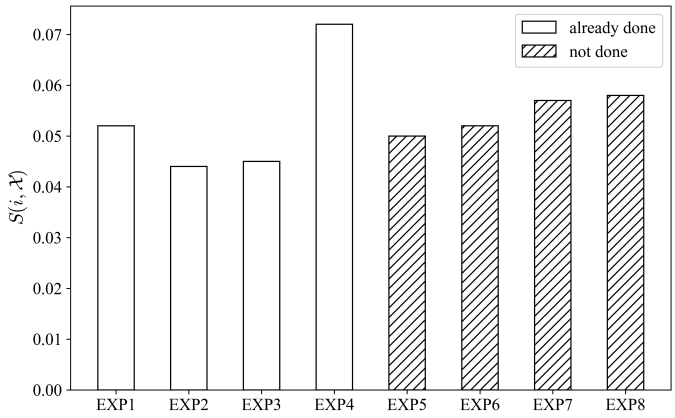

Figure 11: Normalized standard deviation of vectors $\vec{\psi}_{GP}^{(i)}(\vec{x}_j^*)$ on $\mathcal{X}$ for all 8 experiments. Experiments 1 – 4 are already done; experiments 5 – 8 are additionally designed experiments that are not done yet but to be analyzed first.

proportional to the duration of the experiment. That is why we need such a long time as 1160 ms to calculate a stress relaxation history using the constitutive model for EXP 4: for EXP 4, prediction for 1 million parameter vectors would take 1.6 min using SVD and Gaussian process metamodel while it would take up to 322 h to evaluate using the constitutive model. For more complicated constitutive models and longer strain histories, it is impractical to calculate the stress history for millions of parameter sets. *On the other hand, using our method, the computational time required to determine the stress history is independent of the duration of the experiment.* As shown in Table 2, the calculation time of our metamodel is practically identical for the cyclic and relaxation tests.

Table 2: Time to computer stress history per parameter vector

	Constitutive model (ms)	Metamodel (ms)	Constitutive/Meta
EXP 1	41.0	0.094	436
EXP 2	8.26	0.096	86
EXP 3	2.74	0.096	29
EXP 4	1160	0.098	11836

After obtaining 1 million sets of stress histories, the time needed to find the optimal parameters is a few seconds since the calculations are purely matrix multiplications. More importantly, if the parameters determined by the metamodel cannot fit the experiments well, one can confidently conclude that with very high probability there does not exist such a set of parameters that can fit the experimental data in Ω. Such conclusions can hardly be made if the fitting is done manually or using gradient-based optimization methods.

It is important to note that the time needed to run the metamodel does not include the time required to calculate $\boldsymbol{\theta}_k$ and $\left[[K_{XX}]_k + \delta^2[I]\right]^{-1}$ (Eqs. (16b) and (21)). The time needed to calculate

these quantities largely depends on the size of the training set, ω, and is independent of the number of $\vec{x}_j^*$ or η. It is known that a universal issue with GP models is that the basic computation complexity for the training process is $O(\omega^3)$ [21]. This complexity can be prohibitive for large data sets, and thus the size of the training set is usually less than 10000. The inherent unscalability of GPs will limit the size of Ω. There are two ways to tackle this problem. The first is to use theory or physical intuition to narrow Ω down to a domain of manageable size. For example, some of material parameters in the constitutive model may be directly measurable from experiments and this will reduce the dimension of the parameter space. The second is to use approximation methods to reduce the computation complexity. Recent developments have shown several approximation methods for GPs that have higher computational efficiency [32], and parallel computing can also increase the performance of Gaussian process dramatically [33].

7. Conclusion

An algorithm based on SVD and Gaussian process machine learning is used to build metamodels of constitutive models. SVD is used to compress the stress histories and extract the information in them. Gaussian process-based metamodels are used to efficiently and rapidly predict stress histories for a huge number of parameter vectors that are not in the training set. This enables us to select multiple sets of material parameters for a given constitutive model that fits all our experimental data. This is an advantage of our method over the traditional methods of constitutive model fitting. Besides, our approach can be automated and it is computationally efficient. Further, our approach allows exploration of the correlations between different material parameters in the constitutive model. Such information can provide useful guidance to the underlying micro-mechanics that governs the mechanical behavior of materials. This is also important for generative design of materials when the constitutive model accurately captures the physics.

Although we use the PVA gel as a demonstration of our algorithm, our method can be generalized to study other material systems. Indeed, the training set can be generated for any constitutive model. One limitation of our method is that the basic computational complexity of Gaussian process is the size of the training set ω to the third power, i.e., $O(\omega^3)$. When the size of the parameter space becomes larger, larger training set is required to effectively represent the parameter space, which will potentially lead to low computational efficiency. Fortunately, there are ways to bypass this limitation, such as using more efficient approximation algorithms of Gaussian Process or using active learning. Incorporating these advanced algorithms into the current method will be the focus of our future work.

Acknowledgement

The authors acknowledge the valuable advice from Prof. Austin R. Benson. This work is supported by the National Science Foundation, under Grant No. CMMI-1903308 and Grant No. CMMI-2038057.

References

[1] S. J. Buwalda, K. W. Boere, P. J. Dijkstra, J. Feijen, T. Vermonden, W. E. Hennink, Hydrogels in a historical perspective: From simple networks to smart materials, Journal of controlled release 190 (2014) 254–273.

[2] 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.

[3] 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).

[4] 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.

[5] X. Zhao, Multi-scale multi-mechanism design of tough hydrogels: building dissipation into stretchy networks, Soft matter 10 (5) (2014) 672–687.

[6] 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.

[7] 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.

[8] 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.

[9] S. Jung, J. Ghaboussi, Neural network constitutive model for rate-dependent materials, Computers & Structures 84 (15-16) (2006) 955–963.

[10] M. Cilla, I. Pérez-Rey, M. A. Martínez, E. Peña, J. Martínez, On the use of machine learning techniques for the mechanical characterization of soft biological tissues, International journal for numerical methods in biomedical engineering 34 (10) (2018) e3121.

[11] M. Guo, J. S. Hesthaven, Reduced order modeling for nonlinear structural analysis using gaussian process regression, Computer methods in applied mechanics and engineering 341 (2018) 807–826.

[12] M. Guo, J. S. Hesthaven, Data-driven reduced order modeling for time-dependent problems, Computer methods in applied mechanics and engineering 345 (2019) 75–99.

[13] 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.

[14] A. L. Frankel, R. E. Jones, L. P. Swiler, Tensor basis gaussian process models of hyperelastic materials, Journal of Machine Learning for Modeling and Computing 1 (01).

[15] 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.

[16] S. Zheng, Z. Liu, The machine learning embedded method of parameters determination in the constitutional models and potential applications for hydrogels, International Journal of Applied Mechanics.

[17] C. Zhai, T. Li, H. Shi, J. Yeo, Discovery and design of soft polymeric bio-inspired materials with multiscale simulations and artificial intelligence, Journal of Materials Chemistry B.

[18] 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.

[19] 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.

[20] 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.

[21] C. E. Rasmussen, C. K. I. Williams, Gaussian processes for machine learning, MIT Press, Cambridge, MA, 2006.

[22] E. Menga, M. J. Sánchez, I. Romero, Anisotropic meta-models for computationally expensive simulations in nonlinear mechanics, International Journal for Numerical Methods in Engineering 121 (5) (2020) 904–924.

[23] J. L. de Pablos, E. Menga, I. Romero, A methodology for the statistical calibration of complex constitutive material models: Application to temperature-dependent elasto-visco-plastic materials, Materials 13 (19) (2020) 4402.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[25] R. H. Byrd, P. Lu, J. Nocedal, C. Zhu, A limited memory algorithm for bound constrained optimization, SIAM Journal on scientific computing 16 (5) (1995) 1190–1208.

[26] C. Zhu, R. H. Byrd, P. Lu, J. Nocedal, Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization, ACM Transactions on Mathematical Software (TOMS) 23 (4) (1997) 550–560.

[27] R. Long, K. Mayumi, C. Creton, T. Narita, C.-Y. Hui, Rheology of a dual crosslink self-healing gel: Theory and measurement using parallel-plate torsional rheometry, Journal of Rheology 59 (3) (2015) 643–665.

[28] J. Guo, M. Liu, A. T. Zehnder, J. Zhao, T. Narita, C. Creton, C.-Y. Hui, Fracture mechanics of a self-healing hydrogel with covalent and physical crosslinks: A numerical study, Journal of the Mechanics and Physics of Solids 120 (2018) 79–95.

[29] M. D. McKay, R. J. Beckman, W. J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics 42 (1) (2000) 55–61.

[30] J. L. Deutsch, C. V. Deutsch, Latin hypercube sampling with multidimensional uniformity, Journal of Statistical Planning and Inference 142 (3) (2012) 763–772.

[31] W.-L. Loh, et al., On latin hypercube sampling, The annals of statistics 24 (5) (1996) 2058–2080.

[32] K. Dong, D. Eriksson, H. Nickisch, D. Bindel, A. G. Wilson, Scalable log determinants for gaussian process kernel learning, in: Advances in Neural Information Processing Systems, 2017, pp. 6327–6337.

[33] K. Wang, G. Pleiss, J. Gardner, S. Tyree, K. Q. Weinberger, A. G. Wilson, Exact gaussian processes on a million data points, in: Advances in Neural Information Processing Systems, 2019, pp. 14648–14659.