J. Mech. Phys. Solids 203 (2025) 106212

Contents lists available at ScienceDirect

Journal of the Mechanics and Physics of Solids

journal homepage: www.elsevier.com/locate/jmps

Constitutive Kolmogorov–Arnold Networks (CKANs): Combining accuracy and interpretability in data-driven material modeling

Kian P. Abdolazizi [a], Roland C. Aydin [a,b], Christian J. Cyron [a,b,*], Kevin Linka [a,*]

[a] Institute for Continuum and Material Mechanics, Hamburg University of Technology, Eißendorfer Straße 42, 21073 Hamburg, Germany
[b] Institute of Material Systems Modeling, Helmholtz-Zentrum Hereon, Max–Planck-Straße 1, 21502 Geesthacht, Germany

ARTICLE INFO

Keywords:
Kolmogorov–Arnold Networks
Data-driven mechanics
Physics-informed machine learning
Constitutive Artificial Neural Networks
Soft materials
Symbolic regression
Interpretable machine learning

ABSTRACT

Hybrid constitutive modeling integrates two complementary approaches for describing and predicting a material's mechanical behavior: purely data-driven black-box methods and physically constrained, theory-based models. While black-box methods offer high accuracy, they often lack interpretability and extrapolability. Conversely, physics-based models provide theoretical insight and generalizability but may not capture complex behaviors with the same accuracy. Traditionally, hybrid modeling has required a trade-off between these aspects. In this paper, we show how recent advances in symbolic machine learning — specifically Kolmogorov–Arnold Networks (KANs) — help to overcome this limitation. We introduce Constitutive Kolmogorov–Arnold Networks (CKANs) as a new class of hybrid constitutive models. By incorporating a post-processing symbolification step, CKANs combine the predictive accuracy of data-driven models with the interpretability and extrapolation capabilities of symbolic expressions, bridging the gap between machine learning and physical modeling.

1. Introduction

To describe the mechanical behavior of a material — specifically, the relationship between mechanical stress and strain — a constitutive model is required. Developing constitutive models that are accurate, predictive, and interpretable remains a persistent challenge. Historically, these models have primarily consisted of manually derived symbolic expressions that define functions or functionals mapping stress to strain, or vice versa (Treloar, 1944a; Rivlin and Saunders, 1951; Ogden, 1972; Fung, 1981; Holzapfel et al., 2000; Ehret and Itskov, 2007). However, this approach has three significant shortcomings. First, manually deriving symbolic expressions is both complex and time-consuming. Second, the limited complexity of manually tractable expressions often fails to capture the intricate behavior of materials. Third, while these symbolic expressions can describe known materials, they generally struggle to predict the behavior of new materials.

To overcome these shortcomings of traditional constitutive modeling, data-driven methods have attracted increasing attention over the last decade. The distance-minimizing method (Kirchdoerfer and Ortiz, 2016; Carrara et al., 2020), black-box neural networks (Ghaboussi et al., 1991; Hashash et al., 2004) or methods based on spline interpolations (Sussman and Bathe, 2009; Latorre and Montáns, 2013, 2014; Crespo et al., 2017; Dal et al., 2023; Wiesheier et al., 2024) were suggested. They overcome the need for manual derivation of symbolic equations. However, such purely data-driven approaches have great difficulties in extrapolating beyond known material data or predicting the behavior of new materials. Moreover, they are typically not directly interpretable.

* Corresponding authors.
E-mail addresses: christian.cyron@hereon.de (C.J. Cyron), kevin.linka@tuhh.de (K. Linka).

https://doi.org/10.1016/j.jmps.2025.106212
Received 27 February 2025; Received in revised form 7 May 2025; Accepted 21 May 2025
Available online 11 June 2025

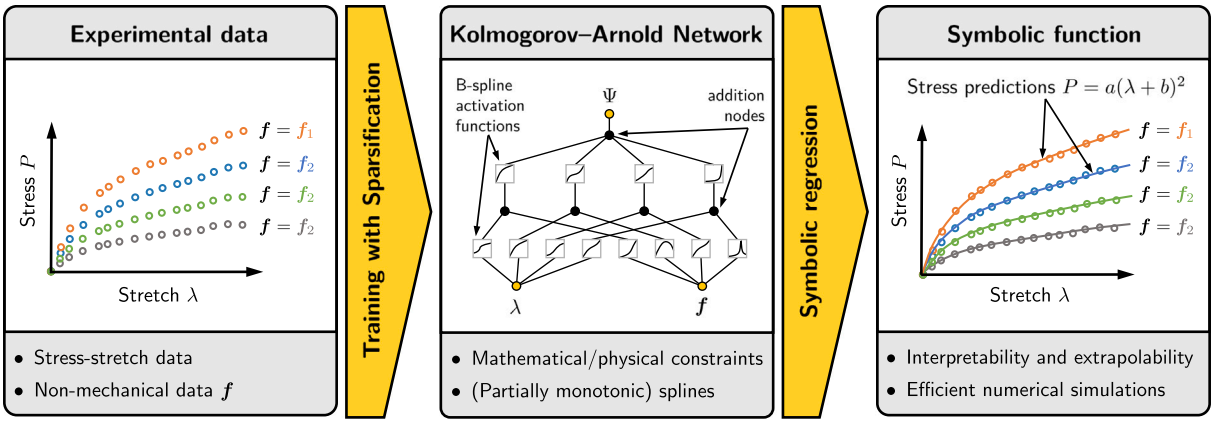


Fig. 1. Constitutive Kolmogorov-Arnold Networks (CKANs). Given experimental stretch λ and stress P data and potentially also non-mechanical data f (such as microstructural imaging data or information on materials processing), a CKAN learns the strain energy function Ψ of a hyperelastic material. It inherently satisfies essential mathematical or physical requirements, such as objectivity, symmetry properties, or thermodynamic consistency. Its activation functions are univariate (partially monotonic) splines adapting during training. Combined with sparsification techniques, this leads to an accurate and efficient model. Finally, the activation functions are symbolized, yielding an interpretable and compact symbolic expression for the strain energy function Ψ.

This lack of interpretability limits the applicability in particular in safety-critical areas where it is crucial to understand the precise nature of an applied model and to prove also certain bounds within which its output will remain under certain conditions to prevent situations where a model failure could cause major damage (Goldberg et al., 2024).

Therefore, hybrid constitutive models were recently introduced to bridge the gap between theory-based symbolic models and data-driven approaches (Fuhg et al., 2024a). These models often embed domain knowledge into machine learning (ML) frameworks using physics-based loss functions or hard constraints. Examples include Physics-Augmented Neural Networks (PANNs), which integrate hyperelastic constraints into neural architectures (Linden et al., 2023), neural ordinary differential equations that enforce physical plausibility (Taç et al., 2023), and models explicitly designed to ensure polyconvexity (As'ad et al., 2022). Constitutive Artificial Neural Networks (CANNs) further expand the scope of data-driven constitutive modeling by incorporating mechanical and non-mechanical inputs (Linka et al., 2021; Abdolazizi et al., 2023). Hybrid constitutive models offer several key advantages. First, like all data-driven models, they eliminate the need for manual model derivation. Second, their flexible architectures allow them to accurately capture highly complex material behavior, given sufficient data is available. Third, integrating theoretical principles improves extrapolation capabilities significantly compared to classical black-box models. Fourth, certain hybrid models, such as CANNs (Linka et al., 2022), can leverage non-mechanical data — such as microstructural imaging or processing parameters — to predict the behavior of novel materials. Despite these strengths, hybrid constitutive models still face a critical limitation. Although embedding physical knowledge makes them more transparent than purely data-driven black-box models, they often retain large, uninterpretable parts, such as deep neural networks. This residual opacity remains a significant drawback, especially in high-stakes fields such as biomedical and aerospace engineering, where reliability and traceability are crucial (Goldberg et al., 2024).

To address this challenge, a complementary trend has focused on interpretable data-driven modeling, aiming to derive constitutive laws directly from experimental data. One approach combines large libraries of physics-inspired building blocks with sparse regression to identify relevant terms efficiently (Flaschel et al., 2021). Furthermore, advances in sparsification and symbolic regression have been crucial for developing interpretable and efficient models. Recent contributions have established comprehensive frameworks for sparsification in data-driven mechanics (Brunton et al., 2016; Fuhg et al., 2024c). Another strategy employs specialized neural networks that embed fundamental constitutive structures (Linka and Kuhl, 2023; Holthusen et al., 2024; Pierre et al., 2023; McCulloch et al., 2024; Holthusen et al., 2025), enabling the automatic selection of dominant terms. Additionally, symbolic regression, particularly through genetic programming, has gained traction for generating concise, human-readable equations (Wang et al., 2019; Abdusalamov et al., 2023). While these methods enhance model interpretability, they come with drawbacks. Limited expressiveness can hinder their ability to capture complex material behavior, while high computational costs and sensitivity to data quality pose significant challenges.

This paper aims to merge the strengths of hybrid constitutive models, such as Constitutive Artificial Neural Networks (CANNs), with the benefits of interpretable symbolic data-driven methods. To achieve this, we employ Kolmogorov–Arnold Networks (KANs), which bridge the gap between purely data-driven approaches and symbolic interpretation by enabling the extraction of interpretable features, modular structures, and closed-form expressions (Liu et al., 2024b). KANs leverage the Kolmogorov–Arnold representation theorem, which states that any continuous multivariate function can be represented as a composition of univariate functions and addition. This property makes KANs universal approximators with inherently interpretable architectures. Recent applications of KANs include physics-informed learning of partial differential equations (Wang et al., 2024; Shukla et al., 2024; Howard et al., 2024) and surrogate modeling in multiscale simulations (Le Clézio et al., 2024). Because KANs provide functionally simple yet expressive representations of complex nonlinear relationships (Liu et al., 2024a), they are particularly well-suited for discovering

constitutive laws. Building on this foundation, we introduce a novel framework that merges the symbolic capabilities of KANs with the flexibility of CANNs, addressing the longstanding trade-off between interpretability and performance.

In this paper, we introduce Constitutive Kolmogorov–Arnold Networks (CKANs), a novel approach to data-driven constitutive modeling that leverages Kolmogorov–Arnold Networks (KANs) within a hybrid modeling framework. By systematically incorporating physically inspired constraints and pruning strategies, our method attains high predictive accuracy while generating transparent, symbolic constitutive laws. The overall workflow of the CKAN framework is illustrated in Fig. 1. This paper is organized as follows: Section 2 introduces the necessary background from continuum mechanics. In Section 3, we present the theoretical foundations of the CKAN framework, while Section 4 details the sparsification and symbolification process necessary for deriving compact and interpretable symbolic expressions. We then demonstrate the effectiveness of our methodology through various numerical experiments and real-world data in Section 5. Finally, Section 6 summarizes the main findings and suggests possible future directions. To reduce the complexity of the discussion herein, we focus in this paper on isotropic hyperelastic materials. However, the ideas developed herein can easily be transferred to much broader classes of materials.

2. Continuum mechanics

2.1. Kinematics

Herein, we rely on the general framework of nonlinear continuum mechanics as outlined, for example, in Holzapfel (2000). We consider a continuum body composed of material points $\mathbf{X}$ in the undeformed reference configuration. During a motion, the material points are mapped at time t to the current position $\mathbf{x}(\mathbf{X}, t)$ in the deformed current configuration. To quantify the deformation of the material in the body, one often uses the so-called deformation gradient $\mathbf{F}$ and its spectral decomposition:

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \sum_{\alpha=1}^{3} \lambda_\alpha \, \boldsymbol{n}_\alpha \otimes \boldsymbol{N}_\alpha. \tag{1}$$

Here $\lambda_\alpha > 0$ are the principal stretches of the material, and $\boldsymbol{N}_\alpha$ and $\boldsymbol{n}_\alpha$ are the corresponding principal stretch directions in the reference and current configuration. The larger λ_α, the more the material is stretched in the corresponding principal stretch direction. The Jacobian (determinant) $J = \det(\mathbf{F}) = \lambda_1 \lambda_2 \lambda_3$ characterizes local volume changes. For incompressible materials $J = 1$. Based on the deformation gradient, one can define the so-called right Cauchy–Green deformation tensor $\mathbf{C}$ with its cofactor $\operatorname{cof}(\mathbf{C})$:

$$\mathbf{C} = \mathbf{F}^{\mathrm{T}}\mathbf{F} = \sum_{\alpha=1}^{3} \lambda_\alpha^2 \, \boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha, \qquad \operatorname{cof}(\mathbf{C}) = \det(\mathbf{C})\mathbf{C}^{-\mathrm{T}} = \sum_{\alpha=1}^{3} \nu_\alpha^2 \, \boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha, \tag{2}$$

For an undeformed body, the deformation gradient and right Cauchy–Green deformation tensor are equal to the identity tensor $\mathbf{I}$; $\nu_\alpha = J/\lambda_\alpha$ is the local stretch of the area orthogonal to the α-direction and can be computed as

$$\nu_1 = \lambda_2 \lambda_3, \quad \nu_2 = \lambda_1 \lambda_3, \quad \nu_3 = \lambda_1 \lambda_2. \tag{3}$$

The principal invariants of $\mathbf{C}$, in terms of the principal stretches, are

$$I_1 = \operatorname{tr}(\mathbf{C}) = \lambda_1^2 + \lambda_2^2 + \lambda_3^2, \qquad I_2 = \operatorname{tr}(\operatorname{cof} \mathbf{C}) = \nu_1^2 + \nu_2^2 + \nu_3^2 = \lambda_2^2\lambda_3^2 + \lambda_1^2\lambda_3^2 + \lambda_1^2\lambda_2^2, \qquad I_3 = \det \mathbf{C} = J^2 = \lambda_1^2\lambda_2^2\lambda_3^2. \tag{4}$$

To describe the behavior of rubber-like materials, it was found that using other sets of kinematic variables can help to bridge the gap between the kinematics on the micro- and macro-scale. For instance, the modified invariants

$$\imath_1 = \sqrt{\frac{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}{3}} = \sqrt{\frac{I_1}{3}}, \qquad \imath_2 = \sqrt[3]{\frac{\nu_1^2 + \nu_2^2 + \nu_3^2}{3}} = \sqrt[3]{\frac{I_2}{3}}, \tag{5}$$

can be interpreted as representing the rubber network's average stretch and tube contraction, respectively (Dal et al., 2023; Kearsley, 1989; Arruda and Boyce, 1993).

Each of the three sets of kinematic variables introduced above, i.e., $\{\lambda_1, \lambda_2, \lambda_3\}$, $\{I_1, I_2, I_3\}$, and $\{\imath_1, \imath_2, I_3\}$, provides a sufficient functional basis to completely characterize the strain energy function of an isotropic hyperelastic material. However, depending on which functional basis one uses, the functions required to describe a specific material behavior may take on a relatively complex or rather simple form. This is illustrated in Fig. 2 by considering three specific deformation modes of a thin incompressible material sample, namely, uniaxial tension, equibiaxial tension, and pure shear (see Fig. A.14 for details). Using $\{\lambda_1, \lambda_2, \lambda_3\}$ as a functional basis, equibiaxial tension and pure shear can be expressed as linear functions, whereas uniaxial tension and simple shear require more complex functional forms. By contrast, using $\{I_1, I_2, I_3\}$, pure and simple shear can be represented by linear functions, whereas uniaxial and equibiaxial tension require more complex functional forms. Note that various tensor basis formulations have been explored to enhance neural network representations in isotropic hyperelasticity. For further details, we refer the reader to Fuhg et al. (2024b).

This article is devoted to the automated derivation of simple and yet accurate symbolic expressions to characterize the mechanical behavior of materials. Apparently, it is worth examining how the results of the framework developed herein depend on the functional basis chosen ($\{\lambda_1, \lambda_2, \lambda_3\}$, $\{I_1, I_2, I_3\}$ or $\{\imath_1, \imath_2, I_3\}$). This comparison thus forms an integral part of the discussion below.

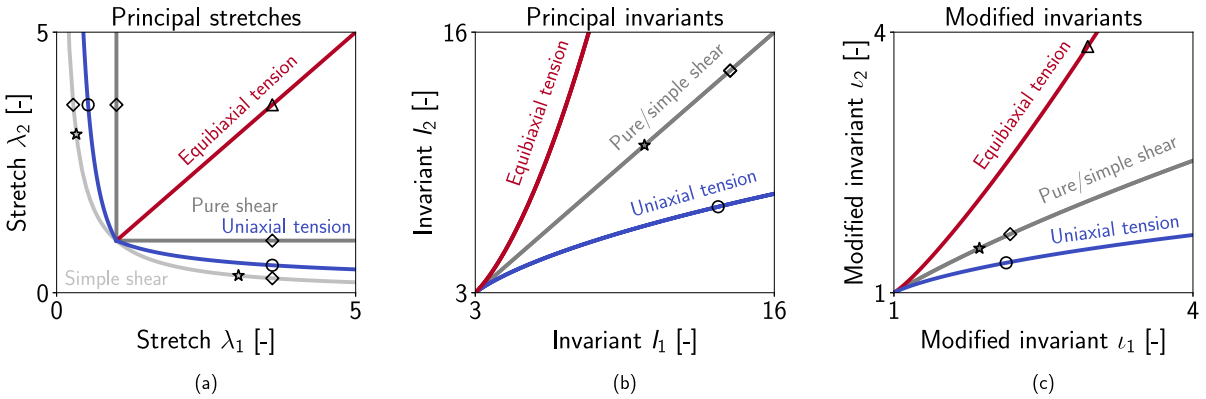


Fig. 2. Representation of special planar modes of deformation of a thin incompressible membrane using three different functional bases: the complexity of the functional forms required to represent uniaxial tension, equibiaxial tension, and pure or simple shear vastly differs depending on whether one relies on representing them in terms of (a) principal stretches $\{\lambda_1, \lambda_2\}$, (b) principal invariants $\{I_1, I_2\}$, (c) modified invariants $\{\iota_1, \iota_2\}$. The markers indicate identical deformation states. Notably, not all deformation states shown in sub figure (a) fall within the range depicted in the sub figure (b), resulting in fewer visible markers in (b).

2.2. Isotropic hyperelasticity

The primary objective of this work is to illustrate how Kolmogorov–Arnold Networks can be applied to hybrid constitutive modeling of materials. For clarity, we focus on isotropic hyperelastic materials, exhibiting the same properties in all directions.

Strain energy function and stress. Hyperelasticity postulates the existence of a strain energy function $\Psi = \Psi(\mathbf{F})$. The constitutive equation for the first Piola–Kirchhoff stress $\mathbf{P}$ of a hyperelastic material directly follows from the second law of thermodynamics generally as (Truesdell, 1984)

$$\mathbf{P} = \frac{\partial \Psi(\mathbf{F})}{\partial \mathbf{F}}. \tag{6}$$

Objectivity of the strain energy function is inherently ensured by using the reduced formulation (Ogden, 1997)

$$\Psi = \Psi(\mathbf{F}) = \Psi(\mathbf{C}), \tag{7}$$

which is based on $\mathbf{C}$ and can thus be shown to be observer-independent (objective). The (symmetric) so-called second Piola–Kirchhoff stress tensors $\mathbf{S}$ is related to the first Piola–Kirchhoff stress tensor $\mathbf{P}$ and the strain energy through (Holzapfel, 2000):

$$\mathbf{P} = \mathbf{F}\mathbf{S}, \qquad \mathbf{S} = 2\frac{\partial \Psi(\mathbf{C})}{\partial \mathbf{C}}. \tag{8}$$

Functional bases. The strain energy function of an isotropic hyperelastic material must be a scalar-valued isotropic tensor function of $\mathbf{C}$ (Truesdell, 1984). The principal invariants I_i, the principal stretches λ_i, and the modified invariants ι_i form so-called functional bases, in terms of which all scalar-valued isotropic tensor functions of $\mathbf{C}$ can be expressed (Truesdell, 1984). Thus, the strain energy function can be written as:

$$\Psi = \Psi(I_1, I_2, I_3) = \Psi(\lambda_1, \lambda_2, \lambda_3) = \Psi(\iota_1, \iota_2, I_3). \tag{9}$$

The appropriate functional form of Ψ for a specific material generally depends on material features, such as the composition, microstructure, or processing parameters. This information may be expressed in terms of some feature vector $\boldsymbol{f}$, leading to

$$\Psi = \Psi\left(I_1, I_2, I_3, \boldsymbol{f}\right) = \Psi\left(\lambda_1, \lambda_2, \lambda_3, \boldsymbol{f}\right) = \Psi\left(\iota_1, \iota_2, I_3, \boldsymbol{f}\right). \tag{10}$$

Incompressibility. Many important materials can be regarded as incompressible so that $J = 1$. In this case, one has only two independent deformation variables, that is, ($\{\lambda_1, \lambda_2\}$, $\{I_1, I_2\}$ or $\{\iota_1, \iota_2\}$), while the respective third one is determined via the incompressibility constraint $J = I_3 = 1$ and thus $\lambda_3 = (\lambda_1\lambda_2)^{-1}$. For incompressible isotropic hyperelastic materials, the strain energy functions can be postulated as

$$\Psi = \hat{\Psi}(I_1, I_2, \boldsymbol{f}) - \frac{p}{2}(I_3 - 1) = \hat{\Psi}(\lambda_1, \lambda_2, \lambda_3, \boldsymbol{f}) - p(J - 1) = \hat{\Psi}(\iota_1, \iota_2, \boldsymbol{f}) - \frac{p}{2}(I_3 - 1), \tag{11}$$

where p denotes the hydrostatic pressure in the material determined from the boundary conditions. For the remainder of this work, the overset hat $\hat{(\cdot)}$ explicitly denotes strain energy functions for the incompressible case. In Eq. (11), we use the same symbol p for the Lagrange multiplier ensuring incompressibility for all three different functional bases. However, the value that results for p differs for these three cases, even if the same boundary value problem is considered, see Appendix A.2.

Separable principal stretch-based strain energy functions. Since the properties of isotropic materials do not depend on direction, principal stretch-based strain energy functions $\Psi(\lambda_1, \lambda_2, \lambda_3, \boldsymbol{f})$ must be symmetric with respect to the principal stretches. However, the space of all symmetric functions is vast, making it challenging to design suitable candidate functions. This complexity is significantly reduced by adopting a separable form, which is symmetric by definition and has proven effective in practice (Xu et al., 2015):

$$\Psi = \Psi(\lambda_1, \lambda_2, \lambda_3, \boldsymbol{f}) = \sum_{\alpha=1}^{3} \omega_1(\lambda_\alpha, \boldsymbol{f}) + \sum_{\alpha=1}^{3} \omega_{-1}(\nu_\alpha, \boldsymbol{f}) + \Omega(\lambda_1\lambda_2\lambda_3, \boldsymbol{f}), \tag{12}$$

where ω_1, ω_{-1}, and Ω are univariate strain energy functions related to the changes in line, area, and volume elements, respectively. For incompressible materials, this simplifies further to:

$$\Psi = \hat{\Psi}(\lambda_1, \lambda_2, \lambda_3, \boldsymbol{f}) - p(J-1) = \sum_{\alpha=1}^{3} \hat{\omega}_1(\lambda_\alpha, \boldsymbol{f}) + \sum_{\alpha=1}^{3} \hat{\omega}_{-1}(\lambda_\alpha^{-1}, \boldsymbol{f}) - p(J-1). \tag{13}$$

Often, even $\hat{\omega}_{-1}$ is discarded, reducing the strain energy function to a sum of a single scalar function $\hat{\omega}_1$ evaluated at the three principal stretches λ_α. This additive separability of the strain energy function is known as the Valanis–Landel hypothesis (Valanis and Landel, 1967), and has been widely adopted (Ogden, 1972; Heinrich and Kaliske, 1997; Shariff, 2000; Attard and Hunt, 2004).

While the CKAN framework also supports non-separable strain energy functions, these require explicit symmetrization by averaging the strain energy function over all permutations of the principal stretches. Since this increases computational cost and typically compromises interpretability, the non-separable approach is not considered here.

Mixed invariant- and principal stretch-based strain energy functions. In view of (4), the principal invariants can be expressed as quadratic functions of the principal stretches, while the squared principal stretches are the roots of the characteristic polynomial of $\mathbf{C}$, which is expressed in terms of the principal invariants (Itskov, 2015). Consequently, any strain energy function based on principal invariants can, in principle, be equivalently expressed in terms of principal stretches, and vice versa (Ehret and Stracuzzi, 2022). One may thus represent the strain energy in terms of $\{\lambda_1, \lambda_2, \lambda_3\}$, $\{I_1, I_2, I_3\}$, or $\{\iota_1, \iota_2, I_3\}$. However, these reformulations often result in more complex expressions and may introduce numerical instabilities (Xu et al., 2015). To mitigate these issues, it can be helpful to adopt a fourth, mixed functional basis $\{\lambda_1, \lambda_2, \lambda_3, I_1, I_2, I_3\}$. Using this basis, one often assumes an additive decomposition of the strain energy function (Kaliske and Heinrich, 1999; Xiang et al., 2018; Davidson and Goulbourne, 2013):

$$\Psi(I_1, I_2, I_3, \lambda_1, \lambda_2, \lambda_3, \boldsymbol{f}) = \Psi_I(I_1, I_2, I_3, \boldsymbol{f}) + \Psi_\lambda(\lambda_1, \lambda_2, \lambda_3, \boldsymbol{f}). \tag{14}$$

While the first three functional bases we introduced are irreducible, the mixed fourth basis introduced in the context of (14) is reducible (Itskov, 2015), meaning it contains more elements than strictly necessary to span the space of interest.

3. Constitutive-Kolmogorov–Arnold Networks (CKANs)

This section summarizes the fundamentals of Kolmogorov–Arnold Networks (KANs) and introduces the Constitutive Kolmogorov–Arnold Network (CKAN) framework for interpretable and predictive material modeling. Standard types of neural networks, like feedforward neural networks (FFNNs), recurrent neural networks (RNNs), convolutional neural networks (CNNs), and others, are powerful approximators. However, their black-box nature limits interpretability and complicates adherence to physical constraints. KANs address these challenges by leveraging the Kolmogorov–Arnold representation theorem and representing the strain energy function using B-spline curves with learnable control points of local B-spline basis functions. This low-dimensional parameterization enables efficient training and direct incorporation of physical prior knowledge (Liu et al., 2024a,b).

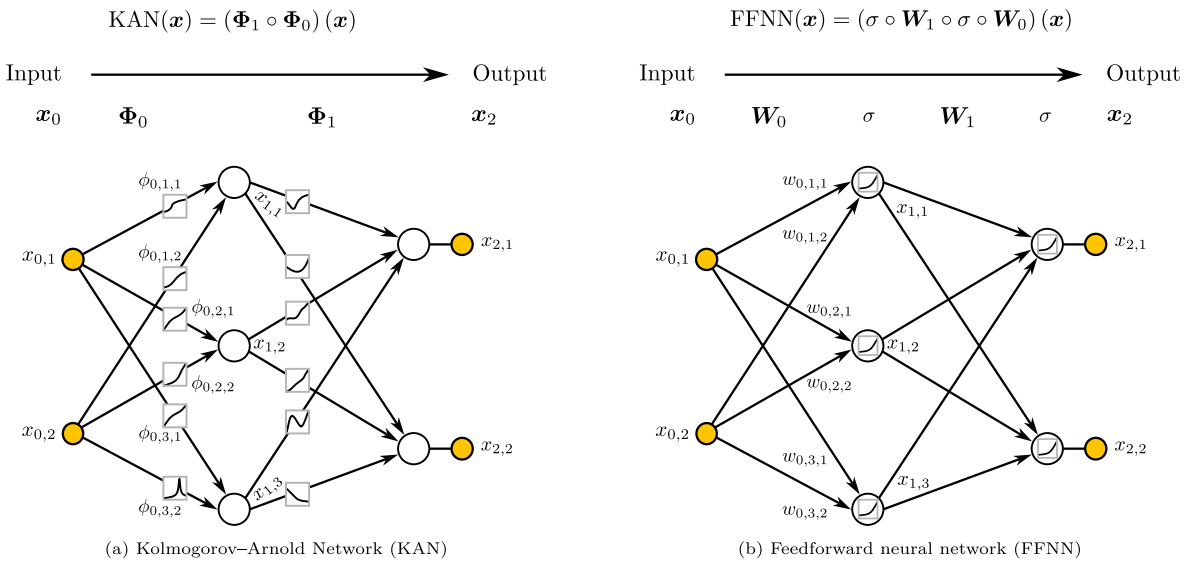
3.1. Kolmogorov–Arnold Networks (KANs)

Kolmogorov–Arnold Networks (KANs) (Liu et al., 2024a,b) draw inspiration from the Kolmogorov–Arnold representation theorem (KART), which states that any continuous multivariate function $f : [0,1]^n \to \mathbb{R}$ can be represented as a finite composition of univariate continuous functions and addition operations:

$$f(\boldsymbol{x}_0) = f(x_{0,1}, x_{0,2}, \dots, x_{0,n}) = \sum_{j=1}^{2n+1} \phi_{1,1,j} \left(\sum_{i=1}^{n} \phi_{0,j,i}(x_{0,i}) \right), \tag{15}$$

where the $x_{0,1}$ are the elements of the vector $\boldsymbol{x}_0$, and the $\phi_{0,j,i} : [0,1] \to \mathbb{R}$ and $\phi_{1,1,j} : \mathbb{R} \to \mathbb{R}$ are univariate continuous functions. Eq. (15) can be understood as a neural network with n input nodes, a middle layer with $2n+1$ nodes, and a single output node. The functions $\phi_{0,j,i}$ connect the input nodes to the nodes in the middle layer, and the functions $\phi_{1,j,i}$ connect the nodes in the middle layer to the output nodes, with a summation performed at each node over all incoming inputs. Interpreting Eq. (15) as a neural network directly leads to the concept of KANs. Generally, a KAN may consist of L layers, each containing n_l nodes with $l = 0, \dots, L-1$. Its topology can thus be characterized by the vector

$$\mathcal{N} = [n_0, n_1, \dots, n_{L-1}]. \tag{16}$$



(a) Kolmogorov–Arnold Network (KAN) (b) Feedforward neural network (FFNN)

Fig. 3. Comparison of a Kolmogorov–Arnold Network (KAN) and a feedforward neural network (FFNN). A KAN (a) can be interpreted as an FFNN (b) where the weights are no longer constants but nonlinear functions, typically spline functions of higher order, and where, as a consequence, the intra-neural (typically nonlinear) activation function σ is no longer needed. In an FFNN, the weights $w_{l,j,i}$ are adjusted during training to learn a specific input–output relation. In a KAN, however, it is the functions $\phi_{l,j,i}$, not the weights, that are adjusted. Orange nodes represent input and output nodes.

The output of layer l is denoted by the vector $\boldsymbol{x}_l$. Specifically, the element $x_{l,i}$ is the output of node i in layer l and, after passing through the function $\phi_{l,j,i}$, becomes an input for node j in layer $l+1$. The output of node j in layer $l+1$ is the sum of all its inputs from layer l, that is:

$$x_{l+1,j} = \sum_{i=1}^{n_l} \phi_{l,j,i}(x_{l,i}), \qquad j = 1, 2, \dots, n_{l+1}. \tag{17}$$

In the context of KANs, the functions $\phi_{l,j,i}$ are usually referred to as activation functions. For the ease of notation, one can arrange them in a matrix $\boldsymbol{\Phi}_l$ so that

$$\boldsymbol{x}_{l+1} = \underbrace{\begin{bmatrix} \phi_{l,1,1}(\cdot) & \phi_{l,1,2}(\cdot) & \dots & \phi_{l,1,n_l}(\cdot) \\ \phi_{l,2,1}(\cdot) & \phi_{l,2,2}(\cdot) & \dots & \phi_{l,2,n_l}(\cdot) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{l,n_{l+1},1}(\cdot) & \phi_{l,n_{l+1},2}(\cdot) & \dots & \phi_{l,n_{l+1},n_l}(\cdot) \end{bmatrix}}_{\boldsymbol{\Phi}_l :=} \boldsymbol{x}_l, \tag{18}$$

Generally, a KAN with L layers then maps its input $\boldsymbol{x}_0$ to its output $\boldsymbol{x}_L$ via

$$\boldsymbol{x}_L = \mathrm{KAN}(\boldsymbol{x}_0) = (\boldsymbol{\Phi}_{L-1} \circ \boldsymbol{\Phi}_{L-2} \circ \cdots \circ \boldsymbol{\Phi}_1 \circ \boldsymbol{\Phi}_0)(\boldsymbol{x}_0). \tag{19}$$

Fig. 3 compares the architecture of a KAN with that of a feedforward neural network (FFNN) with the same topology. They both consist of layers of neurons, where each neuron in one layer generally provides input to each neuron in the subsequent layer. However, while in the FFNN this input is simply multiplied by a scalar weight $w_{l,j,i}$, in the KAN, it passes through a function $\phi_{l,j,i}$. This gives KANs the ability to approximate nonlinear relationships, eliminating the need for the intra-neural activation function σ required in FFNNs. The FFNN and KAN share a similar training process in which the $w_{l,j,i}$ and $\phi_{l,j,i}$, respectively, are adjusted so that, after training, the FFNN and KAN can capture a certain input–output relation. A scalar weight $w_{l,j,i}$ can be interpreted as a constant function, whereas in KANs the $\phi_{l,j,i}$ are usually chosen as spline functions of higher order. In this sense, a KAN can be interpreted as an FFNN whose weights are no longer constant scalars (spline functions of order zero) but spline functions of higher order. This added flexibility allows KANs to directly approximate nonlinear relationships between layers, rendering the intra-neural nonlinear activation functions σ unnecessary and effectively replacing them with the identity function. Formally, the input–output relation of a KAN is given by Eq. (19), while for an FFNN with L layers, it is described by the following equation:

$$\boldsymbol{x}_L = \mathrm{FFNN}(\boldsymbol{x}_0) = (\sigma \circ \boldsymbol{W}_{L-1} \circ \sigma \circ \boldsymbol{W}_{L-2} \circ \cdots \circ \sigma \circ \boldsymbol{W}_1 \circ \sigma \circ \boldsymbol{W}_0)(\boldsymbol{x}_0), \tag{20}$$

where the matrices $\boldsymbol{W}_l$ assemble the scalar weights $w_{l,j,i}$, analogous to how the matrices $\boldsymbol{\Phi}_l$ in KANs assemble the functions $\phi_{l,j,i}$.

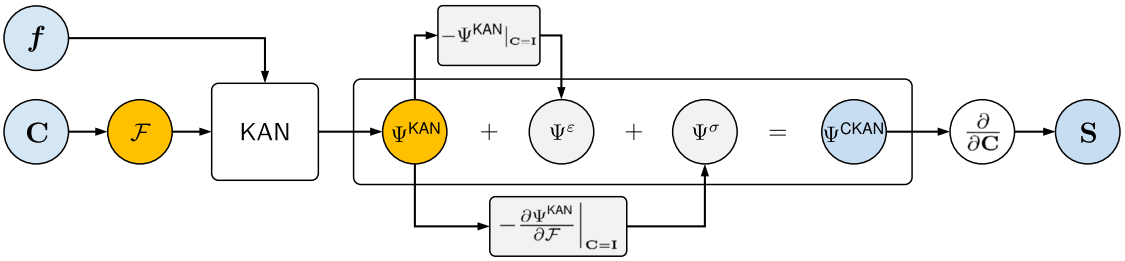


Fig. 4. Schematic overview of the Constitutive Kolmogorov–Arnold Network (CKAN) framework: The right Cauchy Green deformation tensor $\mathbf{C}$ and the feature vector $\boldsymbol{f}$ are input to the CKAN. Depending on the chosen functional basis $\mathcal{F}$ (principal stretches, principal invariants, modified invariants, mixed basis) and the feature vector, the strain energy function Ψ^{KAN} is represented by a KAN. For example, in a principal invariant-based CKAN, one of the KANs from Fig. 5 can be substituted in the box KAN. The correction terms Ψ^{σ} and Ψ^{ϵ} ensure the normalization of the stress and strain energy function in the undeformed configuration ($\mathbf{C}=\mathbf{I}$), yielding the total strain energy function Ψ^{CKAN}. The correction terms depend on Ψ^{KAN} and its derivatives with respect to the elements of the functional basis $\partial\Psi^{\mathrm{KAN}}/\partial\mathcal{F}$, evaluated in the undeformed configuration. They iteratively adapt during training. Finally, the stress $\mathbf{S}$ is obtained as the gradient of Ψ^{CKAN} with respect to $\mathbf{C}$.

3.2. Constitutive modeling with KANs

In this work, we aim to learn the strain energy function of an isotropic hyperelastic material from experimental data. To do so, we have developed Constitutive Kolmogorov–Arnold Networks (CKANs), a machine learning architecture designed to model the constitutive behavior of hyperelastic materials. Fig. 4 provides a schematic overview of the CKAN architecture.

The first design choice in constructing CKANS is selecting an appropriate functional basis $\mathcal{F}$ to describe the deformation state as input. In this paper, we compare the four functional bases introduced in Section 2: $\{\lambda_1,\lambda_2,\lambda_3\}$, $\{I_1,I_2,I_3\}$, $\{\iota_1,\iota_2,I_3\}$, and $\{\lambda_1,\lambda_2,\lambda_3,I_1,I_2,I_3\}$. Additionally, we include any supplementary material features, denoted by vector $\boldsymbol{f}$ into the input space. For instance, $\boldsymbol{f}$ may contain information about the material's processing conditions or microstructure, or about certain properties that are not measured in classical strain-stress tests, such as the material's hardness. The CKAN's output should be an approximation of the material's stress response derived from the strain energy function Ψ^{CKAN}, which fully characterizes the material's nonlinear hyperelasticity. This strain energy function is defined by:

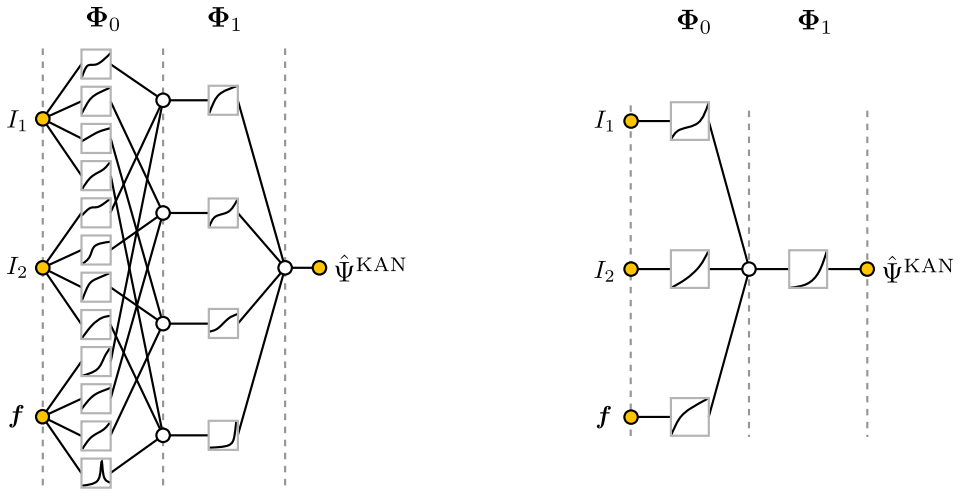
$$\Psi^{\mathrm{CKAN}}(\mathcal{F},\boldsymbol{f}) := \Psi^{\mathrm{KAN}}(\mathcal{F},\boldsymbol{f}) + \Psi^{\sigma}(J,\boldsymbol{f}) + \Psi^{\epsilon}(\boldsymbol{f}). \tag{21}$$

The strain energy contribution Ψ^{KAN} is represented by a KAN and captures the material's basic constitutive behavior. To ensure that both stress and strain energy vanish in the undeformed reference configuration, the correction terms Ψ^{σ} and Ψ^{ϵ} are introduced. These terms are not modeled by a KAN; rather, they depend on Ψ^{KAN} and its derivatives with respect to the elements of the functional basis, evaluated in the undeformed state, see Fig. 4. Details on each contribution to the total strain energy function Ψ^{CKAN} are provided below.

The CKAN framework with its resulting total strain energy function Ψ^{CKAN} inherently ensures thermodynamic consistency, objectivity, material symmetry, symmetry of the second Piola–Kirchhoff stress tensor, as well as zero stress and zero strain energy in the undeformed state. Although beyond the scope of this work, CKANs can easily be extended to anisotropic materials by introducing strain invariants formulated in terms of structure tensors (Kalina et al., 2025; Linka et al., 2021; Boehler, 1987; Zhang and Rychlewski, 1990).

Training of CKANs. As mentioned above, only the contribution Ψ^{KAN} is represented by a KAN. Consequently, the only trainable parameters of the total strain energy function Ψ^{CKAN} are the control points of the univariate spline activation functions associated with Ψ^{KAN}. Thus, training the CKAN is reduced to optimizing these control points. In practice, direct measurements of the strain energy are usually unavailable. Instead, only stress data corresponding to specific tested strain states are accessible. Consequently, the loss function for training the CKAN cannot be based on a direct comparison between Ψ^{CKAN} and experimentally measured strain energies. Instead, it must rely on comparing their derivatives—i.e., the stresses. This requires differentiating Ψ^{CKAN}, yielding the stress tensors via (8), see Fig. 4. Detailed expressions for these stress tensors, corresponding to the different functional bases, are provided in Appendix A.1. Since all mathematical operations in a CKAN are fully differentiable, the equations from Appendix A.1 can be applied directly using automatic differentiation. The computed stresses are then compared with the stress values obtained experimentally. Training is performed by minimizing the loss function $\mathcal{L}_{\mathrm{data}}$, which is defined in Appendix E.1.

Basic strain energy represented by KAN. The basic constitutive behavior of the material is captured by the strain energy contribution Ψ^{KAN} represented by a KAN of arbitrary width and depth. The KAN takes as input the elements of the functional basis $\mathcal{F}$ and the feature vector $\boldsymbol{f}$. For example, Fig. 5 shows two possible KAN architectures, both of which use the principal invariants as the functional basis. A strain energy function that relies on principal stretches as its functional basis requires a more intricate architecture to ensure that Ψ^{KAN} remains symmetric with respect to its arguments. This architecture is built upon the separable ansatz (12) and (13). Further details are provided in Appendix C. The only parameters optimized during training are the control points of the spline activation functions in the KAN representing Ψ^{KAN}.



(a) CKAN with topology $\mathcal{N} = [3, 4, 1]$. (b) CKAN with topology $\mathcal{N} = [3, 1, 1]$

Fig. 5. Examples of Kolmogorov–Arnold Networks (KANs) representing the strain energy function Ψ^{KAN}, where the functional basis $\mathcal{I} = \{I_1, I_2, I_3\}$ was chosen and reduced to the setting of an incompressible material so that I_3 is no longer required as an explicit part of the input. Depending on the complexity of the constitutive behavior of the material of interest, the number of layers and neurons per layer can be adjusted.

Normalization of stress. It can be expected that Ψ^{KAN} will approximate stresses with reasonable quality. Minor inaccuracies are generally acceptable for most practical applications. However, there is one important exception: at zero strain, the stresses should be exactly zero to avoid numerical problems in computer simulations using CKAN-based strain energy functions. Therefore, from a practical point of view, it is advantageous to add a correction term Ψ^{σ} to Ψ^{KAN} to ensure that the stress at zero strain is also zero within numerical precision. One can determine Ψ^{σ} by computing the stress resulting from Ψ^{KAN} at zero strain and constructing a potential Ψ^{σ} such that the stresses resulting from its first derivatives exactly cancel those resulting from Ψ^{KAN} at zero strain, cf. Dal et al. (2023) and Linden et al. (2023). Appendix D demonstrates that Ψ^{σ} generally takes the form

$$\Psi^{\sigma}(J, \boldsymbol{f}) := -\Xi(\boldsymbol{f})\,(J - 1). \tag{22}$$

Notably, $\Psi^{\sigma}(J, \boldsymbol{f})$ is zero in the undeformed state, where $J = 1$. Furthermore, Appendix D provides equations for computing Ξ depending on the chosen functional basis. However, it is important to note that Ξ depends on the derivative of Ψ^{KAN} with respect to the functional basis, evaluated in the reference configuration. Consequently, Ξ continuously adapts during training as Ψ^{KAN} evolves and is *not* computed in a post-processing step, but an integral part of the network architecture, cf. Fig. 4. For isotropic incompressible materials, the stress normalization condition is satisfied through the hydrostatic pressure p and its associated stress contribution, allowing Ψ^{σ} to be set to zero.

Normalization of strain energy. Since training the KAN relies on minimizing the difference between the model's stresses and the experimentally observed values, and since stresses are related to the first derivatives of Ψ^{KAN}, Ψ^{KAN} may differ from the actual strain energy function by an arbitrary constant after training. By definition, Ψ^{σ} vanishes in the undeformed configuration. Thus, applying Ψ^{σ} ensures only zero stresses in the undeformed configuration, not zero strain energy. Therefore, one must evaluate Ψ^{KAN} at zero strain and subtract this constant Ψ^{ϵ} from Ψ^{KAN} to ensure zero strain energy in the undeformed state, i.e.,

$$\Psi^{\epsilon}(\boldsymbol{f}) := -\Psi^{\mathrm{KAN}}|_{\mathbf{C}=\mathbf{I}}. \tag{23}$$

Implementation. The CKAN framework is based on the KAN implementation by Liu et al. (2024b), as modified by Polo-Molina et al. (2024), who replaced the original B-spline activation functions with cubic Hermite splines and imposed constraints on the control points to enforce partial monotonicity. In this context, partial monotonicity allows the user to specify, for each input individually, whether the KAN's output should be monotonic (either non-decreasing or non-increasing) or remain unrestricted with respect to that input. Additionally, the cubic Hermite spline is linearly extrapolated outside its domain, using slopes at the left and right ends that match the spline's slopes at the corresponding endpoints. This approach ensures that the spline remains monotonic not only within its original domain, but throughout $\mathbb{R}$ (Polo-Molina et al., 2024). This modification was adopted in the CKAN framework. For example, enforcing non-decreasing monotonicity of the strain energy function with respect to the strain inputs can improve robustness and extrapolation. Further technical details on the implementation are provided in Appendix E.

4. Symbolic constitutive modeling

Deriving an interpretable symbolic constitutive model requires two additional steps. First, during training, the dense CKAN must be sparsified by iteratively removing redundant activation functions. This resuls in a more compact network that reduces overfitting

and enhances generalizability. Second, the remaining activation functions must be expressed in symbolic form through symbolic regression.

4.1. Sparsification

As depicted in Fig. 5, KAN layers with varying widths can be stacked arbitrarily. In practice, the optimal topology $\mathcal{N}$ of the CKAN for a given dataset is often unknown unless the dataset is derived from a known symbolic expression, such as a predefined strain energy function. To address this issue, we propose starting with a large CKAN and using sparsity regularization during training to automatically determine its optimal topology. In standard FFNNs, L_1 regularization is applied to weights to promote sparsity (Abdolazizi et al., 2023; McCulloch et al., 2024). In analogy, we regularize the L_1 norm of activation functions $\phi_{l,j,i}$. Passing an input batch of n_s samples to a KAN results in n_s inputs $x^s_{l,i}$ to each activation function $\phi_{l,j,i}$. Following Liu et al. (2024b), the L_1 norm of an activation function $\phi_{l,j,i}$ is then the average magnitude over these n_s samples:

$$|\phi_{l,j,i}|_1 := \frac{1}{n_s}\sum_{s=1}^{n_s}\left|\phi_{l,j,i}(x^s_{l,i})\right|. \tag{24}$$

Based on the L_1 norm of a single activation function, the L_1 norm of a KAN layer $\mathbf{\Phi}_l$ with n_{in} inputs and n_{out} outputs is given by:

$$|\mathbf{\Phi}_l|_1 := \sum_{i=1}^{n_{\text{in}}}\sum_{j=1}^{n_{\text{out}}}|\phi_{l,j,i}|_1. \tag{25}$$

Based on the findings that L_1 regularization alone is insufficient for sparsification, Liu et al. (2024b) introduced the entropy S of a KAN layer $\mathbf{\Phi}_l$ as:

$$S(\mathbf{\Phi}_l) = -\sum_{i=1}^{n_{\text{in}}}\sum_{j=1}^{n_{\text{out}}}\frac{|\phi_{l,j,i}|_1}{|\mathbf{\Phi}_l|_1}\log_2\left(\frac{|\phi_{l,j,i}|_1}{|\mathbf{\Phi}_l|_1}\right). \tag{26}$$

Then, the total loss function used for training the network is the sum of the data loss $\mathcal{L}_{\text{data}}$ (characterizing the difference between network output and training data) and the regularization loss

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{data}} + \Lambda\left(\mu_1\sum_{l=0}^{L-1}|\mathbf{\Phi}_l|_1 + \mu_2\sum_{l=0}^{L-1}S(\mathbf{\Phi}_l)\right), \tag{27}$$

where the parameters μ_1 and μ_2 are usually set to $\mu_1 = \mu_2 = 1$, and the hyperparameter Λ controls the overall regularization magnitude.

Evaluating the importance of edges using the L_1 norm of an activation function may be limiting, as it considers only local information. The most recent MultKAN paper (Liu et al., 2024a) addresses this limitation by introducing attribution scores for edges and nodes. These scores are computed iteratively, propagating importance backward from the output layer to the input layer. Although we have successfully obtained sparse and highly accurate CKANs using only L_1 and entropy regularization, we anticipate that incorporating these attribution scores into our framework in future research will further improve the interpretability and performance of CKANs.

4.2. Pruning

After training with the sparsification regularization (27), the network may be pruned on the node level. Following Liu et al. (2024b), pruning of node (l,i) is based on its incoming and outgoing scores $I_{l,i}$ and $O_{l,i}$, respectively, defined by:

$$I_{l,i} = \max_k\{|\phi_{l-1,i,k}|_1\}, \qquad O_{l,i} = \max_j\{|\phi_{l,j,i}|_1\}. \tag{28}$$

If either $I_{l,i}$ or $O_{l,i}$ is below a certain threshold hyperparameter θ, the corresponding node (l,i) is pruned. However, in our application, pruning was not beneficial, as it often removed nodes that had minimal impact on the absolute value of the strain energy function but substantially affected its derivatives, which are critical for accurate stress predictions. Additionally, pruning introduces the additional hyperparameter θ. Therefore, we relied entirely on sparsification regularization. If a spline-based activation function was best approximated by the zero function during symbolification, then the spline function was replaced by the zero function. The symbolification process is described in the following section. Nonetheless, exploring alternative pruning strategies in future research — such as those based on the non-local attribution scores proposed in Liu et al. (2024a) — could be valuable.

4.3. Symbolification

In principle, a user could manually inspect the spline activation functions $\phi_{l,j,i}$ and attempt to identify a symbolic counterpart. However, for large CKAN architectures, automatic identification is generally more practical. To determine a suitable symbolic replacement, the fitting performance of symbolic univariate candidate functions $f_{l,j,i}$ from a predefined function library $\mathcal{B}$ is evaluated for a given $\phi_{l,j,i}$ (Liu et al., 2024b). This function library includes common elementary functions such as the zero function, monomials, and the exponential function, e.g., $\mathcal{B} = \{0, x, x^2, \dots, \ln(x), \exp(x), \cos(x), \dots\}$. If the user has prior knowledge and expects a specific symbolic function, it can also be included in $\mathcal{B}$.

Even if the graphs of the activation function $\phi_{l,j,i}$ and a candidate function $f_{l,j,i}$ appear similar, directly replacing $\phi_{l,j,i}$ with $f_{l,j,i}$ rarely yields a good fit. Rather, shifting and scaling the input and output are additionally required. Thus, identifying a suitable symbolic expression translates to finding a suitable candidate function $f_{l,j,i} \in \mathcal{B}$ and corresponding affine parameters — $a_{l,j,i}$, $b_{l,j,i}$, $c_{l,j,i}$, $d_{l,j,i} \in \mathbb{R}$ — such that (Liu et al., 2024b)

$$\phi_{l,j,i}(x_{l,i}) \approx y_{l,j,i}(x_{l,i}) := c_{l,j,i}\, f_{l,j,i}(a_{l,j,i}\, x_{l,i} + b_{l,j,i}) + d_{l,j,i}. \tag{29}$$

To solve this optimization problem, we first evaluate the activation function $\phi_{l,j,i}$ at n_s sample points $x^s_{l,i}$ to obtain $\phi^s_{l,j,i} = \phi_{l,j,i}(x^s_{l,i})$. The sampling range from which the tuples $\{(x^s_{l,i}, \phi^s_{l,j,i})\}_{s=1}^{n_s}$ are drawn, as well as the spacing between these tuples, is crucial for accurately identifying the symbolic function $f_{l,j,i}$ and the associated affine parameters. Details of the sampling strategy are discussed in Appendix E.3. After sampling, the affine parameters are determined for each $f_{l,j,i} \in \mathcal{B}$, by minimizing the loss function

$$\mathcal{L}_{\text{affine}} = \frac{1}{n_s}\sum_{s=1}^{n_s}\left(\frac{\phi^s_{l,j,i} - y_{l,j,i}(x^s_{l,i})}{\phi^s_{l,j,i}}\right)^2 + \frac{1}{n_s}\sum_{s=1}^{n_s}\left(\frac{\phi'^s_{l,j,i} - y'_{l,j,i}(x^s_{l,i})}{\phi'^s_{l,j,i}}\right)^2, \tag{30}$$

where we propose the second term to incorporate information on the derivative

$$y'_{l,j,i}(x^s_{l,i}) = a_{l,j,i}\, c_{l,j,i}\, f'_{l,j,i}(a_{l,j,i}\, x^s_{l,i} + b_{l,j,i}), \tag{31}$$

which can be computed analytically since $f'_{l,j,i}$ is available in symbolic form. The derivatives $\phi'^s_{l,j,i}$ can be computed either numerically or analytically using established formulas for spline derivatives. The goodness of the fit of all $f_{l,j,i} \in \mathcal{B}$ and the corresponding affine parameters is ranked according to a suitable performance metric, such as the loss $\mathcal{L}_{\text{affine}}$, and the best is selected as the symbolic counterpart of $\phi^s_{l,j,i}$.

In the original KAN papers (Liu et al., 2024a,b), this fitting process is performed through an initial iterative grid search over $a_{l,j,i}$ and $b_{l,j,i}$, followed by linear regression to determine $c_{l,j,i}$ and $d_{l,j,i}$. In contrast, we optimized these parameters simultaneously using the BFGS-B optimizer. In our examples, this approach resulted in a more accurate fit of the symbolized activation functions and significantly increased computational speed, particularly when working with an extensive function library and a large number of extracted sample points n_s. Moreover, this method enables application of individual box constraints to the affine parameters, thereby enhancing flexibility and control in the optimization process. For instance, the monotonicity of the spline activation functions can be maintained in the symbolic counterpart by selecting candidate functions $f_{l,j,i}$ from a function library containing only monotonic functions and by restricting $a_{l,j,i}$, $c_{l,j,i} \in \mathbb{R}_{\geq 0}$. As a result, the monotonicity of the strain energy function with respect to its inputs is also inherited by its symbolic counterpart, given that a KAN consists of a composition of univariate functions and positively weighted sums, cf. (17).

Symbolifying spline-based activation functions involves a trade-off between interpretability and approximation accuracy. To improve their suitability for symbolification, regularization can encourage smoother, more interpretable spline shapes. In this work, we enforced monotonicity (Polo-Molina et al., 2024), which prevents unwanted oscillations and thus enhances the robustness of the symbolic approximation. Moreover, the KAN implementation we use (Liu et al., 2024b) provides an extra regularization option that penalizes large differences between neighboring control points, further promoting smoothness. Although we did not use this regularization in the current study, it may aid symbolification in other cases. In all numerical examples in the present study, however, the symbolic strain energy functions retained high accuracy. This suggests that, at least for the materials and deformation regimes considered here, the symbolification step yields symbolic models that are both expressive and physically meaningful.

5. Numerical examples

In this section, we assess the performance and capabilities of the proposed Constitutive Kolmogorov–Arnold Networks (CKANs) using four experimental datasets. First, we examine Treloar's (Treloar, 1944b) and Kawabata's (Kawabata et al., 1981) experimental data for rubber materials, which are widely regarded as benchmark datasets in hyperelasticity. Next, we study human brain (cortex) tissue data (Budday et al., 2017b), which exhibits complex nonlinear behavior and tension-compression asymmetry. Finally, we use data from Ecoflex silicone polymers with varying Shore hardness values (Liao et al., 2020) to demonstrate how CKANs cannot only describe but also predict the mechanical properties of materials.

To train a CKAN, we minimize the difference between the predicted stress and the stress determined from experiments conducted under homogeneous deformation and quasi-static loading conditions. In such settings, the first Piola–Kirchhoff stress can be computed from the externally applied forces and the specimen's cross-sectional area. In this study, we focus on the deformation modes of biaxial tension and simple shear. While Appendix A.1 outlines the general stress formulations for various functional bases, the detailed expressions for biaxial tension and simple shear under homogeneous deformation — necessary for reproducing the numerical examples — are given in Appendix A.2. In all examples, we assume incompressibility, a common assumption for rubber and brain tissue. The hydrostatic pressure p is determined from the assumption of plane stress, which is justified by the loading conditions and the specimens' shapes in the experiments.

For simplicity, this section and the related appendices only report $\hat{\Psi}^{\text{KAN}}$, as the normalization terms of the strain energy Ψ^{σ} and Ψ^{ϵ} directly follow from (22) and (23), respectively. For an isotropic incompressible material, the stress-free reference configuration is guaranteed by the stress contribution of the hydrostatic pressure p. For brevity, we drop the feature vector $\boldsymbol{f}$ from the notation where it is not explicitly needed.

In Appendix B, we report the hyperparameters and optimizers of the (sub-)CKANs used in this section.

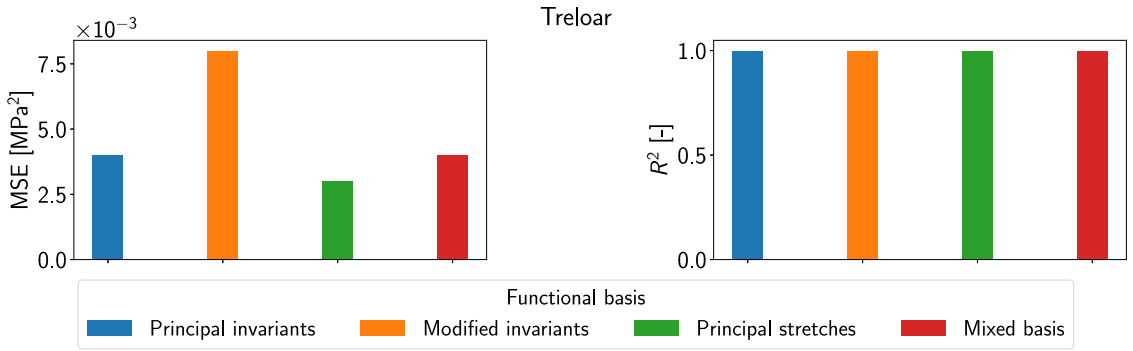


Fig. 6. Comparison of the descriptive performance on Treloar's data: Mean squared error (MSE) and coefficient of determination (R^2) of each functional basis for fitting the stress component P_{11}.

5.1. *Multi-axial loading of vulcanized rubber: Treloar's and Kawabata's experiments*

The performance of constitutive models for rubber-like materials is often evaluated using the classical experimental dataset on vulcanized rubber reported by Treloar (Treloar, 1944b). Several studies have used this prototypical dataset of a highly nonlinear, elastic, and incompressible material to assess and rank the validity of material models (Steinmann et al., 2012; Marckmann and Verron, 2006). Treloar's dataset includes uniaxial tension, equibiaxial tension, and pure shear experiments. While many material models in the literature fit a single deformation mode well, most struggle to capture all three deformation modes simultaneously.

Kawabata et al. (1981) conducted a series of general biaxial experiments on vulcanized rubber. In these tests, the stretch λ_1 in one direction was fixed at various values while the stretch λ_2 in the orthogonal planar direction was varied. During the experiments, the nominal stresses $P_{11}(\lambda_1, \lambda_2)$ and $P_{22}(\lambda_1, \lambda_2)$ in the first and second directions, respectively, were recorded. Due to the material's incompressibility, any kinematically admissible deformation is fully characterized by two independent variables, such as λ_1 and λ_2 or I_1 and I_2. Thus, Kawabata's dataset fully characterizes the material behavior under any multi-axial deformation mode within the prescribed stretch range. Indeed, the results of uniaxial, equibiaxial, and pure shear tests can in principle be extracted also from the general biaxial test data, as shown in Fig. 8.

As reported in Marckmann and Verron (2006), Treloar and Kawabata et al. used vulcanized rubber with the same chemical composition; only slight deviations were observed in the experimental data (Amores et al., 2020). Therefore, the materials can be considered as identical. Consequently, if a model trained on Treloar's data accurately predicts Kawabata's data, it demonstrates genuine generalization. Thus, following Marckmann and Verron (2006), Amores et al. (2020) and Khiêm and Itskov (2016), we trained the CKAN architecture using Treloar's dataset alone and derived a symbolic strain energy function. Note that during training, the CKAN was only provided with data for P_{11}, *not* P_{22}. We subsequently evaluated the discovered strain energy function for generalization performance on Kawabata's dataset, predicting both P_{11} *and* P_{22}. We considered the four distinct functional bases for the strain energy function introduced above: the principle stretches $\{\lambda_1, \lambda_2, \lambda_3\}$, the principal invariants $\{I_1, I_2, I_3\}$, the modified invariants $\{\iota_1, \iota_2, I_3\}$, and a mixed set of principal stretches and principal invariants $\{\lambda_1, \lambda_2, \lambda_3, I_1, I_2, I_3\}$. For each functional basis, we trained the CKAN, derived symbolic expressions for the strain energy function (see Table 1), and assessed the models' training (fitting) performance on Treloar's data and their validation (prediction) performance on Kawabata's data.

CKAN performance comparison across four functional bases. For each of the four functional bases, the trained CKAN exhibited high descriptive accuracy on Treloar's dataset, all achieving $R^2 \geq 0.996$, Fig. 6. However, their ability to generalize beyond the training data differed substantially when tested on Kawabata's biaxial data. Fig. 7 shows the mean normalized mean squared error[1] (MNMSE) on Kawabata's dataset and demonstrates that the CKAN using the mixed functional basis outperformed the others. Fig. 8(a) shows the fitting results of the mixed functional basis on Treloar's data, and Figs. 8(c)–8(f) show the generalization performance on Kawabata's data. The CKAN using the principal-stretch basis performed nearly as well as the one using the mixed basis. The modified invariant basis ranked third, while the principal invariant basis exhibited the weakest generalization. These findings align with previous studies that found for vulcanized rubber, reliance on only I_1 and I_2 may be insufficient to generalize from Treloar's data to Kawabata's multi-axial data (Dal et al., 2023; Marckmann and Verron, 2006).

Details on the mixed basis. The mixed basis enabled top performance (average $R^2 = 0.999$ on Treloar's data). Its symbolic strain energy function (Table 1a) resembles an Ogden-type term alongside a quadratic dependence on I_2. However, the larger network architecture prolongs training because separate sub-networks handle the contributions from I_1, I_2 and from $\lambda_\alpha, \lambda_\alpha^{-1}$. Despite this computational cost, the mixed approach excels in both descriptive and generalization performance (Figs. 8(a),8(c)–8(f)).

[1] First, the normalized mean squared errors (NMSE) for each stretch level λ_1 are separately computed, then these NMSE are averaged for each functional basis and stress component in the small and large stretch regime individually.

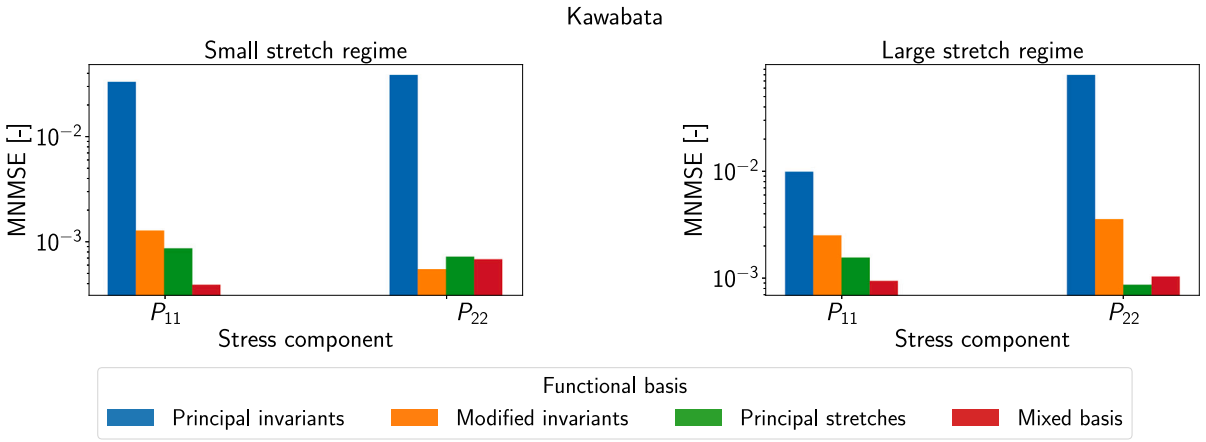


Fig. 7. Comparison of the generalization performance on Kawabata's data: Mean normalized mean squared error (MNMSE) of each functional basis for predicting the stress components P_{11} and P_{22} in the small and large stretch regimes of Kawabata's data.

Table 1
Symbolic strain energy functions and material parameters for vulcanized rubber discovered by the (a) mixed principal invariant- and principal stretch-based (b) principal stretch-based, (c) modified invariant-based, and (d) principal invariant-based CKAN framework trained on Treloar's data (Treloar, 1944b).

(a) Mixed principal invariant- and principal stretch-based

Strain energy function	$\hat{\Psi}^{\text{KAN}} = \hat{\Psi}_I^{\text{KAN}}(I_1, I_2) + \sum_{\alpha=1}^{3} \hat{\omega}_1^{\text{KAN}}(\lambda_\alpha), \quad \hat{\Psi}_I^{\text{KAN}}(I_1, I_2) = a(b\, I_2 + 1)^2, \quad \hat{\omega}_1^{\text{KAN}}(\lambda_\alpha) = c\, \left(d\, \lambda_\alpha + 1\right)^{32} + e\, \left(f\, \lambda_\alpha + 1\right)^{10}$					
Material parameters	a [MPa]	b [–]	c [MPa]	d [–]	e [MPa]	f [–]
	2.8333	$7.6034 \cdot 10^{-4}$	$4.9633 \cdot 10^{-3}$	0.0315	2.3312	0.0219

(b) Principal stretch-based

Strain energy function	$\hat{\Psi}^{\text{KAN}} = \sum_{\alpha=1}^{3} \hat{\omega}_1^{\text{KAN}}(\lambda_\alpha) + \sum_{\alpha=1}^{3} \hat{\omega}_{-1}^{\text{KAN}}(\lambda_\alpha^{-1}), \quad \hat{\omega}_1^{\text{KAN}}(\lambda_\alpha) = a\,(b\, \lambda_\alpha + 1)^{30} + c\,(d\, \lambda_\alpha + 1)^5, \quad \hat{\omega}_{-1}^{\text{KAN}}(\lambda_\alpha^{-1}) = e\,(f\, \lambda_\alpha^{-1} + 1)^{15}$					
Material parameters	a [MPa]	b [–]	c [MPa]	d [–]	e [MPa]	f [–]
	0.0006	0.0441	1.1489	0.0736	0.3981	0.0067

(c) Modified invariant-based

Strain energy function	$\hat{\Psi}^{\text{KAN}}(\iota_1, \iota_2) = a\left(b\, \iota_2 + 1\right)^2 - c\, \operatorname{atan}\left(d(-e\, \iota_1 - 1)^3 + f\right)$					
Material parameters	a [MPa]	b [MPa]	c [MPa]	d [MPa]	e [–]	f [–]
	2.1873	0.0868	1024.9	1.3653	0.4019	65.423

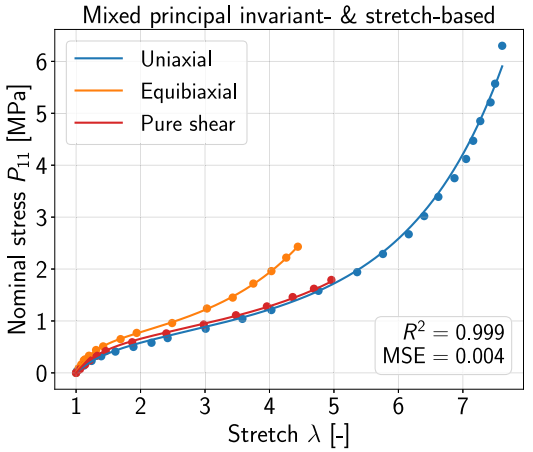
(d) Principal invariant-based

Strain energy function	$\hat{\Psi}^{\text{KAN}}(I_1, I_2) = a\left(b\, I_1 + c\, I_2 + 1\right)^3 + d \exp(e\, I_1)$				
Material parameters	a [MPa]	b [–]	c [–]	d [MPa]	e [–]
	468.37	$1.1034 \cdot 10^{-4}$	$2.2632 \cdot 10^{-6}$	0.0362	0.0774

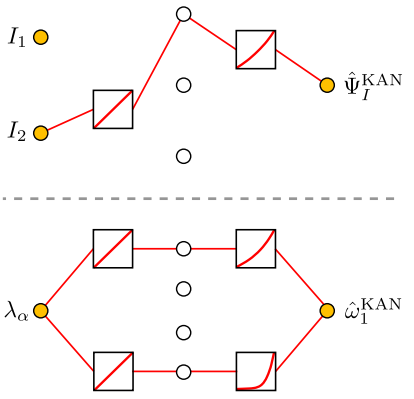
Details on the principal stretch basis. The purely principal stretch-based CKAN also demonstrated strong generalization, narrowly trailing the mixed basis in generalization performance on Kawabata's data (see Appendix, Fig. F.1). Training time is similar to that of the mixed basis approach (due to the multiple evaluations of $\hat{\omega}_1^{\text{KAN}}$ and $\hat{\omega}_{-1}^{\text{KAN}}$). The recovered symbolic expression (Table 1b) shows a close resemblance to the Ogden model (Ogden, 1972).

Details on the modified invariant basis. The CKAN relying on the modified invariants achieved a descriptive performance comparable to the other bases on Treloar's data (cf. Appendix, Fig. F.2). However, it fell short of the mixed basis and the principal stretch-based formulations in generalization performance. The resulting symbolic function (Table 1c) better distinguishes between uniaxial tension and pure shear and thus outperforms the principal invariant basis when extrapolating to Kawabata's biaxial scenario.

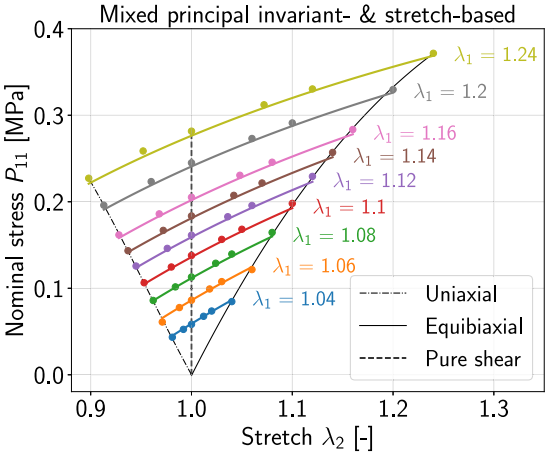
Details on the principal invariant basis. While the CKAN with the principal invariant-based formulation an $R^2 = 0.996$ on Treloar's data (cf. Appendix, Fig. F.3), it yielded the largest MNMSE on Kawabata's dataset (Fig. 7). These results align with previous findings that the principal invariants I_1 and I_2 alone may not adequately capture multi-axial deformation in vulcanized rubber. On a positive note, the simpler network architecture (with only one sub-network) reduces training time. The final symbolic expression (Table 1d) incorporates polynomial terms in I_1 and I_2, and an exponential term in I_1.



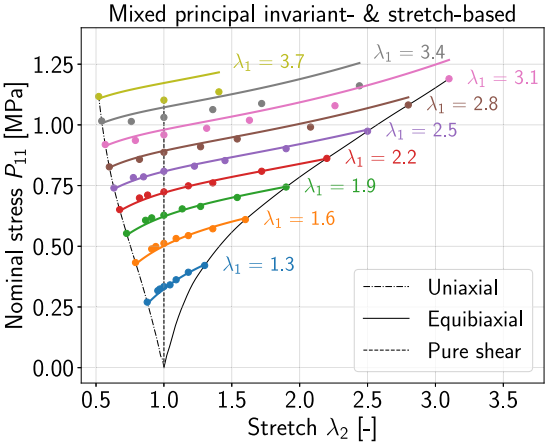
(a) Fitting performance on Treloar's data.



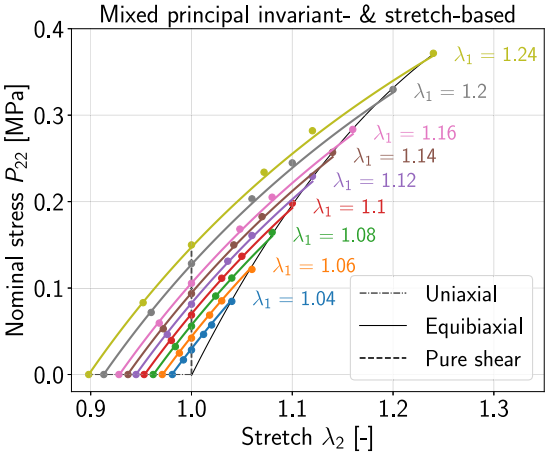
(b) CKAN architecture.



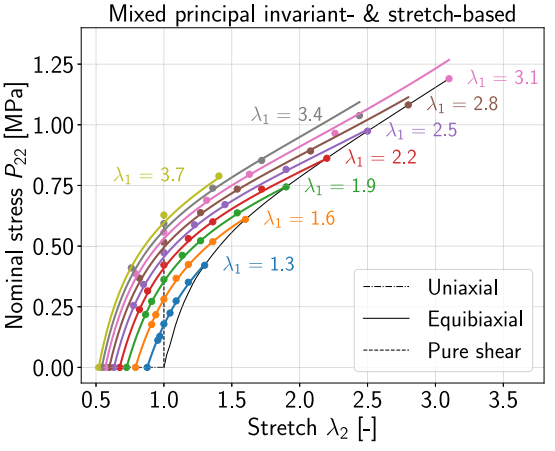
(c) P_{11} generalization: small stretch regime of Kawabata's data.



(d) P_{11} generalization: large stretch regime of Kawabata's data.

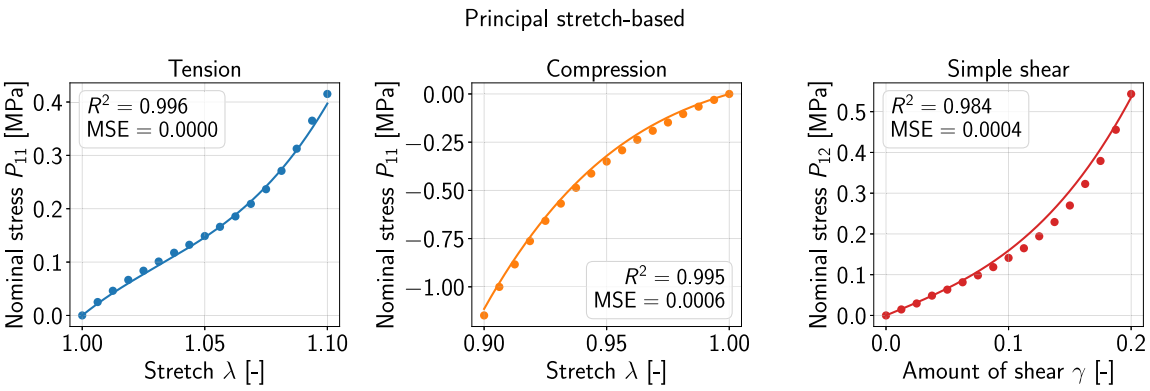


(e) P_{22} generalization: small stretch regime of Kawabata's data.



(f) P_{22} generalization: large stretch regime of Kawabata's data.

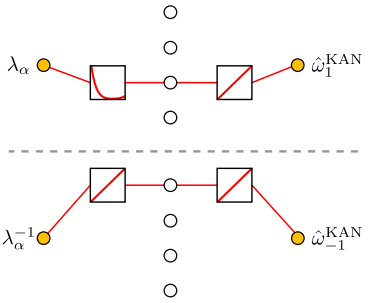
Fig. 8. Training and validation results for vulcanized rubber. (a) Fitting performance on Treloar's data (Treloar, 1944b). (b) Architecture of the final CKAN after training; unconnected nodes indicate pruned activation functions. (c)–(f) Generalization performance of the *mixed principal invariant- and principal stretch-based* CKAN on the biaxial dataset by Kawabata et al. (1981), after the CKAN was trained on Treloar's data only. The solid lines represent the model response. The dots represent the experimental data.



(a) Fitting results: experimental data (dots) and performance of CKAN (solid lines)

Strain energy function	$\hat{\Psi}^{\mathrm{KAN}} = \sum_{\alpha=1}^{3} \hat{\omega}_1^{\mathrm{KAN}}(\lambda_\alpha) + \sum_{\alpha=1}^{3} \hat{\omega}_{-1}^{\mathrm{KAN}}(\lambda_\alpha^{-1})$ $\hat{\omega}_1^{\mathrm{KAN}}(\lambda_\alpha) = a\,(b\,\lambda_\alpha - 1)^4,$ $\hat{\omega}_{-1}^{\mathrm{KAN}}(\lambda_\alpha^{-1}) = c\,\exp(d\,\lambda_\alpha^{-1})$			
Material parameters	a [MPa]	b [-]	c [MPa]	d [-]
	104.0147	0.9639	14.4781	0.1410

(b) Discovered strain energy function and material parameters.



(c) CKAN architecture.

Fig. 9. Descriptive performance of CKAN based on principal stretches for human brain (cortex) tissue (Budday et al., 2017a): (a) Performance of CKAN simultaneously fitted to uniaxial tension, compression, and simple shear data. (b) Resulting strain energy functions and material parameters. (c) Architecture of the final CKAN after training; unconnected nodes indicate pruned activation functions.

Although all formulations fit Treloar's data well, their generalization to Kawabata's biaxial results varied considerably. The mixed basis and the principal stretch-based CKANs had the best generalization performance. The modified invariant basis offered a balanced compromise between generalization performance and computational simplicity. In contrast, the principal invariant basis, though fastest to train, was least effective in predicting multi-axial deformations. Ultimately, the optimal functional basis must be selected for each material individually, considering both the relevant deformation regime and available computational resources. In this regard, CKANs are a powerful and flexible tool.

5.2. Human brain tissue

To demonstrate how the CKAN framework can handle complex biological tissues with limited experimental data and still produce interpretable strain energy functions, we consider human brain tissue experiments from Budday et al. (2017a). Specifically, we examine the homogeneous deformation modes of uniaxial tension (UT), uniaxial compression (UC), and simple shear (SS). These datasets were obtained from cubic samples of the cortex region, excised from ten different human brains. For UT and UC, the experimental measurements consist of the in-plane stretches and corresponding nominal stresses. Simple shear data comprise the amount of shear γ and nominal shear stress. All tests were performed within the elastic regime, but the dataset remains sparse, covering only a limited range of deformations. Unlike the more extensive rubber datasets discussed in Section 5.1, the primary focus here is on how effectively the identified strain energy functions describe each observed deformation mode rather than on wide-range extrapolation.

Principal stretches as the most effective basis for brain tissue. Among the three functional bases tested for this example (principal stretches, principal invariants, and modified invariants), the principal stretch-based formulation provided the best descriptive performance for UT, UC, and SS. Fig. 9 illustrates this performance. Fig. 10 compares the mean squared error (MSE) and R^2 metrics for each basis, showing that principal stretches more accurately capture both tension-compression asymmetry and shear response. The final symbolic strain energy function comprised terms involving both λ_α and their reciprocals λ_α^{-1}, mirroring the behavior noted in Ogden-type models (Budday et al., 2017b; Mihai et al., 2017; Anssari-Benam et al., 2022).

Increasing sparsity (Section 4) often compressed the strain energy to a single function of the reciprocal stretches, e.g.,

$$\hat{\Psi}^{\mathrm{KAN}} = \sum_{\alpha=1}^{3} \hat{\omega}_{-1}^{\mathrm{KAN}}\big(\lambda_\alpha^{-1}\big), \quad \text{with} \quad \hat{\omega}_{-1}^{\mathrm{KAN}}(\lambda_\alpha^{-1}) = a\,(\lambda_\alpha^{-1} + b)^{18}.$$

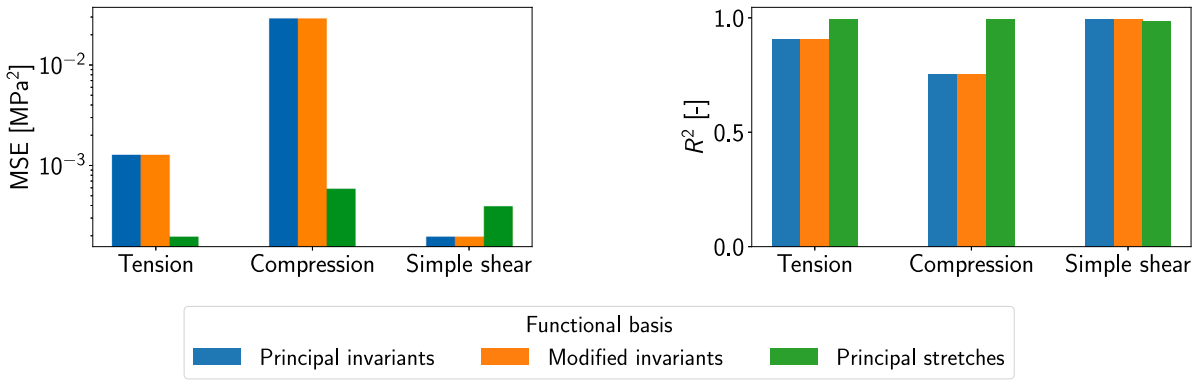


Fig. 10. Comparison of the descriptive performance on the brain tissue data. Mean squared error (MSE) and coefficient of determination (R^2) of the symbolic strain energy functions for the human brain (cortex) each using a different functional basis.

Although these single-term models had some difficulty with the S-shaped stress response of UT, they provide a decent overall description of the brain tissue's mechanical behavior. This highlights the important role of λ_α^{-1} in accurately capturing the tension-compression asymmetry. The fitting results of the one-term strain energy function, along with the corresponding material parameters, are presented in the Appendix in Fig. G.1.

Alternative bases: principal and modified invariants. The principal invariant-based CKAN converged to a strain energy function that was solely dependent on the second invariant I_2. This choice agrees with earlier observations that the brain's distinctive shape-change-dominated behavior can be modeled effectively by functions of I_2 (Kuhl and Goriely, 2024). While this approach performed well for uniaxial tension and simple shear, it struggled with uniaxial compression, as demonstrated in the Appendix in Fig. G.2. Similarly, the modified invariant-based formulation produced nearly identical results, relying exclusively on $\imath_2$ (cf. Appendix, Fig. G.3). Consequently, both invariant-based approaches captured certain deformation modes, yet neither fully described the tension-compression asymmetry observed in brain tissue.

Overall, these findings show that choosing the right functional basis is critical for model performance, particularly for materials with pronounced tension-compression asymmetry. While principal invariant- and modified invariant-based CKANs suffice for uniaxial tension and simple shear, the principal stretch-based formulation provides a more comprehensive description of brain tissue mechanics. Unlike the rubber datasets of Section 5.1, this dataset does not allow for a rigorous evaluation of generalizability due to the absence of multi-axial or biaxial tension data in brain tissue. Nevertheless, the significant differences in descriptive performance across the three considered functional bases underscore the importance of selecting an appropriate representation for the targeted material and deformation modes.

5.3. Ecoflex silicone polymer

Silicone elastomers are highly versatile materials extensively used in biomedical engineering (Bernardi et al., 2017), soft robotics (Case et al., 2015), and wearable devices (Jiang et al., 2018) due to their biocompatibility, thermal stability, and stretchability. Ecoflex, a commercial silicone elastomer, is available in various Shore hardness levels that directly correlate with its stiffness. In Liao et al. (2020), extensive uniaxial tension tests were conducted, including uniaxial cyclic loading-unloading tests on Ecoflex with varying Shore hardness levels.

To demonstrate the utility of auxiliary features in constitutive modeling, we analyzed virgin loading curves from these cyclic tests and incorporated Shore hardness as an additional input to the CKAN. Specifically, we considered Shore hardness values $s \in \{\text{OO-10}, \text{OO-20}, \text{OO-30}, \text{OO-50}\}$ and maximal uniaxial stretch $\lambda_{\max} = 6$. Due to the small dataset, we employed leave-one-out cross-validation, withholding one Shore hardness level for validation while training on the others. This setup allowed us to evaluate the model's predictive capability in interpolating and extrapolating material behavior across different Shore hardness levels.

Feature augmentation for predictive constitutive modeling. In many elastomeric and collagenous tissue models, the influence of additional parameters, such as filler volume fraction (Mullins and Tobin, 1965) or cross-link density (Linka et al., 2018), is often captured through *strain amplification factors* χ. These factors modify the principal invariants as follows:

$$\hat{I}_i = \chi(s)\big(I_i - 3\big) + 3, \quad i = 1, 2, \tag{32}$$

allowing for the direct parameterization of the constitutive model by an external feature. However, determining an appropriate functional form for χ can be challenging and is often highly material-specific. In this study, we take a simpler approach to incorporate the Shore hardness s. First, s is normalized using

$$\bar{s} = \frac{s - s_{\min}}{s_{\max} - s_{\min}}, \tag{33}$$

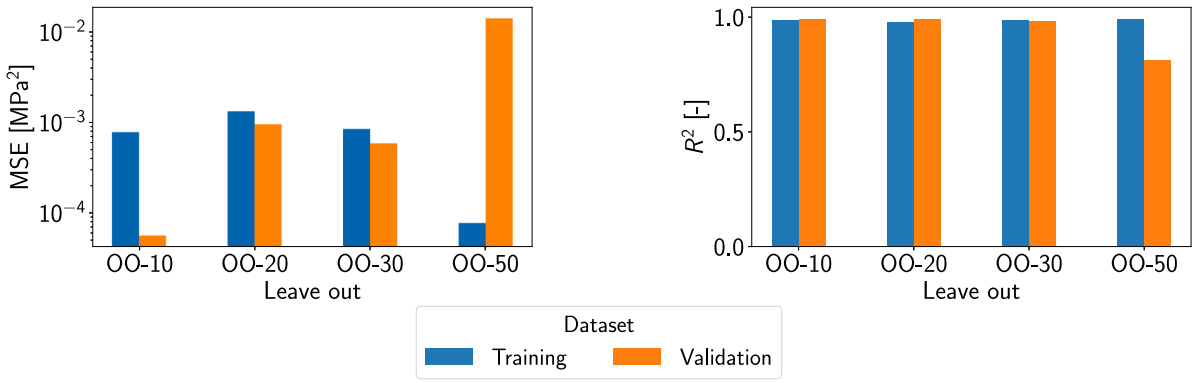


Fig. 11. Comparison of the predictive performance on the Ecoflex silicone polymer data. Mean squared error (MSE) and coefficient of determination (R^2) of the stress curves generated from the discovered symbolic principal invariant-based strain energy functions for Ecoflex silicone on the training and validation set using leave-one-out cross-validation.

Table 2
Strain energy function and material parameters discovered by the principal invariant-based CKAN trained on Ecoflex silicone polymer data from (Liao et al., 2020) using leave-one-out cross-validation.

Strain energy function	$\hat{\Psi}^{\text{KAN}} = a\left[(b\,\bar{s}+1)^2 + c\cosh\left(d\,I_1+e\right)+f\right]^3$					
Leave out	Material parameters					
	a [MPa]	b [–]	c [–]	d [–]	e [–]	f [–]
Shore OO-10	9.6046	0.3722	0.2392	0.0092	0.0805	−0.3287
Shore OO-20	6.4998	0.3427	0.4190	0.0100	0.0800	−0.6707
Shore OO-30	6.4998	0.3427	0.4190	0.0100	0.0800	−0.6707
Shore OO-50	5.4116	0.4997	0.1585	0.0131	0.1045	−0.1422

where $s_{\max} = 50$ and $s_{\min} = 10$. The resulting dimensionless variable $\bar{s}$ is then included via the feature vector $\boldsymbol{f}$ as input to the CKAN. A basic 2-layer CKAN architecture with the topology $\mathcal{N} = [3, 1, 1]$ is utilized to learn a strain energy function that explicitly incorporates the normalized Shore hardness $\bar{s}$. Since the dataset consists exclusively of uniaxial tension data and the primary objective is to evaluate generalizability across different Shore hardness levels, we use the computationally efficient principal invariant-based CKAN formulation. Due to the small dataset and the well-documented relationship that higher Shore hardness generally corresponds to stiffer material behavior (Liao et al., 2020), we additionally impose a monotonicity constraint on the relation between $\bar{s}$ and the strain energy, see Appendix E.2 for details.

Training results (descriptive performance) and validation results (predictive performance). Figs. 11 and 12 visualize the results of the identified symbolic strain energy function on the training and validation dataset. The symbolic strain energy functions demonstrate excellent agreement with the experimental data, achieving accurate fits to the training dataset and reliable predictions for the Shore hardness levels withheld in the respective training process. Especially, in extrapolating the stress–stretch curve for the withheld Shore hardness levels OO-10 and OO-20 in Fig. 12(a) and (b), respectively, precise predictions are achieved.

Symbolic strain energy functions. The symbolic strain energy functions independently identified by the CKANs across the four training datasets are listed in Table 2. Notably, these identified strain energy functions share the same structure: a hyperbolic cosine of I_1 and a quadratic function of $\bar{s}$. This similarity is also evident in the learned network topologies and activation functions shown in Fig. 13. The following characteristics of the identified strain energy functions stand out: First, during the training process, the second principal invariant I_2 is effectively excluded from the input space, probably because of its strong correlation with I_1 in uniaxial tensile tests. In fact, in such tests, I_2 can be uniquely expressed as a function of I_1. Second, the identified material parameters d and e, which operate directly on I_1 within the hyperbolic cosine function, are very similar across all training data sets.

The discovered symbolic expression underscores the capability of the CKAN framework to identify interpretable and predictive material models. Additionally, our findings demonstrate the efficacy of feature augmentation in enabling the predictive modeling of materials with variable properties, such as Ecoflex silicone elastomers.

6. Conclusions

Existing approaches to mechanical constitutive modeling often require a trade-off between accuracy (fitting performance), interpretability, and extrapolability, while also struggling to predict the properties of unknown materials. In this paper, we introduced Constitutive Kolmogorov–Arnold Networks (CKANs), a new framework for discovering symbolic mechanical constitutive laws for materials. CKANs build on Kolmogorov–Arnold Networks (KANs) and incorporate symbolic regression as a post-processing

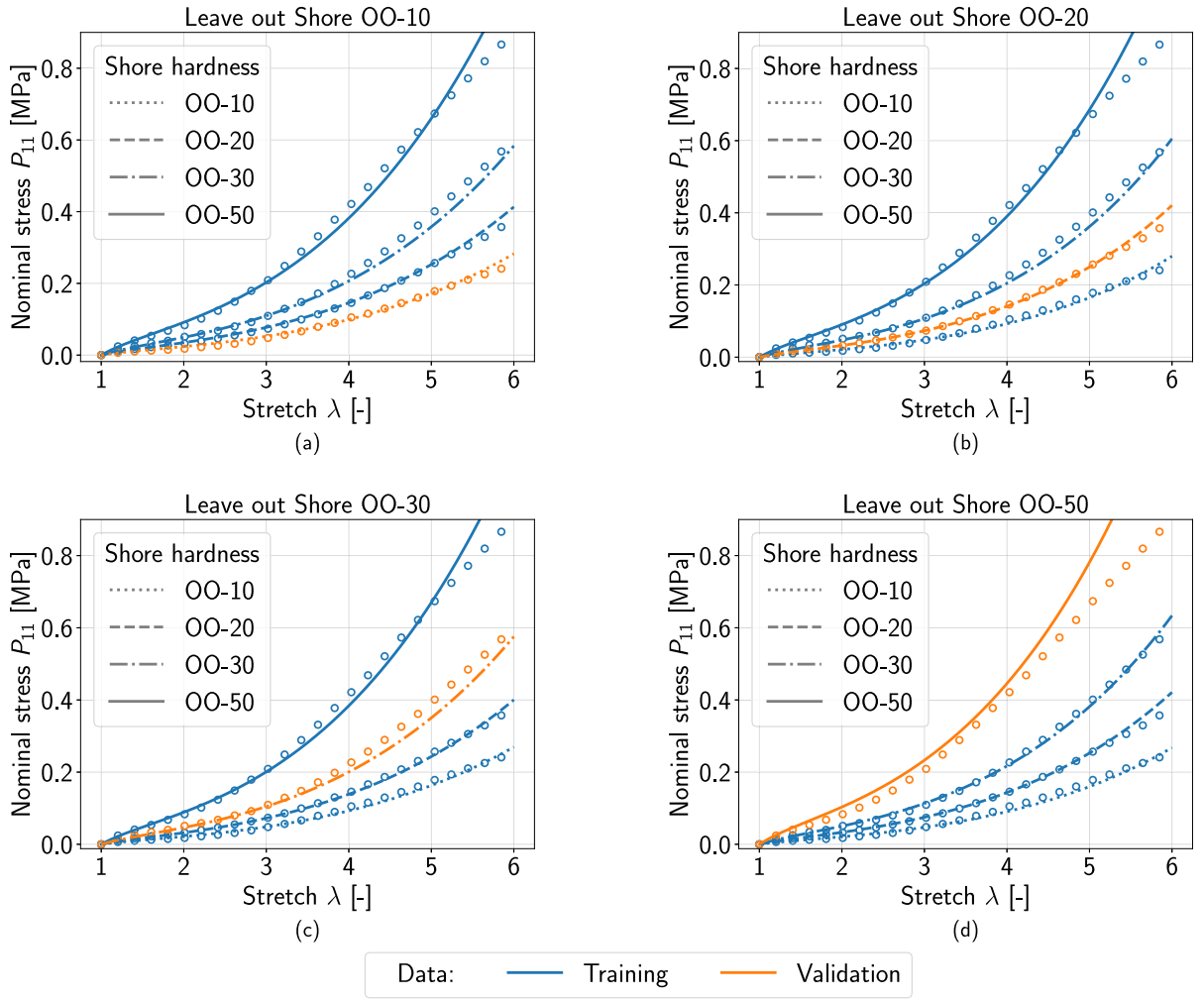


Fig. 12. Training results (descriptive performance) and validation results (predictive performance) for Ecoflex silicone polymer data from Liao et al. (2020) for principal invariant-based symbolic CKANs applied in leave-one-out cross-validation scheme. The lines represent the model response, and the dots the experimental data. Blue indicates data provided during the training (and fitting performance of the model), and orange indicates data withheld during the respective training process. The orange lines thus indicate the ability of the trained CKAN to predict the behavior of a material type (with a different Shore hardness level) for which no testing data was provided within the training data.

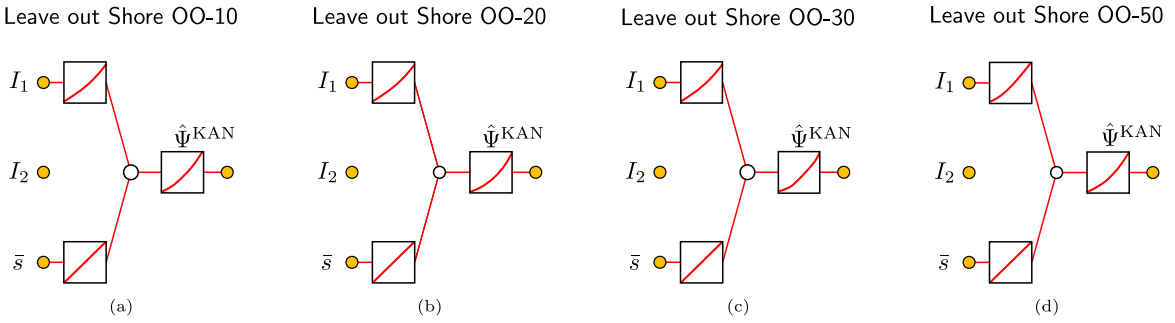


Fig. 13. CKAN architectures obtained during the leave-one-out cross-validation test for Ecoflex silicone polymer data. The red graphs illustrate the symbolic activation functions $f_{i,j,k}$, and the red lines are the only remaining connections after the training. Unconnected nodes indicate activation functions identical to zero. The principal invariants I_1 and I_2, along with the normalized Shore hardness $\bar{s}$ as an additional feature, formed the input to the CKANs. However, the second principal invariant I_2 was discarded during training across all datasets. That is, it was identified as irrelevant.

step. This combination achieves high accuracy, interpretability, and extrapolability simultaneously. Furthermore, thanks to their flexible neural network architecture, CKANs demonstrate strong generalization capabilities, making them well-suited for predicting the behavior of previously unseen materials.

Interpretability. CKANs provide symbolic expressions for the mechanical constitutive behavior of materials. These symbolic expressions are as easily interpretable as traditional, manually derived symbolic constitutive equations, provided that the complexity of the expressions and the number of parameters remain manageable. This can be achieved through regularization strategies that promote simpler model forms, as demonstrated in Appendix G. In particular, CKANs can automatically recover insights from experimental data that where previously obtained through theoretical derivations. For example, in our study of human brain tissue, CKANs using principal or modified invariants identified the second invariant as the dominant variable, aligning with prior theoretical findings (Kuhl and Goriely, 2024; Linka et al., 2023). Similarly, CKANs based on principal stretches recovered an Ogden-type form, consistent with previous results (Budday et al., 2017a; Mihai et al., 2017). These results suggest that simple functional bases can suffice to capture the essential mechanical behavior, reducing both computational cost and interpretive effort.

Computational efficiency. Using sparsification techniques during training, the CKAN architecture effectively eliminates non-essential connections, resulting in sparse symbolic expressions. This sparsity enhances computational efficiency, making CKANs well-suited for finite element software implementation.

Extrapolability. CKANs build on the concept of constitutive artificial neural networks (CANNs) originally introduced in Linka et al. (2021) and Abdolazizi et al. (2023), inheriting their strong extrapolation capabilities. For instance, when trained solely on Treloar's data (uniaxial tension, equibiaxial tension, pure shear), CKANs accurately predict stress–strain curves in Kawabata's multi-axial dataset.

Predictive modeling. Like CANNs, CKANs are very flexible regarding their input. By incorporating additional information beyond stress–strain data, they can predict the mechanical behavior of unknown materials. We demonstrated this by using Shore hardness data for Ecoflex polymers. However, as shown for CANNs, CKANs could also leverage microstructural imaging data or processing information to enhance material property predictions (Linka et al., 2022). In contrast, previous data-driven *model discovery* approaches rely on a predefined library of building block expressions (Flaschel et al., 2021; Linka and Kuhl, 2023; Brunton et al., 2016; Brunton and Kutz, 2022), which limits the variety of possible constitutive models.

Limitations and future directions. Despite the advantages shown here, certain limitations remain. Some manual intervention is required for hyperparameter tuning and symbolic regression, highlighting the need for automated solutions. AI-driven meta-optimization (Hospedales et al., 2021) could streamline these steps, accelerating the discovery of near-optimal models. In a related direction, further constraining the model architecture through theoretical priors, such as polyconvexity, offers promising avenues. A recent extension of the CKAN framework introduced input-convex constraints to enforce polyconvexity and demonstrated the approach's effectiveness on synthetic full-field displacement data (Thakolkaran et al., 2025). Another avenue for future research is the extension of CKANs to more complex materials, e.g., anisotropic (Linka et al., 2021; Kalina et al., 2025) or inelastic materials (Abdolazizi et al., 2023; Holthusen et al., 2024), and multi-physics problems (Franke et al., 2023), potentially aided by a next-generation neural architecture (Liu et al., 2024a). Finally, to firmly establish CKANs as a robust tool for constitutive modeling, extensive validation under real-world conditions and incorporation of uncertainty quantification will be vital.

Summary. Reliable constitutive equations are crucial in computational mechanics, especially for high-stakes applications, such as biomedical and aerospace engineering. While black-box models can provide high predictive accuracy, their lack of transparency limits their use in settings where interpretability and reliability are paramount. The CKAN framework presented here is a compelling alternative that combines robust performance and predictive capabilities with symbolically interpretable strain energy functions. These features makes CKANs ideal for future virtual material design and developing digital twins of complex systems for high-stakes applications.

CRediT authorship contribution statement

Kian P. Abdolazizi: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Roland C. Aydin:** Writing – review & editing. **Christian J. Cyron:** Writing – review & editing, Supervision, Resources, Funding acquisition, Conceptualization. **Kevin Linka:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Funding acquisition, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used OpenAI's ChatGPT-4o to improve the language and readability of the text. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for its publication.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We gratefully acknowledge ZiSheng Liao (South China University of Technology, Guangzhou) for providing the experimental data of Ecoflex silicone polymer. Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 533187597, 517243167.

Appendix A. First Piola–Kirchhoff stresses

Here, we summarize the derivation of the first Piola–Kirchhoff stresses $\mathbf{P}$ for strain energy functions expressed in terms of the different functional bases considered in this paper. Subsequently, the general expressions of $\mathbf{P}$ are specified for the homogeneous deformation modes of biaxial tension and simple shear, which are essential for training our CKANs.

A.1. General deformations

A.1.1. Principal invariant-based strain energy functions

Inserting the principal invariant-based strain energy functions $(10)_1$ and $(11)_1$ into (8), and applying the chain rule along with the differentiation relations

$$\frac{\partial I_1}{\partial \mathbf{C}} = \mathbf{I}, \qquad \frac{\partial I_2}{\partial \mathbf{C}} = I_1\mathbf{I} - \mathbf{C}, \qquad \frac{\partial I_3}{\partial \mathbf{C}} = I_3\mathbf{C}^{-1}, \tag{A.1}$$

yields the following first Piola–Kirchhoff stresses for compressible and incompressible materials, respectively:

$$\mathbf{P} = 2\left[\left(\frac{\partial \Psi}{\partial I_1} + I_1\frac{\partial \Psi}{\partial I_2}\right)\mathbf{F} - \frac{\partial \Psi}{\partial I_2}\mathbf{F}\mathbf{C} + I_3\frac{\partial \Psi}{\partial I_3}\mathbf{F}^{-\mathrm{T}}\right], \tag{A.2}$$

$$\mathbf{P} = -p\mathbf{F}^{-\mathrm{T}} + 2\left[\left(\frac{\partial \Psi}{\partial I_1} + I_1\frac{\partial \Psi}{\partial I_2}\right)\mathbf{F} - \frac{\partial \Psi}{\partial I_2}\mathbf{F}\mathbf{C}\right]. \tag{A.3}$$

A.1.2. Modified invariant-based strain energy functions

For a strain energy function expressed in terms of the modified invariants ι_1 and ι_2, inserting the differentiation relations

$$\frac{\partial \hat{\Psi}}{\partial I_1} = \frac{\partial \hat{\Psi}}{\partial \iota_1}\frac{\partial \iota_1}{\partial I_1} = \frac{1}{6\iota_1}\frac{\partial \hat{\Psi}}{\partial \iota_1}, \qquad \frac{\partial \hat{\Psi}}{\partial I_2} = \frac{\partial \hat{\Psi}}{\partial \iota_2}\frac{\partial \iota_2}{\partial I_2} = \frac{1}{9\iota_2^2}\frac{\partial \hat{\Psi}}{\partial \iota_2}, \tag{A.4}$$

into (A.2) and (A.3) yields the following first Piola–Kirchhoff stresses for compressible and incompressible materials, respectively:

$$\mathbf{P} = 2\left[\left(\frac{1}{6\iota_1}\frac{\partial \Psi}{\partial \iota_1} + I_1\frac{1}{9\iota_2^2}\frac{\partial \Psi}{\partial \iota_2}\right)\mathbf{F} - \frac{1}{9\iota_2^2}\frac{\partial \hat{\Psi}}{\partial \iota_2}\mathbf{F}\mathbf{C} + I_3\frac{\partial \Psi}{\partial I_3}\mathbf{F}^{-\mathrm{T}}\right], \tag{A.5}$$

$$\mathbf{P} = -p\mathbf{F}^{-\mathrm{T}} + 2\left[\left(\frac{1}{6\iota_1}\frac{\partial \hat{\Psi}}{\partial \iota_1} + I_1\frac{1}{9\iota_2^2}\frac{\partial \hat{\Psi}}{\partial \iota_2}\right)\mathbf{F} - \frac{1}{9\iota_2^2}\frac{\partial \hat{\Psi}}{\partial \iota_2}\mathbf{F}\mathbf{C}\right]. \tag{A.6}$$

A.1.3. Principal stretch-based strain energy functions

We substitute the principal stretch-based strain energy functions $(10)_2$ and $(11)_2$ into (8). Applying the chain rule along with the relations

$$\frac{\partial \lambda_\alpha}{\partial \mathbf{C}} = \frac{1}{2\lambda_\alpha}\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha, \qquad \frac{\partial J}{\partial \lambda_\alpha} = \frac{J}{\lambda_\alpha}, \qquad \mathbf{F}\boldsymbol{N}_\alpha = \lambda_\alpha \boldsymbol{n}_\alpha, \tag{A.7}$$

yields the spectral representation of the first Piola–Kirchhoff stress for compressible and incompressible materials, respectively,

$$\mathbf{P} = \sum_{\alpha=1}^{3} \underbrace{\frac{\partial \Psi}{\partial \lambda_\alpha}}_{=P_\alpha} \boldsymbol{n}_\alpha \otimes \boldsymbol{N}_\alpha, \tag{A.8}$$

$$\mathbf{P} = \sum_{\alpha=1}^{3} \underbrace{\left(-\frac{p}{\lambda_\alpha} + \frac{\partial \Psi}{\partial \lambda_\alpha}\right)}_{\hat{P}_\alpha} \boldsymbol{n}_\alpha \otimes \boldsymbol{N}_\alpha, \tag{A.9}$$

where P_α and $\hat{P}_\alpha$ denote the principal stresses.

Finally, we specialize the general expressions (A.8) and (A.9) for the compressible and incompressible separable strain energy functions (12) and (13). In the compressible case,

$$\Psi(\lambda_1, \lambda_2, \lambda_3, \boldsymbol{f}) = \sum_{\alpha=1}^{3} \omega_1(\lambda_\alpha, \boldsymbol{f}) + \sum_{\alpha=1}^{3} \omega_{-1}(\nu_\alpha, \boldsymbol{f}) + \Omega(\lambda_1\lambda_2\lambda_3, \boldsymbol{f}) \tag{A.10}$$

and the corresponding first Piola–Kirchhoff stress is

$$\mathbf{P} = 2\mathbf{F}\frac{\partial \Psi}{\partial \mathbf{C}} = 2\mathbf{F}\left[\left(\sum_{\alpha=1}^{3}\frac{\partial \omega_1}{\partial \lambda_\alpha}\frac{\partial \lambda_\alpha}{\partial \mathbf{C}} + \sum_{\alpha=1}^{3}\frac{\partial \omega_{-1}}{\partial \nu_\alpha}\frac{\partial \nu_\alpha}{\partial \mathbf{C}}\right) + \sum_{\alpha=1}^{3}\frac{\partial \Omega}{\partial J}\frac{\partial J}{\partial \lambda_\alpha}\frac{\partial \lambda_\alpha}{\partial \mathbf{C}}\right]. \tag{A.11}$$

Using (A.7) and

$$\frac{\partial \nu_\alpha}{\partial \mathbf{C}} = \frac{\nu_\alpha}{2}\left(\mathbf{C}^{-1} - \frac{1}{\lambda_\alpha^2}\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha\right), \quad \mathbf{C}^{-1} = \sum_{\alpha=1}^{3}\frac{1}{\lambda_\alpha^2}\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha, \tag{A.12}$$

we obtain

$$\begin{aligned}\mathbf{P} \;=\; \mathbf{F}\Bigg[&\sum_{\alpha=1}^{3}\frac{1}{\lambda_\alpha}\frac{\partial \omega_1}{\partial \lambda_\alpha}\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha + \sum_{\beta=1}^{3}\nu_\beta\frac{\partial \omega_{-1}}{\partial \nu_\beta}\left(\sum_{\alpha=1}^{3}\frac{1}{\lambda_\alpha^2}\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha\right)\\ &- \sum_{\alpha=1}^{3}\frac{\nu_\alpha}{\lambda_\alpha^2}\frac{\partial \omega_{-1}}{\partial \nu_\alpha}\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha + \sum_{\alpha=1}^{3}\frac{\partial \Omega}{\partial J}\frac{J}{\lambda_\alpha^2}\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha\Bigg]\end{aligned} \tag{A.13}$$

$$= \sum_{\alpha=1}^{3}\left[\frac{\partial \omega_1}{\partial \lambda_\alpha} + \frac{1}{\lambda_\alpha}\sum_{\beta=1}^{3}\nu_\beta\frac{\partial \omega_{-1}}{\partial \nu_\beta} - \frac{\nu_\alpha}{\lambda_\alpha}\frac{\partial \omega_{-1}}{\partial \nu_\alpha} + \nu_\alpha\frac{\partial \Omega}{\partial J}\right]\boldsymbol{n}_\alpha \otimes \boldsymbol{N}_\alpha. \tag{A.14}$$

Comparing (A.14) and (A.8) finally yields

$$P_\alpha = \frac{\partial \Psi}{\partial \lambda_\alpha} = \frac{\partial \omega_1}{\partial \lambda_\alpha} + \frac{1}{\lambda_\alpha}\sum_{\beta=1}^{3}\nu_\beta\frac{\partial \omega_{-1}}{\partial \nu_\beta} - \frac{\nu_\alpha}{\lambda_\alpha}\frac{\partial \omega_{-1}}{\partial \nu_\alpha} + \nu_\alpha\frac{\partial \Omega}{\partial J}. \tag{A.15}$$

For the incompressible case, the strain energy function takes on the form

$$\Psi = \hat{\Psi}(\lambda_1, \lambda_2, \lambda_3, \boldsymbol{f}) - p(J-1) = \sum_{\alpha=1}^{3}\hat{\omega}_1(\lambda_\alpha, \boldsymbol{f}) + \sum_{\alpha=1}^{3}\hat{\omega}_{-1}(\lambda_\alpha^{-1}, \boldsymbol{f}) - p(J-1). \tag{A.16}$$

Using

$$\frac{\partial \lambda_\alpha^{-1}}{\partial \mathbf{C}} = -\frac{1}{2\lambda_\alpha^3}\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha \tag{A.17}$$

and following the same steps as in the compressible case, we obtain

$$P_\alpha = -\frac{p}{\lambda_\alpha} + \frac{\partial \hat{\Psi}}{\partial \lambda_\alpha} = -\frac{p}{\lambda_\alpha} + \frac{\partial \hat{\omega}_1}{\partial \lambda_\alpha} - \frac{1}{\lambda_\alpha^2}\frac{\partial \hat{\omega}_{-1}}{\partial \lambda_\alpha^{-1}}. \tag{A.18}$$

A.2. Special deformation modes

In the numerical examples presented in Section 5, we use experimentally determined stress data from biaxial tensile and simple shear tests on rubber and human brain tissue, conducted under conditions of homogeneous deformation and quasi-static loading. Under these conditions, the first Piola–Kirchhoff stress tensor can be computed from the external forces and the specimen's cross-sectional area. During training, the error between the experimental stress data and the stress predicted by the CKAN is minimized, thereby implicitly learning the strain energy function. Consequently, we specify the stress tensor associated with the strain energy function for these special load cases.

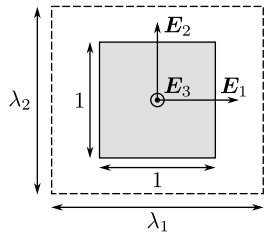
In the following, the component indices of tensors refer to the fixed Cartesian coordinate system with basis vectors $\{\boldsymbol{E}_1, \boldsymbol{E}_2, \boldsymbol{E}_3\}$ illustrated in Fig. A.14. For the ease of notation, we equate tensors with the matrices representing their components. This notation is unambiguous and thus justified because we always refer to the above fixed and well-defined basis.

A.2.1. Biaxial tension

In biaxial tensile tests, we stretch a thin material specimen in two orthogonal directions $\boldsymbol{E}_1$ and $\boldsymbol{E}_2$ by stretches λ_1 and λ_2, Fig. A.14(a). The incompressibility condition determines the stretch in the thickness direction $\boldsymbol{E}_3$: $\lambda_3 = (\lambda_1\lambda_2)^{-1}$. Although λ_1 and λ_2 can be independently controlled in a biaxial tension test (Kawabata et al., 1981), specific test protocols in which λ_1 and λ_2 follow predefined functional relationships are often employed, Fig. A.14(b). For a biaxial tensile test, the deformation gradient in matrix notation and the principal invariants are:

$$\mathbf{F} = \begin{bmatrix}\lambda_1 & 0 & 0\\ 0 & \lambda_2 & 0\\ 0 & 0 & (\lambda_1\lambda_2)^{-2}\end{bmatrix}, \qquad I_1 = \lambda_1^2 + \lambda_2^2 + (\lambda_1\lambda_2)^{-2}, \quad I_2 = \lambda_1^2\lambda_2^2 + \lambda_1^{-2} + \lambda_2^{-2}, \quad I_3 = 1. \tag{A.19}$$

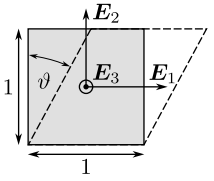
The only non-zero stress components are P_{11} and P_{22} and depending on the employed functional basis for the strain energy function, we arrive at the following expressions:



(a) General biaxial tensile test.

Deformation mode	λ_1	λ_2	λ_3
Uniaxial tension	λ	$\lambda^{-1/2}$	$\lambda^{-1/2}$
Equibiaxial tension	λ	λ	λ^{-2}
Pure shear	λ	1	λ^{-1}

(b) Special biaxial tensile test.



(c) Simple shear test.

Fig. A.14. Homogeneous deformations. (a) In a general biaxial tensile test, a thin incompressible material specimen is stretched in the $\boldsymbol{E}_1$- and $\boldsymbol{E}_2$-directions by two independent stretches λ_1 and λ_2, respectively. The stretch λ_3 in the thickness direction $\boldsymbol{E}_3$ is determined by the incompressibility condition. (b) Common special cases of the general biaxial tensile test are characterized by fixed ratios between the stretches λ_1 and λ_2. (c) In a simple shear test, the top and bottom surfaces of a material specimen are sheared relative to each other by an angle ϑ, while maintaining their initial distance in the $\boldsymbol{E}_2$-direction.

Principal invariant-based strain energy function. Inserting the deformation gradient and first principal invariant of a biaxial tensile deformation from (A.19), along with the incompressibility condition, $\lambda_3 = (\lambda_1\lambda_2)^{-1}$, into (A.3) allows us to determine the hydrostatic pressure p from the plane stress assumption, $\mathrm{P}_{33} = 0$:

$$p = \frac{2}{\lambda_1^2\lambda_2^2}\left[\frac{\partial\hat{\Psi}}{\partial I_1} + \frac{\partial\hat{\Psi}}{\partial I_2}\left(\lambda_1^2 + \lambda_2^2\right)\right], \tag{A.20}$$

Substituting p back into (A.3) results in

$$\mathbf{P} = \begin{bmatrix} 2\left(\lambda_1 - \frac{1}{\lambda_1^3\lambda_2^2}\right)\left(\frac{\partial\hat{\Psi}}{\partial I_1} + \lambda_2^2\frac{\partial\hat{\Psi}}{\partial I_2}\right) & 0 & 0 \\ 0 & 2\left(\lambda_2 - \frac{1}{\lambda_1^2\lambda_2^3}\right)\left(\frac{\partial\hat{\Psi}}{\partial I_1} + \lambda_1^2\frac{\partial\hat{\Psi}}{\partial I_2}\right) & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{A.21}$$

Modified invariant-based strain energy function. Inserting the differentiation relations (A.4) into (A.21) yields the expression for the modified invariant formulation:

$$\mathbf{P} = \begin{bmatrix} 2\left(\lambda_1 - \frac{1}{\lambda_1^3\lambda_2^2}\right)\left(\frac{1}{6\iota_1}\frac{\partial\hat{\Psi}}{\partial\iota_1} + \lambda_2^2\frac{1}{9\iota_2^2}\frac{\partial\hat{\Psi}}{\partial\iota_2}\right) & 0 & 0 \\ 0 & 2\left(\lambda_2 - \frac{1}{\lambda_1^2\lambda_2^3}\right)\left(\frac{1}{6\iota_1}\frac{\partial\hat{\Psi}}{\partial\iota_1} + \lambda_1^2\frac{1}{9\iota_2^2}\frac{\partial\hat{\Psi}}{\partial\iota_2}\right) & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{A.22}$$

Principal stretch-based strain energy function. In a biaxial tensile test of a homogeneous isotropic material, only normal stresses arise, and consequently, the principal directions $\boldsymbol{N}_\alpha$ and $\boldsymbol{n}_\alpha$, and the basis vectors $\boldsymbol{E}_\alpha$ coincide. In this case, the Cartesian stress components directly equate with the principal stresses. Utilizing the plane stress assumption, $\mathrm{P}_{33} = 0$, to determine the hydrostatic pressure p from (A.9) gives:

$$p = \frac{1}{\lambda_1\lambda_2}\frac{\partial\hat{\Psi}}{\partial\lambda_3}, \tag{A.23}$$

therefore, substituting p back into (A.9), gives:

$$\mathbf{P} = \begin{bmatrix} \frac{\partial\hat{\Psi}}{\partial\lambda_1} - \frac{1}{\lambda_1^2\lambda_2}\frac{\partial\hat{\Psi}}{\partial\lambda_3} & 0 & 0 \\ 0 & \frac{\partial\hat{\Psi}}{\partial\lambda_2} - \frac{1}{\lambda_1\lambda_2^2}\frac{\partial\hat{\Psi}}{\partial\lambda_3} & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{A.24}$$

A.2.2. Simple shear

In a simple shear test, we shear the top and bottom surfaces of the specimen relative to each other by the angle $\vartheta = \tan(\gamma)$, while maintaining their initial distance in the $\boldsymbol{E}_2$-direction, Fig. A.14(c). Here, γ denotes the amount of shear. We obtain

$$\mathbf{F} = \begin{bmatrix} 1 & \gamma & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad \lambda_{1,2} = \frac{\sqrt{4+\gamma^2} \pm \gamma}{2}, \quad \lambda_3 = 1, \qquad I_1 = I_2 = 3 + \gamma^2, \quad I_3 = 1. \tag{A.25}$$

Depending on the employed functional basis for the strain energy function, we arrive at the following expressions for the first Piola–Kirchhoff stress tensor.

Principal invariant-based strain energy function. Inserting the deformation gradient and first principal invariant of a simple shear deformation from (A.25), along with the incompressibility condition, $\lambda_3 = (\lambda_1\lambda_2)^{-1} = 1$, into (A.3) allows us to determine the hydrostatic pressure p from the plane stress assumption, $\mathrm{P}_{33} = 0$:

$$p = 2\left[\frac{\partial\hat{\Psi}}{\partial I_1} + \frac{\partial\hat{\Psi}}{\partial I_2}(\gamma^2 + 2)\right]. \tag{A.26}$$

Substituting p back into (A.3) results in

$$\mathbf{P} = \begin{bmatrix} -2\frac{\partial\hat{\Psi}}{\partial I_2}\gamma^2 & 2\gamma\left(\frac{\partial\hat{\Psi}}{\partial I_1} + \frac{\partial\hat{\Psi}}{\partial I_2}\right) & 0 \\ 2\gamma\left(\frac{\partial\hat{\Psi}}{\partial I_1} + \frac{\partial\hat{\Psi}}{\partial I_2}(\gamma^2+1)\right) & -2\frac{\partial\hat{\Psi}}{\partial I_2}\gamma^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{A.27}$$

Modified invariant-based strain energy function. Substituting the differentiation relations (A.4) into (A.27) gives:

$$\mathbf{P} = \begin{bmatrix} -2\frac{1}{9I_2^2}\frac{\partial\hat{\Psi}}{\partial I_2}\gamma^2 & 2\gamma\left(\frac{1}{6I_1}\frac{\partial\hat{\Psi}}{\partial I_1} + \frac{1}{9I_2^2}\frac{\partial\hat{\Psi}}{\partial I_2}\right) & 0 \\ 2\gamma\left(\frac{1}{6I_1}\frac{\partial\hat{\Psi}}{\partial I_1} + \frac{1}{9I_2^2}\frac{\partial\hat{\Psi}}{\partial I_2}(\gamma^2+1)\right) & -2\frac{1}{9I_2^2}\frac{\partial\hat{\Psi}}{\partial I_2}\gamma^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{A.28}$$

Principal stretch-based strain energy function. In a simple shear test, the principal directions $\boldsymbol{N}_\alpha$ and $\boldsymbol{n}_\alpha$ do not align with the basis vectors $\boldsymbol{E}_\alpha$, except in the out-of-plane direction. The principal directions in the reference configuration $\boldsymbol{N}_\alpha$ are obtained by solving the eigenvector problem of $\mathbf{C}$:

$$\boldsymbol{N}_1 = \left[\frac{1}{\sqrt{\lambda_1^2+1}}, \quad \frac{\lambda_1}{\sqrt{\lambda_1^2+1}}, \quad 0\right]^{\mathrm{T}}, \qquad \boldsymbol{N}_2 = \left[\frac{1}{\sqrt{\lambda_2^2+1}}, \quad -\frac{\lambda_2}{\sqrt{\lambda_2^2+1}}, \quad 0\right]^{\mathrm{T}}, \qquad \boldsymbol{N}_3 = \left[0, \quad 0, \quad 1\right]^{\mathrm{T}}. \tag{A.29}$$

Utilizing $(A.7)_2$, we can calculate the principal directions in the current configuration $\boldsymbol{n}_\alpha$:

$$\boldsymbol{n}_1 = \left[\frac{\gamma\lambda_1+1}{\lambda_1\sqrt{\lambda_1^2+1}}, \quad \frac{1}{\sqrt{\lambda_1^2+1}}, \quad 0\right]^{\mathrm{T}}, \qquad \boldsymbol{n}_2 = \left[\frac{-\gamma\lambda_2+1}{\lambda_2\sqrt{\lambda_2^2+1}}, \quad -\frac{1}{\sqrt{\lambda_2^2+1}}, \quad 0\right]^{\mathrm{T}}, \qquad \boldsymbol{n}_3 = \left[0, \quad 0, \quad 1\right]^{\mathrm{T}}. \tag{A.30}$$

Inserting the principal directions $\boldsymbol{N}_\alpha$ and $\boldsymbol{n}_\alpha$ into (A.9), and applying the plane stress assumption, $\mathrm{P}_{33} = 0$, enables us to determine the hydrostatic pressure:

$$p = \frac{\partial\hat{\Psi}}{\partial\lambda_3}. \tag{A.31}$$

Substituting p along with $\boldsymbol{N}_\alpha$ and $\boldsymbol{n}_\alpha$ back into (A.8), and observing that $1 + \gamma\lambda_1 = \lambda_1^2$ and $1 - \gamma\lambda_2 = \lambda_2^2$, leads to:

$$\mathbf{P} = \begin{bmatrix} \frac{1}{\lambda_1^2+1}\frac{\partial\hat{\Psi}}{\partial\lambda_1} + \frac{1}{\lambda_2^2+1}\frac{\partial\hat{\Psi}}{\partial\lambda_2} - \frac{\partial\hat{\Psi}}{\partial\lambda_3} & \frac{\lambda_1^2}{\lambda_1^2+1}\frac{\partial\hat{\Psi}}{\partial\lambda_1} - \frac{\lambda_2^2}{\lambda_2^2+1}\frac{\partial\hat{\Psi}}{\partial\lambda_2} & 0 \\ \frac{1}{\lambda_1^2+1}\frac{\partial\hat{\Psi}}{\partial\lambda_1} - \frac{1}{\lambda_2^2+1}\frac{\partial\hat{\Psi}}{\partial\lambda_2} + \gamma\frac{\partial\hat{\Psi}}{\partial\lambda_3} & \frac{\lambda_1}{\lambda_1^2+1}\frac{\partial\hat{\Psi}}{\partial\lambda_1} + \frac{\lambda_2}{\lambda_2^2+1}\frac{\partial\hat{\Psi}}{\partial\lambda_2} - \frac{\partial\hat{\Psi}}{\partial\lambda_3} & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{A.32}$$

Appendix B. Hyperparameters

Table B.1 lists the hyperparameters of the CKANs from Section 5. For CKANs using the mixed functional basis, the sub-KANs based on principal invariants and principal stretches are likewise parametrized according to Table B.1.

Appendix C. Internal architecture of a principal stretch-based CKAN

Fig. C.1 schematically shows the internal architecture of a principal stretch-based CKAN. To preserve the symmetry of the strain energy function Ψ^{KAN}, parameter sharing is employed.

Table B.1
Hyperparameters of the principal invariant-, modified invariant-, and principal stretch-based (sub-)CKANs used in the numerical examples in Section 5.

Hyperparameter	Value	
	Principal & modified invariant-based	Principal stretch-based
Topology $\mathcal{N}$	[2,3,1]	[1,4,1]
Order of splines	3	3
Grid intervals	1	1
Number of trainable activation functions	9	8
Number of parameters per trainable activation functions	4	4
Optimizer	Adam	Adam
Learning rate	0.001	0.001
Overall regularization magnitude Λ	0.0005	0.0005
L_1 regularization magnitude μ_1	1	1
Entropy regularization magnitude μ_2	1	1

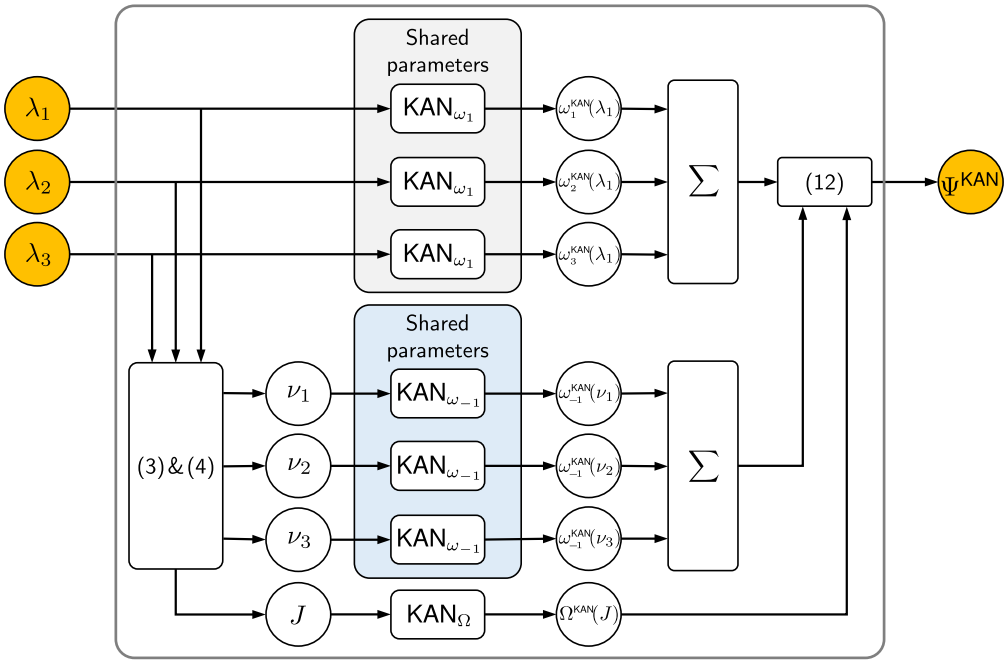


Fig. C.1. Internal architecture of a principal stretch-based Constitutive Kolmogorov–Arnold Network (CKAN): Following the ansatz of separable functions from (12) and (13), the λ_α and the ν_α are processed by distinct KANs, denoted as KAN_{ω_1} and $\mathsf{KAN}_{\omega_{-1}}$. However, to ensure material symmetry (isotropy), all three λ_α have to pass through the same mapping KAN_{ω_1} and all three ν_α through the same mapping KAN_{ω_1}. This is implemented through KANs with shared parameters. For clarity, the feature vector $\boldsymbol{f}$ is not shown, but it would be processed by each KAN, namely KAN_{ω_1}, $\mathsf{KAN}_{\omega_{-1}}$, and KAN_{Ω}.

Appendix D. Normalization terms for a stress-free reference configuration

As outlined in Section 3.2, the stress $\mathbf{S}^{\mathrm{KAN}}$ associated with Ψ^{KAN} does not in general vanish in the undeformed reference configuration. Here, we provide the specific stress normalization terms Ψ^{σ} and the corresponding stress tensors $\mathbf{S}^{\sigma}$ to offset $\mathbf{S}^{\mathrm{KAN}}$ in the reference configuration for each CKAN formulation considered in this work. By construction, Ψ^{σ} vanishes in the undeformed reference configuration.

D.1. Principal invariant-based strain energy function

In view of (A.2), the second Piola–Kirchhoff stress is

$$\mathbf{S}^{\mathbf{KAN}} = 2\frac{\partial \Psi^{\mathrm{KAN}}}{\partial \mathbf{C}} = 2\left[\left(\frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_1} + I_1\frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_2}\right)\mathbf{I} - \frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_2}\mathbf{C} + I_3\frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_3}\mathbf{C}^{-1}\right], \tag{D.1}$$

and becomes in the undeformed reference configuration ($\mathbf{C}=\mathbf{I}$, $I_1 = I_2 = 3$, $I_3 = 1$):

$$\mathbf{S}^{\mathrm{KAN}}|_{\mathbf{C}=\mathbf{I}} = 2\left[\frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_1} + 2\frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_2} + \frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_3}\right]\Big|_{\mathbf{C}=\mathbf{I}}\mathbf{I}. \tag{D.2}$$

To offset this generally non-zero stress, we introduce the stress normalization term

$$\Psi^{\sigma}(J, \boldsymbol{f}) := -\Xi(\boldsymbol{f})(J-1) \tag{D.3}$$

where

$$\Xi(\boldsymbol{f}) := 2\left[\frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_1} + 2\frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_2} + \frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_3}\right]\Big|_{\mathbf{C}=\mathbf{I}}, \tag{D.4}$$

is determined by the derivatives of Ψ^{KAN} with respect to the principal invariants, evaluated in the undeformed reference configuration. However, during training, Ξ continuously adapts as Ψ^{KAN} evolves with each update of the trainable spline parameters and is *not* computed in a post-processing step. Evaluating the corresponding stress contribution

$$\mathbf{S}^{\sigma} = -\Xi\, J\mathbf{C}^{-1} \tag{D.5}$$

for $\mathbf{C} = \mathbf{I}$ confirms that $\mathbf{S}^{\sigma}$ indeed corrects (D.2), ensuring a stress-free reference configuration. For an isotropic incompressible material, the

D.2. Modified invariant-based strain energy function

Following the same reasoning, the stress normalization term for the modified invariant-based formulation is defined as

$$\Psi^{\sigma}(J, \boldsymbol{f}) := -\Xi(\boldsymbol{f})(J-1), \tag{D.6}$$

where

$$\Xi(\boldsymbol{f}) := 2\left[\frac{1}{6}\frac{\partial \Psi^{\mathrm{KAN}}}{\partial \iota_1} + \frac{2}{9}\frac{\partial \Psi^{\mathrm{KAN}}}{\partial \iota_2} + \frac{\partial \Psi^{\mathrm{KAN}}}{\partial I_3}\right]\Big|_{\mathbf{C}=\mathbf{I}}. \tag{D.7}$$

The corresponding stress contribution, ensuring a stress-free reference configuration, is given by

$$\mathbf{S}^{\sigma} = -\Xi J\mathbf{C}^{-1}. \tag{D.8}$$

D.3. Principal stretch-based strain energy function

In view of (A.14), the second Piola–Kirchhoff stress tensor is

$$\mathbf{S}^{\mathrm{KAN}} = \sum_{\alpha=1}^{3}\left[\frac{1}{\lambda_\alpha}\frac{\partial \omega_1^{\mathrm{KAN}}}{\partial \lambda_\alpha} + \frac{1}{\lambda_\alpha^2}\sum_{\beta=1}^{3}\nu_\beta\frac{\partial \omega_{-1}^{\mathrm{KAN}}}{\partial \nu_\beta} - \frac{\nu_\alpha}{\lambda_\alpha^2}\frac{\partial \omega_{-1}^{\mathrm{KAN}}}{\partial \nu_\alpha} + \frac{\nu_\alpha}{\lambda_\alpha}\frac{\partial \Omega^{\mathrm{KAN}}}{\partial J}\right]\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha \tag{D.9}$$

and becomes in the undeformed reference configuration ($\mathbf{C}=\mathbf{I}$, $\lambda_\alpha = \nu_\alpha = J = 1$):

$$\mathbf{S}^{\mathrm{KAN}}|_{\mathbf{C}=\mathbf{I}} = \left[\frac{\partial \omega_1^{\mathrm{KAN}}}{\partial \lambda_\alpha} + 2\frac{\partial \omega_{-1}^{\mathrm{KAN}}}{\partial \nu_\alpha} + \frac{\partial \Omega^{\mathrm{KAN}}}{\partial J}\right]\Big|_{\mathbf{C}=\mathbf{I}}\mathbf{I}, \tag{D.10}$$

where we exploited that $\sum_{\alpha=1}^{3}\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\alpha = \mathbf{I}$. Thus, we define the normalization term

$$\Psi^{\sigma}(J, \boldsymbol{f}) = -\Xi(\boldsymbol{f})(J-1), \tag{D.11}$$

where

$$\Xi(\boldsymbol{f}) := \left[\frac{\partial \omega_1^{\mathrm{KAN}}}{\partial \lambda_\alpha} + 2\frac{\partial \omega_{-1}^{\mathrm{KAN}}}{\partial \nu_\alpha} + \frac{\partial \Omega^{\mathrm{KAN}}}{\partial J}\right]\Big|_{\mathbf{C}=\mathbf{I}}. \tag{D.12}$$

The corresponding stress contribution, ensuring a stress-free reference configuration, is

$$\mathbf{S}^{\sigma} = -\Xi J \mathbf{C}^{-1}. \tag{D.13}$$

Appendix E. Implementation

The computational framework presented here builds on Liu et al. (2024b), implemented in PyTorch, and extended by Polo-Molina et al. (2024), which enables partial monotonicity constraints. Adapting KANs for constitutive modeling of hyperelastic materials required several modifications. Key implementation aspects include designing the loss function, Appendix E.1, enforcing monotonicity constraints, Appendix E.2, and—as discussed in Sections 4.1 and 4.2—applying sparsification and pruning. While we described the general symbolification process already in Section 4.3, we discuss in Appendix E.3 caveats of the symbolification specific to the functional bases considered herein.

E.1. Loss function

The objective of CKANs is, in a first step, to learn the spline parameters that minimize the error between the stress prediction of the CKAN $\mathbf{P}$ and the experimental stress data $\tilde{\mathbf{P}}$. In all numerical examples in Section 5, we consider the special biaxial load cases uniaxial tension (UT) and compression (UC), equibiaxial tension (ET), and pure shear (PS), Fig. A.14(b). Each load case comprises n_a data tuples: $\{(\lambda^a_{1,i}, \lambda^a_{2,i}, \tilde{\mathrm{P}}^a_{11,i})\}_{i=1}^{n_a}$, where $a \in \{\mathrm{UT}, \mathrm{UC}, \mathrm{ET}, \mathrm{PS}\}$. Additionally, n_{SS} data pairs are obtained from simple shear (SS) tests: $\{(\gamma_i, \tilde{P}^{\mathrm{SS}}_{12,i})\}_{i=1}^{n_{\mathrm{SS}}}$.

Then, the objective is to minimize the following general (data) loss function:

$$\mathcal{L}_{\mathrm{data}} = \sum_{a\in\{\mathrm{UT,UC,ET,PS}\}} \frac{1}{n_a} \sum_{i=1}^{n_a} \left(\frac{\mathrm{P}^a_{11,i}(\lambda^a_{1,i}, \lambda^a_{2,i}) - \tilde{\mathrm{P}}^a_{11,i}}{\tilde{\mathrm{P}}^a_{11,i}} \right)^2 + \frac{1}{n_{\mathrm{SS}}} \sum_{i=1}^{n_{\mathrm{SS}}} \left(\frac{\mathrm{P}^{\mathrm{SS}}_{12,i}(\gamma_i) - \tilde{\mathrm{P}}^{\mathrm{SS}}_{12,i}}{\tilde{\mathrm{P}}^{\mathrm{SS}}_{12,i}} \right)^2. \tag{E.1}$$

Often, not all load cases are considered simultaneously during training, but only a subset is used, while the withheld load cases validate the model's generalization capabilities. In addition, the loss function could be augmented with other stress components, if available. However, typically only the P_{11} and P_{12} components are reported for the special biaxial load cases and simple shear, respectively, and used in this work.

E.2. (Partially) monotonic cubic Hermite splines

To maintain a monotonic strain energy function in hyperelastic material modeling, we build on the work by Polo-Molina et al. (2024). This approach extends the Kolmogorov–Arnold Network (KAN) by imposing (partial) monotonicity constraints on the spline activation functions. This enforces a physically meaningful increase in strain energy as strain increases. This monotonicity also mitigates numerical pathologies, such as stress oscillations or material instabilities, thus enhancing both robustness and extrapolation capabilities.

While the original KAN formulation (Liu et al., 2024b) uses B-splines as activation functions, Polo-Molina et al. (2024) proposed cubic Hermite splines and imposed constraints on their control points to ensure monotonicity. Let $\mathbf{X} = \{x_k\}_{k=0}^n$, $\mathbf{Y} = \{y_k\}_{k=0}^n$, and $\mathbf{M} = \{m_k\}_{k=0}^n$ be the knots, their associated control points, and derivatives, respectively. Defining the index set $I = \{0, \dots, n\}$, a cubic Hermite spline p consists of n cubic polynomials such that

$$p(x_k) = y_k, \quad p'(x_k) = m_k, \qquad \forall k \in I. \tag{E.2}$$

The cubic Hermite spline p can be expressed for a segment $I_k = [x_k, x_{k+1}]$ by the restriction

$$p_k(t) := p|_{I_k}(t) = h_{00}(t)\, y_k + h_{10}(t)\,(x_{k+1} - x_k)\, m_k + h_{01}(t)\, y_{k+1} + h_{11}(t)\,(x_{k+1} - x_k)\, m_{k+1}, \tag{E.3}$$

where $t = \frac{x - x_k}{x_{k+1} - x_k}$ and the Hermite basis functions $h_{ij}(t)$ are:

$$h_{00}(t) = 2t^3 - 3t^2 + 1, \qquad h_{10}(t) = t^3 - 2t^2 + t, \tag{E.4}$$

$$h_{01}(t) = -2t^3 + 3t^2, \qquad h_{11}(t) = t^3 - t^2. \tag{E.5}$$

For a monotonically increasing (decreasing) spline, it is necessary that $y_k \le y_{k+1}$ ($y_k \ge y_{k+1}$), $\forall k \in I$. Then, a sufficient condition for p_k being monotonically increasing (decreasing) in I_k is (Polo-Molina et al., 2024; Fritsch and Carlson, 1980):

$$\alpha_k := \frac{m_k}{d_k} \ge 0 \quad \left(\alpha_k := \frac{m_k}{d_k} \le 0\right), \quad \beta_k := \frac{m_{k+1}}{d_k} \ge 0 \quad \left(\beta_k := \frac{m_{k+1}}{d_k} \le 0\right) \quad \text{and} \quad \alpha_k^2 + \beta_k^2 \le 9, \tag{E.6}$$

where $d_k = (y_{k+1} - y_k)/(x_{k+1} - x_k)$ denotes the slope of the secant line between two knots x_{k+1} and x_k. These conditions ensure that the cubic Hermite spline remains monotonic across its entire domain.

During training, the parameters $\mathbf{Y}$ and $\mathbf{M}$ are updated at each optimization step according to the algorithm proposed in Polo-Molina et al. (2024), while satisfying the above-stated conditions. By imposing monotonic behavior, the CKAN framework adheres to essential physical requirements of hyperelastic modeling while maintaining the predictive and interpretive benefits of the KAN

architecture. If no monotonicity constraints are imposed on the inputs, the CKAN automatically reverts to B-splines as in Liu et al. (2024b). For the construction of B-splines we refer to de Boor (1980).

E.3. Functional basis-specific symbolification

When an input vector $\boldsymbol{x}_0^s$ is passed to a CKAN, information propagates from the input layer to the output layer by computing the node outputs $x_{l,i}^s$ which serve as inputs to the activation functions $\phi_{l,j,i}$, cf. (17). The values of $x_{l,i}^s$ and $\phi_{l,j,i}(x_{l,i}^s)$, based on the input vector $\boldsymbol{x}_0^s$, are stored in the corresponding layers during the final forward pass of $\boldsymbol{x}_0^s$ and subsequently serve as sample points for the symbolification process outlined in Section 4.3. Thus, it is crucial to select $\boldsymbol{x}_0^s$ wisely in the final forward pass before symbolizing the activation functions. To this end, we propose dedicated sampling strategies for principal and modified invariant-based CKANs, as well as for principal stretch-based CKANs. For CKANs that rely on a mixed functional basis, these sampling strategies are applied individually to the respective sub-networks.

Principal and modified invariant-based CKANs. When the CKAN is trained using mini-batches, each of which contains data from a single load case or a subset thereof, the symbolification process relies on only a small portion of the training data. Consequently, the symbolic form and the affine parameters derived from this limited sample set may be suboptimal. For CKANs that use principal invariants or modified invariants as the functional basis, one solution is to pass the entire training dataset as a full batch during the final forward pass through the CKAN, immediately before symbolification. This ensures that the symbolification captures a more comprehensive representation of the training data, improving the accuracy and robustness of the symbolic model.

Principal stretch-based CKANs. For principal stretch-based CKANs, passing the entire training set as a full batch does not solve the problem. According to standard `Python` operator precedence, the two sums of the separable strain energy function (13) are evaluated from left to right. Therefore, the univariate strain energy functions $\omega_1(\lambda_\alpha)$ and $\omega_{-1}(\nu_\alpha)$ are always evaluated last with λ_3 and ν_3 as inputs, respectively. For example, in a uniaxial tension test of an incompressible material, as shown in Fig. A.14(a), where the specimen is stretched in the $\boldsymbol{E}_1$-direction by $\lambda_1 \geq 1$, the lateral contractions are $0 < \lambda_3, \lambda_2 \leq 1$. Performing symbolic regression of $\hat{\omega}_1$ using $x_{l,i}$ and $\phi_{l,j,i}(x_{l,i})$ obtained by evaluating the spline activation with $0 < \lambda_3 \leq 1$ is unlikely to yield accurate results if the symbolic forms are later evaluated with $\lambda_1 \geq 1$. To ensure that the symbolification accounts for all λ_α and ν_α present in the training data, we stack all these values and pass them as a full batch to ω_1 and ω_{-1}, respectively. Alternatively, one could identify the minimum and maximum values, $\lambda_{\min} = \min\{\lambda_{\alpha,i}^a\}$ and $\lambda_{\max} = \max\{\lambda_{\alpha,i}^a\}$, from the training dataset and use `torch.linspace` to generate a sequence from $\lambda_{\min}$ to $\lambda_{\max}$ with n_s points. This sequence is then passed to ω_1 before symbolification. This approach allows flexibility in choosing the number of sampling points n_s, independent of the actual size of the training data, which can be advantageous when the derivatives $\phi'^s_{l,j,i}$ in (31) are computed numerically, for instance, using finite differences. The same procedure applies to the area stretches ν_α and the strain energy function ω_{-1}.

After symbolifying an activation function, only its affine parameters remain trainable during subsequent training. However, this can lead to violations of constraints — such as monotonicity — that were imposed on the spline activation function and initially inherited by its symbolic counterpart through constraints on the affine parameters. To address this issue, we took a pragmatic approach and fixed the affine parameters after symbolification.[2] Symbolifying one activation function at a time and continuing training is often beneficial, particularly if the affine parameters are fixed post-symbolification. This strategy allows the remaining splines to adapt to the previously symbolized activation function. Once all activation functions in the network have been symbolified, all the affine parameters can be optimized simultaneously in a final step. Optimizers such as BFGS-B are particularly useful for this final step, as they can enforce constraints directly on the affine parameters. This ensures that any constraints imposed on the strain energy function — such as monotonicity — are satisfied. Incorporating this final optimization step substantially improved the models' overall performance in all the numerical examples considered in Section 5.

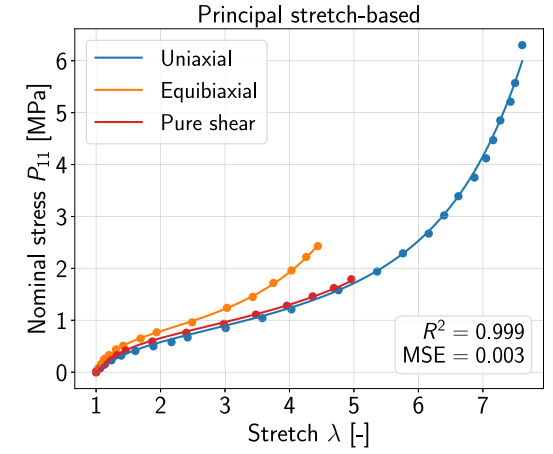
Appendix F. Treloar's and Kawabata's experiments

See Figs. F.1–F.3.

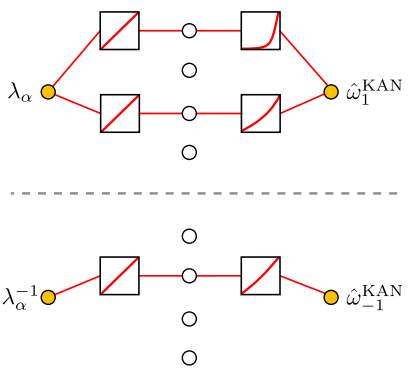
Appendix G. Human brain tissue

See Figs. G.1–G.3.

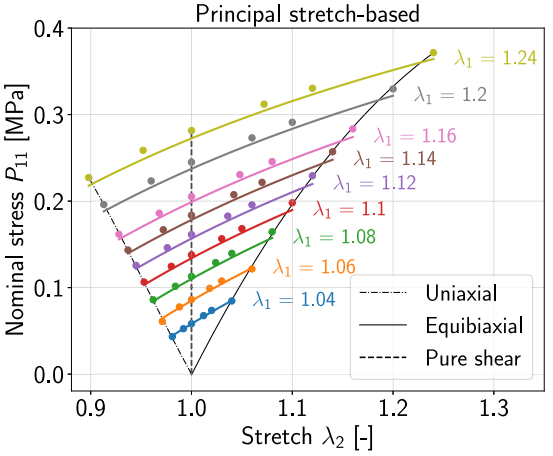
[2] This can be implemented, for example, by defining the `affine` attribute of the `Symbolic_KANLayer` as a `torch.Tensor` instead of a trainable `torch.nn.Parameter`.



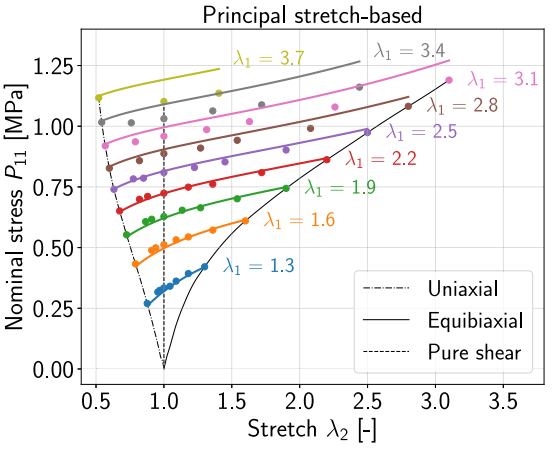
(a) Fitting performance on Treloar's data.



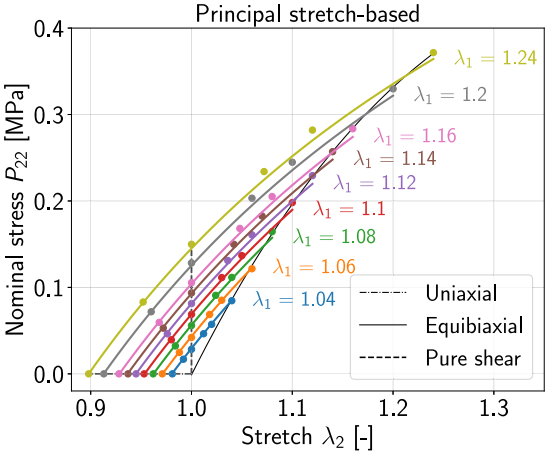
(b) *Principal stretch-based* CKAN architecture.



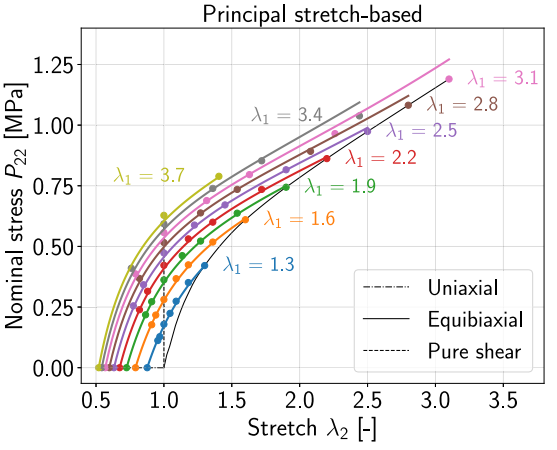
(c) P_{11} generalization: small stretch regime of Kawabata's data.



(d) P_{11} generalization: large stretch regime of Kawabata's data.

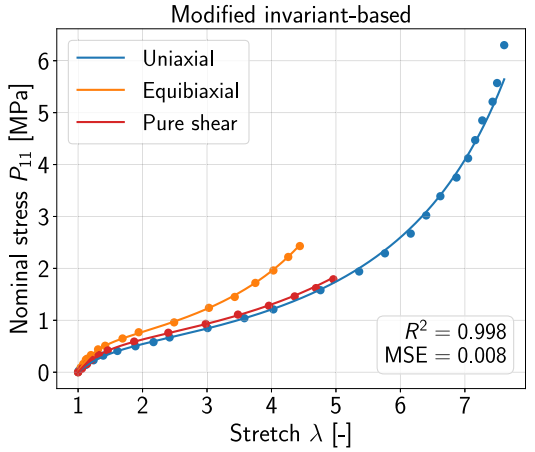


(e) P_{22} generalization: small stretch regime of Kawabata's data.

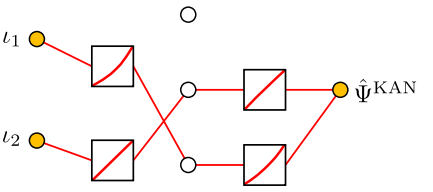


(f) P_{22} generalization: large stretch regime of Kawabata's data.

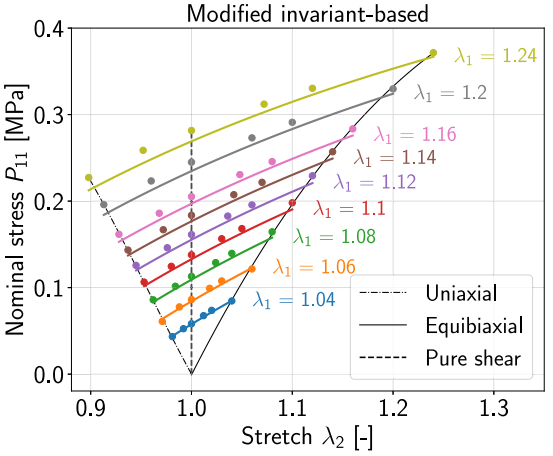
Fig. F.1. Principal stretch-based CKAN for rubber. (a) Fitting performance on Treloar's data (Treloar, 1944b). (b) Architecture of the final CKAN after training; unconnected nodes indicate activation functions identical to zero. (c)–(f) Predictions on the biaxial dataset by Kawabata et al. (1981) using the CKAN trained on Treloar's data only. The solid lines represent the model response. The dots represent the experimental data.



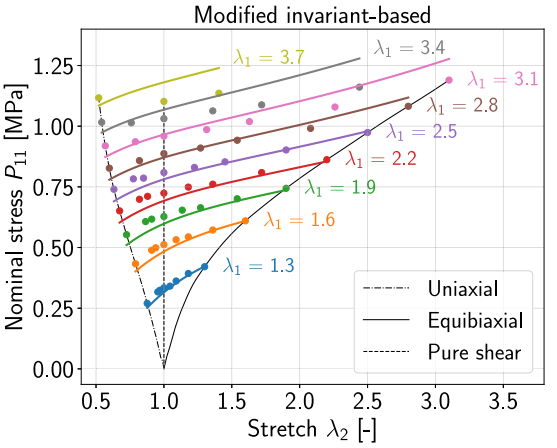
(a) Fitting performance on Treloar's data.



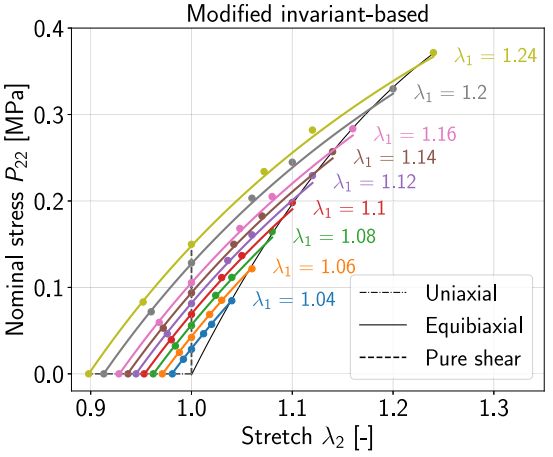
(b) *Modified invariant-based* CKAN architecture.



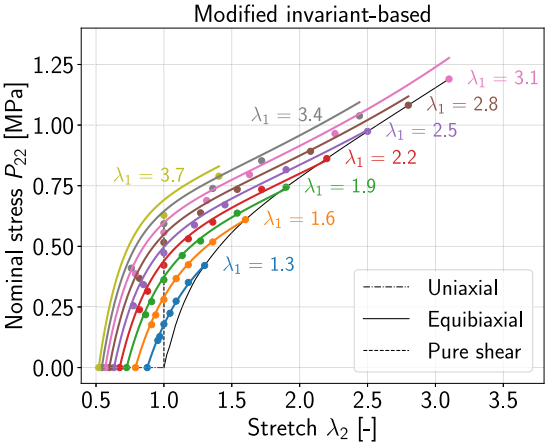
(c) P_{11} generalization: small stretch regime of Kawabata's data.



(d) P_{11} generalization: large stretch regime of Kawabata's data.

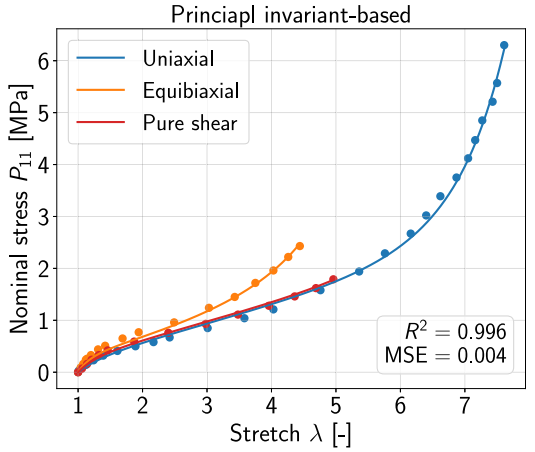


(e) P_{22} generalization: small stretch regime of Kawabata's data.

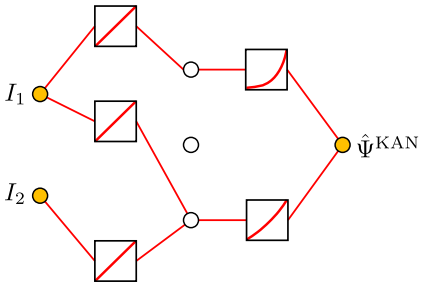


(f) P_{22} generalization: large stretch regime of Kawabata's data.

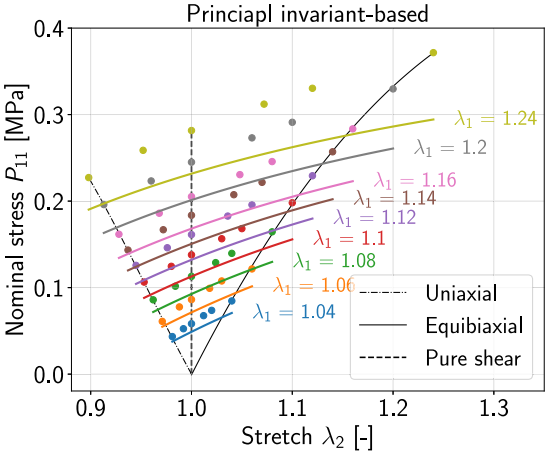
Fig. F.2. Modified invariant-based CKAN for rubber. (a) Fitting performance on Treloar's data (Treloar, 1944b). (b) Architecture of the CKAN after training; unconnected nodes indicate activation functions identical to zero. (c)–(f) Predictions on the biaxial dataset by Kawabata et al. (1981) using the CKAN trained on Treloar's data only. The solid lines represent the model response. The dots represent the experimental data.



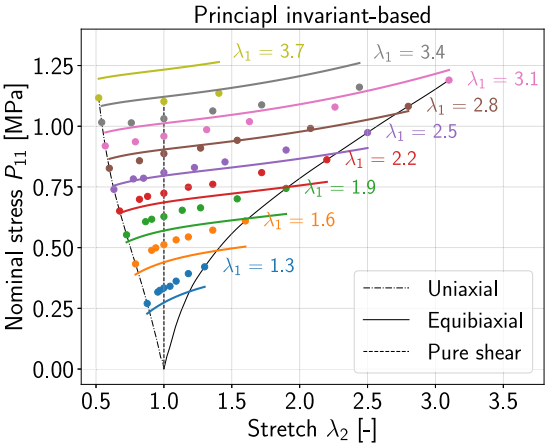
(a) Fitting performance on Treloar's data.



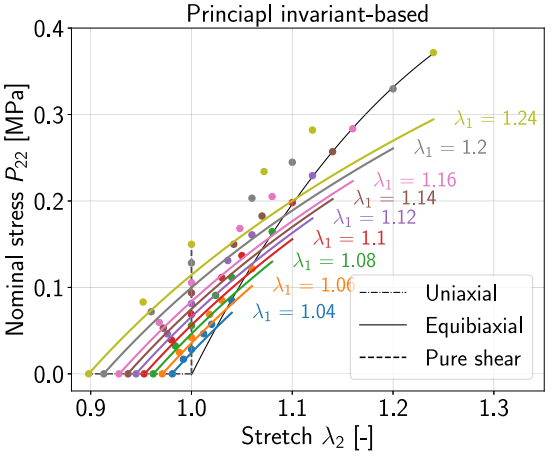
(b) *Principal invariant-based* CKAN architecture.



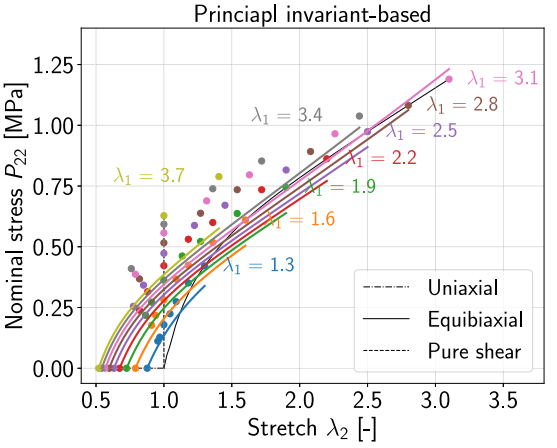
(c) P_{11} generalization: small stretch regime of Kawabata's data.



(d) P_{11} generalization: large stretch regime of Kawabata's data.

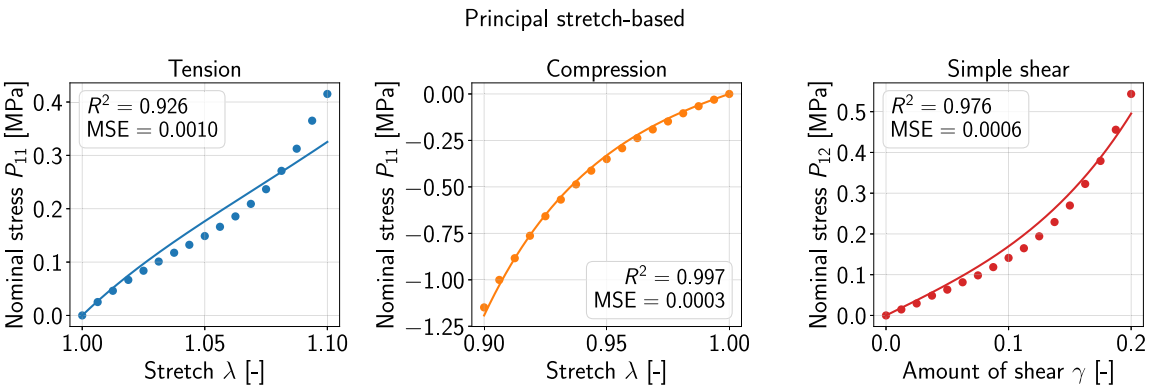


(e) P_{22} generalization: small stretch regime of Kawabata's data.



(f) P_{22} generalization: large stretch regime of Kawabata's data.

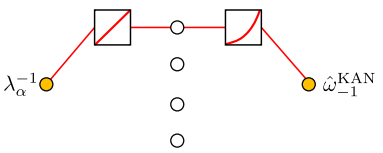
Fig. F.3. Principal invariant-based CKAN for rubber. (a) Fitting performance on Treloar's data (Treloar, 1944b). (b) Architecture of the final CKAN after training; unconnected nodes indicate activation functions identical to zero. (c)–(f) Predictions on the biaxial dataset by Kawabata et al. (1981) using the CKAN trained on Treloar's data only. The solid lines represent the model response. The dots represent the experimental data.



(a) Fitting results; the solid lines represent the model response, the dots represent the experimental data.

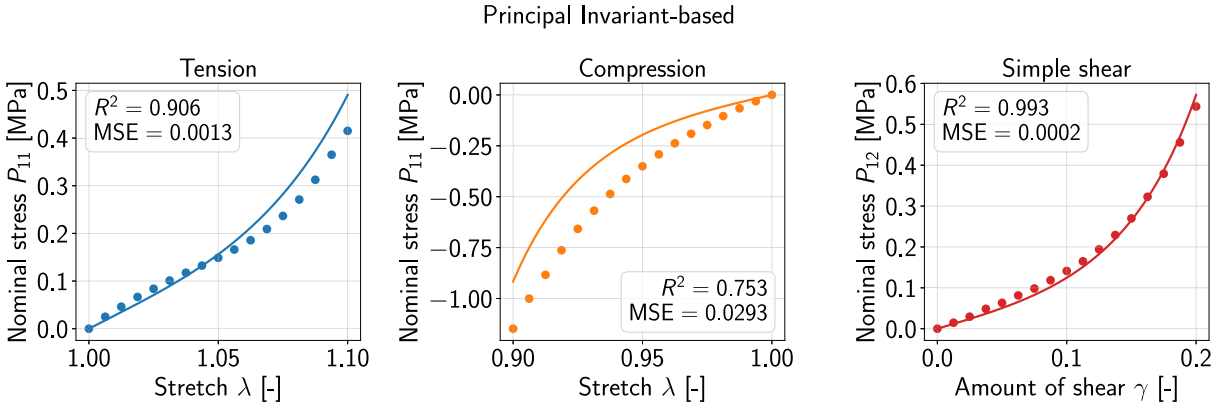
Strain energy function	$\hat{\Psi}^{\mathrm{KAN}} = \sum_{\alpha=1}^{3} \hat{\omega}_1^{\mathrm{KAN}}(\lambda_\alpha) + \sum_{\alpha=1}^{3} \hat{\omega}_{-1}^{\mathrm{KAN}}(\lambda_\alpha^{-1})$ $\hat{\omega}_1^{\mathrm{KAN}}(\lambda_\alpha) = 0, \quad \hat{\omega}_{-1}^{\mathrm{KAN}}(\lambda_\alpha^{-1}) = a\,(\lambda_\alpha^{-1} + b)^{18}$	
Material parameters	a [MPa]	b [-]
	0.01872	-0.04393

(b) Discovered strain energy function and material parameters.



(c) CKAN architecture.

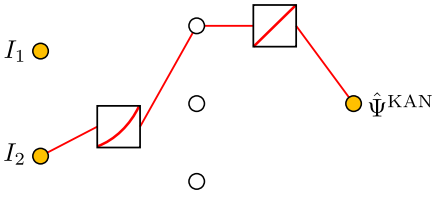
Fig. G.1. Results of the one-term principal stretch-based CKAN. (a) simultaneously fitted to uniaxial tension, compression, and simple shear data of human brain (cortex) tissue data reported in Budday et al. (2017a). (b) Resulting strain energy functions and material parameters. (c) Architecture of the final CKAN after training; unconnected nodes indicate activation functions identical to zero.



(a) Fitting results; the solid lines represent the model response, the dots represent the experimental data.

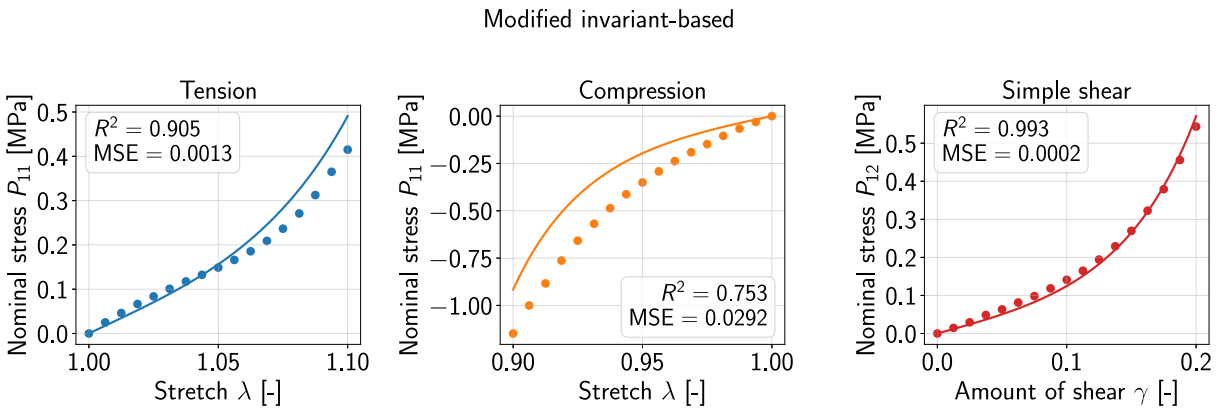
Strain energy function	$\hat{\Psi}^{\mathrm{KAN}}(I_2) = a\,(b\,I_2 - 1)^{24}$	
Material parameters	a [MPa]	b [-]
	$4.26077 \cdot 10^8$	0.45586

(b) Discovered strain energy function and material parameters.



(c) CKAN architecture.

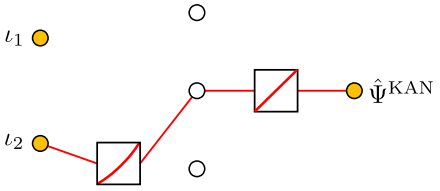
Fig. G.2. Results of the principal invariant-based CKAN. (a) simultaneously fitted to uniaxial tension, compression, and simple shear data of human brain (cortex) tissue data reported in Budday et al. (2017a). (b) Resulting strain energy functions and material parameters. (c) Architecture of the final CKAN after training; unconnected nodes indicate activation functions identical to zero.



(a) Fitting results; the solid lines represent the model response, the dots represent the experimental data.

Strain energy function	$\hat{\Psi}^{\mathrm{KAN}}(\iota_2) = a\,(\iota_2 + b)^{23}$	
Material parameters	a [MPa]	b [-]
	$7.27523 \cdot 10^{22}$	-0.91540128

(b) Discovered strain energy function and material parameters.



(c) CKAN architecture.

Fig. G.3. Results of the modified invariant-based CKAN. (a) simultaneously fitted to uniaxial tension, compression, and simple shear data of human brain (cortex) tissue data reported in Budday et al. (2017a). (b) Resulting strain energy functions and material parameters. (c) Architecture of the final CKAN after training; unconnected nodes indicate activation functions identical to zero.

Data availability

Data will be made available on request.

References

Abdolazizi, Kian P., Linka, Kevin, Cyron, Christian J., 2023. Viscoelastic constitutive artificial neural networks (vCANNs) – A framework for data-driven anisotropic nonlinear finite viscoelasticity. J. Comput. Phys. 499, 112704.

Abdusalamov, Rasul, Hillgärtner, Markus, Itskov, Mikhail, 2023. Automatic generation of interpretable hyperelastic material models by symbolic regression. Internat. J. Numer. Methods Engrg. 124 (9), 2093–2104.

Amores, Víctor J., Benítez, José M., Montáns, Francisco J., 2020. Data-driven, structure-based hyperelastic manifolds: A macro-micro-macro approach to reverse-engineer the chain behavior and perform efficient simulations of polymers. Comput. Struct. 231, 106209.

Anssari-Benam, Afshin, Destrade, Michel, Saccomandi, Giuseppe, 2022. Modelling brain tissue elasticity with the ogden model and an alternative family of constitutive models. Philos. Trans. Ser. A, Math. Phys. Eng. Sci. 380 (2234), 20210325.

Arruda, Ellen M., Boyce, Mary C., 1993. A three-dimensional constitutive model for the large stretch behavior of rubber elastic materials. J. Mech. Phys. Solids 41 (2), 389–412.

As'ad, Faisal, Avery, Philip, Farhat, Charbel, 2022. A mechanics-informed artificial neural network approach in data-driven constitutive modeling. Internat. J. Numer. Methods Engrg. 123 (12), 2738–2759.

Attard, Mario M., Hunt, Giles W., 2004. Hyperelastic constitutive modeling under finite strain. Int. J. Solids Struct. 41 (18–19), 5327–5350.

Bernardi, Laura, Hopf, Raoul., Sibilio, Daniela, Ferrari, Aldo, Ehret, Alexander E., Mazza, Eduardo, 2017. On the cyclic deformation behavior, fracture properties and cytotoxicity of silicone-based elastomers for biomedical applications. Polym. Test. 60, 117–123.

Boehler, Jean-Paul, 1987. In: Boehler, J.P. (Ed.), Applications of Tensor Functions in Solid Mechanics. Springer Vienna, Vienna.

Brunton, Steven L., Kutz, J. Nathan, 2022. Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control. Cambridge University Press.

Brunton, Steven L., Proctor, Joshua L., Kutz, J. Nathan, 2016. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proc. Natl. Acad. Sci. 113 (15), 3932–3937.

Budday, Silvia, Sommer, Gerhard, Birkl, Christoph, Langkammer, Christian, Haybaeck, Johannes, Kohnert, J., Bauer, Melanie, Paulsen, Friedrich P., Steinmann, Paul, Kuhl, Ellen, Holzapfel, Gerhard A., 2017a. Mechanical characterization of human brain tissue. Acta Biomater. 48, 319–340.

Budday, Silvia, Sommer, Gerhard, Haybaeck, Johannes, Steinmann, Paul, Holzapfel, Gerhard A., Kuhl, Ellen, 2017b. Rheological characterization of human brain tissue. Acta Biomater. 60, 315–329.

Carrara, Pietro, De Lorenzis, Laura, Stainier, Laurent, Ortiz, Michael, 2020. Data-driven fracture mechanics. Comput. Methods Appl. Mech. Engrg. 372.

Case, Jennifer C., White, Edward L., Kramer, Rebecca K., 2015. Soft material characterization for robotic applications. Soft Robot. 2 (2), 80–87.

Crespo, José, Latorre, Marcos, Montáns, Francisco Javier, 2017. WYPIWYG hyperelasticity for isotropic, compressible materials. Comput. Mech. 59 (1), 73–92.

Dal, Hüsnü, Denli, Funda Aksu, Açan, Alp Kağan, Kaliske, Michael, 2023. Data-driven hyperelasticity, part I: A canonical isotropic formulation for rubberlike materials. J. Mech. Phys. Solids 179, 105381.
Davidson, Jacob D., Goulbourne, N.C., 2013. A nonaffine network model for elastomers undergoing finite deformations. J. Mech. Phys. Solids 61 (8), 1784–1797.
de Boor, Carl, 1980. A Practical Guide to Splines. vol. 27, Springer New York, NY, p. 325.
Ehret, Alexander E., Itskov, Mikhail, 2007. A polyconvex hyperelastic model for fiber-reinforced materials in application to soft tissues. J. Mater. Sci. 42 (21), 8853–8863.
Ehret, Alexander E., Stracuzzi, Alberto, 2022. Variations on Ogden's model: Close and distant relatives. Philos. Trans. R. Soc. A: Math. Phys. Eng. Sci. 380 (2234).
Flaschel, Moritz, Kumar, Siddhant, De Lorenzis, Laura, 2021. Unsupervised discovery of interpretable hyperelastic constitutive laws. Comput. Methods Appl. Mech. Engrg. 381, 113852.
Franke, Marlon, Klein, Dominik K., Weeger, Oliver, Betsch, Peter, 2023. Advanced discretization techniques for hyperelastic physics-augmented neural networks. Comput. Methods Appl. Mech. Engrg. 416, 116333.
Fritsch, Frederick N., Carlson, Ralph E., 1980. Monotone piecewise cubic interpolation. SIAM J. Numer. Anal. 17 (2), 238–246.
Fuhg, Jan N., Anantha Padmanabha, Govinda, Bouklas, Nikolaos, Bahmani, Bahador, Sun, WaiChing, Vlassis, Nikolaos N., Flaschel, Moritz, Carrara, Pietro, De Lorenzis, Laura, 2024a. A review on data-driven constitutive laws for solids. Arch. Comput. Methods Eng. 1–43.
Fuhg, Jan N., Bouklas, Nikolaos, Jones, Reese E., 2024b. Stress representations for tensor basis neural networks: alternative formulations to Finger–Rivlin–Ericksen. J. Comput. Inf. Sci. Eng. 24 (11).
Fuhg, Jan Niklas, Jones, Reese Edward, Bouklas, Nikolaos, 2024c. Extreme sparsification of physics-augmented neural networks for interpretable model discovery in mechanics. Comput. Methods Appl. Mech. Engrg. 426, 116973.
Fung, Yuan-Cheng, 1981. Biomechanics - Mechanical Properties of Living Tissues. Springer New York, New York, NY.
Ghaboussi, Jamshid, Garrett, James H., Wu, Xiping, 1991. Knowledge-based modeling of material behavior with neural networks. J. Eng. Mech. 117 (1), 132–153.
Goldberg, Carey B., et al., 2024. To do no harm—and the most good—with AI in health care. Nature Med. 30 (3), 623–627.
Hashash, Youssef. M.A., Jung, Sungmoon, Ghaboussi, Jamshid, 2004. Numerical implementation of a neural network based material model in finite element analysis. Internat. J. Numer. Methods Engrg. 59 (7), 989–1005.
Heinrich, Gert, Kaliske, Michael, 1997. Theoretical and numerical formulation of a molecular based constitutive tube-model of rubber elasticity. Comput. Theor. Polym. Sci. 7 (3–4), 227–241.
Holthusen, Hagen, Brepols, Tim, Linka, Kevin, Kuhl, Ellen, 2025. Automated model discovery for tensional homeostasis: Constitutive machine learning in growth and remodeling. Comput. Biol. Med. 186, 109691.
Holthusen, Hagen, Lamm, Lukas, Brepols, Tim, Reese, Stefanie, Kuhl, Ellen, 2024. Theory and implementation of inelastic constitutive artificial neural networks. Comput. Methods Appl. Mech. Engrg. 428, 117063.
Holzapfel, Gerhard A., 2000. Nonlinear Solid Mechanics: a Continuum Approach for Engineering Science. John Wiley & Sons Ltd., p. 456.
Holzapfel, Gerhard A., Gasser, T. Christian, Ogden, Ray W., 2000. A new constitutive framework for arterial wall mechanics and a comparative study of material models. J. Elasticity 61 (1–3), 1–48.
Hospedales, Timothy, Antoniou, Antreas, Micaelli, Paul, Storkey, Amos, 2021. Meta-learning in neural networks: A survey. IEEE Trans. Pattern Anal. Mach. Intell. 44 (9), 5149–5169.
Howard, Amanda A., Jacob, Bruno, Murphy, Sarah H., Heinlein, Alexander, Stinis, Panos, 2024. Finite basis Kolmogorov–Arnold networks: domain decomposition for data-driven and physics-informed problems.
Itskov, Mikhail, 2015. Tensor algebra and tensor analysis for engineers. Mathematical Engineering. Springer Cham, pp. 1–295.
Jiang, Yuting, Wang, Yang, Mishra, Yogendra Kumar, Adelung, Rainer, Yang, Ya, 2018. Stretchable CNTs-ecoflex composite as variable-transmittance skin for ultrasensitive strain sensing. Adv. Mater. Technol. 3 (12), 1800248.
Kalina, Karl A., Brummund, Jörg, Sun, WaiChing, Kästner, Markus, 2025. Neural networks meet anisotropic hyperelasticity: A framework based on generalized structure tensors and isotropic tensor functions. Comput. Methods Appl. Mech. Engrg. 437, 117725.
Kaliske, Michael, Heinrich, Gert, 1999. An extended tube-model for rubber elasticity: Statistical-mechanical theory and finite element implementation. Rubber Chem. Technol. 72 (4), 602–632.
Kawabata, Sueo, Matsuda, Masatoshi, Tei, K., Kawai, Hiromichi, 1981. Experimental survey of the strain energy density function of isoprene rubber vulcanizate. Macromolecules 14 (1), 154–162.
Kearsley, Elliot A., 1989. Note: Strain invariants expressed as average stretches. J. Rheol. 33 (5), 757–760.
Khiêm, Vu Ngoc, Itskov, Mikhail, 2016. Analytical network-averaging of the tube model: Rubber elasticity. J. Mech. Phys. Solids 95, 254–269.
Kirchdoerfer, Trenton, Ortiz, Michael, 2016. Data-driven computational mechanics. Comput. Methods Appl. Mech. Engrg. 304, 81–101.
Kuhl, Ellen, Goriely, Alain, 2024. I too I_2: A new class of hyperelastic isotropic incompressible models based solely on the second invariant. J. Mech. Phys. Solids 188, 105670.
Latorre, Marcos, Montáns, Francisco Javier, 2013. Extension of the sussman-bathe spline-based hyperelastic model to incompressible transversely isotropic materials. Comput. Struct. 122, 13–26.
Latorre, Marcos, Montáns, Francisco Javier, 2014. What-you-prescribe-is-what-you-get orthotropic hyperelasticity. Comput. Mech. 53 (6), 1279–1298.
Le Clézio, Helen, Karapiperis, Konstantinos, Kochmann, Dennis M., 2024. Nonlinear two-scale beam simulations accelerated by thermodynamics-informed neural networks. Extrem. Mech. Lett. 73.
Liao, Zisheng, Hossain, Mokarram, Yao, Xiaohu, 2020. Ecoflex polymer of different shore hardnesses: Experimental investigations and constitutive modelling. Mech. Mater. 144, 103366.
Linden, Lennart, Klein, Dominik K., Kalina, Karl A., Brummund, Jörg, Weeger, Oliver, Kästner, Markus, 2023. Neural networks meet hyperelasticity: A guide to enforcing physics. J. Mech. Phys. Solids 179, 105363.
Linka, Kevin, Cavinato, Cristina, Humphrey, Jay D., Cyron, Christian J., 2022. Predicting and understanding arterial elasticity from key microstructural features by bidirectional deep learning. Acta Biomater. 147, 63–72.
Linka, Kevin, Hillgärtner, Markus, Abdolazizi, Kian P., Aydin, Roland C., Itskov, Mikhail, Cyron, Christian J., 2021. Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning. J. Comput. Phys. 429, 110010.
Linka, Kevin, Hillgärtner, Markus, Itskov, Mikhail, 2018. Fatigue of soft fibrous tissues: Multi-scale mechanics and constitutive modeling. Acta Biomater. 71, 398–410.
Linka, Kevin, Kuhl, Ellen, 2023. A new family of constitutive artificial neural networks towards automated model discovery. Comput. Methods Appl. Mech. Engrg. 403, 115731.
Linka, Kevin, St. Pierre, Sarah R., Kuhl, Ellen, 2023. Automated model discovery for human brain using constitutive artificial neural networks. Acta Biomater. 160, 134–151.
Liu, Ziming, Ma, Pingchuan, Wang, Yixuan, Matusik, Wojciech, Tegmark, Max, 2024a. KAN 2.0: Kolmogorov–Arnold networks meet science.
Liu, Ziming, Wang, Yixuan, Vaidya, Sachin, Ruehle, Fabian, Halverson, James, Soljačić, Marin, Hou, Thomas Y., Tegmark, Max, 2024b. KAN: Kolmogorov–Arnold networks. arXiv preprint arXiv:2404.19756.
Marckmann, Gilles, Verron, Erwan, 2006. Comparison of hyperelastic models for rubber-like materials. Rubber Chem. Technol. 79 (5), 835–858.

McCulloch, Jeremy A., St. Pierre, Skyler R., Linka, Kevin, Kuhl, Ellen, 2024. On sparse regression, lp-regularization, and automated model discovery. Internat. J. Numer. Methods Engrg. 125 (14), e7481.
Mihai, L. Angela, Budday, Silvia, Holzapfel, Gerhard A., Kuhl, Ellen, Goriely, Alain, 2017. A family of hyperelastic models for human brain tissue. J. Mech. Phys. Solids 106, 60–79.
Mullins, Leonard, Tobin, N.R., 1965. Stress softening in rubber vulcanizates. Part I. Use of a strain amplification factor to describe the elastic behavior of filler-reinforced vulcanized rubber. J. Appl. Polym. Sci. 9 (9), 2993–3009.
Ogden, Ray W., 1972. Large deformation isotropic elasticity – on the correlation of theory and experiment for incompressible rubberlike solids. Proc. R. Soc. A 326 (1567), 565–584.
Ogden, Ray W., 1997. Nonlinear Elastic Deformations. Dover Publications.
Pierre, Skyler R.S.t, Rajasekharan, Divya, Darwin, Ethan C., Linka, Kevin, Levenston, Marc E., Kuhl, Ellen, 2023. Discovering the mechanics of artificial and real meat. Comput. Methods Appl. Mech. Engrg. 415, 116236.
Polo-Molina, Alejandro, Alfaya, David, Portela, Jose, 2024. MonoKAN: Certified monotonic Kolmogorov–Arnold network. arXiv preprint arXiv:2409.11078.
Rivlin, Ronald S., Saunders, D.W., 1951. Large elastic deformations of isotropic materials VII. Experiments on the deformation of rubber. Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Sci. 243 (865), 251–288.
Shariff, Mohd H.B.M., 2000. Strain energy function for filled and unfilled rubberlike material. Rubber Chem. Technol. 73 (1), 1–18.
Shukla, Khemraj, Toscano, Juan Diego, Wang, Zhicheng, Zou, Zongren, Karniadakis, George Em, 2024. A comprehensive and FAIR comparison between MLP and kan representations for differential equations and operator networks. Comput. Methods Appl. Mech. Engrg. 431, 117290.
Steinmann, Paul, Hossain, Mokarram, Possart, Gunnar, 2012. Hyperelastic models for rubber-like materials: Consistent tangent operators and suitability for Treloar's data. Arch. Appl. Mech. 82 (9), 1183–1217.
Sussman, Theodore, Bathe, Klaus J., 2009. A model of incompressible isotropic hyperelastic material behavior using spline interpolations of tension-compression test data. Commun. Numer. Methods Eng. 25 (1), 53–63.
Taç, Vahidullah, Rausch, Manuel K., Costabal, Francisco Sahli, Tepole, Adrian Buganza, 2023. Data-driven anisotropic finite viscoelasticity using neural ordinary differential equations. Comput. Methods Appl. Mech. Engrg. 411, 116046.
Thakolkaran, Prakash, Guo, Yaqi, Saini, Shivam, Peirlinck, Mathias, Alheit, Benjamin, Kumar, Siddhant, 2025. Can KAN CANs? Input-convex Kolmogorov–Arnold networks (KANs) as hyperelastic constitutive artificial neural networks (CANs). arXiv preprint arXiv:2503.05617.
Treloar, Leslie R.G., 1944a. The elasticity of a network of long-chain molecules. II. Rubber Chem. Technol. 17 (2), 296–302.
Treloar, Leslie R.G., 1944b. Stress-strain data for vulcanised rubber under various types of deformation. Trans. Faraday Soc. 40, 59–70.
Truesdell, Clifford, 1984. The Elements of Continuum Mechanics. Springer Berlin Heidelberg, Berlin, Heidelberg.
Valanis, Kirk C., Landel, Robert F., 1967. The strain-energy function of a hyperelastic material in terms of the extension ratios. J. Appl. Phys. 38 (7), 2997–3002.
Wang, Yizheng, Sun, Jia, Bai, Jinshuai, Anitescu, Cosmin, Eshaghi, Mohammad Sadegh, Zhuang, Xiaoying, Rabczuk, Timon, Liu, Yinghua, 2024. Kolmogorov Arnold informed neural network: A physics-informed deep learning framework for solving forward and inverse problems based on Kolmogorov Arnold networks.
Wang, Yiqun, Wagner, Nicholas, Rondinelli, James M., 2019. Symbolic regression in materials science. MRS Commun. 9 (3), 793–805.
Wiesheier, Simon, Moreno-Mateos, Miguel Angel, Steinmann, Paul, 2024. Versatile data-adaptive hyperelastic energy functions for soft materials. Comput. Methods Appl. Mech. Engrg. 430, 117208.
Xiang, Yuhai, Zhong, Danming, Wang, Peng, Mao, Guoyong, Yu, Honghui, Qu, Shaoxing, 2018. A general constitutive model of soft elastomers. J. Mech. Phys. Solids 117, 110–122.
Xu, Hongyi, Sin, Funshing, Zhu, Yufeng, Barbič, Jernej, 2015. Nonlinear material design using principal stretches. ACM Trans. Graph. 34 (4), 1–11.
Zhang, Jin Min, Rychlewski, Jan, 1990. Structure tensors for anisotropic solids. Arch. Mech. 42 (3), 267–277.