TU Delft

Delft University of Technology

On automated model discovery and a universal material subroutine for hyperelastic materials

Peirlinck, Mathias; Linka, Kevin; Hurtado, Juan A.; Kuhl, Ellen

DOI
10.1016/j.cma.2023.116534

Publication date
2024

Document Version
Final published version

Published in
Computer Methods in Applied Mechanics and Engineering

Citation (APA)
Peirlinck, M., Linka, K., Hurtado, J. A., & Kuhl, E. (2024). On automated model discovery and a universal material subroutine for hyperelastic materials. *Computer Methods in Applied Mechanics and Engineering*, *418*, Article 116534. https://doi.org/10.1016/j.cma.2023.116534

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

https://www.openaccess.nl/en/you-share-we-take-care

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

Computer Methods in Applied Mechanics and Engineering 418 (2024) 116534

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

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

On automated model discovery and a universal material subroutine for hyperelastic materials

Mathias Peirlinck [a], Kevin Linka [b], Juan A. Hurtado [c], Ellen Kuhl [b,*]

[a] Department of Biomechanical Engineering, Delft University of Technology, Delft, The Netherlands
[b] Department of Mechanical Engineering, Stanford University, Stanford, CA, United States
[c] Dassault Systèmes, Providence, RI, United States

ARTICLE INFO

Dataset link: https://github.com/LivingMatterLab/CANN

Keywords:
Automated model discovery
Constitutive neural networks
Constitutive modeling
Hyperelasticity
Material subroutine

ABSTRACT

Constitutive modeling is the cornerstone of computational and structural mechanics. In a finite element analysis, the constitutive model is encoded in the material subroutine, a function that maps local strains onto stresses. This function is called within every finite element, at each integration point, within every time step, at each Newton iteration. Today's finite element packages offer large libraries of material models to choose from. However, the scientific criteria for appropriate model selection remain highly subjective and prone to user bias. Here we fully automate the process of model selection, autonomously discover the best model and parameters from experimental data, encode all possible discoverable models into a single material subroutine, and seamlessly integrate this universal material subroutine into a finite element analysis. We prototype this strategy for incompressible, isotropic, hyperelastic soft matter systems that we characterize through a combination of twelve possible terms. These terms feature the first and second invariants, raised to the first and second powers, embedded in the identity, exponential, and logarithmic functions, generating $2^{2\times2\times3} = 4096$ models in total. We demonstrate how to integrate these models into a single universal material subroutine that features the classical neo Hooke, Blatz Ko, Mooney Rivlin, Demiray, Gent, and Holzapfel models as special cases. Finite element simulations with our new universal material subroutine show that it specializes well to these widely used models, generalizes well to newly discovered models, and agrees excellently with both experimental data and previous simulations. It also performs well within realistic finite element simulations and accurately predicts stress concentrations in the human brain for six different head impact scenarios. We anticipate that integrating automated model discovery into a universal material subroutine will generalize naturally to more complex compressible, anisotropic, inelastic materials and to other nonlinear finite element platforms. Replacing dozens of individual material subroutines by a single universal material subroutine that is populated directly via automated model discovery – entirely without human interaction – makes finite element analyses more accessible, more robust, and less vulnerable to human error. This could forever change how we simulate materials and structures.

Our source code, data, and examples are available at https://github.com/LivingMatterLab.

* Corresponding author.
E-mail address: ekuhl@stanford.edu (E. Kuhl).

https://doi.org/10.1016/j.cma.2023.116534
Received 17 July 2023; Received in revised form 29 September 2023; Accepted 9 October 2023
Available online 17 October 2023
0045-7825/© 2023 Elsevier B.V. All rights reserved.

1. Motivation

Material modeling lies at the heart of a finite element analysis and selecting the appropriate material model is key to a successful finite element simulation [1]. A material model takes the local strains as input and calculates the stresses and their derivatives as output [2,3]. Nonlinear finite element programs evaluate the material model locally, within every finite element, at each integration point, within every time step, at each Newton iteration [4,5]. The local stresses and their derivatives then enter the global force vector and stiffness matrix to calculate the nodal displacements [1]. Modern finite element packages typically provide a comprehensive suite of built-in material models – linear, polynomial, exponential, or logarithmic – with dozens of models to choose from [6–10]. This raises the question how we can select the best model and, probably more importantly, to which extent can we remove user bias throughout this selection process?

Admittedly, selecting the appropriate material model is a difficult task. This is especially true for unexperienced users or experienced scientists from other disciplines. For hyperelastic materials alone, commercial finite element packages offer the neo Hooke [11], Blatz Ko [12], Mooney Rivlin [13,14], Yeoh [15], Gent [16], Demiray [17], Holzapfel [18], Ogden [19], and Valanis Landel [20] models, and continue to add new models as new releases emerge. To complicate matters, most finite element packages offer their users the flexibility to define their own custom-designed *user material subroutines* [21]. A user material subroutine is a modular software component that empowers the user to define and simulate complex material behaviors that cannot be captured by standard built-in material models. For example, our group has recently characterized different types of artificial meat and discovered material models that have never been used in any material library [22]. By implementing our own material subroutine, we can not only accurately model this newly discovered complex material behavior, but also design and functionalize new materials [23–25]. This customization enhances the fidelity and accuracy of the simulation and enables the analysis of cutting-edge engineering problems for which standard material models fall short [22,26,27]. With this added flexibility in mind, do we now have to implement a new material subroutine every time we study a new material? And how do we discover the appropriate functional form that best describes the material behavior?

Recently, a trend has emerged to autonomously discover the model and parameters that best describe a specific material from experimental data, without any prior domain knowledge or user bias [28,29]. There are several different strategies to achieve this. Most of them harness the power and robustness of algorithms developed for machine learning [30]. While some discover models that are interpretable, others do not.

Non-interpretable approaches closely follow traditional neural networks and typically discover functions of rectified linear unit, softplus, or hyperbolic tangent type [31]. The first representative of this category uses *tensor basis Gaussian process regression*, a special type of regression for isotropic hyperelastic materials that harnesses the representation theorem to a priori ensures objectivity [32]. In a rather abstract sense, it learns a 3×3 mapping that maps the three isotropic invariants onto the three coefficients of the stress tensor representation [33]. The second uses invariant-based *constitutive artificial neural networks* that a priori satisfy thermodynamic consistency by learning a free energy function from which it derives the stress [34,35]. The third uses *neural ordinary differential equations*, special neural networks that a priori satisfy objectivity and polyconvexity by directly learning the derivatives of the free energy function that enter the stress definition [36,37]. While these approaches are straightforward, provide an excellent approximation of the data, and can be integrated manually within finite element software packages [31,38], they learn non-interpretable models and parameters and teach us little about the underlying material.

Interpretable approaches discover models that are made up of a library of functional building blocks that resemble traditional constitutive models. The first representative of this category uses *sparse regression* and adopts unsupervised learning to discover interpretable models from a feature library of candidate functions [39,40]. The second uses *symbolic regression* and genetic programming to discover mathematical expressions for invariant-based models in the form of rooted trees [41]. Here we combine features of all five approaches: We use a custom-designed invariant-based constitutive artificial neural network that a priori satisfies objectivity, thermodynamic consistency, and polyconvexity and autonomously discovers a free energy function that features popular constitutive terms and parameters with a clear physical interpretation [42–44]. In practice, interpretable models are limited by their functional form. This implies that they generally approximate data less perfectly than non-interpretable models. At the same time, interpretable models are a generalization of popular existing constitutive models that – by design – translate smoothly into user material subroutines for a finite element analysis.

The objective of this work is to integrate *automated model discovery* into the finite element workflow by creating a single *universal material subroutine* that seamlessly incorporates thousands of possible constitutive models. Here we prototype this strategy for incompressible, isotropic, hyperelastic soft matter systems, but expect that it will generalize naturally to compressible, anisotropic, inelastic materials. While we motivate this material subroutine from constitutive neural networks [43], the concept generalizes well to material models discovered via symbolic regression [41] or sparse regression or from feature libraries [39]. In Section 2, we briefly summarize the governing kinematic and constitutive equations. In Section 3, we introduce our constitutive neural network for automated model discovery. In Section 4, we translate all possible models of our network into a universal material subroutine and illustrate its pseudocode within the invariant-based `UANISOHYPER_INV` environment of the finite element package Abaqus. In Section 5, we illustrate the features of our user material subroutine by means of three types of examples: four benchmarks with popular constitutive models, two benchmarks with newly discovered models, and six realistic finite element simulations. We discuss our results in Section 6 and close with a brief conclusion and outlook in Section 7.

2. Governing equations

To set the stage, we briefly summarize the governing kinematic and constitutive equations [2] and specialize the general equations to the homogeneous deformation modes of uniaxial tension, uniaxial compression, and simple shear [43].

2.1. Kinematics

We consider finite deformations and introduce the deformation map $\boldsymbol{\varphi}$ that maps material particles $\boldsymbol{X}$ from the undeformed configuration to particles, $\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X})$, in the deformed configuration [5]. To characterize relative deformations, we introduce the deformation gradient $\boldsymbol{F}$, the gradient of the deformation map $\boldsymbol{\varphi}$ with respect to the undeformed coordinates $\boldsymbol{X}$, and its Jacobian J,

$$\boldsymbol{F} = \nabla_{\boldsymbol{X}} \boldsymbol{\varphi} \qquad \text{with} \qquad J = \det(\boldsymbol{F}) > 0\,. \tag{1}$$

In the following, we consider isotropic materials and introduce the three principal invariants I_1, I_2, I_3 and their derivatives $\partial_{\boldsymbol{F}} I_1$, $\partial_{\boldsymbol{F}} I_2$, $\partial_{\boldsymbol{F}} I_3$,

$$\begin{aligned} I_1 &= \boldsymbol{F} : \boldsymbol{F} & & & \partial_{\boldsymbol{F}} I_1 &= 2\,\boldsymbol{F} \\ I_2 &= \tfrac{1}{2}\,[\,I_1^2 - [\,\boldsymbol{F}^{\mathrm{t}} \cdot \boldsymbol{F}\,] : [\,\boldsymbol{F}^{\mathrm{t}} \cdot \boldsymbol{F}\,]\,] & \text{with} & & \partial_{\boldsymbol{F}} I_2 &= 2\,[\,I_1\,\boldsymbol{F} - \boldsymbol{F} \cdot \boldsymbol{F}^{\mathrm{t}} \cdot \boldsymbol{F}\,] \\ I_3 &= \det(\boldsymbol{F}^{\mathrm{t}} \cdot \boldsymbol{F}) = J^2 & & & \partial_{\boldsymbol{F}} I_3 &= 2\,I_3\,\boldsymbol{F}^{-\mathrm{t}}\,. \end{aligned} \tag{2}$$

For isotropic, perfectly incompressible materials, the third invariant always remains identical to one, $I_3 = J^2 = 1$. This reduces the set of invariants to two, I_1 and I_2.

Tension and compression. For the special homogeneous deformation of uniaxial tension and compression, we apply a stretch λ in the loading direction, $F_{11} = \lambda_1 = \lambda$. For an isotropic, perfectly incompressible material, we can determine the stretches orthogonal to the loading direction, $F_{22} = \lambda_2 = \lambda^{-1/2}$ and $F_{33} = \lambda_3 = \lambda^{-1/2}$, from the constant third invariant, $I_3 = \lambda_1^2\,\lambda_2^2\,\lambda_3^2 = 1$. The deformation gradient takes a diagonal form, $\boldsymbol{F} = \mathrm{diag}\{\ \lambda, \lambda^{-1/2}, \lambda^{-1/2}\ \}$, and introduces the following first and second invariants and their derivatives,

$$I_1 = \lambda^2 + \frac{2}{\lambda} \quad \text{and} \quad I_2 = 2\lambda + \frac{1}{\lambda^2} \quad \text{with} \quad \frac{\partial I_1}{\partial \lambda} = 2\left[\lambda - \frac{1}{\lambda^2}\right] \quad \text{and} \quad \frac{\partial I_2}{\partial \lambda} = 2\left[1 - \frac{1}{\lambda^3}\right]. \tag{3}$$

Shear. For the special homogeneous deformation of simple shear, we apply a shear strain γ in one direction, $F_{12} = \gamma$. For an isotropic, perfectly incompressible material with $F_{11} = F_{22} = F_{33} = 1$, we obtain the following explicit expressions for the first and second invariants and derivatives,

$$I_1 = 3 + \gamma^2 \quad \text{and} \quad I_2 = 3 + \gamma^2 \quad \text{with} \quad \frac{\partial I_1}{\partial \lambda} = 2\,\gamma \quad \text{and} \quad \frac{\partial I_2}{\partial \lambda} = 2\,\gamma\,. \tag{4}$$

2.2. Constitutive equations

Constitutive equations relate a stress like the Piola or nominal stress $\boldsymbol{P}$, the force per undeformed area, to a deformation measure like the deformation gradient $\boldsymbol{F}$. For a hyperelastic material that satisfies the second law of thermodynamics, the Piola stress, $\boldsymbol{P} = \partial\psi(\boldsymbol{F})/\partial\boldsymbol{F}$, is the derivative of the free energy $\psi(\boldsymbol{F})$ with respect to the deformation gradient $\boldsymbol{F}$, modified by a pressure term, $-p\,\boldsymbol{F}^{-\mathrm{t}}$, that ensures perfect incompressibility [5],

$$\boldsymbol{P} = \frac{\partial \psi}{\partial \boldsymbol{F}} - p\,\boldsymbol{F}^{-\mathrm{t}}\,. \tag{5}$$

Here, the hydrostatic pressure, $p = -\frac{1}{3}\,\boldsymbol{P} : \boldsymbol{F}$, acts as a Lagrange multiplier that that we determine from the boundary conditions. Instead of formulating the free energy function directly in terms of the deformation gradient $\psi(\boldsymbol{F})$, it proofs convenient to express it in terms of the invariants, $\psi(I_1, I_2)$, to yield the following explicit representation of the Piola stress,

$$\boldsymbol{P} = \frac{\partial \psi}{\partial I_1}\frac{\partial I_1}{\partial \boldsymbol{F}} + \frac{\partial \psi}{\partial I_2}\frac{\partial I_2}{\partial \boldsymbol{F}} - p\,\boldsymbol{F}^{-\mathrm{t}} = 2\left[\frac{\partial \psi}{\partial I_1} + I_1\frac{\partial \psi}{\partial I_2}\right]\boldsymbol{F} - 2\frac{\partial \psi}{\partial I_2}\boldsymbol{F} \cdot \boldsymbol{F}^{\mathrm{t}} \cdot \boldsymbol{F} - p\,\boldsymbol{F}^{-\mathrm{t}}\,. \tag{6}$$

Tension and compression. For the special homogeneous deformation of uniaxial tension and compression, we evaluate the nominal uniaxial stress P_{11} using the general stress–stretch relationship for perfectly incompressible materials, $P_{ii} = [\partial\psi/\partial I_1]\,[\partial I_1/\partial\lambda_i] + [\partial\psi/\partial I_2]\,[\partial I_2/\partial\lambda_i] - [1/\lambda_i]\,p$, for $i = 1,2,3$ with the invariants in tension and compression from Eq. (3). Here, p denotes the hydrostatic pressure that we determine from the zero stress condition in the transverse directions, $P_{22} = 0$ and $P_{33} = 0$, as $p = [2/\lambda]\,\partial\psi/\partial I_1 + [2\lambda + 2/\lambda^2]\,\partial\psi/\partial I_2$. This results in the following explicit uniaxial stress–stretch relation,

$$P_{11} = 2\left[\frac{\partial \psi}{\partial I_1} + \frac{1}{\lambda}\frac{\partial \psi}{\partial I_2}\right]\left[\lambda - \frac{1}{\lambda^2}\right]. \tag{7}$$

Shear. For the special homogeneous deformation of simple shear, we evaluate the nominal shear stress P_{12} using the general stress–stretch relationship for perfectly incompressible materials with the invariants for shear from Eq. (4). This results in the following explicit shear stress–strain relation,

$$P_{12} = 2\left[\frac{\partial \psi}{\partial I_1} + \frac{\partial \psi}{\partial I_2}\right]\gamma\,. \tag{8}$$

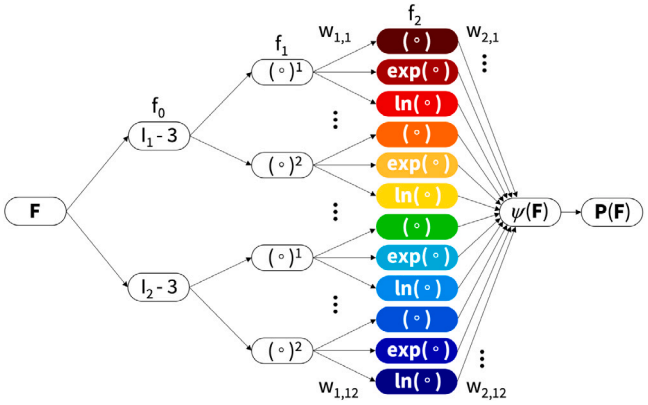

Fig. 1. Constitutive neural network for isotropic, perfectly incompressible, hyperelastic materials. The network has two hidden layers with four and twelve nodes and 24 weights. It takes the deformation gradient $\boldsymbol{F}$ as input and computes the first and second invariants $[I_1-3]$ and $[I_2-3]$. The first layer generates powers $(\circ)^1$ and $(\circ)^2$ of the two invariants and multiplies them by the network weights $w_{1,1..12}$. The second layer applies the identity $(\circ)$, the exponential function, $(\exp(\circ)-1)$, and the natural logarithm, $(-\ln(1-(\circ)))$, to these powers, multiplies them by the network weights $w_{2,1..12}$ and sums them up to calculate the strain energy function $\psi(\boldsymbol{F})$, which defines the Piola stress, $\boldsymbol{P}=\partial\psi/\partial\boldsymbol{F}$. The network is selectively connected by design to a priori satisfy the condition of polyconvexity.

3. Neural network modeling

Motivated by these kinematic and constitutive considerations, we reverse-engineer a family of invariant-based neural networks that satisfy the conditions of thermodynamic consistency, material objectivity, material symmetry, incompressibility, constitutive restrictions, and polyconvexity by design [2,5]. Yet, instead of building these constraints into the loss function of the neural network [45,46], we hardwire them directly into our network input, output, architecture, and activation functions [42–44] to explicitly satisfy the fundamental laws of physics.

Fig. 1 illustrates our constitutive neural network for isotropic, perfectly incompressible, hyperelastic materials. The network has two hidden layers with four and twelve nodes and a total of 24 weights. It takes the deformation gradient $\boldsymbol{F}$ as input and computes the first and second invariants $[I_1-3]$ and $[I_2-3]$. The first layer applies activation functions $f_{1,1}$ and $f_{1,2}$ by generating the first and second powers $(\circ)^1$ and $(\circ)^2$ of the two invariants, and multiplies them by the network weights $w_{1,1..12}$. The second layer applies activation functions $f_{2,1}$ and $f_{2,2}$ and $f_{2,3}$ by generating the identity $(\circ)$, the exponential function, $(\exp(\circ)-1)$, and the natural logarithm, $(-\ln(1-(\circ)))$, from these powers, and multiplies them by the network weights $w_{2,1..12}$. The sum of all twelve terms defines the strain energy function $\psi(\boldsymbol{F})$, from which the network calculates its output, the Piola stress, $\boldsymbol{P}=\partial\psi/\partial\boldsymbol{F}$. Importantly, the network is only selectively connected to a priori satisfy the condition of polyconvexity. The set of equations for this network takes the following explicit representation,

$$\begin{aligned}\psi(I_1,I_2) = {} & w_{2,1}\ w_{1,1}\ [I_1-3] + w_{2,2}\ [\exp(w_{1,2}\ [I_1-3]\ )-1] - w_{2,3}\ \ln(1-w_{1,3}\ [I_1-3]\ ) \\ & + w_{2,4}\ w_{1,4}\ [I_1-3]^2 + w_{2,5}\ [\exp(w_{1,5}\ [I_1-3]^2)-1] - w_{2,6}\ \ln(1-w_{1,6}\ [I_1-3]^2) \\ & + w_{2,7}\ w_{1,7}\ [I_2-3] + w_{2,8}\ [\exp(w_{1,8}\ [I_2-3]\ )-1] - w_{2,9}\ \ln(1-w_{1,9}\ [I_2-3]\ ) \\ & + w_{2,10}\,w_{1,10}\ [I_2-3]^2 + w_{2,11}\ [\exp(w_{1,11}\ [I_2-3]^2)-1] - w_{2,12}\ \ln(1-w_{1,12}\ [I_2-3]^2)\,.\end{aligned} \tag{9}$$

Here one of the first two weights of each row becomes redundant, and we can reduce the set of network parameters from 24 to 20, $\mathbf{w}=[(w_{1,1}w_{2,1}),w_{1,2},w_{2,2},w_{1,3},w_{2,3},(w_{1,4}w_{2,4}),w_{1,5},w_{2,5},w_{1,6},w_{2,6},(w_{1,7}w_{2,7}),w_{1,8},w_{2,8},w_{1,9},w_{2,9},(w_{1,10}w_{2,10}),w_{1,11},w_{2,11},w_{1,12},w_{2,12}]$. Using the second law of thermodynamics, we can derive an explicit expression for the Piola stress from Eq. (6), $\boldsymbol{P}=\partial\psi/\partial I_1\cdot\partial I_1/\partial\boldsymbol{F}+\partial\psi/\partial I_2\cdot\partial I_2/\partial\boldsymbol{F}-p\,\boldsymbol{F}^{-\mathrm{t}}$,

$$\begin{aligned}\boldsymbol{P} = {} & [\,w_{2,1}\ w_{1,1} + w_{2,2}\ w_{1,2}\ \exp(w_{1,2}\ [I_1-3]\ ) + w_{2,3}\ w_{1,3}\ /[\,1-w_{1,3}\ [I_1-3]\ ] \\ & + 2\,[I_1-3]\,[\,w_{2,4}\ w_{1,4} + w_{2,5}\ w_{1,5}\ \exp(w_{1,5}\ [I_1-3]^2)] + w_{2,6}\ w_{1,6}\ /[\,1-w_{1,6}\ [I_1-3]^2\ ]]\ \partial I_1/\partial\boldsymbol{F} \\ & + [\,w_{2,7}\ w_{1,7} + w_{2,8}\ w_{1,8}\ \exp(w_{1,8}\ [I_2-3]\ ) + w_{2,9}\ w_{1,9}\ /[\,1-w_{1,9}\ [I_2-3]\ ] \\ & + 2\,[I_2-3]\,[\,w_{2,10}w_{1,10} + w_{2,11}w_{1,11}\ \exp(w_{1,11}\ [I_2-3]^2)] + w_{2,12}w_{1,12}\ /[\,1-w_{1,12}\ [I_2-3]^2\ ]]\ \partial I_2/\partial\boldsymbol{F},\end{aligned} \tag{10}$$

and correct it by the pressure term, $-p\,\boldsymbol{F}^{-\mathrm{t}}$, with $p=-\frac{1}{3}\boldsymbol{P}:\boldsymbol{F}$. The constitutive neural network learns its weights, $\boldsymbol{\theta}=\{w_{1..2,1..12}\}$, by minimizing a loss function L that penalizes the error between the model we want to discover and the experimental data. We characterize this error as the mean squared error, the L_2-norm of the difference between model $\boldsymbol{P}(\boldsymbol{F}_i)$ and data $\hat{\boldsymbol{P}}_i$, divided by the number of training points n_{trn},

$$L(\boldsymbol{\theta};\boldsymbol{F}) = \frac{1}{n_{\mathrm{trn}}}\sum_{i=1}^{n_{\mathrm{trn}}} \|\,\boldsymbol{P}(\boldsymbol{F}_i)-\hat{\boldsymbol{P}}_i\,\|^2 \rightarrow \min\,. \tag{11}$$

We train the network by minimizing the loss function (11) and learn the network weights $\theta = \{w_{1..2,1..12}\}$ using the ADAM optimizer, a robust adaptive algorithm for gradient-based first-order optimization. To comply with physical constraints, we constrain all weights to always remain non-negative, $w_{i,j} \geq 0$.

4. Universal material subroutine

Our objective is to create a seamless simulation pipeline from experiment – via discovered model and parameters – to simulation. To smoothly integrate our discovered model and parameters, we create a universal material subroutine that translates the local deformation, for example in the form of the deformation gradient $\boldsymbol{F}$, into the current stress, for example the Piola stress $\boldsymbol{P}$. This subroutine operates on the integration point level. Conveniently, some finite element codes provide the option to define a material subroutine that works directly with the strain invariants and returns the first and second derivatives of the free energy function to calculate the stresses and their derivatives [21]. Towards this goal, we express the free energy function ψ from Eq. (9) in the following abstract form.

$$\psi = \sum_{k=1}^{n} w_{2,k}\, f_{2,k}\left(f_{1,j}\left(f_{0,i}\right); w_{1,k}\right) \qquad \text{with} \qquad f_2 = \begin{Bmatrix} w_1\,(\circ) \\ \exp(w_1\,(\circ)) - 1 \\ -\ln(1 - w_1\,(\circ)) \\ \vdots \end{Bmatrix} \quad f_1 = \begin{Bmatrix} (\circ)^1 \\ (\circ)^2 \\ (\circ)^3 \\ \vdots \end{Bmatrix} \quad f_0 = \begin{Bmatrix} [\,I_1 - 3\,] \\ [\,I_2 - 3\,] \\ [\,I_3 - 1\,] \\ \vdots \end{Bmatrix} \tag{12}$$

Here f_0 maps the deformation gradient $\boldsymbol{F}$ onto a set of shifted invariants, $[I_1 - 3]$, $[I_2 - 3]$, $[I_3 - 1]$, $[I_4 - 1]$, $[I_5 - 1]$, that ensure that the strain energy function is zero in the underformed reference configuration, f_1 raises these invariants to the first, second, or any higher order power, $(\circ)^1$, $(\circ)^2$, $(\circ)^3$, and f_2 applies the identity $(\circ)$, exponential, $(\exp(\circ) - 1)$, logarithm, $-\ln(1 - (\circ))$, or any other thermodynamically admissible function to these powers. The material subroutine can then calculate the Piola stress following Eq. (10) by using the derivatives of the individual functions f_0, f_1, f_2.

$$\boldsymbol{P} = \sum_{k=1}^{n} w_{2,k}\, \frac{\partial f_{2,k}}{\partial(\circ)}\, \frac{\partial f_{1,j}}{\partial(\circ)}\, \frac{\partial f_{0,i}}{\partial \boldsymbol{F}} \qquad \text{with} \qquad \frac{\partial f_2}{\partial(\circ)} = \begin{Bmatrix} w_1 \\ w_1 \exp(w_1\,(\circ)) \\ w_1\,/(1 - w_1\,(\circ)) \\ \vdots \end{Bmatrix} \quad \frac{\partial f_1}{\partial(\circ)} = \begin{Bmatrix} 1 \\ 2\,(\circ)^1 \\ 3\,(\circ)^2 \\ \vdots \end{Bmatrix} \quad \frac{\partial f_0}{\partial(\circ)} = \begin{Bmatrix} \partial I_1/\partial \boldsymbol{F} \\ \partial I_2/\partial \boldsymbol{F} \\ \partial I_3/\partial \boldsymbol{F} \\ \vdots \end{Bmatrix} \tag{13}$$

In implicit finite element codes that rely on a global Newton Raphson iteration, the material subroutine also calculates the second derivative for the tangent moduli.

$$\frac{\mathrm{d}\boldsymbol{P}}{\mathrm{d}\boldsymbol{F}} = \sum_{k=1}^{n} w_{2,k} \left[\left[\frac{\partial^2 f_{2,k}}{\partial(\circ)^2} \left[\frac{\partial f_{1,j}}{\partial(\circ)} \right]^2 + \frac{\partial f_{2,k}}{\partial(\circ)} \frac{\partial^2 f_{1,j}}{\partial(\circ)^2} \right] \frac{\partial f_{0,i}}{\partial \boldsymbol{F}} \otimes \frac{\partial f_{0,i}}{\partial \boldsymbol{F}} + \frac{\partial f_{2,k}}{\partial(\circ)} \frac{\partial f_{1,j}}{\partial(\circ)} \frac{\partial^2 f_{0,i}}{\partial \boldsymbol{F} \otimes \partial \boldsymbol{F}} \right]$$

$$\text{with} \qquad \frac{\partial^2 f_2}{\partial(\circ)^2} = \begin{Bmatrix} 0 \\ w_1^2 \exp(w_1(\circ)) \\ w_1^2/(1 - w_1(\circ))^2 \\ \vdots \end{Bmatrix} \quad \frac{\partial^2 f_1}{\partial(\circ)^2} = \begin{Bmatrix} 0 \\ 2 \\ 6\,(\circ) \\ \vdots \end{Bmatrix} \quad \frac{\partial^2 f_0}{\partial(\circ)^2} = \begin{Bmatrix} \partial^2 I_1/\partial \boldsymbol{F}^2 \\ \partial^2 I_2/\partial \boldsymbol{F}^2 \\ \partial^2 I_3/\partial \boldsymbol{F}^2 \\ \vdots \end{Bmatrix} \tag{14}$$

Fig. 2 illustrates the free energy and stress contributions of our universal material subroutine for isotropic, perfectly incompressible, hyperelastic materials. Its free energy function ψ is made up of twelve terms, based on the two invariants, $[I_1 - 3]$ and $[I_2 - 3]$, taken to the first and second powers, $(\circ)^1$ and $(\circ)^2$, and embedded in the identity $(\circ)$, the exponential function, $(\exp(\circ) - 1)$, and the natural logarithm, $(-\ln(1 - (\circ)))$. The odd rows illustrate these twelve terms, $\psi(I_1)$ and $\psi(I_2)$, for the special case of tension and compression with $0.5 \leq \lambda \leq 2.0$, top four rows, and shear with $-2.0 \leq \gamma \leq +2.0$, bottom four rows. The even rows underneath each free energy term illustrate the corresponding Piola stress, $\boldsymbol{P} = \partial\psi/\partial\boldsymbol{F}$, as the derivative of the free energy with respect to the deformation gradient, for the special case of tension and compression P_{11}, top, and shear, P_{12}, bottom.

Our discovered model translates seamlessly into a modular universal material subroutine within any finite element environment. Here we illustrate this translation by means of the Abaqus finite element analysis software suite [21]. We leverage the `UANISOHYPER_INV` user subroutine to introduce our discovered hyperelastic material strain energy function (9) or (12) in terms of the discovered pairs of network weights and activation functions. Specifically, our user subroutine defines the strain energy density, `UA(1)` $= \psi$, and the arrays of its first derivatives, `UI1(NINV)` $= \partial\psi/\partial I_i$, and second derivatives, `UI2(NINV*(NINV+1)/2)` $= \partial^2\psi/\partial I_i \partial I_j$, with respect to the invariants. With a view towards a potential generalization to anisotropic materials, we introduce an array of generalized invariants, `aInv(NINV)` $= I*_i$ with $i = 1, \ldots,$`NINV`, where `NINV` is the total number of isotropic and anisotropic invariants, and adopt the `UANISOHYPER_INV` invariant numbering,

$$\begin{array}{lcllcl} I_1 & \rightarrow & I*_i;\ i = 1 & I_{4(\alpha\beta)} & \rightarrow & I*_i;\ i = 4 + 2\,(\alpha - 1) + \beta\,(\beta - 1)) \\ I_2 & \rightarrow & I*_i;\ i = 2 & I_{5(\alpha\beta)} & \rightarrow & I*_i;\ i = 5 + 2\,(\alpha - 1) + \beta\,(\beta - 1)) \\ J & \rightarrow & I*_i;\ i = 3\,. & & & \end{array}$$

Here, $I_{4(\alpha\beta)} = \boldsymbol{n}_{0\alpha} \cdot (\boldsymbol{F}^{\mathrm{t}} \cdot \boldsymbol{F}) \cdot \boldsymbol{n}_{0\beta}$ and $I_{5(\alpha\beta)} = \boldsymbol{n}_{0\alpha} \cdot (\boldsymbol{F}^{\mathrm{t}} \cdot \boldsymbol{F})^2 \cdot \boldsymbol{n}_{0\beta}$ are anisotropic invariants in terms of the unit vectors $\boldsymbol{n}_{0\alpha}$ and $\boldsymbol{n}_{0\beta}$ that represent the directions of anisotropy in the reference configuration. For example, a transversely isotropic behavior with a single direction of anisotropy, $\alpha = \beta = 1$, introduces two additional invariants, $I_4 = I_{4(11)}$ and $I_5 = I_{5(11)}$ [44].

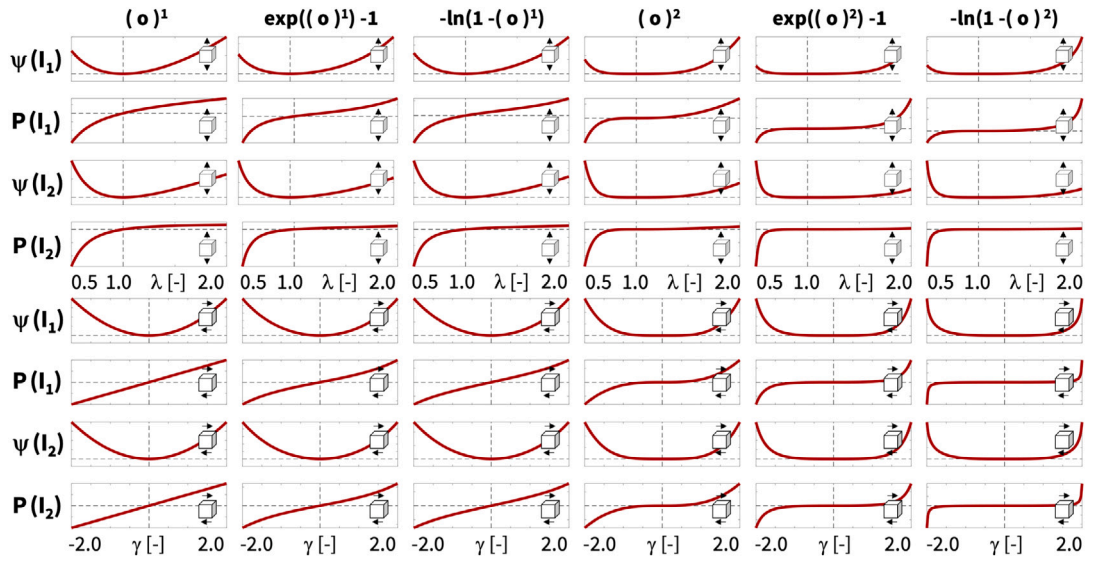

Fig. 2. Universal material subroutine. Free energy and stress contributions. The free energy function ψ of our isotropic, perfectly incompressible, hyperelastic material is made up of powers $(\circ)^1$ and $(\circ)^2$ of the two invariants $[I_1 - 3]$ and $[I_2 - 3]$, embedded in the identity $(\circ)$, the exponential function, $(\exp(\circ) - 1)$, and the natural logarithm, $(-\ln(1 - (\circ)))$, odd rows. The stress $\boldsymbol{P} = \partial\psi/\partial\boldsymbol{F}$ is the derivative of the free energy with respect to the deformation gradient, even rows. The twelve terms represent the twelve nodes of the constitutive neural network in Fig. 1 for tension and compression with $0.5 \leq \lambda \leq 2.0$, top four rows, and shear with $-2.0 \leq \gamma \leq +2.0$, bottom four rows.

Algorithm 1: Pseudocode for universal material subroutine `UANISOHYPER_INV`

```
subroutine UANISOHYPER_INV(aInv,UA,UI1,UI2)
    // initialize energy, its derivatives, reference configuration, and model parameters
    set initial array values for UA, UI1, UI2;
    set reference configuration array UANISOHYPER_INV;
    retrieve discovered parameter table UNIVERSAL_TAB;
    // evaluate all n nodes, i.e., all rows in parameter table
    for k in n do
        // extract invariant, activation functions, and weights
        extract invariant kf0(k);
        extract activation functions kf1(k) and kf2(k);
        extract weights w1(k) and w2(k);
        // compute invariant relative to reference configuration
        xInv = aInv(kf0(k))-3;
        // update energy and its derivatives UA, UI1, UI2
        call uCANN(xInv,kf1(k),kf2(k),w1(k),w2(k),UA,UI1,UI2;)
    // return updated arrays
    return UA, UI1, UI2
```

Algorithm 1 presents the `UANISOHYPER_INV` pseudocode that describes how we compute the `UA(1)`, `UI1(NINV)`, and `UI2(NINV*(NINV+1)/2)` arrays at the integration point level during a finite element analysis. In short, we begin by initializing all relevant arrays and read the activation functions $kf_{1,k}$ and $kf_{2,k}$ and weights $w_{1,k}$ and $w_{2,k}$ of the n color-coded nodes of our constitutive neural network in Fig. 1 from our user-defined parameter table `UNIVERSAL_TAB`. Next, for each node, we evaluate its row in the parameter table `UNIVERSAL_TAB` and additively update the strain energy density function `UA`, its first derivative `UI1`, and its second derivative `UI2` using the invariants `xInv` relative to the reference configuration.

Algorithm 2: Pseudocode to update energy and its derivatives of UANISOHYPER_INV

```
subroutine uCANN(xInv,kf1,kf2,w1,w2,UA,UI1,UI2)
    // process first network layer - determine f1,df1,ddf1
    w0 = 1;
    call uCANN_h1(xInv,w0,kf1,f1,df1,ddf1);
    // process second network layer - determine f2,df2,ddf2
    call uCANN_h2(f1,w1,kf2,f2,df2,ddf2);
    // update energy UA and its derivatives UI1,UI2
    UA  = UA  + w2 * f2;
    UI1 = UI1 + w2 * df2*df1;
    UI2 = UI2 + w2 *(ddf2*df1*df1 + df2*ddf1);
    return UA, UI1, UI2
```

Algorithm 2 details the additive update of the free energy UA and its first and second derivatives UI1 and UI2 within the user material subroutine uCANN.

Algorithm 3: Pseudocode to evaluate first network layer of UANISOHYPER_INV

```
subroutine uCANN_h1(x,w,kf,f,df,ddf)
    // calculate first layer output f,df,ddf for activation function kf
    if kf = 1 then
        f = w * x;
        df = w * 1;
        ddf = w * 0;
    else if kf = 2 then
        f = w**2 * x**2;
        df = w**2 * 2*x;
        ddf = w**2 * 2;
    return f,df,ddf
```

Algorithm 4: Pseudocode to evaluate second network layer of UANISOHYPER_INV

```
subroutine uCANN_h2(x,kf,w,f,df,ddf)
    // calculate second layer output f,df,ddf for activation function kf
    if kf = 1 then
        f = w * x;
        df = w * 1;
        ddf = w * 0;
    else if kf = 2 then
        f = exp(w*x)-1;
        df = w * exp(w*x);
        ddf = w**2 * exp(w*x);
    else if kf = 3 then
        f = -ln(1-w*x);
        df = w / (1-w*x);
        ddf = w**2 / (1-w*x)**2;
    return f,df,ddf
```

Algorithms 3 and 4 provide the pseudocode for the two subroutines uCANN_h1 and uCANN_h2 that evaluate the first and second network layers for each network node with its discovered activation functions and weights. In Abaqus FEA, we define our model parameters in a parameter table. Each row of this parameter table represents one of the color-coded nodes in Fig. 1 and consists of five terms: an integer kf0 that defines the index of the pseudo-invariant xInv, two integers kf1 and kf2 that define the indices of the first- and second-layer activation functions, and two float values w1 and w2 that define the weights of the first and second

layers. We declare this input format using the following parameter table type definition in the UNIVERSA_PARAM_TYPES.INC file.

```
*PARAMETER TABLE TYPE, name="UNIVERSAL_TAB", parameters=5
INTEGER, ,"Index of Pseudo-Invariant, kf0,o"
INTEGER, ,"Index of first hidden layer activation function, kf1,o"
INTEGER, ,"Index of second hidden layer activation function, kf2,o"
FLOAT , ,"Weight of first hidden layer, w1,o"
FLOAT , ,"Weight of second hidden layer, w2,o"
```

Within Abaqus FEA, we include the parameter table type definition using

```
*INCLUDE, INPUT=UNIVERSAL_PARAM_TYPES.INC
```

at the beginning of the input file. We activate our user-defined material model through the command

```
*ANISOTROPIC HYPERELASTIC, USER, FORMULATION=INVARIANT
```

followed by the discovered parameter table entries. For a fully activated constitutive neural network without any zero weights, the header and the twelve rows of this table reads as follows, where terms with zero weight can simply be excluded from the list.

```
*PARAMETER TABLE,  TYPE="UNIVERSAL_TAB"
1,1,1,w1,1 ,w2,1        1,1,2,w1,2 ,w2,2        1,1,3,w1,3 ,w2,3        1,2,1,w1,4 ,w2,4
1,2,2,w1,5 ,w2,5        1,2,3,w1,6 ,w2,6        2,1,1,w1,7 ,w2,7        2,1,2,w1,8 ,w2,8
2,1,3,w1,9 ,w2,9        2,2,1,w1,10 ,w2,10      2,2,2,w1,11 ,w2,11      2,2,3,w1,12 ,w2,12
```

The first index of each row selects between the first and second invariants, I_1 or I_2, the second index raises them to linear or quadratic powers, $(\circ)^1$ or $(\circ)^2$, and the third index selects between the identity, exponential, or logarithmic function, $(\circ)$, $(\exp(\circ)-1)$, or $(-\ln(1-(\circ)))$. Importantly, our user-defined material subroutine is universal by design. Combinations of the 2×12 weights naturally introduce popular and widely used material models as special cases. Here, for illustrative purposes, we only highlight examples in terms of the first and second invariants I_1 and I_2. However, we can easily expand our user-defined material subroutine to include the third invariant I_3 or any combination of the $I_{4\alpha\beta}$ and $I_{5\alpha\beta}$ invariants according to the invariant numbering scheme NINV. Moreover, the modular structure of our material subroutine facilitates a straightforward addition of additional first- and second-layer activation functions $kf1$ and $kf2$ within the uCANN_h1 and uCANN_h2 subroutines, or even completely novel layers with additional activation functions uCANN_h* within the hierarchical uCANN subroutine in Algorithm 2.

5. Results

We illustrate the features of our new user material subroutine in terms of three types of examples: First, we benchmark it with four popular constitutive models, demonstrate how to create the parameter tables for these models, and compare the simulations against the experimental data for gray matter tissue. Second, we benchmark it with two newly discovered models, create their parameter tables, and compare the simulations against both gray and white matter experiments. Finally, we demonstrate how it generalizes to realistic finite element simulations in terms of six different head impact simulations.

5.1. Benchmarking with popular constitutive models.

To demonstrate that our universal material subroutine includes popular constitutive models as special cases, we benchmark our subroutine with four widely used models, translate their network weights $w_{1,\bullet}$ and $w_{2,\bullet}$ of the first and second layers into their model parameters, provide the material table for the input file to our material subroutine, and compare each simulation to experimental data [47]. Table 1 summarizes our discovered non-zero gray matter weights $w_{1,\bullet}$ and $w_{2,\bullet}$ and model parameters μ, μ_1, μ_2, a, b, α, β when training our network with combined tension, compression, and shear data [43] from human gray matter experiments [47].

Neo Hooke model. The neo Hooke model [11] is the simplest of all models. It has a free energy function that is constant in the first invariant, $[\, I_1 - 3 \,]$, scaled by the shear modulus μ. We recover it as a special case from our network free energy (9) as

$$\psi = \tfrac{1}{2}\,\mu\,[\, I_1 - 3 \,] \qquad \text{where} \qquad \mu = 2\, w_{1,1} w_{2,1}\,. \tag{15}$$

The neo Hooke model translates into the following material table for our universal material subroutine,

```
*PARAMETER TABLE, TYPE="UNIVERSAL_TAB"
1,1,1,w1,1 ,w2,1
```

and activates the first term of our model.

Blatz Ko model. The Blatz Ko model [12] has a free energy function that depends on the second and third invariants, $[\, I_2 - 3 \,]$ and $[\, I_3 - 1 \,]$, scaled by the shear modulus μ as $\psi = \frac{1}{2}\mu\,[\, I_2/I_3 + 2\,\sqrt{I_3} - 5 \,]$. For perfectly incompressible materials, $I_3 = 1$, we recover it as a special case of the network free energy (9) as

$$\psi = \tfrac{1}{2}\,\mu\,[\, I_2 - 3 \,] \qquad \text{where} \qquad \mu = 2\, w_{1,7} w_{2,7}\,. \tag{16}$$

Table 1
Neo Hooke, Blatz Ko, Mooney Rivlin, Demiray, Gent, and Holzapfel models and parameters. Discovered non-zero gray matter weights $w_{1,\bullet}$ and $w_{2,\bullet}$ and model parameters μ, μ_1, μ_2, a, b, α, β for training with combined tension, compression, and shear data [43] from human gray matter experiments [47].

	neo Hooke ten+com+shr		**Blatz Ko ten+com+shr**		**Mooney Rivlin ten+com+shr**		**Demiray ten+com+shr**		**Gent ten+com+shr**		**Holzapfel ten+com+shr**	
	gray matter $n = 15, 17, 35$		**gray matter** $n = 15, 17, 35$		**gray matter** $n = 15, 17, 35$		**gray matter** $n = 15, 17, 35$		**gray matter** $n = 15, 17, 35$		**gray matter** $n = 15, 17, 35$	
	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]
$w_{\bullet,1}$	0.7880	1.1522	–	–	0.0026	0.4128	–	–	–	–	–	–
$w_{\bullet,2}$	–	–	–	–	–	–	1.0529	0.8760	–	–	–	–
$w_{\bullet,3}$	–	–	–	–	–	–	–	–	1.8399	0.4782	–	–
$w_{\bullet,5}$	–	–	–	–	–	–	–	–	–	–	4.1833	4.7548
$w_{\bullet,7}$	–	–	1.4156	0.6726	2.2122	0.4253	–	–	–	–	–	–
	$\mu = 1.8159$kPa		$\mu = 1.9043$kPa		$\mu_1 = 0.0021$kPa $\mu_2 = 1.8817$kPa		$a = 1.8447$kPa $b = 1.0529$		$\alpha = 1.7597$kPa $\beta = 1.8399$		$a = 39.7815$kPa $b = 4.1833$	

The Blatz Ko model translates into the following material table for our universal material subroutine,

```
*PARAMETER TABLE, TYPE="UNIVERSAL_TAB"
2,1,1,w_{1,7},w_{2,7}
```

and activates the seventh term of our model.

Mooney Rivlin model. The Mooney Rivlin model [13,14] is a combination of both free energy functions (15) and (16). It accounts for the first and second invariants, $[\,I_1 - 3\,]$ and $[\,I_2 - 3\,]$, scaled by the moduli μ_1 and μ_2 that sum up to the overall shear modulus, $\mu = \mu_1 + \mu_2$. We recover it as a special case of the network free energy (9) as

$$\psi = \tfrac{1}{2}\,\mu_1\,[\,I_1 - 3\,] + \tfrac{1}{2}\,\mu_2\,[\,I_2 - 3\,] \qquad \text{where} \qquad \mu_1 = 2\,w_{1,1}w_{2,1} \;\text{ and }\; \mu_2 = 2\,w_{1,7}w_{2,7}\,. \tag{17}$$

The Mooney Rivlin model translates into the following material table for our universal material subroutine,

```
*PARAMETER TABLE, TYPE="UNIVERSAL_TAB"
1,1,1,w_{1,1},w_{2,1}        2,1,1,w_{1,7},w_{2,7}
```

and activates the first and seventh terms of our model.

Demiray model. The Demiray model [17] uses linear exponentials of the first invariant, $[I_1 - 3]$, in terms of two parameters a and b. We recover it as a special case of the network free energy (9) as

$$\psi = \frac{1}{2}\,\frac{a}{b}\,[\,\exp(\,b\,[\,I_1 - 3\,]\,) - 1\,] \qquad \text{where} \qquad a = 2\,w_{1,2}w_{2,2} \;\text{ and }\; b = w_{1,2}\,. \tag{18}$$

The Demiray model translates into the following material table for our universal material subroutine,

```
*PARAMETER TABLE, TYPE="UNIVERSAL_TAB"
1,1,2,w_{1,2},w_{2,2}
```

and activates the second term of our model.

Gent model. The Gent model [16] uses linear logarithms of the first invariant, $[I_1 - 3]$, in terms of two parameters α and β. We recover it as a special case of the network free energy (9) as

$$\psi = -\frac{1}{2}\,\frac{\alpha}{\beta}\,\ln(\,1 - \beta\,[\,I_1 - 3\,]\,) \qquad \text{where} \qquad \alpha = 2\,w_{1,3}w_{2,3} \;\text{ and }\; \beta = w_{1,3}\,. \tag{19}$$

The Gent model translates into the following material table for our universal material subroutine,

```
*PARAMETER TABLE, TYPE="UNIVERSAL_TAB"
1,1,3,w_{1,3},w_{2,3}
```

and activates the third term of our model.

Holzapfel model. The Holzapfel model [18] uses quadratic exponentials, typically of the fourth invariant, which we adapt here for the first invariant, $[\,I_1 - 3\,]$, in terms of two parameters a and b. We recover it as a special case of the network free energy (9) as

$$\psi = \frac{1}{2}\,\frac{a}{b}\,[\,\exp(\,b[\,I_1 - 3\,]^2\,) - 1\,] \quad \text{where} \quad a = 2\,w_{1,5}w_{2,5} \text{ and } b = w_{1,5}\,. \tag{20}$$

The Holzapfel model translates into the following material table for our universal material subroutine,

```
*PARAMETER TABLE, TYPE="UNIVERSAL_TAB"
1,2,2,w_{1,5},w_{2,5}
```

and activates the fifth term of our model.

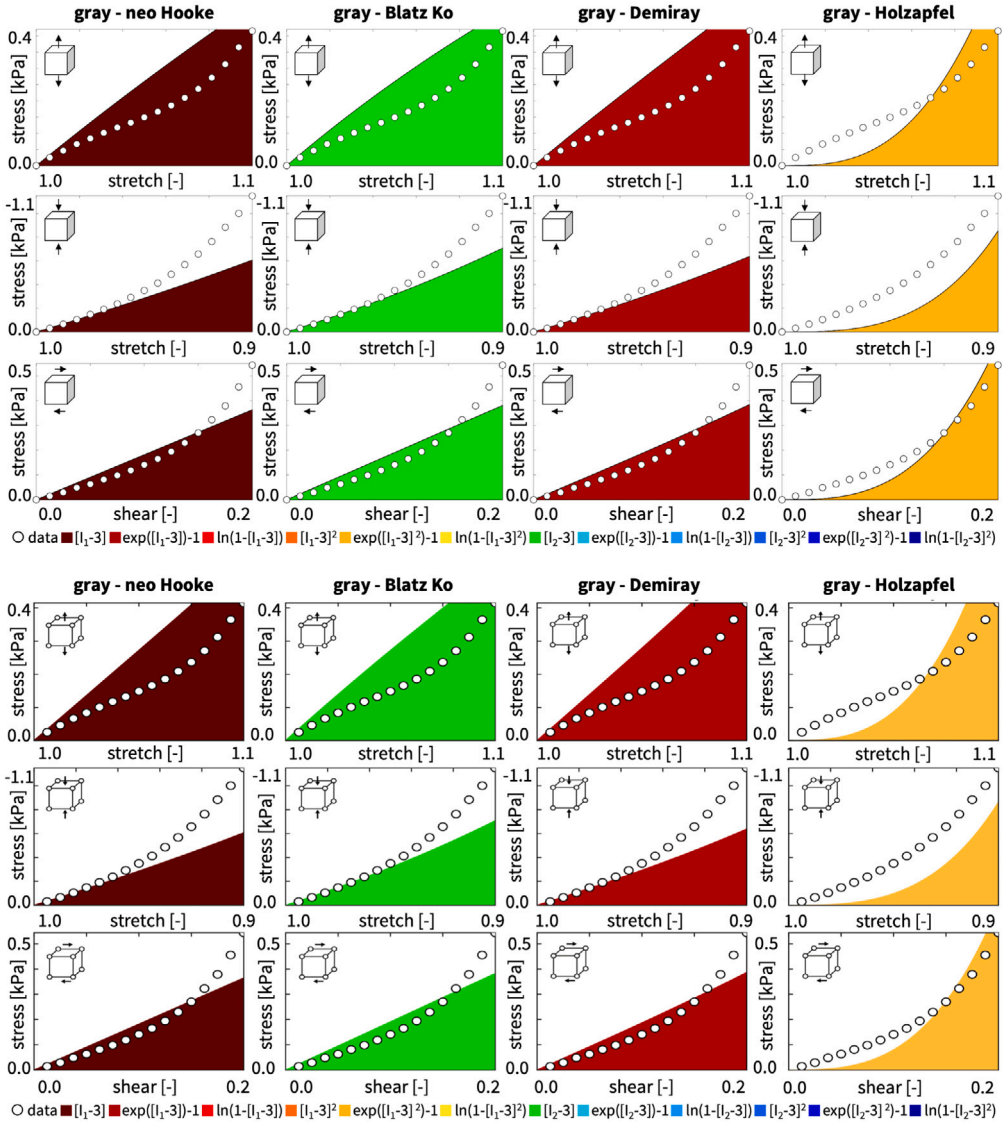

Fig. 3. Neo Hooke, Blatz Ko, Mooney Rivlin, Demiray, Gent, and Holzapfel models and finite element simulation. Nominal stress as a function of stretch and shear strain for the neo Hooke model $\psi = \frac{1}{2}\mu[\,I_1 - 3\,]$ with $1, 1, 1, w_{1,1}, w_{2,1}$, the Blatz Ko model $\psi = \frac{1}{2}\mu[\,I_2 - 3\,]$ with $2, 1, 1, w_{1,7}, w_{2,7}$, the Demiray model $\psi = \frac{1}{2}\frac{a}{b}[\,\exp(b[\,I_1 - 3\,]) - 1\,]$ with $1, 1, 2, w_{1,2}, w_{2,2}$, and the Holzapfel model $\psi = \frac{1}{2}\frac{a}{b}[\,\exp(b[\,I_1 - 3\,]^2) - 1\,]$ with $1, 2, 2, w_{1,5}, w_{2,5}$. Dots illustrate the tension, compression, and shear data [43] from human gray matter experiments [47]; color-coded area highlights the contribution to the stress function using the weights from Table 1; top graphs display the discovered model and bottom graphs display the finite element simulation.

Fig. 3 compares the neo Hooke, Blatz Ko, Demiray, and Holzapfel models and the finite element simulation with our universal material subroutine. The graphs show the nominal stress as a function of the stretch and shear strain for all four models. The dots indicate the tension, compression, and shear data of human gray matter tissue, the color-coded areas highlights the contributions to the stress function. The finite element simulation with our universal material subroutine in the bottom graphs agrees excellently with the discovered models in the top graphs [43] and confirms the correct implementation of the first, seventh, second, and fifth terms of our model. Taken together, these simple examples demonstrate that we can recover popular constitutive functions for which the network weights gain a well-defined physical meaning and the universal material subroutine specializes to widely used constitutive models.

5.2. Benchmarking with newly discovered models.

To illustrate how our universal material subroutine performs for newly discovered models, we benchmark our subroutine with two recently discovered models for gray and white matter tissue [43], translate their network weights $w_{1,\cdot}$ and $w_{2,\cdot}$ of the first and

Table 2
Newly discovered gray and white matter models and parameters. Discovered gray and white matter weights $w_{1,\bullet}$ and $w_{2,\bullet}$ for training with individual and combined tension, compression, and shear data [43] from human gray matter experiments [47].

	tension	compression	shear	ten+com+shr	tension	compression	shear	ten+com+shr
	gray matter $n = 15$	gray matter $n = 17$	gray matter $n = 35$	gray matter $n = 15, 17, 35$	white matter $n = 18$	white matter $n = 18$	white matter $n = 33$	white matter $n = 18, 18, 33$
	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]	$w_{1,\bullet}$ [-]	$w_{2,\bullet}$ [kPa]
$w_{\bullet,1}$	0.314	0.346	0.403	0.198	0.663	0.180	0.000	0.000	0.000	0.000	1.736	0.281	0.364	0.249	0.000	0.000
$w_{\bullet,2}$	0.158	0.110	0.063	0.790	0.242	0.260	0.000	0.000	0.000	0.000	0.000	0.000	0.103	0.240	0.000	0.000
$w_{\bullet,3}$	0.000	0.000	0.000	0.000	0.766	0.184	0.000	0.000	0.000	0.000	0.000	0.000	0.037	0.307	0.000	0.000
$w_{\bullet,4}$	1.130	0.681	2.373	1.109	1.440	1.420	0.000	0.000	0.893	0.147	1.547	1.077	1.394	0.652	0.000	0.000
$w_{\bullet,5}$	1.472	1.562	1.186	2.103	1.336	1.711	0.000	0.000	0.376	0.233	1.142	1.215	1.360	1.103	0.000	0.000
$w_{\bullet,6}$	0.502	0.435	0.000	0.000	0.000	0.000	0.000	0.000	1.308	0.430	1.212	1.148	0.440	0.831	0.000	0.000
$w_{\bullet,7}$	0.952	0.169	1.853	0.290	0.373	0.190	0.000	0.000	1.004	0.072	0.000	0.000	0.035	0.295	1.386	0.160
$w_{\bullet,8}$	0.228	0.207	0.059	0.059	0.261	0.357	0.000	0.000	0.087	0.072	0.003	0.030	0.055	0.391	0.240	0.490
$w_{\bullet,9}$	0.682	0.173	1.947	0.114	0.000	0.000	0.988	0.634	0.840	0.207	0.000	0.000	0.768	0.118	0.000	0.000
$w_{\bullet,10}$	2.264	0.848	2.274	1.130	0.880	1.987	2.774	1.370	0.000	0.000	1.008	1.413	1.055	0.855	0.000	0.000
$w_{\bullet,11}$	0.038	0.357	1.223	2.067	1.735	1.551	1.650	1.888	0.000	0.000	1.219	1.133	0.999	1.074	1.889	1.686
$w_{\bullet,12}$	0.933	0.473	0.000	0.000	0.882	1.425	1.403	1.666	1.105	0.003	2.648	0.823	0.000	0.000	1.179	1.911

second layers into their model parameters, provide the material table for the input file to our material subroutine, and compare each simulation to experimental data [47]. Table 2 summarizes our discovered weights $w_{1,\bullet}$ and $w_{2,\bullet}$ when training our network with individual and combined tension, compression, and shear data [43] from human gray and white matter experiments [47].

Gray matter model. The left columns of Table 2 provide four different models for gray matter tissue, three for training with the individual tension, compression, and shear data, and one for training with all three data sets combined. When trained with the individual data sets for tension, compression, and shear, the neural network in Fig. 1 discovers the majority of terms of the free energy function (9), eleven, nine, and ten terms, while only one, three, and two terms train to zero. When trained with all three data sets combined, our network uniquely discovers a four-term model, while the weights of the other eight terms train to zero,

$$\psi = \frac{1}{2}\,\mu_2[\,I_2 - 3\,]^2 + \frac{1}{2}\,\frac{a_2}{b_2}[\,\exp(\,b_2[\,I_2 - 3\,]^2) - 1\,] - \frac{1}{2}\,\frac{\alpha_1}{\beta_1}\ln(\,1 - \beta_1[\,I_2 - 3\,]) - \frac{1}{2}\,\frac{\alpha_2}{\beta_2}\ln(\,1 - \beta_2[\,I_2 - 3\,]^2)\,. \tag{21}$$

The non-zero weights translate into physically meaningful gray matter parameters with well-defined physical units, the four stiffness-like parameters, $\mu_2 = 2\,w_{1,10}\,w_{2,10} = 7.60$ kPa, $a_2 = 2\,w_{1,11}\,w_{2,11} = 6.23$ kPa, $\alpha_1 = 2\,w_{1,9}\,w_{2,9} = 1.25$ kPa, $\alpha_2 = 2\,w_{1,12}\,w_{2,12} = 4.67$ kPa, and the three nonlinearity parameters, $b_2 = w_{1,11} = 1.65$, $\beta_1 = w_{1,9} = 0.99$, $\beta_2 = w_{1,12} = 1.40$. The newly discovered gray matter model translates into the following material table for our universal material subroutine.

```
*PARAMETER TABLE, TYPE="UNIVERSAL_TAB"
2,1,3,w1,9,w2,9
2,2,1,w1,10,w2,10
2,2,2,w1,11,w2,11
2,2,3,w1,12,w2,12
```

Fig. 4 compares the gray matter model and the finite element simulation with our universal material subroutine. The graphs show the nominal stress as a function of the stretch and shear strain for the gray matter model. The dots indicate the tension, compression, and shear data of human gray matter tissue, the color-coded areas highlight the contributions to the stress function. The finite element simulation with our universal material subroutine in the bottom graphs agrees excellently with the discovered gray matter model in the top graphs [43] and confirms the correct implementation of all twelve terms of our model.

White matter model. The right columns of Table 2 provide four different models for white matter tissue, three for training with the individual tension, compression, and shear data, and one for training with all three data sets combined. When trained with the individual data sets for tension, compression, and shear, the neural network in Fig. 1 discovers the majority of terms of the free energy function (9), seven, eight, and eleven terms, while only five, four, and one terms train to zero. When trained with all three data sets combined, our network uniquely discovers a four-term model, while the weights of the other eight terms train to zero,

$$\psi = \frac{1}{2}\,\mu_1[\,I_2 - 3\,] + \frac{1}{2}\,\frac{a_1}{b_1}[\,\exp(\,b_1[\,I_2 - 3\,]) - 1\,] + \frac{1}{2}\,\frac{a_2}{b_2}\,[\exp(\,b_2[\,I_2 - 3]^2) - 1\,] - \frac{1}{2}\,\frac{\alpha_2}{\beta_2}\ln(1 - \beta_2[\,I_2 - 3]^2). \tag{22}$$

The non-zero weights translate into physically meaningful parameters with well-defined physical units, the four stiffness-like parameters, $\mu_1 = 2\,w_{1,7}\,w_{2,7} = 0.44$ kPa, $a_1 = 2\,w_{1,8}\,w_{2,8} = 0.24$ kPa, $a_2 = 2\,w_{1,11}\,w_{2,11} = 6.37$ kPa, $\alpha_2 = 2\,w_{1,12}\,w_{2,12} = 4.51$ kPa, and the three nonlinearity parameters, $b_1 = w_{1,8} = 0.24$, $b_2 = w_{1,11} = 1.89$, $\beta_2 = w_{1,12} = 1.18$. The newly discovered white matter model translates into the following material table for our universal material subroutine.

```
*PARAMETER TABLE, TYPE="UNIVERSAL_TAB"
2,1,1,w1,7,w2,7
2,1,2,w1,8,w2,8
2,2,2,w1,11,w2,11
2,2,3,w1,12,w2,12
```

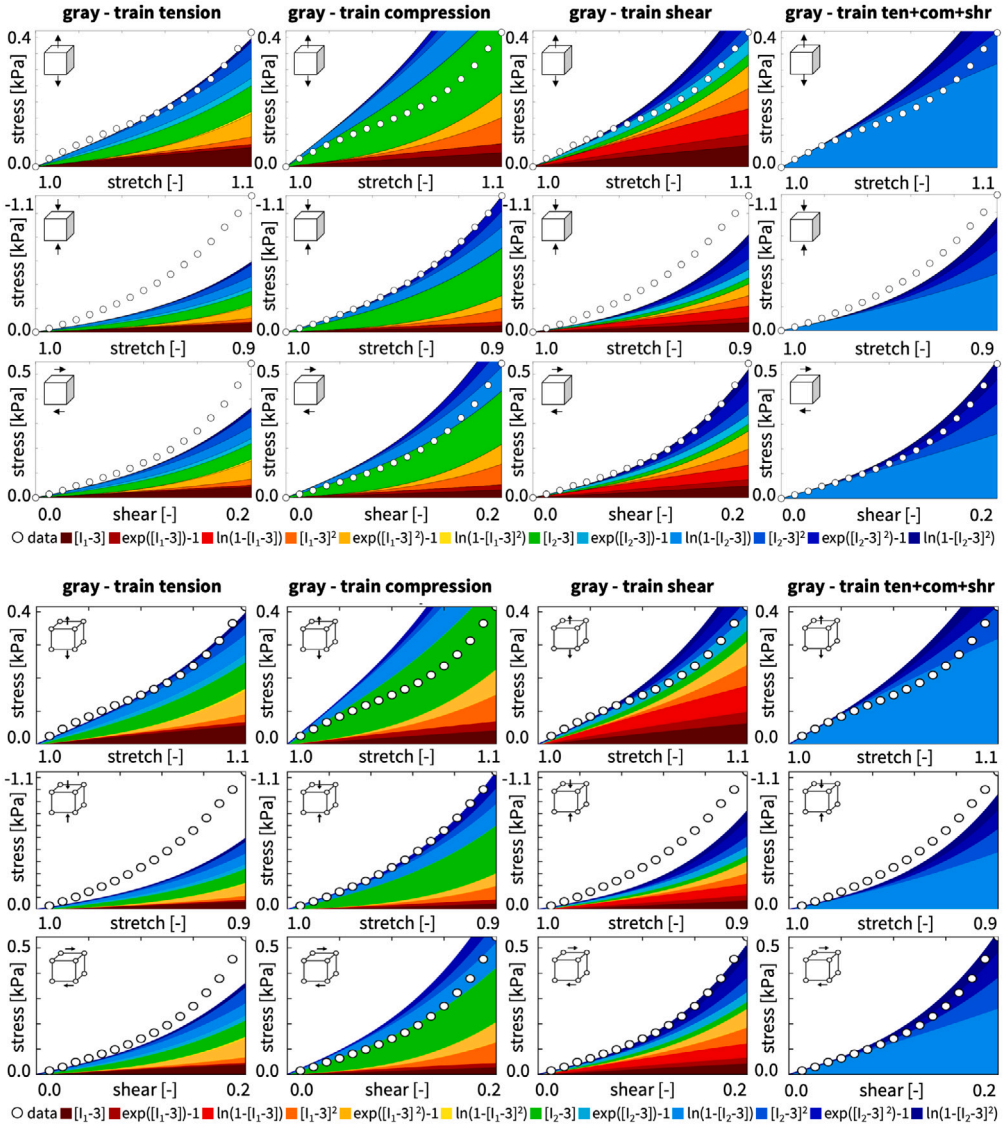

Fig. 4. Gray matter discovered model and finite element simulation. Nominal stress as a function of stretch and shear strain for the isotropic, perfectly incompressible constitutive neural network with two hidden layers, and twelve nodes in Fig. 1. Dots illustrate the tension, compression, and shear data [43] from human gray matter experiments [47]; color-coded areas highlight the twelve contributions to the discovered stress function using the weights from Table 2; top graphs display the discovered model and bottom graphs display the finite element simulation.

Fig. 5 compares the white matter model and the finite element simulation with our universal material subroutine. The graphs show the nominal stress as a function of the stretch and shear strain for the white matter model. The dots indicate the tension, compression, and shear data of human white matter tissue, the color-coded areas highlight the contributions to the stress function. The finite element simulation with our universal material subroutine in the bottom graphs agrees excellently with the discovered white matter model in the top graphs [43] and confirms the correct implementation of the twelve terms of our model. Taken together, these eight examples demonstrate that our proposed method generalizes well to previously undiscovered constitutive functions, which translate smoothly into a universal material subroutine that agrees well with the experimental data and previous simulations.

5.3. Realistic finite element simulations.

To illustrate the performance of our universal material subroutine for our discovered gray and white matter models from Eqs. (21) and (22) within a realistic finite element simulation, we study six different head impact scenarios. Fig. 6 shows our sagittal model that consists of 6182 gray and 5701 white matter linear triangular elements, 6441 nodes, and 12,882 degrees of freedom. Fig. 7 shows coronal model that consists of 7106 gray and 14 196 white matter linear triangular elements, 11 808 nodes, and 23 616

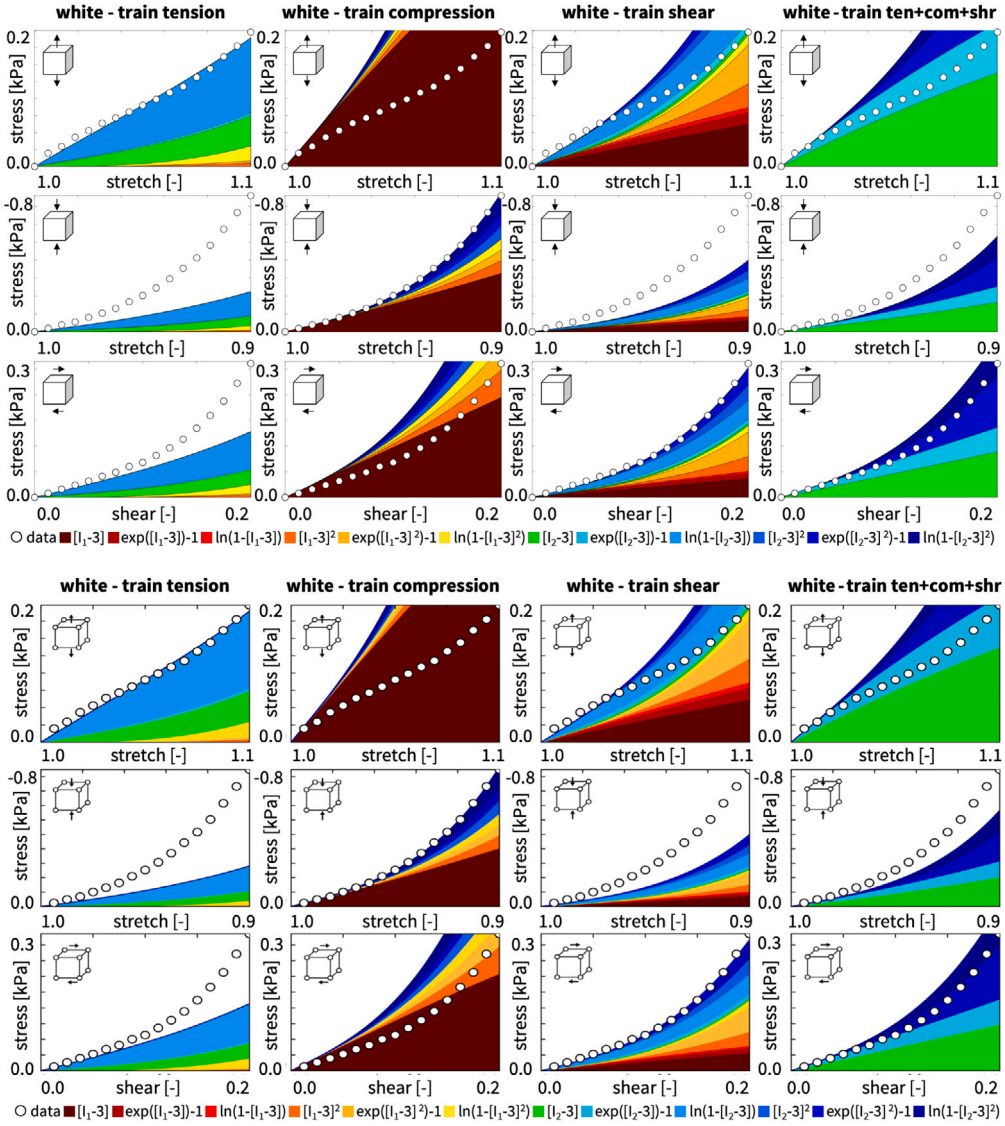

Fig. 5. White matter discovered model and finite element simulation. Nominal stress as a function of stretch and shear strain for the isotropic, perfectly incompressible constitutive neural network with two hidden layers, and twelve nodes in Fig. 1. Dots illustrate the tension, compression, and shear data [43] from human white matter experiments [47]; color-coded areas highlight the twelve contributions to the discovered stress function using the weights from Table 2; top graphs display the discovered model and bottom graphs display the finite element simulation.

degrees of freedom. We embed both models into the skull using spring support at the free boundaries and apply top-of-the-head, diagonal, and frontal impacts to the sagittal model and top-of-the-head, diagonal, and lateral impacts to the coronal model. Figs. 6 and 7 summarize the stress profiles for the six different impact simulations. Clearly, we observe stress concentrations at the gray and white matter interface, between the cortex and the corona radiata. These stress concentrations are common after a hit to the head, and are a result of the structural and mechanical differences between different tissue types: Gray matter consists primarily of neuronal cell bodies and is rather dense, while white matter consists primarily of myelinated axons. Upon an impact to the head, forces are transmitted differently through these tissue types. From Figs. 4 and 5, we conclude that gray matter is almost twice as stiff as white matter, with maximum tensile stresses of 0.4 kPa versus 0.2 kPa for stretches of 1.1, maximum compressive stresses of −1.1 kPa versus −0.8 kPa for stretches of 0.9, and maximum shear stresses of 0.5 kPa versus 0.5 kPa for shear of 0.2. This disparity in mechanical stiffnesses leads to localized stress concentrations at the gray and white matter interface, which can disrupt the structural integrity of the tissue and trigger diffuse axonal injuries. The simulations predict that these stress concentrations occur primarily in the frontal and occipital lobes for top-of-the-head impacts, in the deep white matter tracts for diagonal impacts, in the frontal and parietal lobes for frontal impacts, and in the gray and white matter interface for lateral impacts. Taken together, these six examples demonstrate that our discovered gray and white matter models translate smoothly into a universal material subroutine

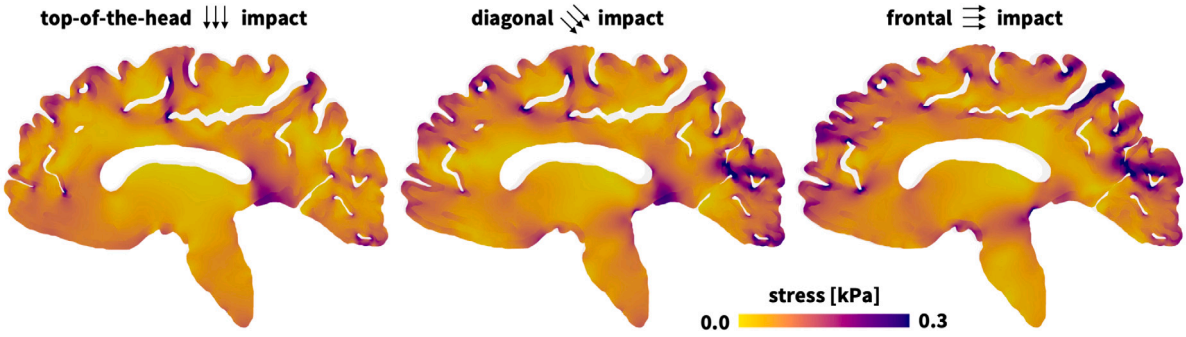

Fig. 6. Stress profiles for top-of-the-head, diagonal, and frontal impact to the human brain. The finite element simulations use our universal material subroutine with the discovered models from Table 2 for gray matter from Eq. (21) with the four stiffness-like parameters $\mu = 7.60$ kPa, $a_2 = 6.23$ kPa, $\alpha_1 = 1.25$ kPa, $\alpha_2 = 4.67$ kPa, and the three nonlinearity parameters, $b_2 = 1.65$, $\beta_1 = 0.99$, $\beta_2 = 1.40$, and for white matter from Eq. (22) with the three stiffness-like parameters $\mu = 0.44$ kPa, $a_1 = 0.24$ kPa, $a_2 = 6.37$ kPa, $\alpha_2 = 4.51$ kPa, and the three nonlinearity parameters, $b_1 = 0.24$, $b_2 = 1.89$, $\beta_2 = 1.18$.

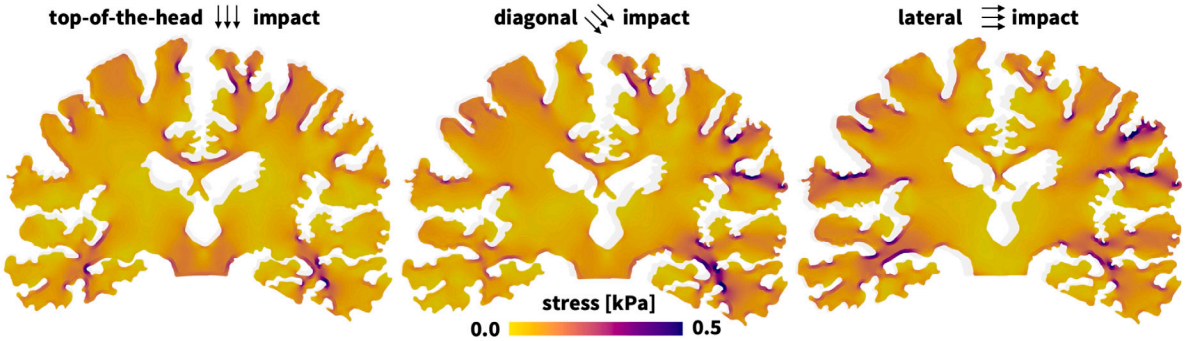

Fig. 7. Stress profiles for to top-of-the-head, diagonal, and lateral impact to the human brain. The finite element simulations use our universal material subroutine with the discovered models from Table 2 for gray matter from equation (21) with the four stiffness-like parameters $\mu = 7.60$ kPa, $a_2 = 6.23$ kPa, $\alpha_1 = 1.25$ kPa, $\alpha_2 = 4.67$ kPa, and the three nonlinearity parameters, $b_2 = 1.65$, $\beta_1 = 0.99$, $\beta_2 = 1.40$, and for white matter from equation (22) with the three stiffness-like parameters $\mu = 0.44$ kPa, $a_1 = 0.24$ kPa, $a_2 = 6.37$ kPa, $\alpha_2 = 4.51$ kPa, and the three nonlinearity parameters, $b_1 = 0.24$, $b_2 = 1.89$, $\beta_2 = 1.18$.

that generalizes from the homogeneous simulations in Figs. 3 through 5 to realistic finite element simulations in Figs. 6 and 7, where it robustly predicts heterogeneous stress profiles across complex structures.

6. Discussion

Our universal material subroutine specializes well to popular constitutive models. To demonstrate that our material subroutine includes popular constitutive models as special cases, we benchmarked it with four widely used models. Fig. 3 compares the neo Hooke [11], Blatz Ko [12], Demiray [17], and Holzapfel [18] models in the top row to finite element simulations with our universal material subroutine in the bottom row. All four models only activate a single term of the subroutine, which translates into a single-row material table, and a single-color stress plot. For all four models, the finite element simulations with our new material subroutine in the bottom twelve graphs of Fig. 3 agree identically with the initial model in the top twelve graphs [43]. These simple benchmark examples demonstrate that we can recover popular constitutive models for which the weights of our constitutive neural network in Fig. 1 gain a well-defined physical meaning and the universal material subroutine takes the functional form of one of the twelve activation functions in Fig. 2 [42]. Importantly, to perform a finite element analysis, we no longer need to select a specific material model; instead, we can simply use our universal material subroutine and selectively activate its relevant terms through the non-zero entries in the material table.

Our universal material subroutine expands naturally to compressible and anisotropic materials. For illustrative purposes, we have only demonstrated the versatility of our material subroutine for *incompressible* and *isotropic* hyperelastic materials [42]. For these, the first index of our parameter table selects between the first and second invariants, the second index raises them to linear or quadratic powers, and the third index selects between the identity, exponential, and logarithmic functions. This setting seamlessly generalizes to *compressible* and *anisotropic* materials by selecting a first index of three, four, five, ..., `NINV` to include terms in the third, fourth, or fifth invariants. For example, the subroutine `UANISOHYPER_INV` supports up to three fiber directions resulting in a total of 15 invariants. It also naturally allows for higher order powers by selecting a second index larger than two and facilitates the integration of additional functional forms through a third index larger than three. In the present study, we have illustrated how to

translate the output of our automated model discovery into the input of our universal material subroutine within Abaqus [21] using the software's invariant-based user material subroutine `UANISOHYPER_INV`. However, the inherent modularity of our approach ensures that this translation will generalize naturally to arbitrary implicit or explicit nonlinear finite element packages, which is part of our current follow up research.

Our proposed method generalizes well to previously undiscovered constitutive functions. Automated model discovery allows us to discover the best possible model, in our case out of $2^{12} = 4096$ possible combinations of terms [43]. Traditionally, model developers have rationalized constitutive models from the shape of experimental curves and then fit their parameters to data. Throughout the past decades, this has generated dozens of models with one [11,12,16,17], two [13,14,18], three [15] or more terms, almost always in terms of the *first invariant*. Recent developments in deep learning now allow us to rapidly screen thousands of possible combinations of terms and discover the best possible fit. However, when only trained with individual tension, compression, or shear data, the network tends to overfit the data and discovers a wide variety of terms [42,48]. Yet, when trained with all three data sets combined, the network robustly and repeatedly discovers a small subset of terms in the *second invariant* for both gray and white matter [43]. Strikingly, these terms have been overlooked by traditional manual model development. In retrospect, it seems obvious that the second invariant is well suited to characterize human brain tissue : While the first invariant, $I_1 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2$, is quadratic in terms of the stretches λ, the second invariant, $I_2 = \lambda_1^2\lambda_2^2 + \lambda_2^2\lambda_3^2 + \lambda_1^2\lambda_3^2$, is quartic and seems better suited to represent nonlinearities [49]. This is particularly relevant in the small stretch regime of $0.9 \leq \lambda \leq 1.1$ that we study here, where the stretches are small and their nonlinear effects remain minor [47]. For rubber-like materials, where the stretches in uniaxial tension, equibiaxial tension, and pure shear can easily reach values of $1.0 \leq \lambda \leq 8.0$, the second invariant explodes and seems less well-suited to characterize the stretch-stress response [50]. The excellent agreement of the finite element simulations with our universal material subroutine in the bottom twelve graphs of Figs. 4 and 5 with the experimental data [47] and the discovered models [43] in the top twelve graphs confirms the correct implementation of our discovered gray and white matter models.

Our universal material subroutine uses interpretable material parameters. A characteristic feature of our proposed modeling strategy is that it features different activation functions, linear and quadratic, $(\circ)^1$ and $(\circ)^2$, embedded in the identity, exponential, and logarithmic functions, $(\circ)$, $(\exp(\circ)-1)$, and $(-\ln(1-(\circ)))$ [42]. This is in stark contrast to previous approaches that have used one and the same activation function across all network nodes, for example of hyperbolic tangent [31,36], exponential linear unit [51], or softplus squared [52] type. While it is theoretically possible to manually embed these models into a finite element workflow, their weights translate into material parameters that have no clear physical interpretation [53]. In contrast, our non-zero weights translate into physically meaningful parameters with well-defined physical units: the stress-like parameters, $w_{1,\bullet}w_{2,\bullet}$, with the unit kilopascal, and the dimensionless parameters, $w_{1,\bullet}$, that govern the exponential [17] and logarithmic [16] nonlinearities. We conclude that our proposed approach generalizes well to previously undiscovered constitutive functions, which translate naturally into a universal material subroutine that agrees well with the experimental data [47] and with previous simulations [42]. Importantly, rather than having to implement a new material subroutine for each newly discovered model, we use a single universal material subroutine that inherently incorporates all $2^{12} = 4096$ possible combinations of terms and activates the relevant model merely by means of the twelve rows of its parameter table.

Our universal material subroutine generalizes well to realistic simulations. When embedded into a finite element simulation, our material subroutine translates the local deformation gradient into stresses and stress derivatives that enter the global force vector and stiffness matrix of the local Newton iteration to solve the balance of motion. To illustrate that our new subroutine not only performs well for the homogeneous examples in Figs. 3 to 5, but also for realistic finite element simulations, we simulate the regional stress distributions across the human brain for six different head impact scenarios [54]. Figs. 6 and 7 emphasize the sensitivity of the stress profiles with respect to the location and direction of the impact. Depending on impact location and severity, individuals may experience a broad spectrum of symptoms ranging from headaches, dizziness, nausea, and vision problems to difficulties with concentration and attention [55]. *Top-of-the-head impacts* in Figs. 6 and 7, left, affect regions of the skull that are usually very thin and vulnerable to skull fractures, brain contusions, and significant brain damage [56]. In agreement with our simulated stress profiles, these impacts primarily affect the frontal region of the brain that plays a crucial role in higher cognitive functions, personality, emotional regulation, and decision-making. Their symptoms may range from cognitive impairment, memory loss, and motor function deficit to long-term consequences such as permanent disability or death. *Diagonal impacts* as in Figs. 6 and 7, middle, can cause rotational forces that many result in diffuse axonal injuries. These injuries occur when brain structures tear in response to elevated shear stresses [57]. Diffuse axonal injuries often involve deep white matter tracts and can affect multiple lobes of the brain, including the frontal, temporal, and parietal lobes, for which our simulation predicts elevated stress levels. These can have profound effects on brain function and lead to cognitive, behavioral, and motor impairments associated with difficulties of attention, memory, problem-solving, and emotional regulation. *Frontal impacts* as in Figs. 6, right, directly affect the frontal region of the brain, the side of impact, through coup injury. Importantly, they can also have severe secondary effects on brain regions opposite to the impact, through countercoup injury, as we conclude from our simulated stress profiles. Frontal impacts commonly results in mild or severe concussions [55] or traumatic brain injuries associated with a wide range of symptoms such as headaches, dizziness, confusion, and memory issues. These can impair higher cognitive functions, personality, and emotional regulation and lead to changes in behavior, mood, and decision-making. *Lateral impacts* as in Figs. 7, right, cause the brain to rotate, which can induce diffuse axonal injuries similar to diagonal impacts [58]. As we conclude from our simulated stress profiles, lateral impacts affect mainly the gray and white matter interface, which experiences much higher stresses than, for example, under top-of-the-head impacts of the same magnitude. Diffuse axonal injuries can result in cognitive, behavioral, and motor impairments, and affect various aspects of daily life. Lateral impacts may also lead to contusions, which can compromise brain function and potentially cause

long-term memory loss, personality changes, and emotional instability. Knowing the precise location and direction of a head impact is critical because impacts to different brain regions can result in varying types and severity of injuries [59]. Understanding the stress profiles in response to different types of impact can help assess the extent of an injury, determine the appropriate treatment, and develop strategies to prevent further head trauma [60].

Limitations. Our results suggest that we can seamlessly integrate automated model discovery into a finite element workflow through a new universal material subroutine. Nonetheless, our study has several limitations that point towards possible future extensions. First, while our current model is incompressible and isotropic, we can easily expand it to include compressibility [61] and anisotropy [62] by adding the third, fourth, fifth, and higher order invariants, that we can simply embed via the first index in our parameter table. Second, we can expand our model and include higher order powers [15], cubic or quartic, via the second index in our parameter table. Third, we could generalize our current network architecture from an additive coupling of the invariants towards a multiplicative coupling [63], which would translate into additional cross-coupling terms in the tangents of our user material subroutine. Fourth, instead of using a purely invariant-based formulation, we could also include principal-stretch-based terms [48] that mimic an Ogden [19] or Valanis–Landel [20] type behavior. Fifth, in addition to the elastic potential that characterizes the hyperelastic behavior, we could also include one or more inelastic potentials that characterize viscosity, plasticity, damage, or growth [64].

7. Conclusion

Constitutive modeling is critical to a successful analysis of materials and structures. However, the scientific criteria for selecting the appropriate model remain insufficiently understood. This work seeks to address the question whether and how we can automate constitutive modeling within a finite element analysis. Our work is made possible by a recent trend in physics-based artificial intelligence, automated model discovery, a new technology that allows us to autonomously discover the best model to explain experimental data. Automated model discovery comes in various flavors and uses sparse regression, symbolic regression, or constitutive neural networks with the common goal to discover constitutive models from thousands of combinations of a few functional building blocks. Here our objective was to integrate automated model discovery into the finite element workflow by creating a single unified user material subroutine that contains $2^{12} = 4096$ constitutive models made up of 12 individual terms. For illustrative purposes, we prototyped this strategy within the `UANISOHYPER_INV` environment of the general-purpose finite element software Abaqus and share our new universal material subroutine publicly on GitHub. For three examples, we demonstrated that our universal material subroutine specializes well to traditional constitutive models, generalizes well to newly discovered models, and performs well within realistic finite element simulations. While we have only prototyped our approach for a specific hyperelastic material model, for a specific type of automated model discovery, and for a specific finite element platform, we are confident that our strategy will generalize naturally to more complex anisotropic, compressible, and inelastic materials, to other types of model discovery, and to other nonlinear finite element analysis platforms. Replacing dozens of individual material subroutines by a single universal material subroutine – populated directly via automated model discovery – makes finite element analyses more accessible, more robust, and less vulnerable to human error. This could induce a paradigm shift in constitutive modeling and forever change how we simulate materials and structures.

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.

Data availability

Our source code, data, and examples are available at https://github.com/LivingMatterLab/CANN.

Acknowledgments

This work was supported by a DAAD Fellowship to Kevin Linka, and by NSF CMMI Award 2320933 *Automated Model Discovery for Soft Matter* to Ellen Kuhl.

References

[1] J.R. Hughes, The Finite Element Method. Linear Static and Dynamic Finite Element Analysis, Prentice-Hall, Eaglewood Cliffs, New Jersey, 1987.
[2] G.A. Holzapfel, Nonlinear Solid Mechanics: A Continuum Approach to Engineering, John Wiley & Sons, Chichester, 2000.
[3] J.E. Marsden, J.R. Hughes, Topics in the mathematical foundations of elasticity, in: R. Knops (Ed.), Nonlinear Analysis and Mechanics, Heriot-Watt Symposium, Vol II: 30-285, Pitman, Boston, Massachusetts, 1978.
[4] J. Bonet, R.D. Wood, Nonlinear Continuum Mechnanics for Finite Element Analysis, Cambridge University Press, Cambridge, 1997.
[5] J.E. Marsden, J.R. Hughes, Mathematical Foundations of Elasticity, Prentice-Hall, Eaglewood Cliffs, New Jersey, 1987.
[6] H. Dal, K. Acikgoz, Y. Badina, On the performance of isotropic hyperelastic constitutive models for rubber-like materials: A state of the art review, Appl. Mech. Rev. 73 (2021) 020802.
[7] S. Hartmann, Paramter estimation of hyperelastic relations of generalized polynomial-type with constraint conditions, Int. J. Solids Struct. 38 (2001) 7999–8018.

[8] S.K. Melley, L. Liu, Y. Liu, J. Leng, A review on material models for isotropic hyperelasticity, Int. J. Mech. Syst. Dyn. 1 (2021) 71–88.
[9] L.A. Mihai, S. Budday, G.A. Holzapfel, E. Kuhl, A. Goriely, A family of hyperelastic models for human brain tissue, J. Mech. Phys. Solids 106 (2017) 60–79.
[10] P. Steinmann, M. Hossain, G. Possart, Hyperelastic models for rubber-like materials: Consistent tangent operators and suitability for Treloar's data, Arch. Appl. Mech. 82 (2012) 1183–1217.
[11] L.R.G. Treloar, Stresses and birefringence in rubber subjected to general homogeneous strain, Proc. Phys. Soc. 60 (1948) 135–144.
[12] P.J. Blatz, W.L. Ko, Application of finite elastic theory to the deformation of rubbery materials, Trans. Soc. Rheol. 6 (1962) 223–251.
[13] M. Mooney, A theory of large elastic deformations, J. Appl. Phys. 11 (1940) 582–590.
[14] R.S. Rivlin, Large elastic deformations of isotropic materials. IV. Further developments of the general theory, Philos. Trans. R. Soc. London Ser. A 241 (1948) 379–397.
[15] O.H. Yeoh, Some forms of the strain energy function for rubber, Rubber Chem. Technol. 66 (1993) 754–771.
[16] A. Gent, A new constitutive relation for rubber, Rubber Chem. Technol. 69 (1996) 59–61.
[17] H. Demiray, A note on the elasticity of soft biological tissues, J. Biomech. 5 (1972) 309–311.
[18] G.A. Holzapfel, T.C. Gasser, R.W. Ogden, A new constitutive framework for arterial wall mechanics and comparative study of material models, J. Elasticity 61 (2000) 1–48.
[19] R.W. Ogen, Large deformation isotropic elasticity – On the correlation of theory and experiment for incompressible rubberlike solids, Proc. R. Soc. London Ser. A 326 (1972) 565–584.
[20] K. Valanis, R.F. Landel, The strain–energy function of a hyperelastic material in terms of the extension ratios, J. Appl. Phys. 38 (1967) 2997–3002.
[21] Abaqus Analysis User's Guide, Dassault Systèmes Simulia Corp., Rhode Island., 2020.
[22] S.R. St Pierre, D. Rajasekharan, E.C. Darwin, K. Linka, M.E. Levenston, E. Kuhl, Discovering the mechanics of artificial and real meat, Comput. Methods Appl. Mech. Engrg. 415 (2023) 116236.
[23] G.W. Liu, Q.B. Li, M.A. Msekh, Z. Zuo, Abaqus implementation of monolithic and staggered schemes for quasi-static and dynamic fracture phase-field model, Comput. Mater. Sci. 121 (2016) 35–47.
[24] Y. Navidtehrani, C. Betegon, E. Martinez-Paneda, A unified abaqus implementation of the phase field fracture method using only a user material subroutine, Materials 14 (1913) (2021).
[25] R. Ostwald, E. Kuhl, A. Menzel, On the implementation of finite deformation gradient-enhanced damage models, Comput. Mech. 64, 847–877.
[26] T.W. Hohenberger, R.J. Winslow, N.M. Pugno, J.J.C. Busfield, A constitutive model for both low and high strain nonlinearities in highly filled elastomers and implementation with user-defined material subroutines in Abaqus, Rubber Chem. Technol. 92 (2020) 653–686.
[27] H. Saini, O. Rohrle, A biophysically guided constitutive law of the musculotendon-complex: Modelling and numerical implementation in Abaqus, Comput. Methods Programs Biomed. 226 (2022) 107152.
[28] N. Atanasova, L. Todorovski, S. Dzeroski, B. Kompare, Application of automated model discovery from data and expert knowledge to a real-world domain: Lake Glumso, Ecol. Model. 212 (2008) 92–98.
[29] J. Bongard, H. Lipson, Automated reverse engineering of nonlinear dynamical systems, Proc. Natl. Acad. Sci. 104 (2007) 9943–9948.
[30] M. Alber, A. Buganza Tepole, W. Cannon, S. De, S. Dura-Bernal, K. Garikipati, G.E. Karniadakis, W.W. Lytton, P. Perdikaris, L. Petzold, E. Kuhl, Integrating machine learning and multiscale modeling: Perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences, npj Digit. Med. 2 (115) (2019).
[31] Y.M.A. Hashash, S. Jung, J. Ghaboussi, Numerical implementation of a neural network based material model in finite element analysis, Int. J. Numer. Methods Eng. 59 (2004) 989–1005.
[32] A.L. Frankel, R.E. Jones, L.P. Swiler, Tensor basis Gaussian process models of hyperelastic materials, J. Mach. Learn. Comput. Model. 1, 1–17.
[33] J.N. Fuhg, N. Bouklas, On physics-informed data-driven isotropic and anisotropic constitutive models through probabilistic machine learning and space-filling sampling, Comput. Methods Appl. Mech. Engrg. 394 (2022) 114915.
[34] G.A. Holzapfel, K. Linka, S. Sherifova, C. Cyron, Predictive constitutive modelling of arteries by deep learning, J. R. Soc. Interface 18 (2021) 20210411.
[35] K. Linka, M. Hillgartner, K.P. Abdolazizi, R.C. Aydin, M. Itskov, C.J. Cyron, Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning, J. Comput. Phys. 429 (2021) 110010.
[36] V. Tac, F. Sahli Costabal, A. Buganza Tepole, Data-driven tissue mechanics with polyconvex neural ordinary differential equations, Comput. Methods Appl. Mech. Engrg. 398 (2022) 115248.
[37] V. Tac, K. Linka, F. Sahli Costabal, E. Kuhl, A. Buganza Tepole, Benchmarking physics-informed frameworks for data-driven hyperelasticity, Comput. Mech. (2023) http://dx.doi.org/10.1007/s00466-023-02355-2.
[38] V. Tac, V.D. Sree, M.K. Rausch, A. Buganza Tepole, Data-driven modeling of the mechanical behavior of anisotropic soft biological tissue, Eng. Comput. 38 (2022) 4167–4182.
[39] M. Flaschel, S. Kumar, L. De Lorenzis, Unsupervised discovery of interpretable hyperelastic constitutive laws, Comput. Methods Appl. Mech. Engrg. 381 (2021) 113852.
[40] M. Flaschel, S. Kumar, L. De Lorenzis, Automated discovery of generalized standard material models with EUCLID, Comput. Methods Appl. Mech. Eng. 405 (2023) 115867.
[41] R. Abdusalamov, M. Hillgartner, M. Itskov, Automatic generation of interpretable hyperelastic models by symbolic regression, Internat. J. Numer. Methods Engrg. 124, 2093–2104.
[42] K. Linka, E. Kuhl, A new family of constitutive artificial neural networks towards automated model discovery, Comput. Methods Appl. Mech. Engrg. 403 (2023) 115731.
[43] K. Linka, S.R. St Pierre, E. Kuhl, Automated model discovery for human brain using constitutive artificial neural networks, Acta Biomater. 160 (2023) 134–151.
[44] K. Linka, A. Buganza Tepole, G.A. Holzapfel, E. Kuhl, Automated model discovery for skin: Discovering the best model, data, and experiment, Comput. Methods Appl. Mech. Eng. 410 (2023) 116007.
[45] R. Eggersmann, L. Stainier, M. Ortiz, S. Reese, Model-free data-driven computational mechanics enhanced by tensor voting, Comput. Methods Appl. Mech. Engrg. 373 (2021) 113499.
[46] S. Rezaei, A. Harandi, A. Moeineddin, B.X. Xu, S. Reese, A mixed formulation for physics- informed neural networks as a potential solver for engineering problems in heterogeneous domains: Comparison with finite element method, Comput. Methods Appl. Mech. Engrg. 401 (2022) 115616.
[47] S. Budday, G. Sommer, C. Birkl, C. Langkammer, J. Jaybaeck, Kohnert, M. Bauer, F. Paulsen, P. Steinmann, E. Kuhl, G.A. Holzapfel, Mechanical characterization of human brain tissue, Acta Biomater. 48 (2017) 319–340.
[48] S.R. St Pierre, K. Linka, E. Kuhl, Principal-stretch-based constitutive neural networks autonomously discover a subclass of Ogden models for human brain tissue, Brain Multiphys. 4 (2023) 100066.
[49] C.O. Horgan, M.G. Smayda, The importance of the second strain invariant in the constitutive modeling of elastomers and soft biomaterials, Mech. Mater. 51 (2012) 43–52.
[50] L.R.G. Treloar, Stress–strain data for vulcanised rubber under various types of deformation, Trans. Faraday Soc. 40 (1944) 59–70.
[51] S. Kakaletsis, E. Lejeune, M.K. Rausch, Can machine learning accelerate soft material parameter identification from complex mechanical test data? Biomech. Model. Mechanobiol. 22 (2023) 57–70.

[52] F. As'ad, P. Avery, C. Farhat, A mechanics-informed artificial neural network approach in data-driven constitutive modeling, Internat. J. Numer. Methods Engrg. 123 (2022) 2738–2759.
[53] D.K. Klein, M. Fernandez, R.J. Martin, P. Neff, O. Weeger, Polyconvex anisotropic hyperelasticity with neural networks, J. Mech. Phys. Solics 159 (2022) 105703.
[54] L. Noel, E. Kuhl, Modeling neurodegeneration in chronic traumatic encephalopathy using gradient damage models, Comput. Mech. 64 (2019) 1375–1387.
[55] T.C. Harris, R. de Rooij, E. Kuhl, The shrinking brain: Cerebral atrophy following traumatic brain injury, Ann. Biomed. Eng. 47 (2019) 1941–1959.
[56] D.W. Sproule, E.T. Campolettano, S. Rowson, Football helmet impact standards in relation to on-field impacts, Proc. Inst. Mech. Eng. Part P / J. Sports Eng. Technol. 231 (2017) 232–317.
[57] A. Goriely, M.G.D. Geers, G.A. Holzapfel, J. Jayamohan, A. Jerusalem, S. Sivaloganathan, W. Squier, J.A.W. van Dommelen, S. Waters, E. Kuhl, Mechanics of the brain: Perspectives, challenges, and opportunities, Biomech. Modeling Mechanobiol. 14 (2015) 931–965.
[58] K.D. Browne, X.H. Chen, D.F. Meaney, D.H. Smith, Mild traumatic brain injury and diffuse axonal injury in swine, J. Neurotrauma 28 (2011) 1747–1755.
[59] A. Post, T.B. Hoshizaki, M.D. Gilchrist, S. Brien, M. Cusimano, S. Marshall, Traumatic brain injuries. The influence of the direction of impact, Nuerosurgery 76 (2015) 81–91.
[60] J. Weickenmeier, C.A.M. Butler, P.G. Young, A. Goriely, E. Kuhl, The mechanics of decompressive craniectomy: Personalized simulations, Comput. Methods Appl. Mech. Engrg. 314 (2017) 180–195.
[61] S. Hartmann, P. Neff, Polyconvexity of generalized polynomial-type hyperelastic strain energy functions for near-incompressibility, Int. J. Solids Struct. 40 (2003) 2767–2791.
[62] A.J.M. Spencer, Theory of invariants, in: A.C. Eringen (Ed.), in: Continuum Physics, vol. 1, Academic Press, New York, 1971, pp. 239–353.
[63] T.C. Gasser, R.W. Ogden, G.A. Holzapfel, Hyperelastic modelling of arterial layers with distributed collagen fibre orientations, J. R. Soc. Interface 3 (2006) 15–35.
[64] L.M. Wang, K. Linka, E. Kuhl, Automated model discovery for muscle using constitutive recurrent neural networks, J. Mech. Behav. Biomed. Mater. 145 (2023) 106021.