NN-EUCLID

Deep-learning hyperelasticity without stress data

Thakolkaran, Prakash; Joshi, Akshay; Zheng, Yiwen; Flaschel, Moritz; De Lorenzis, Laura; Kumar, Siddhant

DOI
10.1016/j.jmps.2022.105076

Publication date
2022

Document Version
Final published version

Published in
Journal of the Mechanics and Physics of Solids

J. Mech. Phys. Solids 169 (2022) 105076

Contents lists available at ScienceDirect

Journal of the Mechanics and Physics of Solids

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

NN-EUCLID: Deep-learning hyperelasticity without stress data

Prakash Thakolkaran [a], Akshay Joshi [a], Yiwen Zheng [a], Moritz Flaschel [b], Laura De Lorenzis [b], Siddhant Kumar [a,*]

[a] Department of Materials Science and Engineering, Delft University of Technology, 2628 CD Delft, The Netherlands
[b] Department of Mechanical and Process Engineering, ETH Zürich, 8092 Zürich, Switzerland

ARTICLE INFO

Dataset link: https://euclid-code.github.io

Keywords:
Constitutive modeling
Unsupervised learning
Hyperelasticity
Neural network
Convexity

ABSTRACT

We propose a new approach for *unsupervised* learning of hyperelastic constitutive laws with physics-consistent deep neural networks. In contrast to supervised learning, which assumes the availability of stress–strain pairs, the approach only uses realistically measurable full-field displacement and global reaction force data, thus it lies within the scope of our recent framework for *Efficient Unsupervised Constitutive Law Identification and Discovery* (EUCLID) and we denote it as NN-EUCLID. The absence of stress labels is compensated for by leveraging a physics-motivated loss function based on the conservation of linear momentum to guide the learning process. The constitutive model is based on input-convex neural networks, which are capable of learning a function that is convex with respect to its inputs. By employing a specially designed neural network architecture, multiple physical and thermodynamic constraints for hyperelastic constitutive laws, such as material frame indifference, material stability, and stress-free reference configuration are automatically satisfied. We demonstrate the ability of the approach to accurately learn several hidden isotropic and anisotropic hyperelastic constitutive laws – including e.g., Mooney–Rivlin, Arruda–Boyce, Ogden, and Holzapfel models – without using stress data. For anisotropic hyperelasticity, the unknown anisotropic fiber directions are automatically discovered jointly with the constitutive model. The neural network-based constitutive models show good generalization capability beyond the strain states observed during training and are readily deployable in a general finite element framework for simulating complex mechanical boundary value problems with good accuracy.

1. Introduction

The merger of data-driven methods (often enabled or empowered by machine learning tools) and constitutive material modeling in solid mechanics is rapidly redefining the way we characterize and model complex material behavior. Traditionally, phenomenological models – parameterized by a few material-specific constants – are calibrated by iterative tuning to match the observations of simple mechanical tests, e.g., uni-/biaxial tension, torsion, and bending tests. Advances in experimental mechanics such as the advent of digital image correlation (DIC) and digital volume correlation (DVC) enabled more sophisticated approaches, such as finite element model updating (FEMU) (Marwala, 2010) or the virtual fields method (VFM) (Pierron and Grédiac, 2012), which rely upon the availability of full-field displacement data and exploit them for calibration of unknown parameters in a priori selected classical material models.

Modern data-driven methods aim to take a further step to overcome the limited expressive power of classical material models. In the (material) model-free paradigm proposed by Kirchdoerfer and Ortiz (2016) and first formulated for elasticity, a material's

* Corresponding author.
E-mail address: Sid.Kumar@tudelft.nl (S. Kumar).

https://doi.org/10.1016/j.jmps.2022.105076
Received 4 May 2022; Received in revised form 27 July 2022; Accepted 15 September 2022
Available online 21 September 2022

state of deformation is directly mapped to a stress state closest to the stress–strain pair available in the dataset (subject to physical compatibility constraints) (Kirchdoerfer and Ortiz, 2016; Ibañez et al., 2017; Kirchdoerfer and Ortiz, 2017; Conti et al., 2018; Nguyen and Keip, 2018; Eggersmann et al., 2019; Carrara et al., 2020; Karapiperis et al., 2021). Alternative approaches keep the concept of a material model and surrogate it by learning an approximate mapping between the strains and stresses (referring again to the easiest case of elasticity) using, e.g., sparse regression with feature engineering (Flaschel et al., 2021, 2022; Joshi et al., 2022; Wang et al., 2021), manifold learning methods and polynomial approximations (Ibañez et al., 2018, 2017; González et al., 2019b), Gaussian process regression (Rocha et al., 2021; Fuhg et al., 2022a), and artificial neural networks (NNs) (Ghaboussi et al., 1991; Fernández et al., 2021; Klein et al., 2022; Vlassis and Sun, 2021; Kumar et al., 2020; Bastek et al., 2022; Zheng et al., 2021; Mozaffar et al., 2019; Bonatti and Mohr, 2021; Vlassis et al., 2020; Kumar and Kochmann, 2021; As'ad et al., 2022; Liang et al., 2022), with the list being roughly in order of decreasing physical interpretability and increasing approximation power. Further hybrid approaches use data to construct automatic corrections to existing models (Ibáñez et al., 2019; González et al., 2019a). Physics-informed neural networks (PINNs) (Raissi et al., 2019) have been used for inverse estimation of constitutive models or differential equations, but they are limited to either constitutive models of known form with unknown parameters or unknown constitutive models but for one-dimensional cases (Huang et al., 2020; Tartakovsky et al., 2018; Haghighat et al., 2020; Chen and Gu, 2021). Consequently, they suffer from limited model expressive power — similar to parametric model calibration with classical FEMU and VFM techniques.

A common bottleneck in most state-of-the-art data-driven methods, be them based on NNs or not, is the requirement of a large number of stress labels, i.e., in machine learning jargon, the methods are *supervised*. In principle, stress labels can be computationally generated via multiscale homogenization techniques; however, this data generation approach relies upon modeling at the lower scale(s) (where model choice and calibration is even more challenging), is associated with a prohibitive computational cost, and its application is restricted to surrogate modeling and acceleration of multiscale simulations. Obtaining stress labels experimentally is a challenge too. Mechanical experiments from which stresses can be inferred, such as uni-/bi-axial tensile tests or bending tests, are too simple to sufficiently probe the high-dimensional stress–strain space and more complex experiments only provide boundary projections of stress tensors in form of force measurements. The recent recognition of this issue motivated the development of the data-driven identification method (Leygue et al., 2018; Dalémat et al., 2019; Cameron and Tasan, 2021), which formulates the inverse problem associated to the approach in Kirchdoerfer and Ortiz (2016). In this light, breaking away from the supervised setting, i.e., reliance on stress labels, is the key to developing truly data-efficient and practically applicable data-driven constitutive models.

To address the aforementioned limitations of classical approaches and supervised data-driven methods, the authors recently proposed a novel framework denoted as *Efficient Unsupervised Constitutive Law Identification and Discovery* (EUCLID) (Flaschel et al., 2021, 2022; Joshi et al., 2022). The method is unsupervised, i.e., it requires no stress data but only global reaction forces and full-field displacement data realistically obtainable through, e.g., DIC or DVC; it uses sparse regression over a large catalog of candidate functions to deliver interpretable constitutive laws given by parsimonious mathematical expressions. EUCLID departs from data-driven learning to achieve physics-driven learning, i.e., the need for stress labels is circumvented by enforcing that the full-field displacement data satisfy the conservation of linear momentum – possibly along with additional physics-based requirements. We recently demonstrated the performance of EUCLID in isotropic hyperelasticity (Flaschel et al., 2021), rate-independent plasticity (Flaschel et al., 2022), anisotropic hyperelasticity and elastodynamics (Joshi et al., 2022). In the latter contribution, we developed Bayesian-EUCLID, a probabilistic framework based on Bayesian learning which automatically discovers material models along with the related uncertainties.

Despite the advantages in terms of interpretability, models discovered by sparse regression are limited to the space spanned by the linear (or non-linear) combination of hand-crafted candidate functions in the feature library. On the other hand NNs, being universal approximators (Hornik et al., 1989), greatly surpass sparse regression in terms of size of the model space. At the expense of interpretability, they can, in principle, represent any generic constitutive behavior — with no limits on expressive power and no need to rely upon domain knowledge. However, the approximation power comes along with a high risk for overfitting; this issue and the high degree of ill-posedness due to the lack of stress labels hindered the development of unsupervised NN-based constitutive models thus far.

In this work, within the scope of the EUCLID framework, we demonstrate the first case of unsupervised NNs-based learning of constitutive models with no stress data. In this first step, we focus on isotropic and anisotropic hyperelasticity. To represent the constitutive model, we use input convex neural networks (ICNN) – recently developed by Amos et al. (2017) and applied to constitutive modeling (As'ad et al., 2022; Klein et al., 2022; Fuhg et al., 2022b; Huang et al., 2022). Their special architecture automatically guarantees material stability for the underlying strain energy density, material objectivity, and stress-free reference configuration. In the absence of stress labels, the NN is trained to a physics-constrained loss based on full-field displacement data satisfying conservation of linear momentum. The approach delivers unsupervised NN-based constitutive models that are physically admissible, can generalize well beyond the strain states observed in the full-field displacement data, and can automatically discover the unknown principal directions of anisotropy whenever applicable. We denote the approach as NN-EUCLID.

2. Unsupervised deep learning of hyperelastic constitutive laws

2.1. Problem setting

Consider a specimen of an unknown hyperelastic material subjected to quasi-static mechanical deformation with two-dimensional reference domain $\Omega \in \mathbb{R}^2$. A complex specimen geometry (e.g., plate with a hole) is specifically chosen to generate heterogeneous

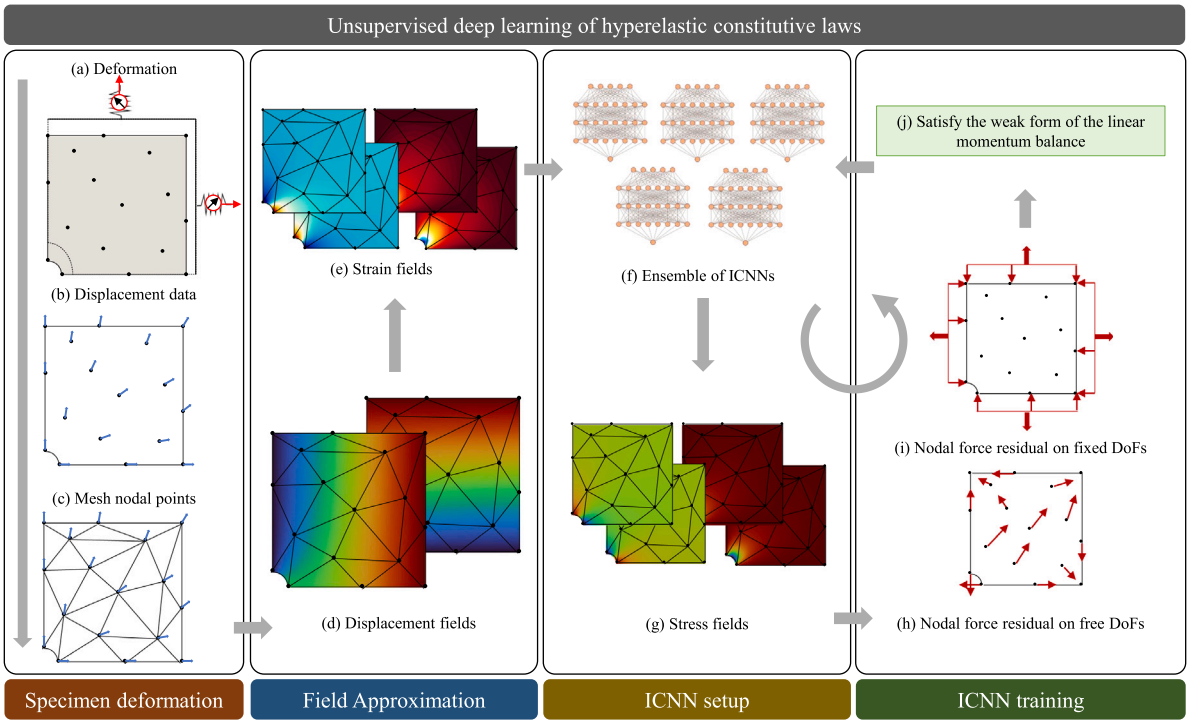

Fig. 1. Schematic of the approach for unsupervised deep learning of hyperelastic constitutive models. (a,b) Point-wise displacements and reaction forces are recorded for a hyperelastic specimen under quasi-static deformation. Using the point-wise displacements and a finite element mesh of the domain (c), continuous displacement and strain fields are obtained (d,e). An ensemble of physics-consistent ICNN-based constitutive models (f) map the strain fields to stress fields (evaluated at the quadrature point of each element) (g). The stress fields are used to compute the internal and external nodal forces (h,i). Based on the weak form of the conservation of linear momentum, the residual forces are minimized (j) — point-wise for the free degrees of freedom (h) and aggregated for the fixed degrees of freedom (under each set of Dirichlet constraint with a measured reaction force) (i). The optimization is carried out iteratively to train the parameters of the ICNN-based constitutive models (f).

and diverse strain states. The material is assumed to be homogeneous and possibly, anisotropic with unknown fiber orientations $\{\alpha_i \in [0,\pi) : i = 1,2,\dots\}$. Note that the fiber orientations have two-fold symmetry and, therefore, are constrained to $[0,\pi)$ instead of $[0,2\pi)$. We apply Dirichlet and Neumann conditions on the boundaries $\partial\Omega_u \subseteq \partial\Omega$ and $\partial\Omega_t = \partial\Omega \backslash \partial\Omega_u$, respectively. Without losing generality, we only consider displacement-controlled conditions (i.e., Dirichlet boundary conditions) in the subsequent discussions, while noting that applied forces in load-controlled conditions are equivalent to reaction forces in displacement-controlled conditions. The observed data consists of n_t snapshots of displacement measurements $\mathcal{U} = \{\boldsymbol{u}^{a,t} \in \mathbb{R}^2 : a = 1,\dots,n_n; t = 1,\dots,n_t\}$ at n_n points with coordinates $\mathcal{X} = \{\boldsymbol{X}^a \in \Omega : a = 1,\dots,n_n\}$ in the reference domain. Additionally, for each snapshot, n_β reaction forces $\{\boldsymbol{R}^{\beta,t} : \beta = 1,\dots,n_\beta; t = 1,\dots,n_t\}$ are assumed to be measured (using, e.g., load cells) at some but not necessarily all the sets of Dirichlet boundary conditions. In the subsequent discussion, we drop the superscript $(\cdot)^t$ for the sake of brevity; however the numerical procedure applies to all snapshots independently.

With this limited data, the objective is to learn the underlying constitutive model governing the stress–strain response. While most modern DIC setups can easily track the displacement of thousands to millions of points, it is impractical to have more than a few load cells attached to a specimen, i.e., $n_\beta \ll n_n$. In addition, since reaction forces are only boundary-aggregated projections of stress tensors, the lack of supervision, i.e., stress labels for constitutive model learning becomes apparent. Note that while the data contains two-dimensional displacements, the constitutive model that will be learnt is three-dimensional. The following sections describe the unsupervised learning approach using the aforementioned data. Fig. 1 illustrates a step-by-step schematic of the approach.

2.2. Approximation of the displacement field from point-wise data

We mesh the points $\mathcal{X}$ in the reference domain using linear triangular elements (with single quadrature point at the barycenter) to obtain the displacement field as

$$\boldsymbol{u}(\boldsymbol{X}) = \sum_{a=1}^{n_n} N^a(\boldsymbol{X})\, \boldsymbol{u}^a. \tag{1}$$

Here $N^a : \Omega \to \mathbb{R}$ is the shape function (corresponding to linear triangular elements) associated with the node at $\boldsymbol{X}^a$. The deformation gradient field is then approximated as

$$\boldsymbol{F}(\boldsymbol{X}) = \boldsymbol{I} + \sum_{a=1}^{n_n} \boldsymbol{u}^a \otimes \nabla N^a(\boldsymbol{X}), \tag{2}$$

where $\boldsymbol{I}$ is the identity matrix and ∇ is the gradient operator with respect to the reference coordinates.

2.3. NN-based constitutive model

A hyperelastic constitutive model is given by a strain energy density $W(\boldsymbol{F})$ from which the first Piola–Kirchhoff stress $\boldsymbol{P}(\boldsymbol{F})$ and the (incremental) tangent modulus $\mathbb{C}(\boldsymbol{F})$ are derived as

$$\boldsymbol{P}(\boldsymbol{F}) = \frac{\partial W(\boldsymbol{F})}{\partial \boldsymbol{F}} \quad \text{and} \quad \mathbb{C}(\boldsymbol{F}) = \frac{\partial \boldsymbol{P}(\boldsymbol{F})}{\partial \boldsymbol{F}}, \quad \forall \quad \boldsymbol{F} \in \mathrm{GL}_+(3), \tag{3}$$

respectively, where $\mathrm{GL}_+(3)$ denotes the set of all invertible second-order tensors with positive determinant. Therefore, the objective reduces to learning $W(\boldsymbol{F})$. However, any learned model for $W(\boldsymbol{F})$ must also satisfy the following physical and thermodynamic constraints:

- The stress must vanish in the case of no deformation (reference configuration), i.e.,

$$\boldsymbol{P}(\boldsymbol{F} = \boldsymbol{I}) = \boldsymbol{0}. \tag{4}$$

- The strain energy density must be objective, i.e.,

$$W(\boldsymbol{R}\boldsymbol{F}) = W(\boldsymbol{F}), \quad \forall \quad \boldsymbol{F} \in \mathrm{GL}_+(3), \quad \boldsymbol{R} \in \mathrm{SO}(3), \tag{5}$$

  where SO(3) denotes the 3D rotation group.
- For material stability, $W(\boldsymbol{F})$ must satisfy quasiconvexity (Morrey, 1952; Schröder, 2010), i.e.,

$$\int_{\mathcal{B}} W(\bar{\boldsymbol{F}} + \nabla \boldsymbol{w}) \mathrm{d}V \geq W(\bar{\boldsymbol{F}}) \int_{\mathcal{B}} \mathrm{d}V \quad \forall \quad \mathcal{B} \subset \mathbb{R}^3, \quad \bar{\boldsymbol{F}} \in \mathrm{GL}_+(3), \quad \boldsymbol{w} \in C_0^\infty(\mathcal{B}) \ \ (\text{i.e., } \boldsymbol{w} = 0 \text{ on } \partial\mathcal{B}). \tag{6}$$

  However, enforcing quasiconvexity is often analytically/numerically intractable (Kumar et al., 2019). Since polyconvexity implies quasiconvexity, the quasiconvexity constraint is commonly relaxed to that of polyconvexity (Ball, 1976; Schröder, 2010). $W(\boldsymbol{F})$ is polyconvex if and only if there exists a convex function $\mathcal{P}$ such that

$$W(\boldsymbol{F}) = \mathcal{P}(\boldsymbol{F}, \mathrm{Cof}\ \boldsymbol{F}, \det \boldsymbol{F}). \tag{7}$$

To satisfy objectivity in (5), a straightforward (and common) way of proceeding is to express the strain energy density as a function of the right Cauchy–Green deformation tensor $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F}$, or, equivalently, of the Green–Lagrange strain tensor $\boldsymbol{E} = (\boldsymbol{F}^T\boldsymbol{F} - \boldsymbol{I})/2$, i.e., as $W(\boldsymbol{E}(\boldsymbol{F}))$, which represents a sufficient condition for objectivity. However, $\boldsymbol{E}$ is not necessarily convex in $\boldsymbol{F}$, hence this choice makes it difficult to enforce polyconvexity in $\boldsymbol{F}$ using (7). Following Yang Gao et al. (2017) and As'ad et al. (2022), we notice that the convexity of W in $\boldsymbol{E}$ implies its local convexity in $\boldsymbol{F}$, which in turn ensures local material stability. Note that local convexity in $\boldsymbol{F}$ is preferable to global convexity in $\boldsymbol{F}$, as the latter would preclude the non-uniqueness of the solution in situations where such non-uniqueness is physical (such as in buckling) and would prevent the satisfaction of the so-called growth condition, i.e. the condition of the strain energy density becoming infinite for $\det \boldsymbol{F} \to 0^+$.

In light of the aforementioned constraints, we consider an ansatz for the strain energy density of the form:

$$W(\boldsymbol{F}) = \underbrace{W^{\mathrm{NN}}_{\mathcal{Q},\mathcal{A}}(\boldsymbol{E}(\boldsymbol{F}))}_{\text{ICNN model}} + \underbrace{W^0}_{\substack{\text{Energy}\\ \text{correction}}} + \underbrace{\boldsymbol{H} : \boldsymbol{E}}_{\substack{\text{Stress}\\ \text{correction}}}, \tag{8}$$

which is adapted with modifications from As'ad et al. (2022) and Klein et al. (2022). Here, $W^{\mathrm{NN}}_{\mathcal{Q},\mathcal{A}}$ represents an ICNN model (Amos et al., 2017) with parameter sets $\mathcal{Q}$ and $\mathcal{A}$ (details described below), W^0 is a constant scalar, and $\boldsymbol{H}$ is a symmetric 3×3 constant matrix. W^0 offsets the energy density such that it vanishes at zero deformation ($\boldsymbol{F} = \boldsymbol{I}$), i.e.,

$$W(\boldsymbol{I}) = 0 \quad \Longrightarrow \quad W^0 = -\ W^{\mathrm{NN}}_{\mathcal{Q},\mathcal{A}}\Big|_{\boldsymbol{F}=\boldsymbol{I}}. \tag{9}$$

From (3), the first Piola–Kirchhoff stress is obtained as

$$\boldsymbol{P}(\boldsymbol{F}) = \frac{\partial W^{\mathrm{NN}}_{\mathcal{Q},\mathcal{A}}(\boldsymbol{E}(\boldsymbol{F}))}{\partial \boldsymbol{F}} + \boldsymbol{F}\boldsymbol{H}. \tag{10}$$

Similar to W^0, $\boldsymbol{H}$ is set such that the stress vanishes at $\boldsymbol{F} = \boldsymbol{I}$ (see (4)), i.e.,

$$\boldsymbol{P}(\boldsymbol{I}) = \boldsymbol{0} \quad \Longrightarrow \quad \boldsymbol{H} = -\left.\frac{\partial W^{\mathrm{NN}}_{\mathcal{Q},\mathcal{A}}}{\partial \boldsymbol{F}}\right|_{\boldsymbol{F}=\boldsymbol{I}}. \tag{11}$$

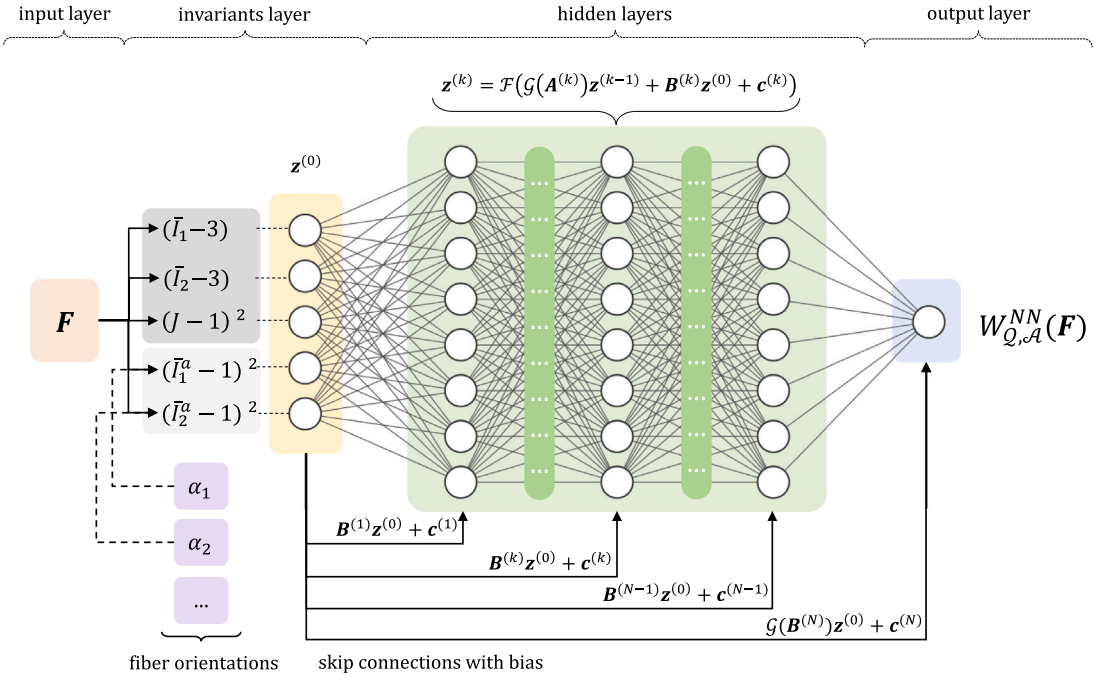

Fig. 2. Schematic of the ICNN-based constitutive model for hyperelasticity.

The tangent modulus is derived from (10) as

$$\mathbb{C}_{ijkl} = \frac{\partial P_{ij}(\boldsymbol{F})}{\partial F_{kl}} = \frac{\partial^2 W^{\text{NN}}_{\mathcal{Q},\mathcal{A}}(\boldsymbol{E}(\boldsymbol{F}))}{\partial F_{ij}\partial F_{kl}} + \delta_{ik}H_{lj}, \tag{12}$$

where we use the Einstein summation convention over the subscripts and δ denotes the Kronecker delta. Note that W^0 and $\boldsymbol{H}$ in the energy and stress corrections, respectively, are updated in situ during each iterative step of the neural network training (discussed later in Section 2.4), i.e., the corrections are not applied *a posteriori*. Further description regarding the training of $W^{\text{NN}}_{\mathcal{Q},\mathcal{A}}$ is provided in Algorithm 1.

$W^{\text{NN}}_{\mathcal{Q},\mathcal{A}}$ – enabled by the special ICNN architecture (Amos et al., 2017) – outputs a strain energy density that exhibits local convexity with respect to the input deformation gradient. Here, we present the architecture of $W^{\text{NN}}_{\mathcal{Q},\mathcal{A}}$ followed by the description of each layer (see Fig. 2 for schematic).

$$\text{Input layer}: \quad \boldsymbol{F} \tag{13a}$$

$$\text{Invariants layer}: \quad \boldsymbol{z}^{(0)} = \left[\tilde{I}_1 - 3,\ \tilde{I}_2 - 3,\ (J-1)^2,\ (\tilde{I}_1^a - 1)^2,\ (\tilde{I}_2^a - 1)^2,\ \dots\right]^T \tag{13b}$$

$$\text{with trainable parameters: } \alpha_1 \in [0,\pi),\ \alpha_2 \in [0,\pi),\ \dots,$$

$$\text{First hidden layer of size } d_1: \quad \boldsymbol{z}^{(1)} = \mathcal{F}\left(\mathcal{G}\left(\boldsymbol{A}^{(1)}\right)\boldsymbol{z}^{(0)} + \boldsymbol{B}^{(1)}\boldsymbol{z}^{(0)} + \boldsymbol{c}^{(1)}\right) \tag{13c}$$

$$\text{with trainable parameters: } \boldsymbol{A}^{(1)} \in \mathbb{R}^{d_1 \times \left|\boldsymbol{z}^{(0)}\right|},\ \boldsymbol{B}^{(1)} \in \mathbb{R}^{d_1 \times \left|\boldsymbol{z}^{(0)}\right|},\ \boldsymbol{c}^{(1)} \in \mathbb{R}^{d_1},$$

$$\vdots$$

$$k\text{th hidden layer of size } d_k: \quad \boldsymbol{z}^{(k)} = \mathcal{F}\Big(\underbrace{\mathcal{G}\left(\boldsymbol{A}^{(k)}\right)\boldsymbol{z}^{(k-1)}}_{\text{convex-linear}} + \underbrace{\boldsymbol{B}^{(k)}\boldsymbol{z}^{(0)}}_{\text{skip-connect}} + \underbrace{\boldsymbol{c}^{(k)}}_{\text{bias}}\Big) \tag{13d}$$

$$\text{with trainable parameters: } \boldsymbol{A}^{(k)} \in \mathbb{R}^{d_k \times d_{k-1}},\ \boldsymbol{B}^{(k)} \in \mathbb{R}^{d_k \times \left|\boldsymbol{z}^{(0)}\right|},\ \boldsymbol{c}^{(k)} \in \mathbb{R}^{d_k},$$

$$\vdots$$

$$\text{Output } (N\text{th}) \text{ layer of size } d_N = 1: \quad W^{\text{NN}}_{\mathcal{Q},\mathcal{A}} = \boldsymbol{z}^{(N)} = \mathcal{G}\left(\boldsymbol{A}^{(N)}\right)\boldsymbol{z}^{(N-1)} + \mathcal{G}\left(\boldsymbol{B}^{(N)}\right)\boldsymbol{z}^{(0)} + \boldsymbol{c}^{(N)} \tag{13e}$$

$$\text{with trainable parameters: } \boldsymbol{A}^{(N)} \in \mathbb{R}^{d_N \times d_{N-1}},\ \boldsymbol{B}^{(N)} \in \mathbb{R}^{d_N \times \left|\boldsymbol{z}^{(0)}\right|},\ \boldsymbol{c}^{(N)} \in \mathbb{R}^{d_N}.$$

$\mathcal{Q} = \{\boldsymbol{A}^{(k)}, \boldsymbol{B}^{(k)}, \boldsymbol{c}^{(k)} : k = 1, \dots, N\}$ and $\mathcal{A} = \{\alpha_i : i = 1, 2, \dots\}$ denote the set of all the trainable ICNN weights and unknown fiber orientations, respectively. The invariants layer receives the deformation gradient $\boldsymbol{F}$ from the input layer and computes strain

invariants as

$$\begin{aligned} \text{volumetric invariant}: &\quad J = \det(\boldsymbol{F}) = I_3^{1/2} \\ \text{isotropic deviatoric invariants}: &\quad \tilde{I}_1 = J^{-2/3} I_1, \quad \tilde{I}_2 = J^{-4/3} I_2, \\ \text{anisotropic deviatoric invariants}: &\quad \tilde{I}_1^a = J^{-2/3} \left(\boldsymbol{a}^{(1)} \cdot \boldsymbol{C}\boldsymbol{a}^{(1)}\right), \quad \tilde{I}_2^a = J^{-2/3} \left(\boldsymbol{a}^{(2)} \cdot \boldsymbol{C}\boldsymbol{a}^{(2)}\right), \quad \dots \quad , \end{aligned} \tag{14}$$

where

$$I_1 = \mathrm{tr}(\boldsymbol{C}), \qquad I_2 = \frac{1}{2}\left[\mathrm{tr}(\boldsymbol{C})^2 - \mathrm{tr}(\boldsymbol{C}^2)\right], \qquad I_3 = \det(\boldsymbol{C}) \tag{15}$$

are the principal invariants of $\boldsymbol{C}$ (equivalently, $\boldsymbol{E} = (\boldsymbol{C} - \boldsymbol{I})/2$). Here, we assume a deviatoric–volumetric split for modeling compressible hyperelasticity. For an anisotropic hyperelastic material with fiber orientations $\{\alpha_i : i = 1, 2, \dots\}$, $\boldsymbol{a}^{(i)} = (\cos\alpha_i, \sin\alpha_i, 0)^T$ denotes the ith fiber direction with third component being normal to the plane of Ω. The fiber orientations in $\mathcal{A}$ are unknown and, therefore, treated as trainable parameters of the ICNN model. The invariants are further shifted by appropriate scalar values – in some cases, also squared – such that both their value and the respective derivatives with respect to $\boldsymbol{F}$ are zero in the case of no deformation, i.e., at $\boldsymbol{F} = \boldsymbol{I}$.

The strain invariants – now vectorized as $\boldsymbol{z}^{(0)}$ – are sequentially transformed through $k = 1, \dots, (N-1)$ hidden layers. In any kth layer of size $d_k > 0$, the output of the previous layer ($\boldsymbol{z}^{(k-1)}$) undergoes a *convex linear transformation* via $\mathcal{G}\left(\boldsymbol{A}^{(k)}\right)$, where $\boldsymbol{A}^{(k)} \in \mathbb{R}^{d_k \times d_{k-1}}$ is a trainable weight matrix and $\mathcal{G}$ denotes a nonlinear *non-negative* function applied element-wise. The layer also includes a *skip connection* that adds the output of the invariants layer, i.e., $\boldsymbol{z}^{(0)}$, after linear transformation via trainable weight matrix $\boldsymbol{B}^{(k)} \in \mathbb{R}^{d_k \times \left|\boldsymbol{z}^{(0)}\right|}$ and bias vector $\boldsymbol{c}^{(k)} \in \mathbb{R}^{d_k}$. Finally, the sum of the convex linear transformation and the skip connection are passed through a nonlinear activation function $\mathcal{F}$ that is *convex* and *non-decreasing* and acts element-wise on the input. The output/N^{th} layer consists of summing the convex linear transforms of both the output of the penultimate layer ($\boldsymbol{z}^{(N-1)}$) and the invariants layer ($\boldsymbol{z}^{(0)}$); however, the output does not pass through the activation function $\mathcal{F}$.

The aforementioned architectural choices are motivated by the following reasons. Since $W^{\text{NN}}_{\mathcal{Q},\mathcal{A}}$ only depends on the strain invariants, which are objective, the objectivity constraint (5) is identically satisfied. In contrast to a classical feed-forward neural network, the combination of convex linear transformations enabled via non-negative $\mathcal{G}$ in the hidden layers (non-negative weighted sums of convex functions are convex) and non-decreasing convex activation function $\mathcal{F}$ ensures that the ICNN output is convex in the strain invariants (see Boyd and Vandenberghe (2004) and Amos et al. (2017) for proofs) and consequently, the local material stability ensured by the overall strain energy density in (8). Without loss of generality, $\mathcal{G}$ and $\mathcal{F}$ are chosen based on the *softplus* function (As'ad et al., 2022) as

$$\mathcal{G}(x) = c_{\mathcal{G}} \underbrace{\log(1 + e^x)}_{\text{softplus}} \qquad \text{and} \qquad \mathcal{F}(x) = c_{\mathcal{F}} \underbrace{(\log(1 + e^x))^2}_{\text{squared-softplus}}, \tag{16}$$

respectively, for any $x \in \mathbb{R}$; where $c_{\mathcal{G}} > 0$ and $c_{\mathcal{F}} > 0$ are hyperparameters. The C^∞ order of continuity of the softplus function ensures that the strain energy density and its derivatives – stress and tangent – are sufficiently smooth and continuous.[1] The squaring of the softplus function in $\mathcal{F}$ mitigates the problem of vanishing second derivative of the softplus function. This not only enables computing the tangent of the strain energy density (see (12)), but also prevents vanishing gradients for backpropagation-based NN training (Goodfellow et al., 2016).[2] The skip connections are added to avoid vanishing gradients, overfitting, and accuracy saturation (or even degradation) as the NN architecture becomes deeper (with increasing number of hidden layers) (He et al., 2016).

2.4. Unsupervised learning of constitutive models

In the absence of energy density/stress labels, we use the fact that the observed data must satisfy the conservation of linear momentum to guide the learning process (Flaschel et al., 2021) and estimate the model parameters $\mathcal{Q}$ and $\mathcal{A}$ of the ICNN. In the case of negligible body forces and quasi-static loading (i.e., negligible inertia), the weak form of the linear momentum balance in the reference domain Ω is given by

$$\int_\Omega \boldsymbol{P} : \nabla \boldsymbol{v} \; \mathrm{d}V - \int_{\partial\Omega_t} \hat{\boldsymbol{t}} \cdot \boldsymbol{v} \; \mathrm{d}S = 0 \quad \forall \;\; \text{admissible} \;\; \boldsymbol{v}, \tag{17}$$

where $\hat{\boldsymbol{t}}$ is the surface traction acting on $\partial\Omega_t$ and $\boldsymbol{v}$ is an admissible test function that is sufficiently regular and vanishes on the Dirichlet boundary $\partial\Omega_u$. Note that $\hat{\boldsymbol{t}} = \boldsymbol{0}$ in the context of purely displacement-controlled testing. We use the weak form of the linear momentum balance instead of the strong form because the latter requires double spatial derivatives, which are more sensitive to noise in the data. For the scope of this work, we consider a plane strain assumption and note that the method may be extended to the three-dimensional case with, e.g., Digital Volume Correlation data. While a plane stress assumption is more appropriate for samples with small out-of-plane thickness and would require a reformulation of the proposed method, for the purpose of developing

[1] Activation functions such as $\mathrm{ReLU}(\cdot) = \max(\cdot, 0)$ and its variants, while common in deep learning, are not recommended in this context as they are not twice differentiable.

[2] Since the loss function is based on the first derivative of the NN, training requires computing the second derivative for optimization of the NN parameters; see Section 2.4 and (22).

and testing the method and given that we consider artificial data here, the plane strain assumption is just as good as any other — provided it is consistent with the way the benchmark data are generated.

Let $\mathcal{D} = \{(a,i) : a = 1,\dots,n_n; i = 1,2\}$ denote the set of degrees of freedom such that the displacement of the ath node in the ith direction, i.e., u_i^a, is included in the observed data. We further divide all the degrees of freedom into the following mutually exclusive subsets of $\mathcal{D}$,

- $\mathcal{D}^{\text{free}}$: degrees of freedom not subjected to Dirichlet constraints, i.e., u_i^a is not fixed;
- $\mathcal{D}_\beta^{\text{fix}}$ (with $\beta = 1,\dots,n_\beta$): degrees of freedom under Dirichlet constraints that collectively contribute to the observed reaction force R^β.

Similar to the displacement field approximation (1), we approximate the test function as

$$\boldsymbol{v}(\boldsymbol{X}) = \sum_{a=1}^{n_n} N^a(\boldsymbol{X})\boldsymbol{v}^a, \qquad \text{with} \qquad v_i^a = 0 \quad \forall \quad (a,i) \in \bigcup_{\beta=1}^{n_\beta} \mathcal{D}_\beta^{\text{fix}} \tag{18}$$

for admissibility. Subsequently, the momentum balance in (17) reduces to

$$\sum_{a=1}^{n_n} v_i^a f_i^a = 0, \qquad \text{where} \qquad f_i^a = \underbrace{\int_\Omega P_{ij}\, \nabla_j N^a \, \mathrm{d}V}_{\text{internal force}} - \underbrace{\int_{\partial\Omega_t} \hat{t}_i N^a \, \mathrm{d}S}_{\text{external force}}. \tag{19}$$

f_i^a can be interpreted as the difference between the internal force arising from the constitutive model and the external force due to the applied tractions on Neumann boundaries. Note that the integrals in (19) are computed using numerical quadrature defined by the mesh over the point set $\mathcal{X}$.

Since the test functions are arbitrary, (19) must be satisfied at each free degree of freedom (where v_i^a does not vanish) independently, i.e.,

$$f_i^a = 0 \qquad \forall \quad (a,i) \in \mathcal{D}^{\text{free}}. \tag{20}$$

However, at the fixed degrees of freedom, the internal and external forces are balanced by the reaction force from the Dirichlet constraints. Point-wise reaction forces cannot be measured experimentally, and are hence assumed to be unavailable. Instead, only global reaction forces along the boundary segments are known. Therefore, the aggregated force balance for each measured reaction force yields

$$\sum_{(a,i)\in\mathcal{D}_\beta^{\text{fix}}} f_i^a = R^\beta \qquad \forall \qquad \beta = 1,\dots,n_\beta, \tag{21}$$

where the summation is performed over the point-wise forces across all the degrees of freedom in the βth Dirichlet constraint, i.e., $\mathcal{D}_\beta^{\text{fix}}$. Recall that $(\cdot)^t$ was dropped for the sake of brevity and the above force balance constraints must be satisfied for all the data snapshots at $t = 1,\dots,n_t$.

The objective now reduces to learning the constitutive model $W(\boldsymbol{F})$ in (8) (equivalently, $\boldsymbol{P}(\boldsymbol{F})$ in (10)), parameterized by the ICNN weights $\mathcal{Q}$ and anisotropic fiber orientations $\mathcal{A}$, such that the full-field displacement and reaction force data satisfy the purely physics-based constraints (20) and (21). We pose this inverse problem as the minimization of a loss function based on the force balance residuals given by

$$\mathcal{Q},\mathcal{A} \leftarrow \arg\min_{\mathcal{Q},\mathcal{A}} \sum_{t=1}^{n_t} \left[\underbrace{\sum_{(a,i)\in\mathcal{D}^{\text{free}}} \left(f_i^{a,t}\right)^2}_{\text{free degrees of freedom}} + \underbrace{\sum_{\beta=1}^{n_\beta} \left(R^{\beta,t} - \sum_{(a,i)\in\mathcal{D}_\beta^{\text{fix}}} f_i^{a,t} \right)^2}_{\text{fixed degrees of freedom}} \right]. \tag{22}$$

The optimization problem in (22) is solved via gradient-based minimization (see Appendix A for implementation details). Since each transformation – starting from the observed displacement and reaction force data to the constitutive model and the physics-constrained loss – is differentiable, the gradient of the objective function with respect to the trainable parameters in $\mathcal{Q}$ and $\mathcal{A}$ is easily computed. This is enabled by *automatic differentiation* (Baydin et al., 2017) – wherein each mathematical operation is tracked in a computational graph and later the gradients are computed via the chain rule for differentiation.

The inverse problem in (22) is highly ill-posed as it admits multiple solutions due to two reasons. *(i)* While the ICNN output is convex with respect to the invariants of $\boldsymbol{C}$, the loss function in (22) is highly non-convex with respect to the trainable parameters of the ICNN. *(ii)* The absence of strain energy density or stress labels and thus the training with indirect labels in the form of the physics-constrained loss implies that several different models can likely explain the limited observed data. For these reasons, the optimization problem in (22) admits several local minima and the final solution is sensitive to the initial state of the trainable parameters, i.e., $\mathcal{Q}$ and $\mathcal{A}$. To address this issue, we consider an ensemble of ICNNs — each trained independently with different initial states chosen randomly; we only accept those models which have distinctively low loss values and reject those for which the loss function gets trapped in bad minima with high loss values (see Appendix A for implementation details). In Section 3.2, we show that despite the multiplicity in the solution/model space, the constitutive responses of the accepted models are accurate and consistent with each other and the proposed approach is robust to this ill-posedness.

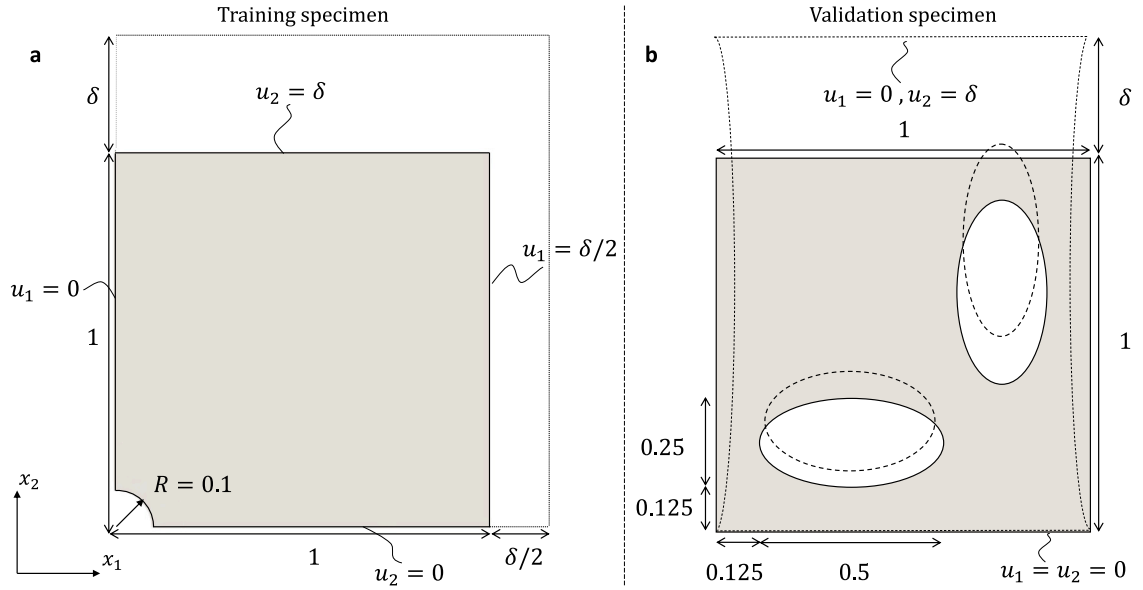

Fig. 3. (a) *Training specimen*: geometry and boundary conditions of a square plate with a hole in the bottom-left corner, subjected to displacement-controlled asymmetric biaxial tension. The resulting data (full-field displacements and reaction forces) including noise are used to train the ICNN-based constitutive models. (b) *Validation specimen*: geometry and boundary conditions of a square plate with two asymmetric elliptical holes, subjected to displacement-controlled uniaxial tension. The specimen is only used for validation of the ICNN-based constitutive models; no related data are used during the training stage. All lengths and displacements are normalized with respect to the side length of the undeformed specimen.

3. Numerical benchmarks

3.1. Data generation

Adapting the benchmarks of Flaschel et al. (2021), we emulate a DIC experiment by using the finite element method (FEM) to simulate the behavior of a hyperelastic square plate with a hole (shown in Fig. 3a) under plane-strain conditions. The specimen undergoes displacement-controlled asymmetric biaxial tension with symmetry boundary conditions on the bottom and left boundaries and loading parameter δ. We employ linear triangular elements and the nodal displacements and reaction forces (horizontal force on the left and right boundaries, vertical force on the upper and lower boundaries) are recorded for n_t load steps. Compared to simple uni-/biaxial tension or torsion tests, the combination of this geometry and loading generates sufficiently diverse and heterogeneous strain states to train a generalizable constitutive model with just a single experiment (see Section 3.2 and Fig. 11). Henceforth, the specimen is referred to as the *training specimen*.

We generate the synthetic data with FEM using the following instances of well-known physical and phenomenological material models.

1. Neo-Hookean (NH) model:

$$W(\boldsymbol{F}) = 0.5(\tilde{I}_1 - 3) + 1.5(J-1)^2. \tag{23}$$

2. Isihara (IH) model (Isihara et al., 1951):

$$W(\boldsymbol{F}) = 0.5(\tilde{I}_1 - 3) + (\tilde{I}_2 - 3) + (\tilde{I}_1 - 3)^2 + 1.5(J-1)^2. \tag{24}$$

3. Haines-Wilson (HW) model (Haines and Wilson, 1979):

$$W(\boldsymbol{F}) = 0.5(\tilde{I}_1 - 3) + (\tilde{I}_2 - 3) + 0.7(\tilde{I}_1 - 3)(\tilde{I}_2 - 3) + 0.2(\tilde{I}_1 - 3)^3 + 1.5(J-1)^2. \tag{25}$$

4. Gent-Thomas (GT) model (Gent and Thomas, 1958):

$$W(\boldsymbol{F}) = 0.5(\tilde{I}_1 - 3) + \log(\tilde{I}_2/3) + 1.5(J-1)^2. \tag{26}$$

5. Arruda–Boyce (AB) model (Arruda and Boyce, 1993):

$$W(\boldsymbol{F}) = 2.5\sqrt{N_c}\left[\beta_c\lambda_c - \sqrt{N_c}\log\left(\frac{\sinh\beta_c}{\beta_c}\right)\right] - c_{\mathrm{AB}} + 1.5(J-1)^2, \tag{27}$$

where $\lambda_c = \sqrt{\tilde{I}_1/3}$, $\beta = \mathcal{L}^{-1}\left(\lambda_c/\sqrt{N_c}\right)$, and $\mathcal{L}^{-1}$ denotes the inverse Langevin function. The constants are set to $N_c = 28$ (denoting number of polymeric chain segments) and $c_{\mathrm{AB}} \approx 3.7910$, the latter is used to offset the energy density to zero at $\boldsymbol{F} = \boldsymbol{I}$ (since the Arruda–Boyce feature does not itself vanish for zero deformation).

6. Ogden model (OG) (Ogden and Hill, 1972):

$$W(\boldsymbol{F}) = \frac{\mu}{\eta}(\lambda_1^\eta + \lambda_2^\eta + \lambda_3^\eta - 3) \tag{28}$$

where $\lambda_1, \lambda_2, \lambda_3$ are the principal stretches and $\mu = \eta = 1.3$.

7. Anisotropic model with one fiber family at $\alpha_1 = 45°$ orientation (abbreviated as: AI45):

$$W(\boldsymbol{F}) = 0.5(\tilde{I}_1 - 3) + 0.75(J-1)^2 + 0.5(\tilde{I}_1^a - 1)^2. \tag{29}$$

8. Anisotropic model with one fiber family at $\alpha_1 = 60°$ orientation (abbreviated as: AI60):

$$W(\boldsymbol{F}) = 0.5(\tilde{I}_1 - 3) + 0.75(J-1)^2 + 0.5(\tilde{I}_1^a - 1)^2. \tag{30}$$

9. Anisotropic Holzapfel model (HZ) (Holzapfel et al., 2000)

$$W(\boldsymbol{F}) = 0.5(\tilde{I}_1 - 3) + \frac{\kappa_1}{2\kappa_2}\left[\exp\left(\kappa_2(\tilde{I}_1^a - 1)^2\right) + \exp\left(\kappa_2(\tilde{I}_2^a - 1)^2\right) - 2\right] + (J-1)^2 \tag{31}$$

with two fiber families at $\alpha_1 = +30°$ and $\alpha_2 = -30°$ orientations and constants $\kappa_1 = 0.9$ and $\kappa_2 = 0.8$.

For benchmarking purposes, the ICNN-based model is then trained to surrogate the aforementioned models by using the NN-EUCLID procedure in Section 2.4. Further implementation details are presented in Appendix A.

To account for the noise in real DIC experiments, we add artificial noise to the synthetic displacement data generated using the FEM simulations. The noise is mainly dependent on the pixel accuracy of the imaging device, which means that the noise level remains constant for every degree of freedom at every load step irrespective of the displacement. The noise is added to the displacement data as

$$u_i^{a,t} = u_i^{\text{fem},a,t} + \varepsilon_i^{a,t} \quad \text{with} \quad \varepsilon_i^{a,t} \sim \mathcal{N}(0, \sigma_u^2) \quad \forall \quad (a,i) \in \mathcal{D},\ t \in \{1, \dots, n_t\}, \tag{32}$$

where $u_i^{\text{fem},a,t}$ and $\varepsilon_i^{a,t}$ denote the true displacement from FEM and random noise, respectively, the latter sampled independently and identically from a zero-centered normal distribution with standard deviation σ_u. Following Flaschel et al. (2021), we consider two noise levels (normalized relative to specimen length): $\sigma_u = 10^{-4}$ (*low noise*) and $\sigma_u = 10^{-3}$ (*high noise*) that are representative of modern DIC setups. As per the common practice in DIC, each snapshot of the noisy displacement data is further spatially denoised. For the scope of this work, we use kernel ridge regression (KRR) for denoising; see Flaschel et al. (2021) for details on the denoising algorithm. The denoised and interpolated displacement data are then used for the ICNN training.

3.2. Results

The detailed ICNN architecture and training protocols for the benchmarks are presented in Appendix A. Due to the ill-posedness and sensitivity to initial guess of the NN weights, we create an ensemble of $n_e = 30$ ICNN models with the same architecture but each trained independently with different random weight initialization. The ICNN models with final loss value (see (22)) within 20% of the lowest loss value across the ensemble are accepted, while the remaining models are rejected. The filtering of models with losses higher than the acceptance criterion allows for a qualitative estimate of the confidence associated with the predicted lowest-loss ICNN model. For example, if it is observed that only the lowest-loss model lies within the acceptance criterion, then it can be inferred that the predicted lowest-loss ICNN model has high associated uncertainty. Conversely, if all the ICNN models in the ensemble lie within the acceptance criterion, the predicted lowest-loss ICNN model has high associated certainty. The choice of acceptance criterion also provides a commensurate qualitative estimate of robustness to the NN initialization and does not influence the lowest-loss (best) ICNN model in any way.

3.2.1. Model accuracy, generalization, and FEM deployment

For all the benchmarks (23)–(31), we evaluate the ICNN-based constitutive models against the ground truth models along six deformation paths: uniaxial tension (UT), uniaxial compression (UC), biaxial tension (BT), biaxial compression (BC), simple shear (SS) and pure shear (PS). The deformation gradients for these paths are given by:

$$\begin{aligned}
&\boldsymbol{F}^{\text{UT}}(\gamma) = \begin{bmatrix} 1+\gamma & 0 \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{F}^{\text{UC}}(\gamma) = \begin{bmatrix} \frac{1}{1+\gamma} & 0 \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{F}^{\text{BT}}(\gamma) = \begin{bmatrix} 1+\gamma & 0 \\ 0 & 1+\gamma \end{bmatrix}, \\
&\boldsymbol{F}^{\text{BC}}(\gamma) = \begin{bmatrix} \frac{1}{1+\gamma} & 0 \\ 0 & \frac{1}{1+\gamma} \end{bmatrix}, \quad \boldsymbol{F}^{\text{SS}}(\gamma) = \begin{bmatrix} 1 & \gamma \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{F}^{\text{PS}}(\gamma) = \begin{bmatrix} 1+\gamma & 0 \\ 0 & \frac{1}{1+\gamma} \end{bmatrix},
\end{aligned} \tag{33}$$

where $\gamma \in [0, 1]$ is a loading parameter. Note that these deformation paths are purely for evaluation purposes and in no way contribute to model training. In all the benchmarks, the strain energy density predictions from the accepted ICNN models – despite the multiplicity in solutions due to the ill-posedness – show good agreement with the ground truth; see Figs. 4, 5, 6 and Figs. 7, 8, 9 for the low and high noise cases, respectively. The ICNN models also show good agreement with the ground truth in terms of the first Piola–Kirchhoff stress along the six deformation paths for both noise levels; see Figs. C.12–C.17.

For further validation, we deploy the ICNN-based constitutive model within a finite element simulation framework. Without loss of generality, we use linear triangular elements and a Newton–Raphson-based nonlinear solver, which requires tangent computations.

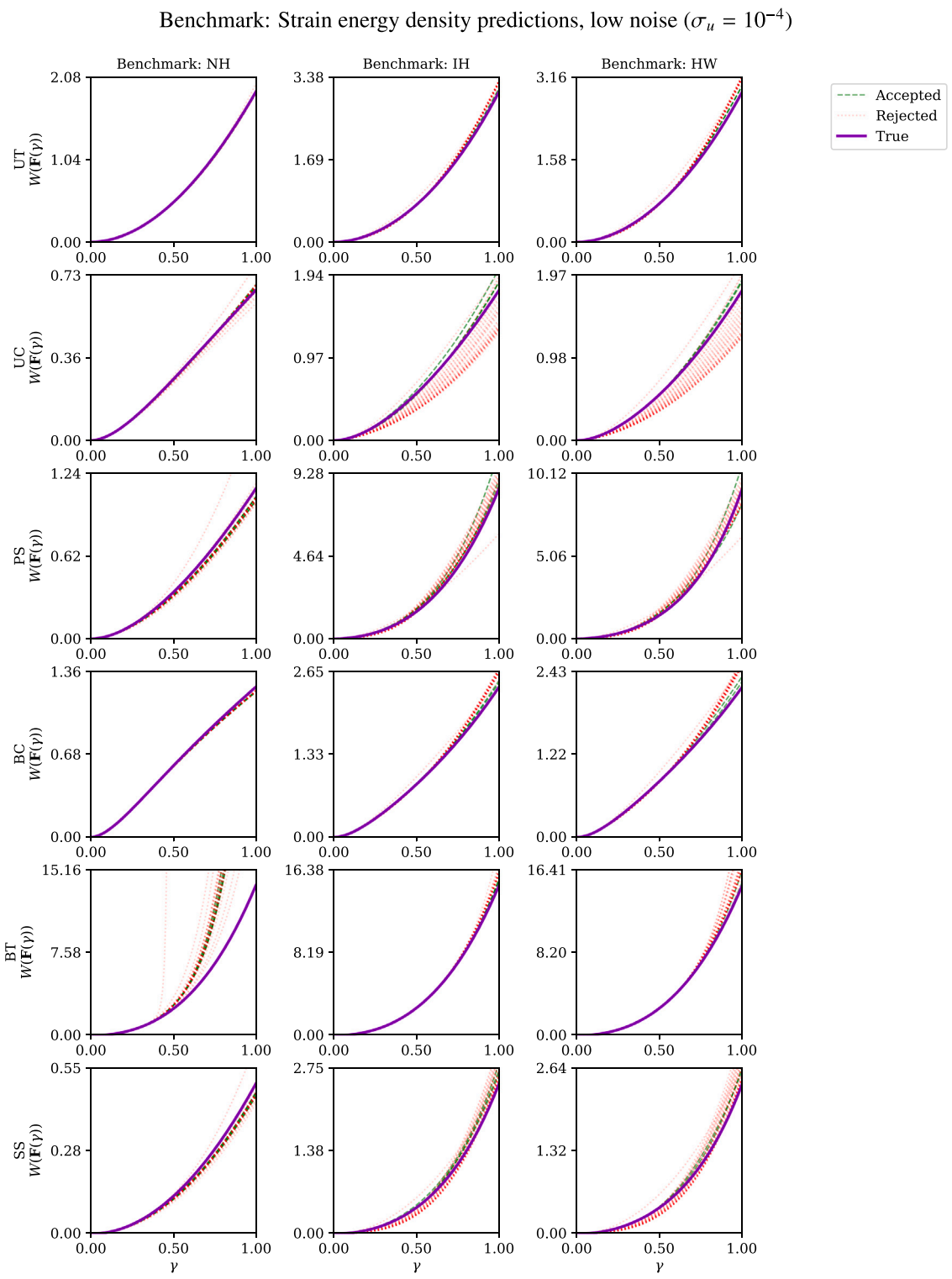

Fig. 4. Strain energy density $W(\boldsymbol{F}(\gamma))$ prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with low noise ($\sigma_u = 10^{-4}$) and benchmarks NH (23), IH (24), and HW (25). The constitutive response of the (hidden) true model is also shown for reference.

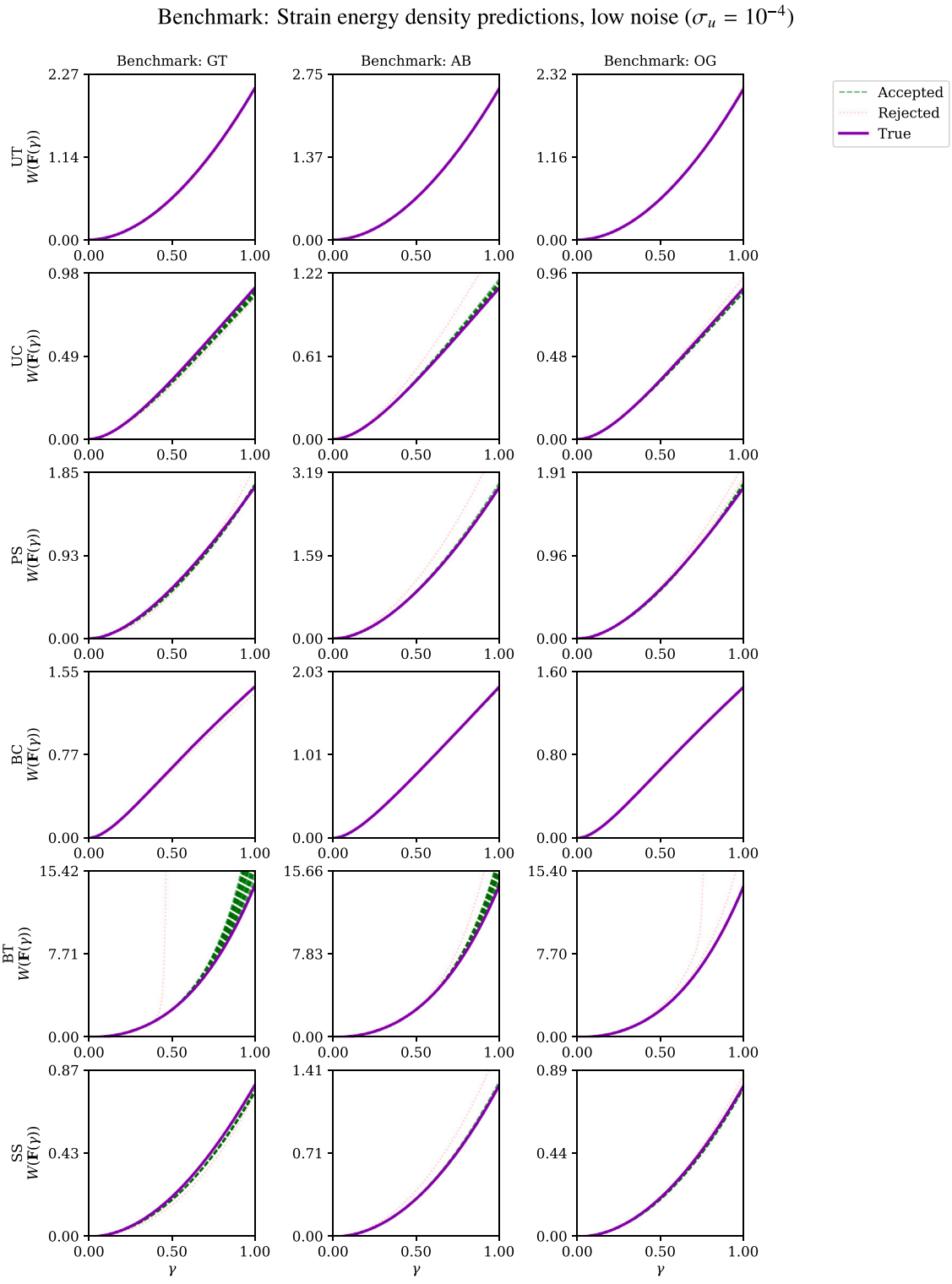

Fig. 5. Strain energy density $W(\boldsymbol{F}(\gamma))$ prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with low noise ($\sigma_u = 10^{-4}$) and benchmarks GT (26), AB (27), and OG (28). The constitutive response of the (hidden) true model is also shown for reference.

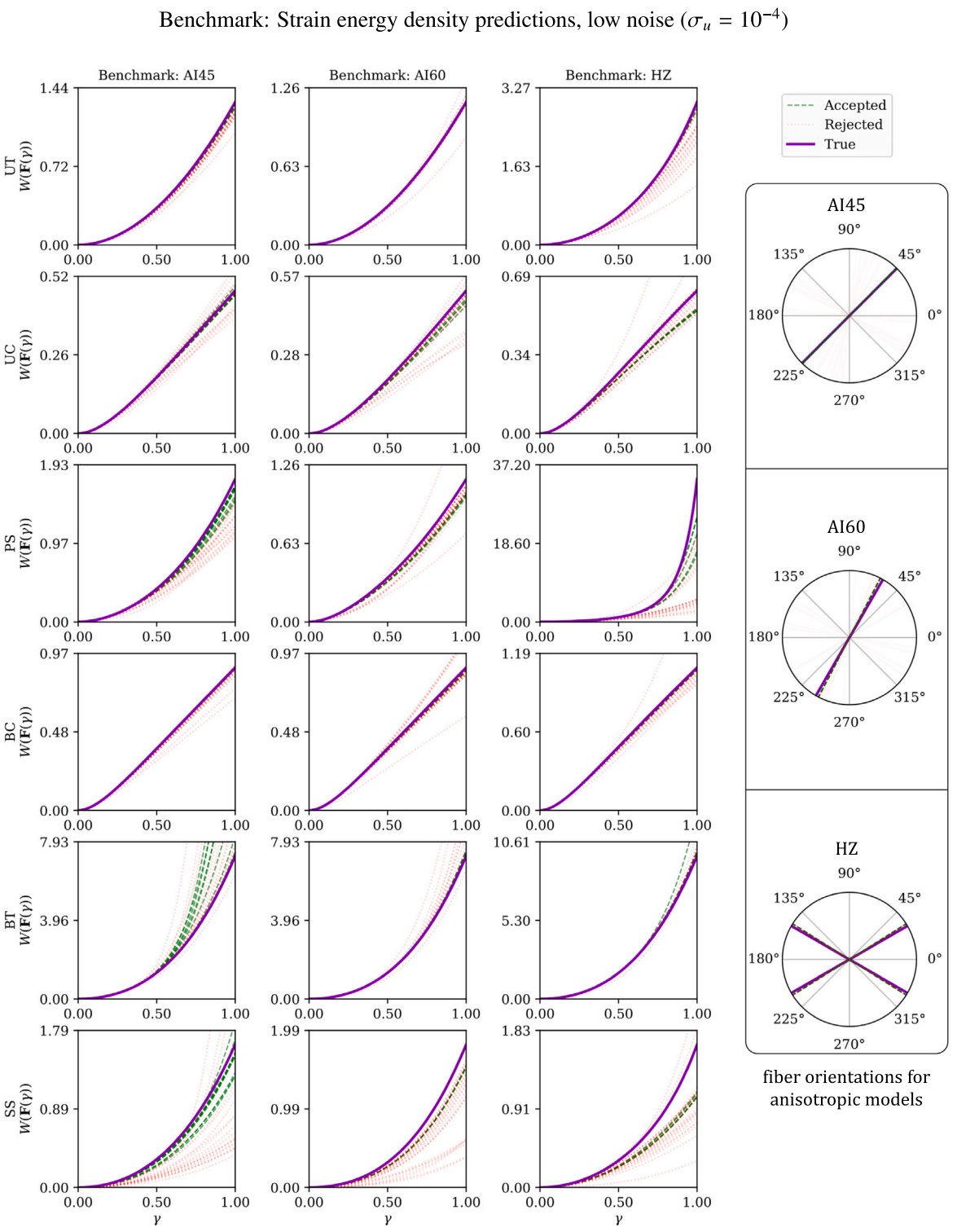

Fig. 6. Strain energy density $W(\boldsymbol{F}(\gamma))$ prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with low noise ($\sigma_u = 10^{-4}$) and benchmarks AI45 (29), AI60 (30), and HZ (31). The constitutive response of the (hidden) true model is also shown for reference. The insets show the fiber orientations discovered by the ICNN-based models and the fiber orientations in the (hidden) true model.

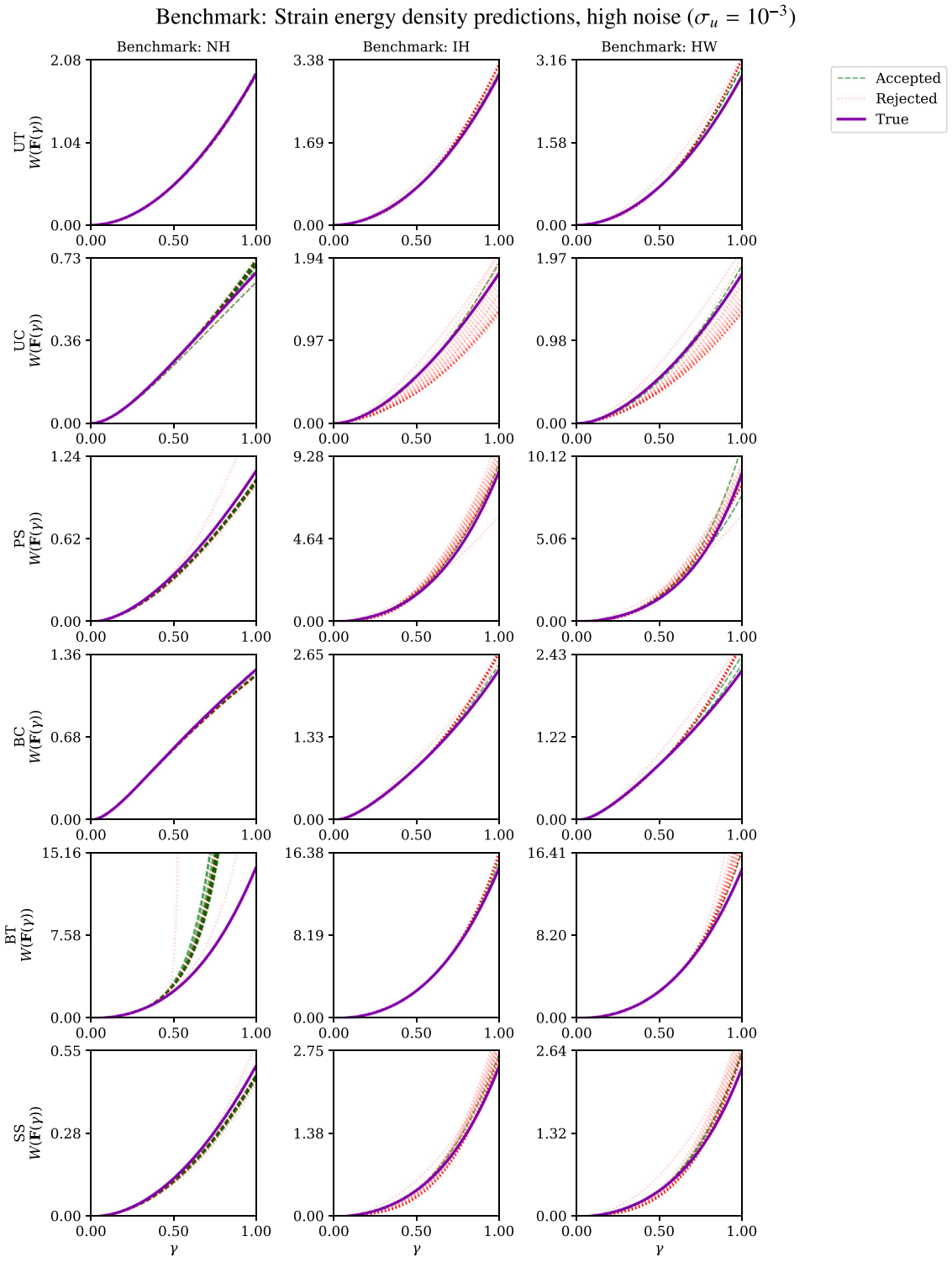

Fig. 7. Strain energy density $W(\boldsymbol{F}(\gamma))$ prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with high noise ($\sigma_u = 10^{-3}$) and benchmarks NH (23), IH (24), and HW (25). The constitutive response of the (hidden) true model is also shown for reference.

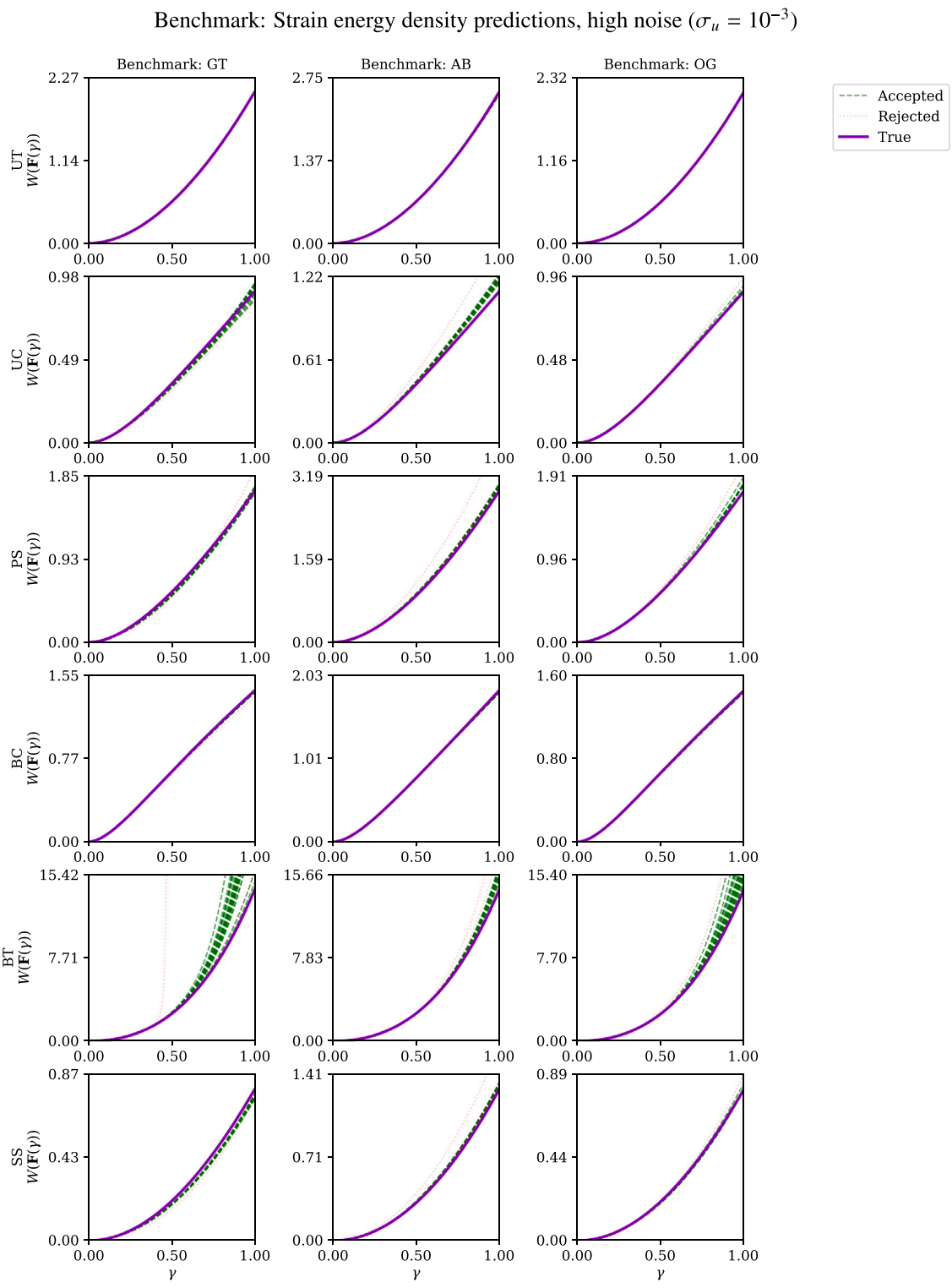

Fig. 8. Strain energy density $W(\boldsymbol{F}(\gamma))$ prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with high noise ($\sigma_u = 10^{-3}$) and benchmarks GT (26), AB (27), and OG (28). The constitutive response of the (hidden) true model is also shown for reference.

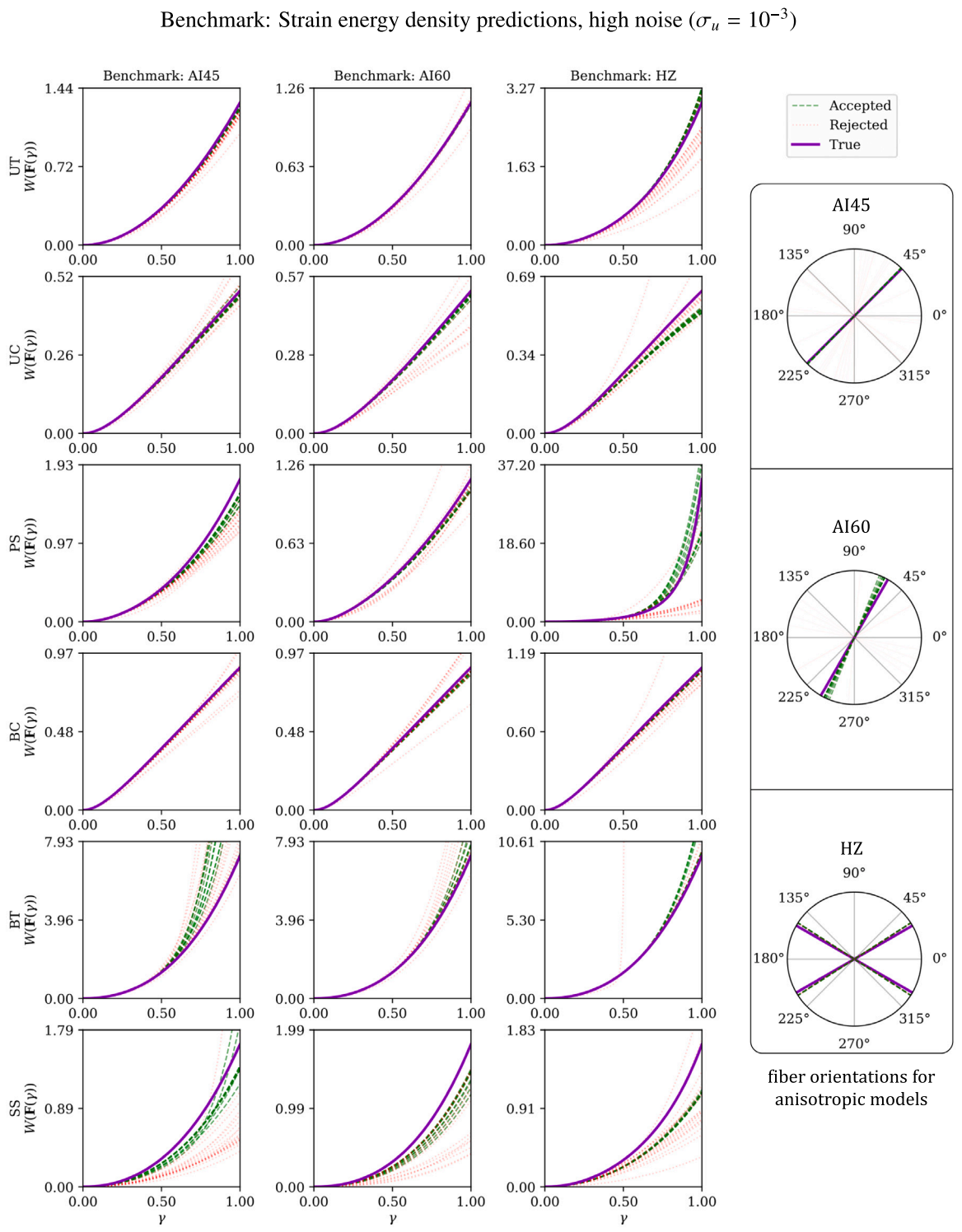

Fig. 9. Strain energy density $W(\boldsymbol{F}(\gamma))$ prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with high noise ($\sigma_u = 10^{-3}$) and benchmarks AI45 (29), AI60 (30), and HZ (31). The constitutive response of the (hidden) true model is also shown for reference. The insets show the fiber orientations discovered by the ICNN-based models and the fiber orientations in the (hidden) true model.

The element-wise stress and tangent modulus are computed using automatic differentiation via (10) and (12), respectively. To ensure that the ICNN models generalize to a simulation that is different from the one they are trained on, we consider a validation specimen

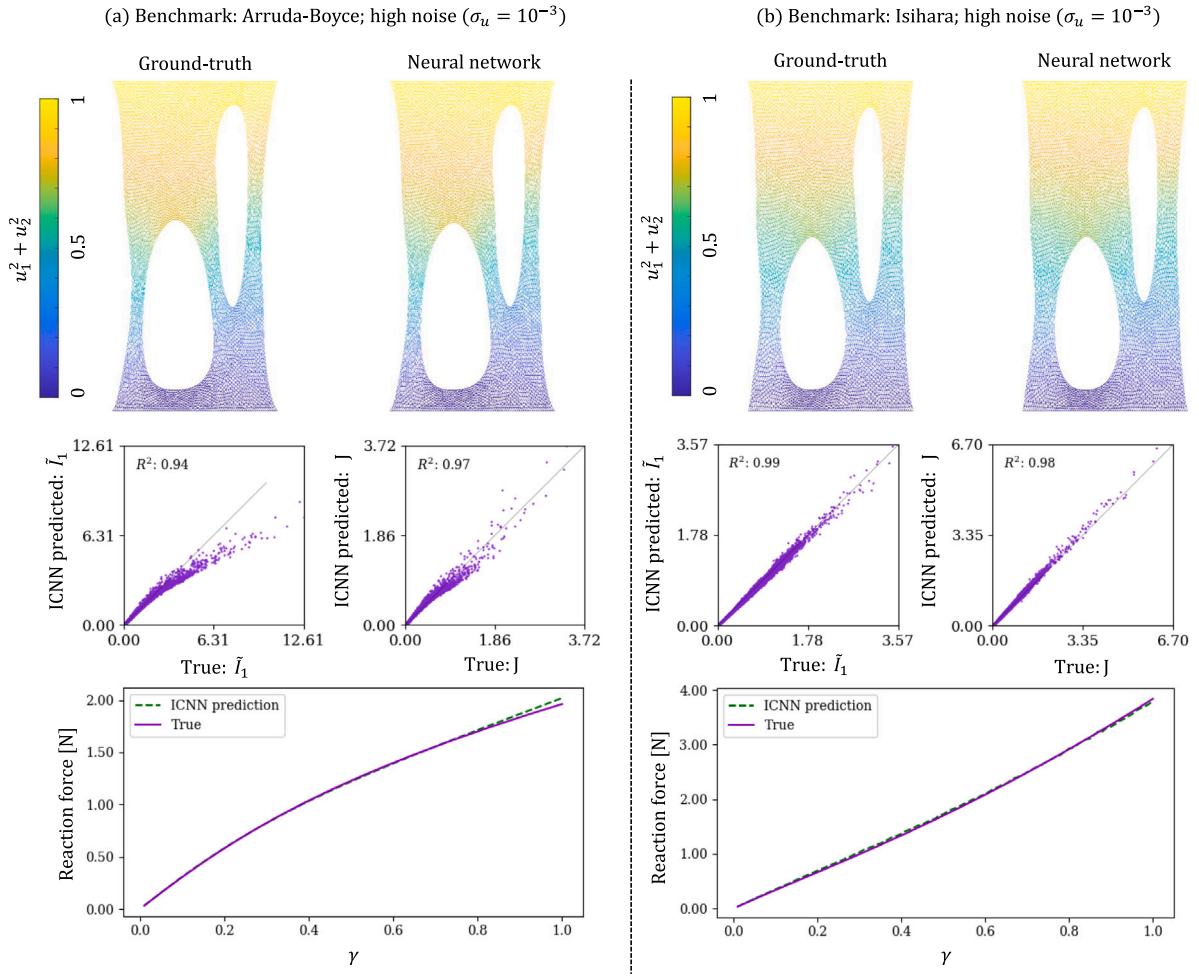

Fig. 10. Comparison of FEM solutions for the validation specimen (Fig. 3b) between the ground-truth and ICNN-based constitutive models for (a) Arruda–Boyce (27) and (b) Isihara (24) benchmarks. (*Top*) Deformed shape (shaded by displacement magnitude) of the specimen at loading parameter $\delta = 1$ (see Fig. 3b for details). (*Middle*) Predicted strain invariants ($\tilde{I}_1$ and J) at each quadrature point from the ICNN-based simulation vs. true strain invariants (from ground-truth simulation) in the validation specimen. In the ideal case, we expect each data point to lie on a line (indicated as red dashed line) with zero intercept and unit slope for perfect accuracy. The coefficient of determination (R^2) with respect to this line serves as a measure of accuracy. (*Bottom*) Comparison of the vertical reaction force on the top surface as a function of loading parameter δ, obtained from the ICNN-based simulation vs. the ground-truth simulation.

(Fig. 3b) with a more complex geometry than the training specimen, featuring two asymmetric elliptical holes and subjected to quasi-static uniaxial loading. For these validation simulations we use the ICNN material models with the lowest loss across the trained ensembles. Fig. 10 shows that the solutions to the above mechanical boundary value problem for two representative ground-truth constitutive models (Arruda–Boyce (27) and Isihara (24)) are in excellent agreement with those based on their respective ICNN ensemble-based constitutive models. This is quantitatively supported by good accuracy in the element-wise strain invariants (coefficient of determination or R^2 scores greater than 0.94 in the high noise case) as well as in the reaction forces between ground-truth and ICNN-based simulations. The ICNN architecture and smooth activations also mitigate spurious, oscillatory, and non-smooth artifacts in the overall stress–strain response and ensure smooth first and second derivatives of the strain energy density for robust deployment in FEM setting. In Appendix D, we demonstrate that a simple feed-forward NN without the input-convex architecture and smoothness is not suitable for constitutive modeling.

For visualizing the generalization capability of the ICNN models, we plot the two-dimensional projections of the principal strain invariants in the training specimen (across all load steps and all elements) – as demonstrated in Fig. 11. The strain invariants generated by the six deformation paths in (33) as well as from the validation specimens (across all load steps and all elements) are also added to the same plot. This confirms the generalization capability of the ICNN-based constitutive models, as the validation strain states go beyond the space of the strain states observed in the training specimen. Uniaxial tension, biaxial compression and uniaxial compression deformation paths are mostly encompassed by the training data, yielding almost perfect results for all the benchmarks. It also explains the significant mismatch between the ground truth and predicted strain energy densities – particularly

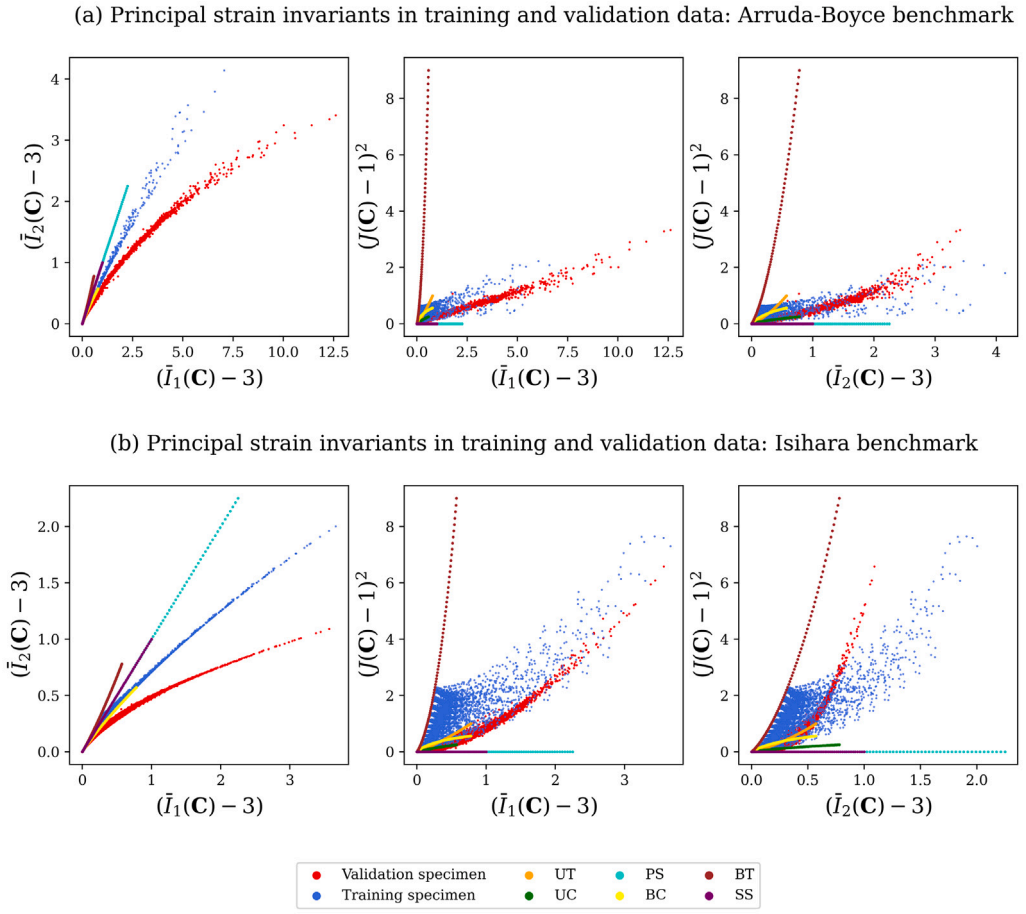

Fig. 11. Two-dimensional projections of the principal strain invariants $(\bar{I}_1-3)$, $(\bar{I}_2-3)$, and $(J-1)^2$ at all the elements in the training specimen (Fig. 3a) across all the loadsteps for (a) Arruda–Boyce (27) and (b) Isihara (24) benchmarks. Also shown are the principal strain invariants of the validation specimen (Fig. 3b) across all elements and loadsteps and the six evaluation deformation paths (33). Note that all the data shown here are based on ground-truth model without noise (including strain invariants of the validation specimen.).

for $\gamma > 0.5$ in simple shear, pure shear, and biaxial tension deformations – as these deformations are too large relative to those observed in the training specimen.

3.2.2. Learning hidden anisotropy

For the benchmarks with single fiber family ((29) and (30)) and Holzapfel model (31) with two fiber families, the fiber orientations in $\mathcal{A}$ are not known a priori but rather learned by the ICNNs as part of the training (see Algorithm 1 for pseudocode). Despite the unsupervised setting, the fiber orientations in $\mathcal{A}$ are accurately identified in both low and high noise cases — as demonstrated in Figs. 6 and 9, respectively.

4. Conclusion and outlook

We developed NN-EUCLID, i.e. an unsupervised learning approach for the encoding of material models in deep NNs with no reliance on stress data, and demonstrated its performance for isotropic and anisotropic hyperelasticity. The proposed framework only requires realistically measurable data, i.e., full-field displacements and global forces. The constitutive model is based on ICNNs whose special structure guarantees the physical requirements of material objectivity, (local) material stability, and stress-free reference configuration. The lack of stress labels in the unsupervised learning setting is compensated for by minimizing a physics-motivated loss which encapsulates the constraint that the observed displacement fields must satisfy the conservation of linear momentum. Through several benchmarks for isotropic and anisotropic hyperelasticity, we demonstrated the capability of accurately learning the underlying material behavior by only providing data from a single experiment under different noise levels that are representative of contemporary DIC setups. The ICNN framework also automatically discovers unknown principal directions of anisotropy (fiber orientations) in anisotropic hyperelastic materials. We further showed that the ICNN-based constitutive models can generalize to

Table A.1
List of parameters and hyperparameters used for the data generation and benchmarks.

Parameter	Notation	Value
Training specimen:		
Number of nodes in mesh for FEM-based data generation	–	63,601
Number of nodes in data available for learning	n_n	1441
Number of reaction force constraints	n_β	4
Number of data snapshots for NH, GT	n_t	3
Number of data snapshots for IH, HW, AI45, AI60, HZ	n_t	8
Number of data snapshots for AB	n_t	10
Number of data snapshots for OG	n_t	6
Loading parameter for NH, GT, IH, HW	δ	$\{0.1 \times t : t = 1, \dots, n_t\}$
Loading parameter for AB, OG, AI45, AI60, HZ	δ	$\{0.05 \times t : t = 1, \dots, n_t\}$
Validation specimen:		
Number of nodes in the mesh	–	4908
Loading parameter	δ	$\{0.01 \times t : t = 1, \dots, 100\}$
ICNN hyperparameters:		
Number of hidden layers	$N-1$	3
Number of neurons in each hidden layer $(k = 1, \dots, N-1)$	d_k	64
Dropout rate in hidden layers	–	20%
Scaling parameter in $\mathcal{G}$	$c_\mathcal{G}$	1.0
Scaling parameter in $\mathcal{F}$	$c_\mathcal{F}$	1/12
Number of models in the ensemble	n_e	30
Number of epochs	–	500
Learning rate schedule	–	cyclic
Base learning rate	–	0.001
Maximum learning rate	–	0.1
Learning rate cycle upward steps	–	50
Learning rate cycle downward steps	–	50

strain states and specimen geometries beyond those of the training specimen; and that they can be deployed within FEM simulations involving both stress and tangent computations with good accuracy (relative to the otherwise unknown ground-truth constitutive model).

In a broader view, the proposed framework offers a departure from the reliance on stress labels for deep learning of constitutive models. Future developments include experimental validation as well as the extension to unsupervised deep learning of inelastic material behavior.

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 and code availability

The data and codes generated in the current study are freely available at https://euclid-code.github.io.

Acknowledgments

MF and LDL would like to acknowledge funding by SNF, Switzerland through grant N. 200021_204316 “Unsupervised data-driven discovery of material laws”.

Appendix A. Implementation details

All the parameters and hyperparameters used in data generation and NN training are listed in Table A.1 with the following related explanations.

A.1. Data generation

All lengths and displacements are normalized with respect to the side length of the undeformed specimen. The training specimen is discretized with a high-resolution mesh (same as Flaschel et al. (2021)) consisting of 63,601 nodes. The noisy displacements of the training specimen are spatially-denoised using the same KRR denoiser and parameters as in Flaschel et al. (2021). For data efficiency, the denoised displacements are then projected onto a coarser mesh with $n_n = 1441$ nodes, which are used for training the ICNN models.

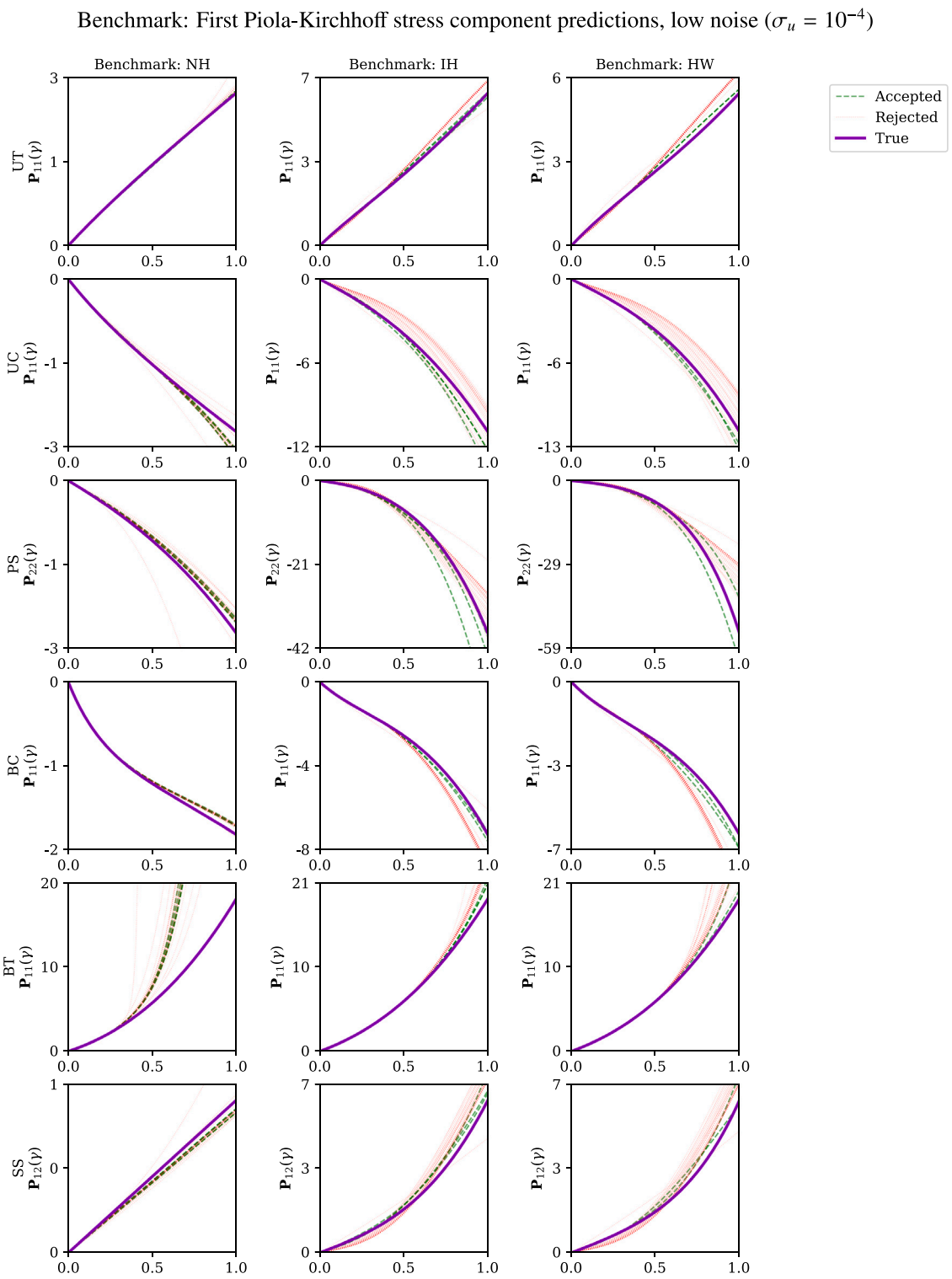

Fig. C.12. First Piola–Kirchhoff stress $\boldsymbol{P}(\boldsymbol{F}(\gamma))$ component prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with low noise ($\sigma_u = 10^{-4}$) and benchmarks NH (23), IH (24), and HW (25). The constitutive response of the (hidden) true model is also shown for reference.

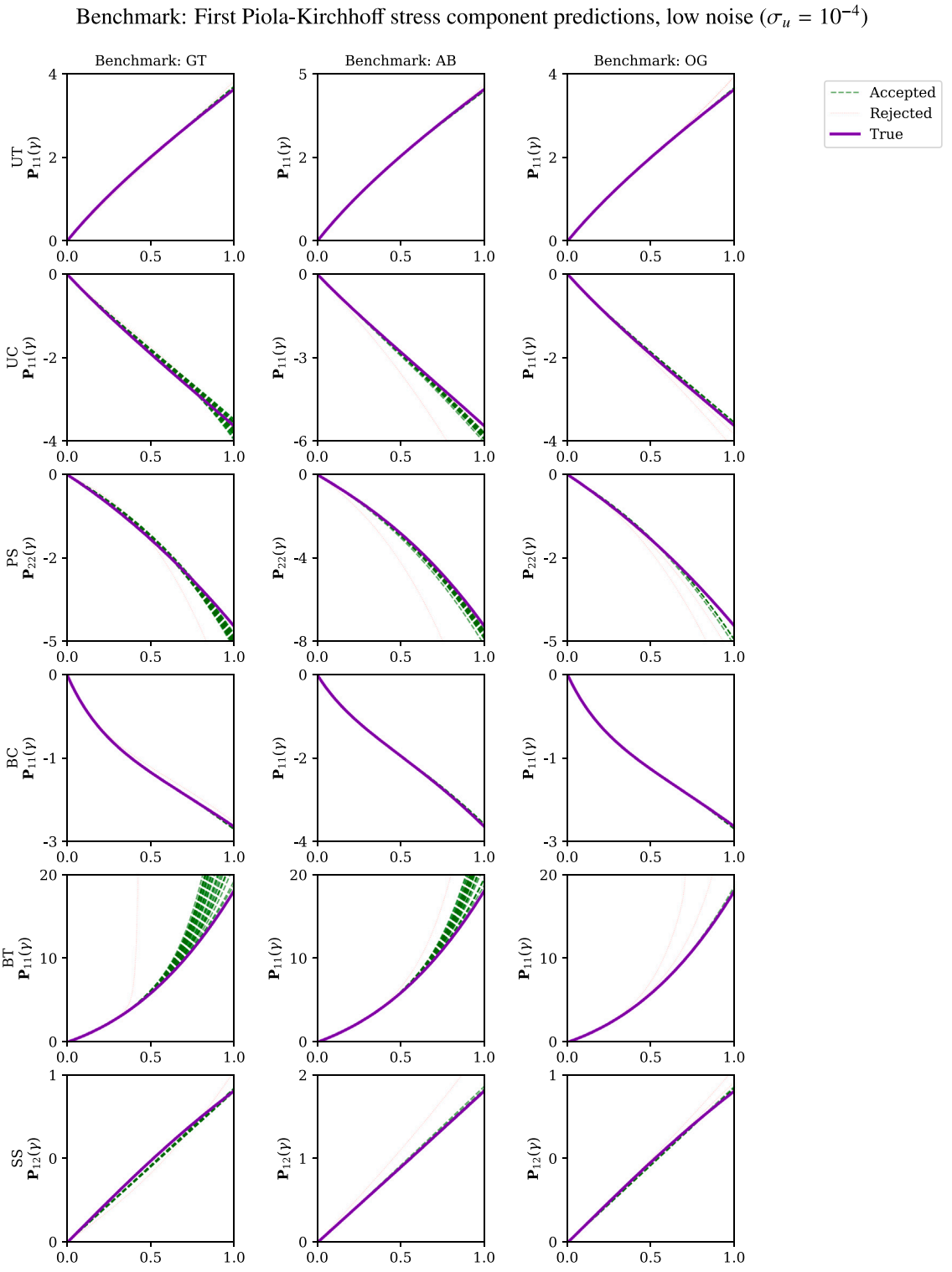

Fig. C.13. First Piola–Kirchhoff stress $\boldsymbol{P}(\boldsymbol{F}(\gamma))$ component prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with low noise ($\sigma_u = 10^{-4}$) and benchmarks GT (26), AB (27), and OG (28). The constitutive response of the (hidden) true model is also shown for reference.

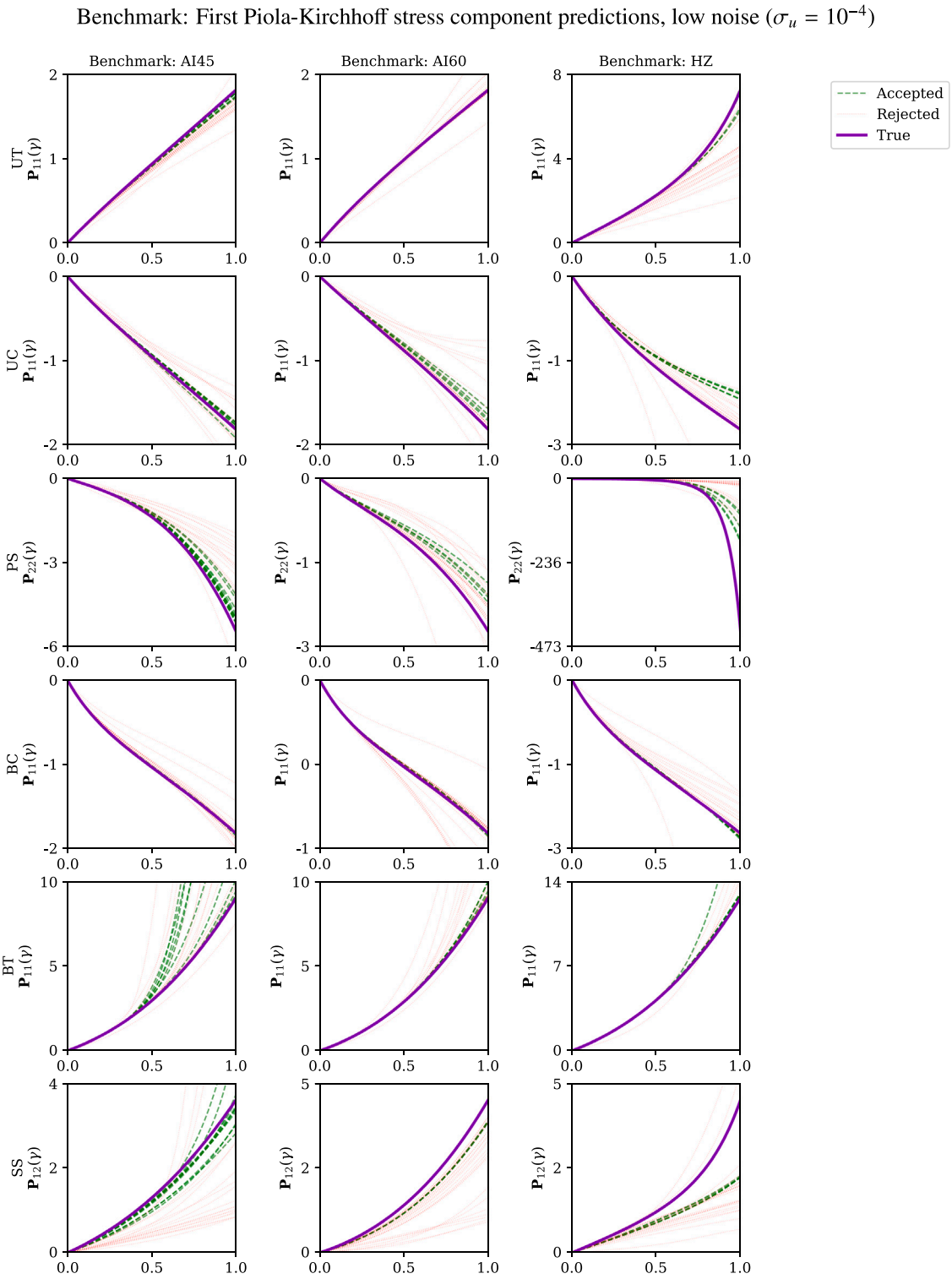

Fig. C.14. First Piola–Kirchhoff stress $\boldsymbol{P}(\boldsymbol{F}(\gamma))$ component prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with low noise ($\sigma_u = 10^{-4}$) and benchmarks AI45 (29), AI60 (30), and HZ (31). The constitutive response of the (hidden) true model is also shown for reference.

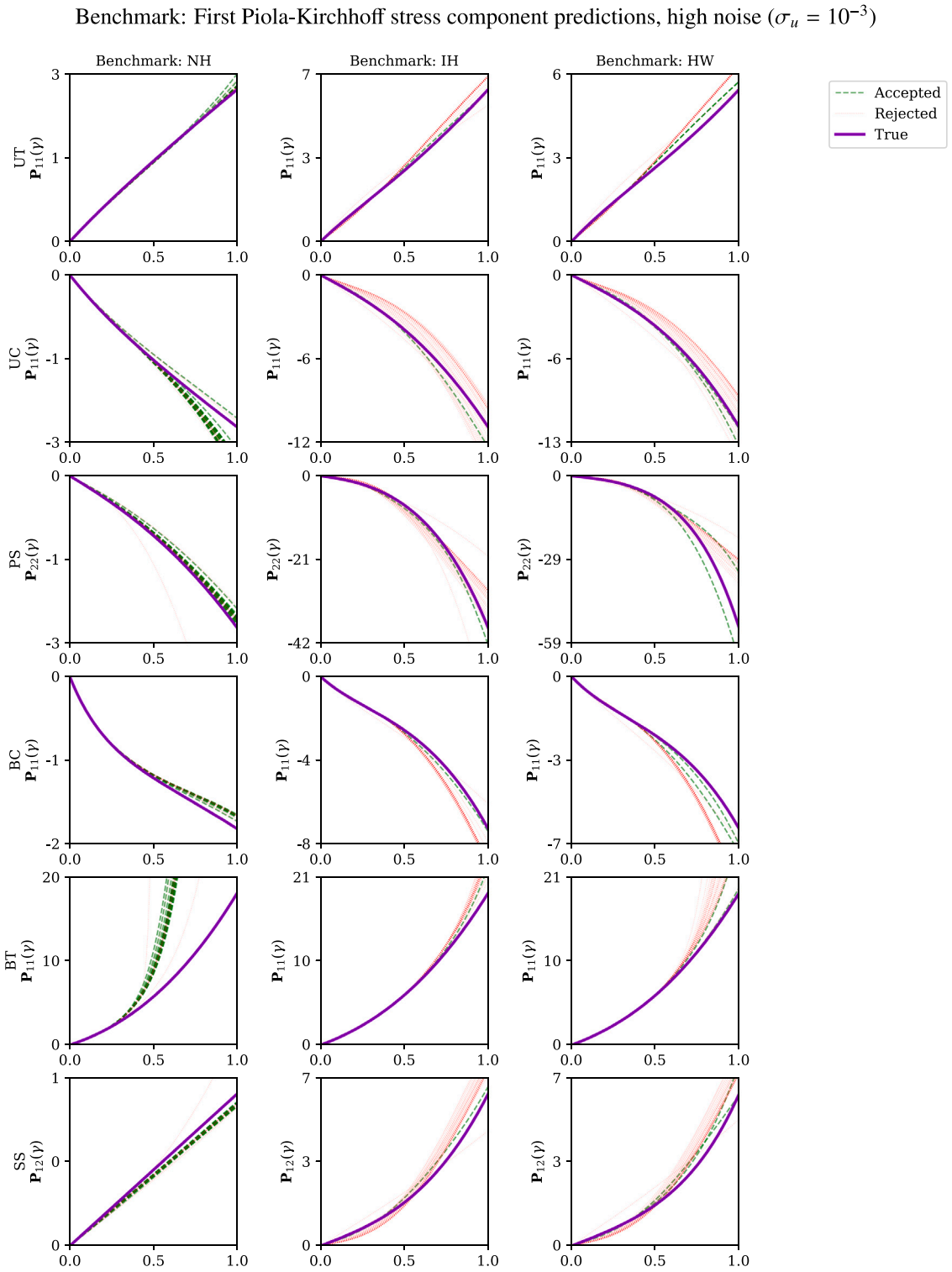

Fig. C.15. First Piola–Kirchhoff stress $\boldsymbol{P}(\boldsymbol{F}(\gamma))$ component prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with high noise ($\sigma_u = 10^{-3}$) and benchmarks NH (23), IH (24), and HW (25). The constitutive response of the (hidden) true model is also shown for reference.

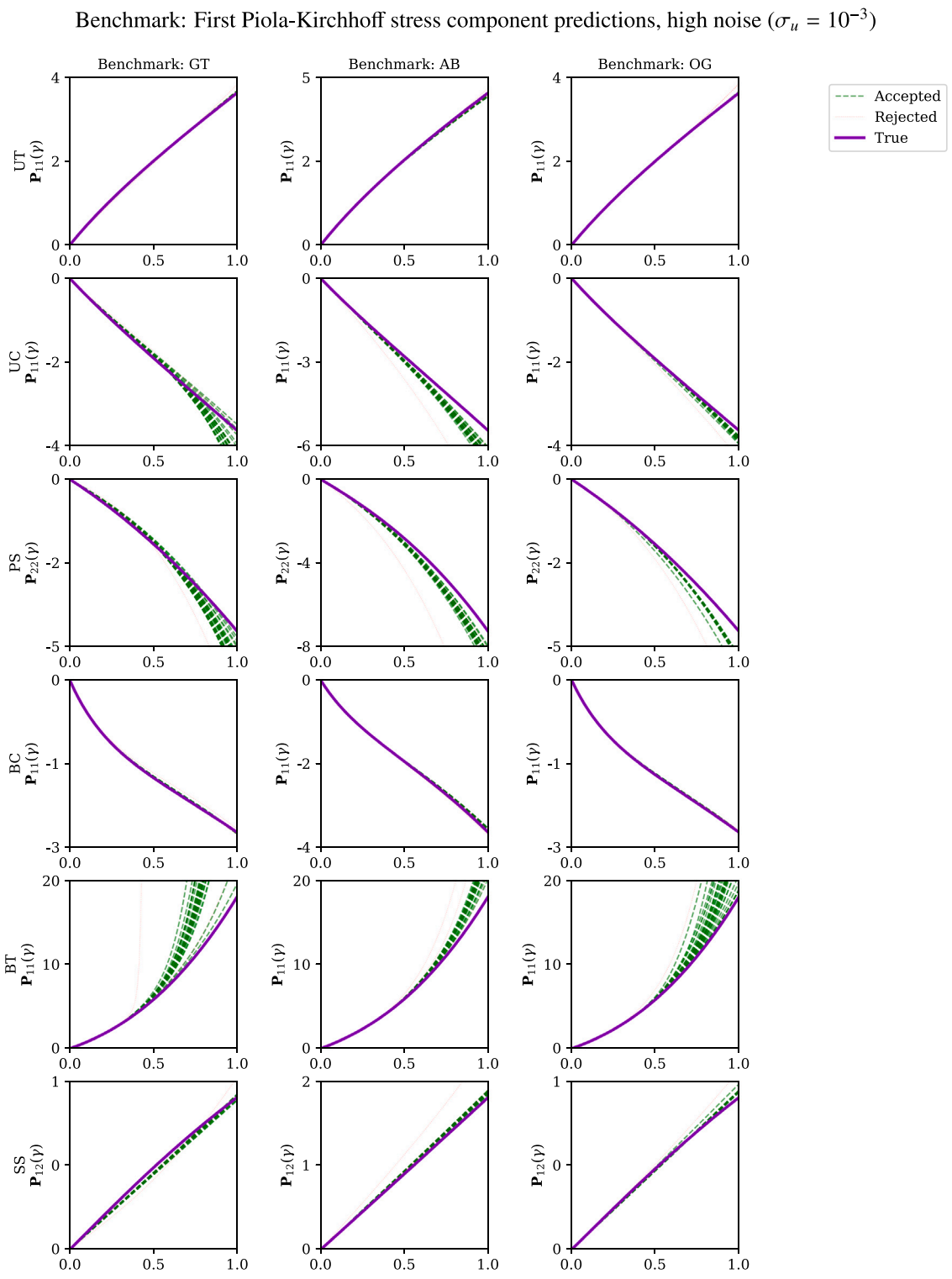

Fig. C.16. First Piola–Kirchhoff stress $\boldsymbol{P}(\boldsymbol{F}(\gamma))$ component prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with high noise ($\sigma_u = 10^{-3}$) and benchmarks GT (26), AB (27), and OG (28). The constitutive response of the (hidden) true model is also shown for reference.

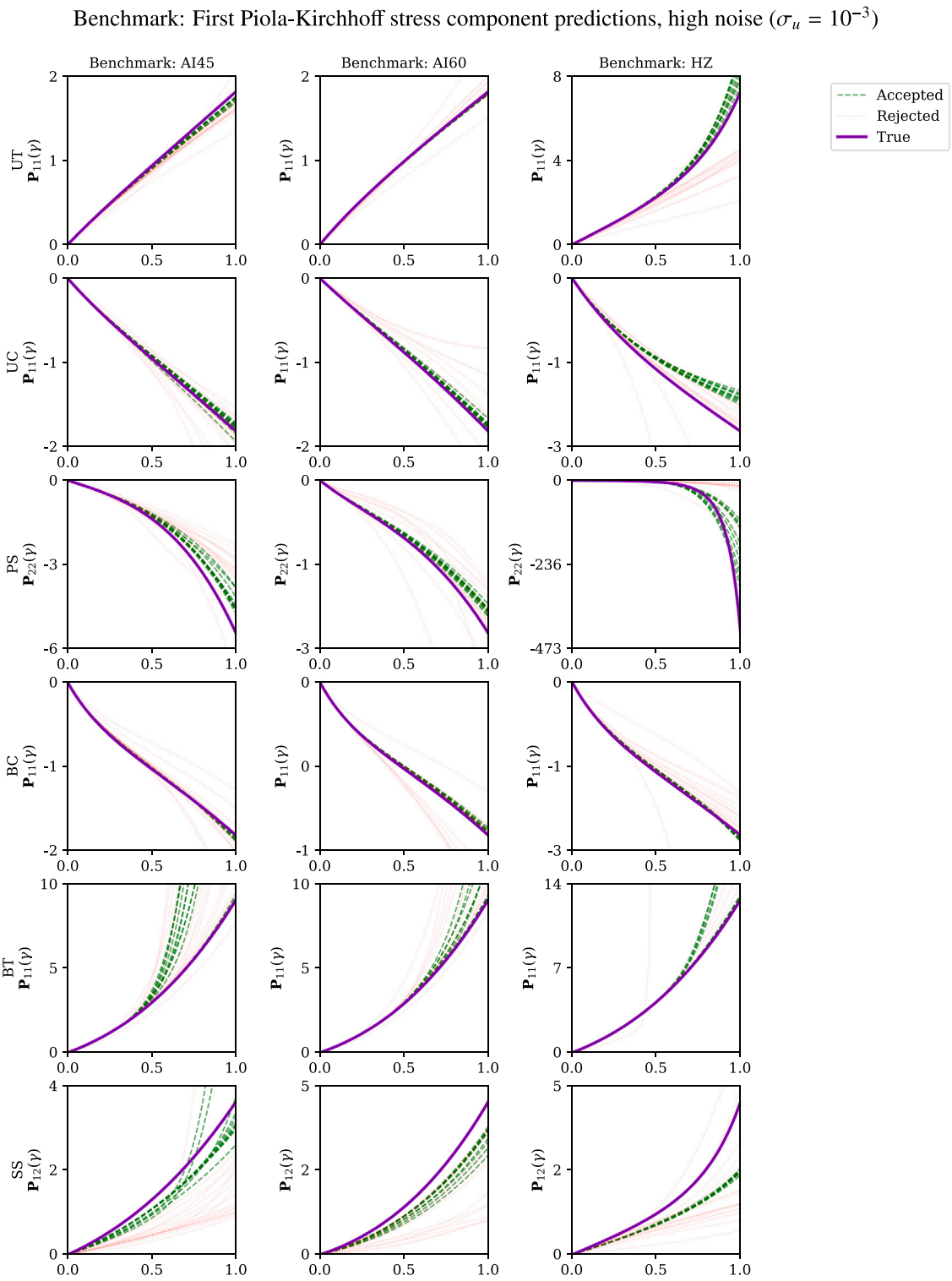

Fig. C.17. First Piola–Kirchhoff stress $\boldsymbol{P}(\boldsymbol{F}(\gamma))$ component prediction along the different deformation paths in (33) using the ICNN-based constitutive models (both accepted and rejected ones) for the case with high noise ($\sigma_u = 10^{-3}$) and benchmarks AI45 (29), AI60 (30), and HZ (31). The constitutive response of the (hidden) true model is also shown for reference.

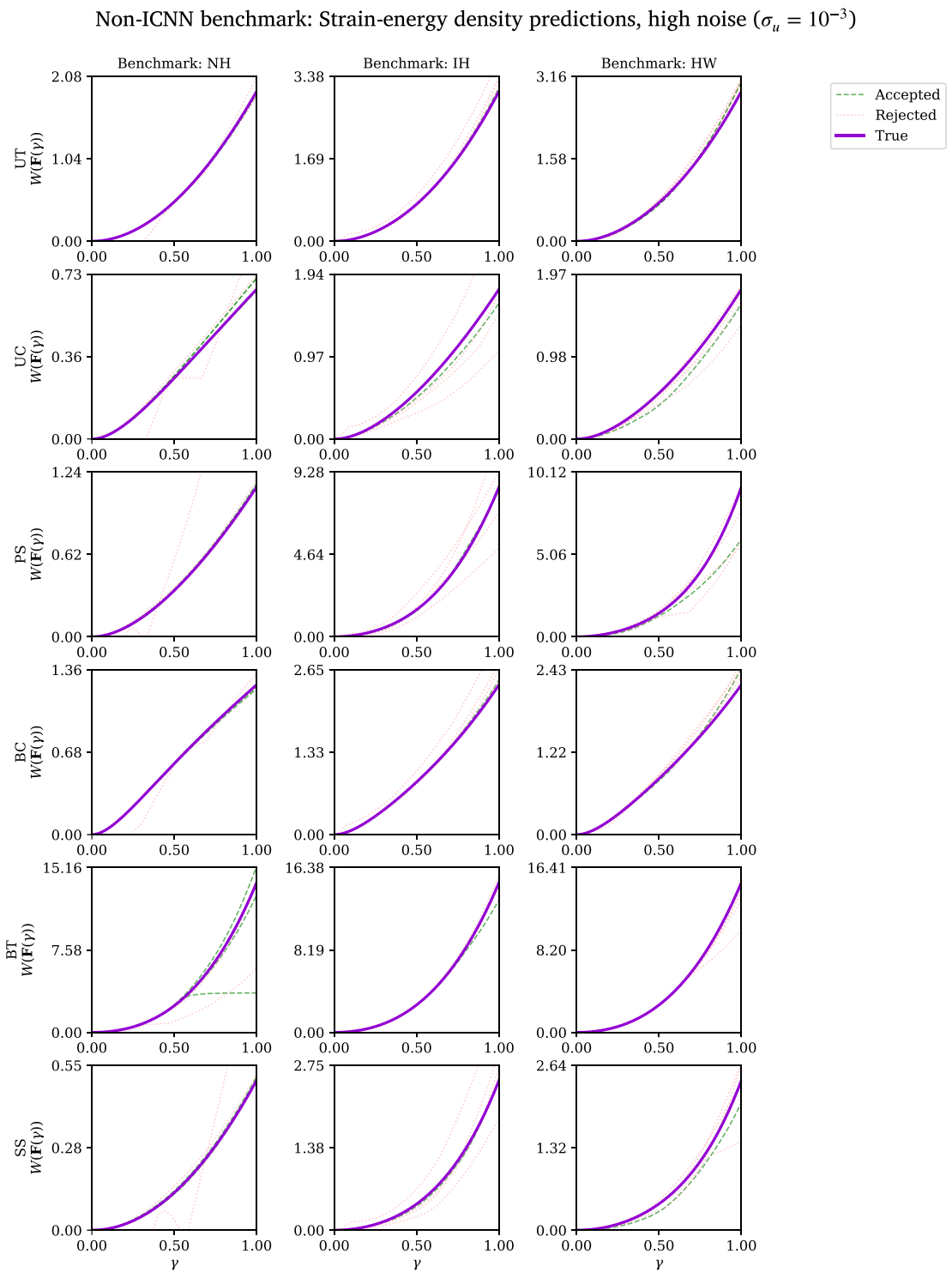

Fig. D.18. Strain energy density $W(\boldsymbol{F}(\gamma))$ prediction along the different deformation paths in (33) using the NN-based (non-convex and non-smooth) constitutive models (both accepted and rejected ones) for the case with high noise ($\sigma_u = 10^{-3}$) and benchmarks NH (23), IH (24), and HW (25). The constitutive response of the (hidden) true model is also shown for reference.

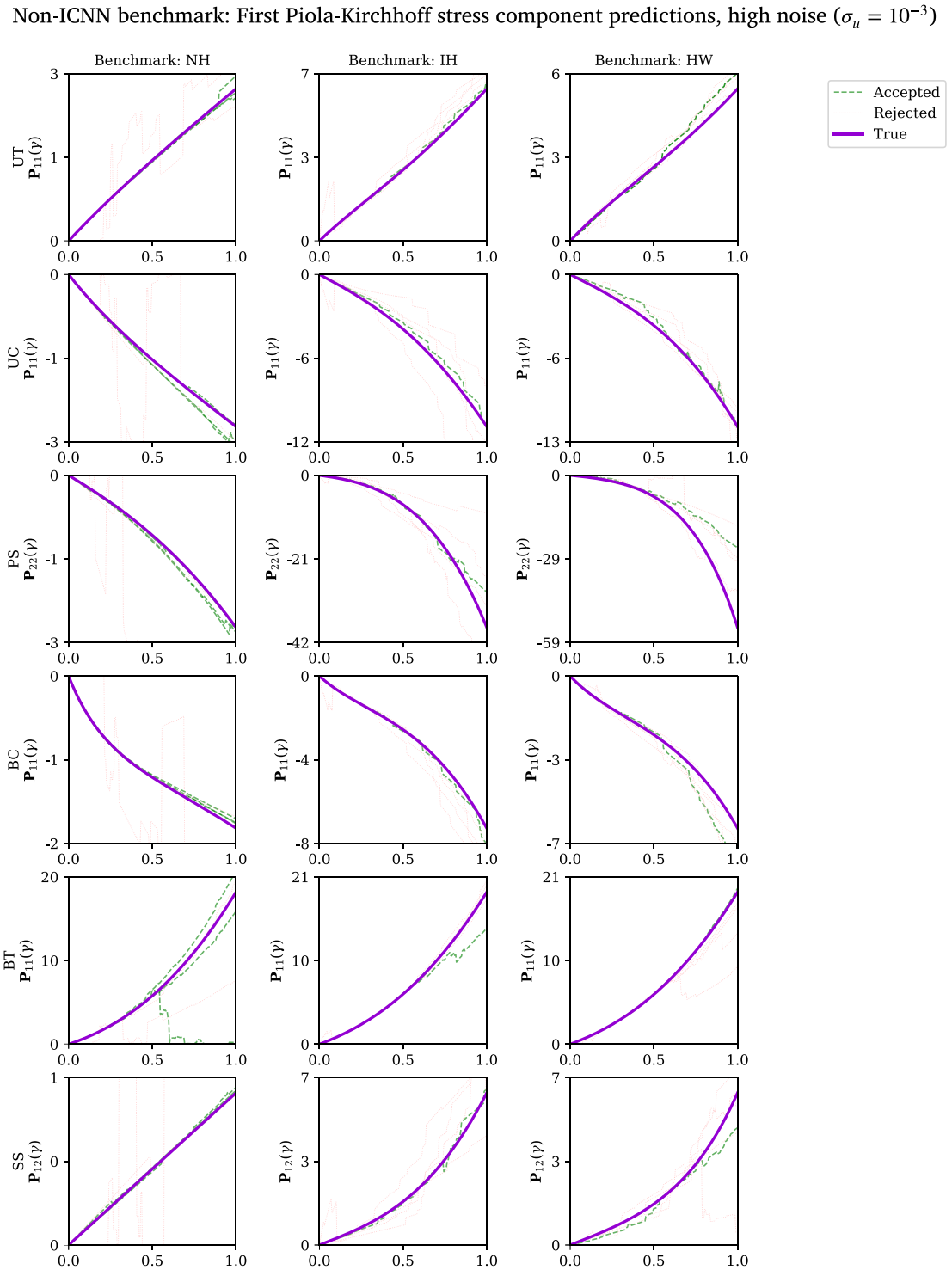

Fig. D.19. First Piola–Kirchhoff stress $\boldsymbol{P}(\boldsymbol{F}(\gamma))$ component prediction along the different deformation paths in (33) using the NN-based (non-convex and non-smooth) constitutive models (both accepted and rejected ones) for the case with high noise ($\sigma_u = 10^{-3}$) and benchmarks NH (23), IH (24), and HW (25). The constitutive response of the (hidden) true model is also shown for reference.

A.2. ICNN

The ICNN architecture consists of three hidden layers ($N = 4$) with 64 neurons each ($d_1 = d_2 = d_3 = 64$). We use the Adam optimizer (Kingma and Ba, 2014) with automatic differentiation-based backpropagation to train the ICNN parameters $\mathcal{Q}$ and $\mathcal{A}$. The

Algorithm 1 Unsupervised training of the ICNN-based constitutive models

Input: Point-wise displacement data $\mathcal{U} = \{\boldsymbol{u}^{a,t} \in \mathbb{R}^2 : a = 1, \dots, n_n; t = 1, \dots, n_t\}$
Input: Global reaction forces $\{R^{\beta,t} : \beta = 1, \dots, n_\beta; t = 1, \dots, n_t\}$
Randomly initialize ICNN parameters: $\mathcal{Q}, \mathcal{A}$.
Initialize learning rate scheduler
Initialize Adam optimizer with parameters $\mathcal{Q}, \mathcal{A}$ and learning rate scheduler
for $e = 1, \dots, n_e$ **do** ▷ Training epochs
  Loss: $\ell \leftarrow 0$ ▷ Initialize loss for current epoch
  for $t = 1, \dots, n_t$ **do** ▷ Iterate over snapshots
    for each element in mesh **do**
      $W^0 \leftarrow W^{\mathrm{NN}}_{\mathcal{Q},\mathcal{A}}|_{\boldsymbol{F}=\boldsymbol{I}}$ ▷ see (9)
      $\boldsymbol{H} \leftarrow -\left.\frac{\partial W^{\mathrm{NN}}_{\mathcal{Q},\mathcal{A}}}{\partial \boldsymbol{F}}\right|_{\boldsymbol{F}=\boldsymbol{I}}$ ▷ see (11)
      $W \leftarrow W^{\mathrm{NN}}_{\mathcal{Q},\mathcal{A}} + W^0 + \boldsymbol{H} : \boldsymbol{E}$ ▷ see (8)
      $\boldsymbol{P} \leftarrow \frac{\partial W}{\partial \boldsymbol{F}}$ ▷ see (10)
    end for
    for $a = 1, \dots, n_a$ **do**
      for $i = 1, 2$ **do**
        Compute force $f_i^{a,t}$ using (19)
      end for
    end for
    for $(a, i) \in \mathcal{D}^{\mathrm{free}}$ **do**
      $\ell \leftarrow \ell + \left(f_i^{a,t}\right)^2$ ▷ force balance at free degrees of freedom; see (22)
    end for
    for $\beta = 1, \dots, n_\beta$ **do**
      $r^{\beta,t} \leftarrow 0$
      for $(a, i) \in \mathcal{D}^{\mathrm{fix}}_\beta$ **do**
        $r^{\beta,t} \leftarrow r^{\beta,t} + f_i^{a,t}$ ▷ see (22)
      end for
      $\ell \leftarrow \ell + \left(R^{\beta,t} - r^{\beta,t}\right)^2$ ▷ force balance at fixed degrees of freedom; see (22)
    end for
  end for
  Compute gradients $\partial \ell / \partial \mathcal{Q}$ and $\partial \ell / \partial \mathcal{A}$ using automatic differentiation
  Update $\mathcal{Q}$ and $\mathcal{A}$ with Adam optimizer using gradients $\partial \ell / \partial \mathcal{Q}$ and $\partial \ell / \partial \mathcal{A}$
  Update learning rate with learning rate scheduler based on epoch number e
end for
Output: Trained ICNN model $W^{\mathrm{NN}}_{\mathcal{Q},\mathcal{A}}$

training is carried out for 500 epochs, which was observed to be sufficient for the loss (22) to converge an acceptable value. To mitigate the network parameters getting stuck in bad local minima, the learning rate of the optimizer is linearly cycled from 0.001 to 0.1 and back every 100 epochs. Dropout rate of 20% is used as regularization to prevent overfitting during training (Srivastava et al., 2014). Note that dropout is turned off for evaluation during validation/testing. The scaling coefficients in (16) are set to $c_{\mathcal{Q}} = 1.0$ and $c_{\mathcal{F}} = 1/12$ to avoid vanishing and/or exploding gradients.

A.3. Anisotropy

For anisotropic hyperelasticity, the NN-based constitutive model requires additional angle parameters $\mathcal{A} = \{\alpha_1, \alpha_2, \dots\}$ to compute the anisotropic invariants $\{\tilde{I}_1^a, \tilde{I}_2^a, \dots\}$. Since α_i is constrained to $[0, \pi)$, this calls for constrained optimization, which is challenging, particularly with NNs. To avoid this, we first initialize a trainable parameter $\zeta_i \in \mathbb{R}$. The fiber orientation is then given by

$$\alpha_i = \frac{\pi}{1 + e^{-\zeta_i}}. \tag{A.1}$$

This change of variables from α_i to ζ_i allows unconstrained optimization, which is easier than constrained optimization.

Appendix B. Pseudocode for unsupervised ICNN training

Algorithm 1 summarizes the unsupervised training of the ICNN-based constitutive models.

Appendix C. Model accuracy in stress predictions

Figs. C.12–C.17 show the comparisons of the first Piola–Kirchhoff stress predictions from the ICNN models vs. the ground-truth along the six deformation paths (33) for both noise levels.

Appendix D. Constitutive models trained without input-convex architecture and smoothness

Figs. D.18 and D.19 show the constitutive response for three representative benchmarks when the ICNN in (8) is replaced with a simple feed-forward neural network (keeping the same architecture) with no constraints on the weights (i.e., no convexity constraints) and ReLU activation functions (i.e., not smooth). Although the strain energy responses are somewhat acceptable, the stress responses exhibit spurious, oscillatory, and non-smooth artifacts. Hence, a simple feed-forward NN (without enforcing smoothness and input-convexity) cannot be deployed in an FEM framework. See As'ad et al. (2022) for more in-depth analysis of the ICNN architecture from a constitutive modeling perspective.

References

Amos, B., Xu, L., Kolter, J.Z., 2017. Input convex neural networks. In: Precup, D., Teh, Y.W. (Eds.), Proceedings of the 34th International Conference on Machine Learning. In: Proceedings of Machine Learning Research, vol. 70, PMLR, pp. 146–155, URL: https://proceedings.mlr.press/v70/amos17b.html.

Arruda, E.M., Boyce, M.C., 1993. A three-dimensional constitutive model for the large stretch behavior of rubber elastic materials. J. Mech. Phys. Solids 41 (2), 389–412. http://dx.doi.org/10.1016/0022-5096(93)90013-6, URL: https://www.sciencedirect.com/science/article/pii/0022509693900136.

As'ad, F., Avery, P., Farhat, C., 2022. A mechanics-informed artificial neural network approach in data-driven constitutive modeling. Internat. J. Numer. Methods Engrg. 123 (12), 2738–2759. http://dx.doi.org/10.1002/nme.6957, URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.6957. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.6957.

Ball, J.M., 1976. Convexity conditions and existence theorems in nonlinear elasticity. Arch. Ration. Mech. Anal. 63 (4), 337–403. http://dx.doi.org/10.1007/BF00279992, URL: http://link.springer.com/10.1007/BF00279992.

Bastek, J.-H., Kumar, S., Telgen, B., Glaesener, R.N., Kochmann, D.M., 2022. Inverting the structure–property map of truss metamaterials by deep learning. Proc. Natl. Acad. Sci. 119 (1), e2111505119. http://dx.doi.org/10.1073/pnas.2111505119.

Baydin, A.G., Pearlmutter, B.A., Radul, A.A., Siskind, J.M., 2017. Automatic differentiation in machine learning: A survey. J. Mach. Learn. Res. 18 (1), 5595–5637.

Bonatti, C., Mohr, D., 2021. One for all: Universal material model based on minimal state-space neural networks. Sci. Adv. 7 (26), http://dx.doi.org/10.1126/sciadv.abf3658.

Boyd, S.P., Vandenberghe, L., 2004. Convex Optimization. Cambridge University Press, Cambridge, UK ; New York.

Cameron, B.C., Tasan, C., 2021. Full-field stress computation from measured deformation fields: A hyperbolic formulation. J. Mech. Phys. Solids 147, 104186. http://dx.doi.org/10.1016/j.jmps.2020.104186.

Carrara, P., De Lorenzis, L., Stainier, L., Ortiz, M., 2020. Data-driven fracture mechanics. Comput. Methods Appl. Mech. Engrg. 372, 113390. http://dx.doi.org/10.1016/j.cma.2020.113390, URL: https://linkinghub.elsevier.com/retrieve/pii/S0045782520305752.

Chen, C.-T., Gu, G.X., 2021. Learning hidden elasticity with deep neural networks. Proc. Natl. Acad. Sci. 118 (31), e2102721118. http://dx.doi.org/10.1073/pnas.2102721118.

Conti, S., Müller, S., Ortiz, M., 2018. Data-driven problems in elasticity. Arch. Ration. Mech. Anal. 229 (1), 79–123. http://dx.doi.org/10.1007/s00205-017-1214-0, URL: http://link.springer.com/10.1007/s00205-017-1214-0.

Dalémat, M., Coret, M., Leygue, A., Verron, E., 2019. Measuring stress field without constitutive equation. Mech. Mater. 136, 103087. http://dx.doi.org/10.1016/j.mechmat.2019.103087, URL: https://linkinghub.elsevier.com/retrieve/pii/S0167663619302376.

Eggersmann, R., Kirchdoerfer, T., Reese, S., Stainier, L., Ortiz, M., 2019. Model-free data-driven inelasticity. Comput. Methods Appl. Mech. Engrg. 350, 81–99. http://dx.doi.org/10.1016/j.cma.2019.02.016, URL: https://linkinghub.elsevier.com/retrieve/pii/S0045782519300878.

Fernández, M., Jamshidian, M., Böhlke, T., Kersting, K., Weeger, O., 2021. Anisotropic hyperelastic constitutive models for finite deformations combining material theory and data-driven approaches with application to cubic lattice metamaterials. Comput. Mech. 67 (2), 653–677. http://dx.doi.org/10.1007/s00466-020-01954-7, URL: http://link.springer.com/10.1007/s00466-020-01954-7.

Flaschel, M., Kumar, S., De Lorenzis, L., 2021. Unsupervised discovery of interpretable hyperelastic constitutive laws. Comput. Methods Appl. Mech. Engrg. 381, 113852. http://dx.doi.org/10.1016/j.cma.2021.113852, URL: https://linkinghub.elsevier.com/retrieve/pii/S0045782521001894.

Flaschel, M., Kumar, S., De Lorenzis, L., 2022. Discovering plasticity models without stress data. arXiv:2202.04916. URL: https://arxiv.org/abs/2202.04916.

Fuhg, J.N., Marino, M., Bouklas, N., 2022a. Local approximate Gaussian process regression for data-driven constitutive models: development and comparison with neural networks. Comput. Methods Appl. Mech. Engrg. 388, 114217. http://dx.doi.org/10.1016/j.cma.2021.114217, URL: https://www.sciencedirect.com/science/article/pii/S004578252100548X.

Fuhg, J.N., van Wees, L., Obstalecki, M., Shade, P., Bouklas, N., Kasemer, M., 2022b. Machine-learning convex and texture-dependent macroscopic yield from crystal plasticity simulations. Materialia 23, 101446. http://dx.doi.org/10.1016/j.mtla.2022.101446, URL: https://www.sciencedirect.com/science/article/pii/S2589152922001296.

Gent, A.N., Thomas, A.G., 1958. Forms for the stored (strain) energy function for vulcanized rubber. J. Polym. Sci. 28 (118), 625–628. http://dx.doi.org/10.1002/pol.1958.1202811814, URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/pol.1958.1202811814. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/pol.1958.1202811814.

Ghaboussi, J., Garrett, J.H., Wu, X., 1991. Knowledge-based modeling of material behavior with neural networks. J. Eng. Mech. 117 (1), 132–153. http://dx.doi.org/10.1061/(ASCE)0733-9399(1991)117:1(132), URL: http://ascelibrary.org/doi/10.1061/%28ASCE%290733-9399%281991%29117%3A1%28132%29.

González, D., Chinesta, F., Cueto, E., 2019a. Learning corrections for hyperelastic models from data. Front. Mater. 6, 14. http://dx.doi.org/10.3389/fmats.2019.00014, URL: https://www.frontiersin.org/article/10.3389/fmats.2019.00014.

González, D., Chinesta, F., Cueto, E., 2019b. Thermodynamically consistent data-driven computational mechanics. Contin. Mech. Thermodyn. 31 (1), 239–253. http://dx.doi.org/10.1007/s00161-018-0677-z, URL: http://link.springer.com/10.1007/s00161-018-0677-z.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press, http://www.deeplearningbook.org.

Haghighat, E., Raissi, M., Moure, A., Gomez, H., Juanes, R., 2020. A deep learning framework for solution and discovery in solid mechanics. arXiv:2003.02751 [cs, stat]. URL: http://arxiv.org/abs/2003.02751. arXiv:2003.02751.

Haines, D.W., Wilson, W.D., 1979. Strain-energy density function for rubberlike materials. J. Mech. Phys. Solids 27 (4), 345–360. http://dx.doi.org/10.1016/0022-5096(79)90034-6, URL: https://www.sciencedirect.com/science/article/pii/0022509679900346.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, http://dx.doi.org/10.1109/cvpr.2016.90.

Holzapfel, G.A., Gasser, T.C., Ogden, R.W., 2000. A new constitutive framework for arterial wall mechanics and a comparative study of material models. J. Elast. Phys. Sci. Solids 61 (1), 1–48. http://dx.doi.org/10.1023/A:1010835316564.

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. Neural Netw. 2 (5), 359–366. http://dx.doi.org/10.1016/0893-6080(89)90020-8.

Huang, S., He, Z., Chem, B., Reina, C., 2022. Variational Onsager Neural Networks (VONNs): A thermodynamics-based variational learning strategy for non-equilibrium PDEs. J. Mech. Phys. Solids 163, 104856. http://dx.doi.org/10.1016/j.jmps.2022.104856, URL: https://www.sciencedirect.com/science/article/pii/S0022509622000692.

Huang, D.Z., Xu, K., Farhat, C., Darve, E., 2020. Learning constitutive relations from indirect observations using deep neural networks. J. Comput. Phys. 416, 109491. http://dx.doi.org/10.1016/j.jcp.2020.109491, URL: https://www.sciencedirect.com/science/article/pii/S0021999120302655.

Ibáñez, R., Abisset-Chavanne, E., González, D., Duval, J.-L., Cueto, E., Chinesta, F., 2019. Hybrid constitutive modeling: data-driven learning of corrections to plasticity models. Int. J. Mater. Form. 12 (4), 717–725. http://dx.doi.org/10.1007/s12289-018-1448-x.

Ibañez, R., Abisset-Chavanne, E., Aguado, J.V., Gonzalez, D., Cueto, E., Chinesta, F., 2018. A manifold learning approach to data-driven computational elasticity and inelasticity. Arch. Comput. Methods Eng. 25 (1), 47–57. http://dx.doi.org/10.1007/s11831-016-9197-9.

Ibañez, R., Borzacchiello, D., Aguado, J.V., Abisset-Chavanne, E., Cueto, E., Ladeveze, P., Chinesta, F., 2017. Data-driven non-linear elasticity: constitutive manifold construction and problem discretization. Comput. Mech. 60 (5), 813–826. http://dx.doi.org/10.1007/s00466-017-1440-1, URL: http://link.springer.com/10.1007/s00466-017-1440-1.

Isihara, A., Hashitsume, N., Tatibana, M., 1951. Statistical theory of rubber-like elasticity. IV. (Two-dimensional stretching). J. Chem. Phys. 19 (12), 1508–1512. http://dx.doi.org/10.1063/1.1748111, URL: https://aip.scitation.org/doi/10.1063/1.1748111. Publisher: American Institute of Physics.

Joshi, A., Thakolkaran, P., Zheng, Y., Escande, M., Flaschel, M., De Lorenzis, L., Kumar, S., 2022. Bayesian-EUCLID: discovering hyperelastic material laws with uncertainties. http://dx.doi.org/10.48550/ARXIV.2203.07422, URL: https://arxiv.org/abs/2203.07422.

Karapiperis, K., Ortiz, M., Andrade, J., 2021. Data-Driven nonlocal mechanics: Discovering the internal length scales of materials. Comput. Methods Appl. Mech. Engrg. 386, 114039. http://dx.doi.org/10.1016/j.cma.2021.114039, URL: https://linkinghub.elsevier.com/retrieve/pii/S0045782521003704.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. http://dx.doi.org/10.48550/ARXIV.1412.6980, URL: https://arxiv.org/abs/1412.6980.

Kirchdoerfer, T., Ortiz, M., 2016. Data-driven computational mechanics. Comput. Methods Appl. Mech. Engrg. 304, 81–101. http://dx.doi.org/10.1016/j.cma.2016.02.001, URL: https://www.sciencedirect.com/science/article/pii/S0045782516300238.

Kirchdoerfer, T., Ortiz, M., 2017. Data-driven computing in dynamics. Internat. J. Numer. Methods Engrg. 113 (11), 1697–1710. http://dx.doi.org/10.1002/nme.5716.

Klein, D.K., Fernández, M., Martin, R.J., Neff, P., Weeger, O., 2022. Polyconvex anisotropic hyperelasticity with neural networks. J. Mech. Phys. Solids 159, 104703. http://dx.doi.org/10.1016/j.jmps.2021.104703, URL: http://arxiv.org/abs/2106.14623. arXiv:2106.14623.

Kumar, S., Kochmann, D.M., 2021. What machine learning can do for computational solid mechanics. arXiv:2109.08419. URL: https://arxiv.org/abs/2109.08419.

Kumar, S., Tan, S., Zheng, L., Kochmann, D.M., 2020. Inverse-designed spinodoid metamaterials. Npj Comput. Mater. 6 (1), 73. http://dx.doi.org/10.1038/s41524-020-0341-6, URL: http://www.nature.com/articles/s41524-020-0341-6.

Kumar, S., Vidyasagar, A., Kochmann, D.M., 2019. An assessment of numerical techniques to find energy-minimizing microstructures associated with nonconvex potentials. Internat. J. Numer. Methods Engrg. 121 (7), 1595–1628. http://dx.doi.org/10.1002/nme.6280.

Leygue, A., Coret, M., Réthoré, J., Stainier, L., Verron, E., 2018. Data-based derivation of material response. Comput. Methods Appl. Mech. Engrg. 331, 184–196. http://dx.doi.org/10.1016/j.cma.2017.11.013, URL: https://linkinghub.elsevier.com/retrieve/pii/S0045782517307156.

Liang, M., Chang, Z., Wan, Z., Gan, Y., Schlangen, E., Šavija, B., 2022. Interpretable ensemble-machine-learning models for predicting creep behavior of concrete. Cem. Concr. Compos. 125, 104295. http://dx.doi.org/10.1016/j.cemconcomp.2021.104295.

Marwala, T., 2010. Finite-Element-Model Updating using Computional Intelligence Techniques. Springer London, http://dx.doi.org/10.1007/978-1-84996-323-7.

Morrey, C.B., 1952. Quasi-convexity and the lower semicontinuity of multiple integrals. Pacific J. Math. 2 (1), 25–53. http://dx.doi.org/10.2140/pjm.1952.2.25.

Mozaffar, M., Bostanabad, R., Chen, W., Ehmann, K., Cao, J., Bessa, M.A., 2019. Deep learning predicts path-dependent plasticity. Proc. Natl. Acad. Sci. 116 (52), 26414–26420. http://dx.doi.org/10.1073/pnas.1911815116.

Nguyen, L.T.K., Keip, M.-A., 2018. A data-driven approach to nonlinear elasticity. Comput. Struct. 194, 97–115. http://dx.doi.org/10.1016/j.compstruc.2017.07.031, URL: https://linkinghub.elsevier.com/retrieve/pii/S0045794917301311.

Ogden, R.W., Hill, R., 1972. Large deformation isotropic elasticity – on the correlation of theory and experiment for incompressible rubberlike solids. Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 326 (1567), 565–584. http://dx.doi.org/10.1098/rspa.1972.0026, URL: https://royalsocietypublishing.org/doi/10.1098/rspa.1972.0026. Publisher: Royal Society.

Pierron, F., Grédiac, M., 2012. The Virtual Fields Method: Extracting Constitutive Mechanical Parameters from Full-Field Deformation Measurements. Springer.

Raissi, M., Perdikaris, P., Karniadakis, G., 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J. Comput. Phys. 378, 686–707. http://dx.doi.org/10.1016/j.jcp.2018.10.045.

Rocha, I., Kerfriden, P., van der Meer, F., 2021. On-the-fly construction of surrogate constitutive models for concurrent multiscale mechanical analysis through probabilistic machine learning. J. Comput. Phys.: X 9, 100083. http://dx.doi.org/10.1016/j.jcpx.2020.100083, URL: https://www.sciencedirect.com/science/article/pii/S2590055220300354.

Schröder, J., 2010. Anisotropie polyconvex energies. In: CISM International Centre for Mechanical Sciences. Springer Vienna, pp. 53–105. http://dx.doi.org/10.1007/978-3-7091-0174-2_3.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (56), 1929–1958, URL: http://jmlr.org/papers/v15/srivastava14a.html.

Tartakovsky, A.M., Marrero, C.O., Perdikaris, P., Tartakovsky, G.D., Barajas-Solano, D., 2018. Learning parameters and constitutive relationships with physics informed deep neural networks. arXiv:1808.03398 [physics]. URL: http://arxiv.org/abs/1808.03398. arXiv:1808.03398.

Vlassis, N.N., Ma, R., Sun, W., 2020. Geometric deep learning for computational mechanics Part I: anisotropic hyperelasticity. Comput. Methods Appl. Mech. Engrg. 371, 113299. http://dx.doi.org/10.1016/j.cma.2020.113299.

Vlassis, N.N., Sun, W., 2021. Sobolev training of thermodynamic-informed neural networks for interpretable elasto-plasticity models with level set hardening. Comput. Methods Appl. Mech. Engrg. 377, 113695. http://dx.doi.org/10.1016/j.cma.2021.113695.

Wang, Z., Estrada, J., Arruda, E., Garikipati, K., 2021. Inference of deformation mechanisms and constitutive response of soft material surrogates of biological tissue by full-field characterization and data-driven variational system identification. J. Mech. Phys. Solids 153, 104474. http://dx.doi.org/10.1016/j.jmps.2021.104474.

Yang Gao, D., Neff, P., Roventa, I., Thiel, C., 2017. On the convexity of nonlinear elastic energies in the right Cauchy-Green tensor. J. Elasticity 127 (2), 303–308.

Zheng, L., Kumar, S., Kochmann, D.M., 2021. Data-driven topology optimization of spinodoid metamaterials with seamlessly tunable anisotropy. Comput. Methods Appl. Mech. Engrg. 383, 113894. http://dx.doi.org/10.1016/j.cma.2021.113894.