



MOX-Report No. 30/2026

The internal law of a material can be discovered from its boundary

Regazzoni, F.

MOX, Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox-dmat@polimi.it

<https://mox.polimi.it>

The internal law of a material can be discovered from its boundary

Francesco Regazzoni¹

¹ MOX, Department of Mathematics, Politecnico di Milano, Milan, Italy

Abstract

Since the earliest stages of human civilization, advances in technology have been tightly linked to our ability to understand and predict the mechanical behavior of materials. In recent years, this challenge has increasingly been framed within the broader paradigm of data-driven scientific discovery, where governing laws are inferred directly from observations. However, existing methods require either stress-strain pairs or full-field displacement measurements, which are often inaccessible in practice. We introduce Neural-DFEM, a method that enables unsupervised discovery of hyperelastic material laws even from partial observations, such as boundary-only measurements. The method embeds a differentiable finite element solver within the learning loop, directly linking candidate energy functionals to available measurements. To guarantee thermodynamic consistency and mathematical well-posedness throughout training, the method employs Hyperelastic Neural Networks, a novel structure-preserving neural architecture that enforces frame indifference, material symmetry, polyconvexity, and coercivity by design. The resulting framework enables robust material model discovery in both two- and three-dimensional settings, including scenarios with boundary-only measurements. Neural-DFEM allows for generalization across geometries and loading conditions, and exhibits unprecedented accuracy and strong resilience to measurement noise. Our results demonstrate that reliable identification of material laws is achievable even under partial observability when strong physical inductive biases are embedded in the learning architecture.

1 Introduction

Materials have shaped technological development throughout human history, from the Stone and Iron Ages to the modern era of silicon and engineered matter. As a matter of fact, the ability to understand and predict the mechanical response of deformable bodies underpins progress in fields spanning structural engineering, soft matter physics, biomechanics, and geophysics [44, 57]. Within modern sciences, this challenge has traditionally been addressed through mathematical models, known as constitutive laws [2, 21]. The parameters of these models are typically calibrated using experimental measurements, but their functional forms are typically human-defined and only partially informed by experimental evidence. More recently, a new data-driven paradigm has emerged, which bypasses predefined functional forms and instead learns the material response directly from data [32, 40, 25]. This shift aligns with a broader movement toward the automated discovery of governing physical laws from data [29, 5, 59, 8, 28], spanning approaches such as sparse identification of nonlinear dynamics [4, 5], symbolic regression [55], and neural networks [46, 7].

Constitutive law identification can be naturally cast as a constrained inverse problem, where the unknown is the governing material law, which must be inferred from observations of the system response. Within this paradigm, a distinction is commonly made between *model-free* and *model discovery* approaches. Model-free methods [32, 6, 33, 49] do not yield an explicit constitutive relationship; instead, a collection of recorded strain–stress pairs is used to infer the material response online, by locally approximating the behavior based on data points closest to the current state of each material point. While effective in certain settings, these approaches cannot readily leverage *a priori* physical knowledge and are typically both data-hungry and memory-intensive, as the full dataset must be queried during the simulation.

In contrast, model discovery methods aim to identify an explicit constitutive model in the classical sense, yet without prescribing its functional form *a priori*. Such models may be based on symbolic regression techniques [12, 38, 13, 14, 15], where the constitutive law is discovered by combining elements from a library

of simple mathematical expressions, or on machine learning models such as neural networks [48, 37, 26, 52], Gaussian processes [16], or neural ordinary differential equations [50]. Symbolic approaches offer enhanced interpretability, whereas machine learning models provide superior expressive power. Moreover, several architectures have been proposed to automatically embed physical and mathematical constraints into the learned model, thereby restricting the hypothesis space *a priori* [38, 50].

Regardless of whether the model is symbolic or black-box, existing methods also differ in the type of data they require. The majority of approaches are supervised in nature, as they rely on paired strain–stress data [39, 39]. However, such data are often unrealistic to obtain in practice: stress measurements are typically available only in simple mechanical tests, such as uniaxial tension, which fail to capture the full complexity of constitutive behavior. As a matter of fact, the simultaneous measurement of all tensorial components of strain and stress is practically infeasible.

A more realistic experimental setting involves direct measurements of the displacement field of a specimen, for instance obtained via digital image correlation (DIC) techniques [18], together with global force measurements recorded during the deformation process. The Virtual Field Method (VFM) [20] has been leveraged for addressing material discovery problems in this setting [26, 12, 13]. The VFM identifies the constitutive model that minimizes the residual of the force balance, given the measured displacement field. Despite the significant progress enabled by this approach, important limitations remain. Since the displacement field is directly used in the fitting procedure, measurement noise can severely affect the results, making denoising techniques necessary. More critically, the method requires access to the complete displacement field, which prevents its application to scenarios involving partial measurements, such as surface-only observations in three-dimensional specimens, where internal displacement measurements are challenging. As a consequence, this class of methods has so far been restricted to two-dimensional benchmark problems under plane strain or plane stress assumptions, and realistic three-dimensional applications capturing complex material behavior remain largely out of reach.

This work fills this gap, by introducing the first material model discovery method capable of operating in a fully unsupervised setting while explicitly handling partial or defective data, such as boundary-only observations. The proposed approach, termed *Neural Differentiable Finite Element Method* (Neural-DFEM), embeds a differentiable finite element solver for the equilibrium equations directly into the training process, thereby establishing a direct link between candidate constitutive laws and the resulting deformation fields. By formulating the loss function in terms of the discrepancy between available measurements and model predictions on the training specimens, Neural-DFEM enables constitutive model discovery from limited data and exhibits exceptional robustness to measurement noise.

Embedding a differentiable solver within the training loop introduces two main challenges. The first challenge arises from the requirement that the elastostatic equilibrium problem associated with each intermediate candidate model remain solvable throughout the optimization process; failure of the finite element solver at any stage would lead in fact to training breakdown. To this end, we here introduce a novel neural-network-based constitutive model ansatz, termed *Hyperelastic Neural Network* (HNN) which, for the first time in the literature, provides mathematically provable guarantees of physical and thermodynamic consistency, as well as mathematical well-posedness of the associated equilibrium problem.

The second challenge concerns computational cost. This issue is addressed through the use of a quasi-Newton optimization strategy, which significantly reduces the number of training epochs required for convergence, combined with a continuation strategy during the initial stages of training. In addition, to mitigate the local convergence properties of the Newton solver employed for equilibrium, a backtracking strategy is incorporated into the line search of the quasi-Newton training algorithm, automatically rejecting parameter updates that would move the system outside the Newton basin of attraction. These numerical strategies together ensure robustness and computational efficiency of Neural-DFEM.

The contributions of this work are threefold:

- A computational framework for unsupervised constitutive discovery under partial observability, enabling material identification from boundary-only measurements.
- A structure-preserving neural constitutive class (HNN) with provable physical consistency and well-posedness guarantees.

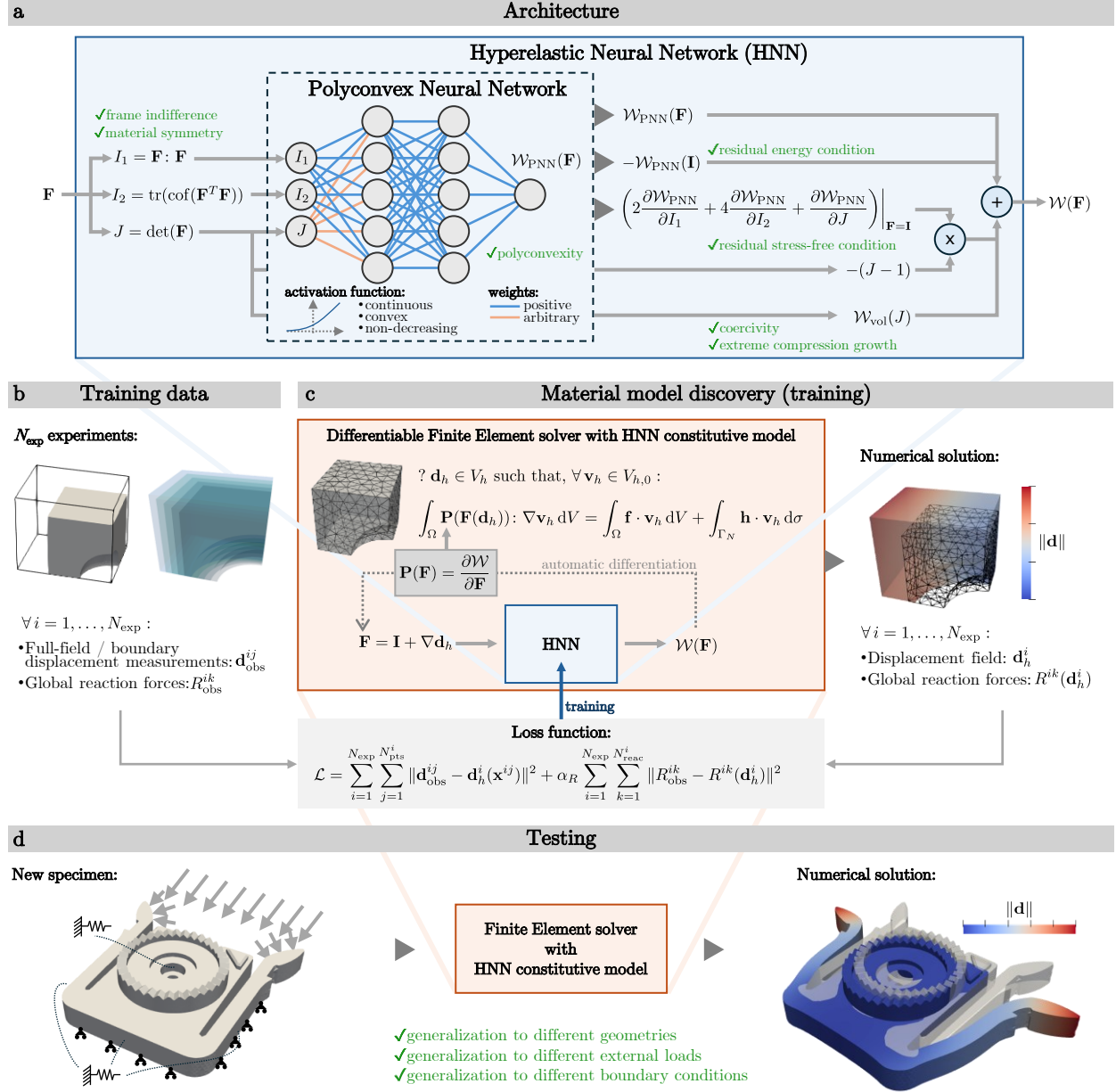


Figure 1: **Overview of the Neural-DFEM framework.** (a) HNNs constitute a novel family of neural networks designed to approximate the strain energy density functional of a hyperelastic isotropic material. By construction, an HNN satisfies the seven physical and mathematical requirements necessary for well-posedness and physical consistency, as indicated in the figure alongside the architectural components that enforce them. The figure highlights the main design traits of HNNs, including weights with controlled sign, and a specific choice of the activation function. Optionally, HNNs can include skip connections, although they are not represented in the figure. (b) The training data consist of (possibly partial) displacement measurements and global reaction forces, coming from N_{exp} experiments. (c) Training is performed by embedding a differentiable finite element solver within the loss evaluation. The loss measures the discrepancy between predicted and observed data, while the Piola-Kirchhoff stress tensor is obtained by differentiating the HNN output with respect to the deformation gradient. (d) Once trained, the HNN can be integrated into standard finite element solvers to predict the deformation of new specimens, enabling generalization across geometries, loading conditions, and boundary conditions.

- An efficient solver-in-the-loop training strategy that efficient resolution of the inverse problem.

The resulting framework is validated across multiple two- and three-dimensional test cases, demonstrating robustness to substantial measurement noise and generalization across geometries and loading conditions. An overview of the Neural-DFEM framework is shown in Fig. 1.

2 Methods

2.1 Notation

We denote with capital bold letters matrices, e.g., $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$, and by lowercase bold letters vectors, e.g., $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$. The Frobenius norm is denoted by $|\cdot|$, that is $|\mathbf{A}| = \text{tr}(\mathbf{A}^T \mathbf{A})^{1/2} = (\sum_{i,j} A_{ij}^2)^{1/2}$. With $\mathbf{a} \succeq \mathbf{b}$ we mean that each component of \mathbf{a} is greater or equal to the corresponding component of \mathbf{b} . We denote by $\mathbf{1}_n \in \mathbb{R}^n$ the n -dimensional vector with all entries equal to 1, and by $\mathbf{1}_{n \times m} \in \mathbb{R}^{n \times m}$ the $n \times m$ matrix with all entries equal to 1.

We denote by $d = 3$ the space dimension. $\mathbf{I} \in \mathbb{R}^{d \times d}$ is the identity tensor, $\mathbf{0} \in \mathbb{R}^{d \times d}$ is the zero tensor. The set of second order tensors with positive determinant is denoted as $\text{GL}^+(d) = \{\mathbf{A} \in \mathbb{R}^{d \times d} \mid \det \mathbf{A} > 0\}$. The special orthogonal group is denoted as $\text{SO}(d) = \{\mathbf{A} \in \mathbb{R}^{d \times d} \mid \mathbf{A}^T \mathbf{A} = \mathbf{I}, \det \mathbf{A} = 1\}$.

2.2 Material model discovery: problem setting

Let us consider a bounded domain $\Omega \subset \mathbb{R}^3$, representing the stress-free reference configuration of an elastic body, and $\chi: \Omega \rightarrow \mathbb{R}^3$ the deformation map such that $\chi(\Omega)$ is the deformed configuration. We denote by $\mathbf{d}(\mathbf{x}) = \chi(\mathbf{x}) - \mathbf{x}$ the displacement field, by $\mathbf{F} = \nabla \chi = \mathbf{I} + \nabla \mathbf{d}$ the deformation gradient tensor, and by $J = \det \mathbf{F}$ its determinant. We denote by $\mathbf{C} = \mathbf{F}^T \mathbf{F}$ the right Cauchy-Green tensor, and its invariants by $I_1 = \text{tr} \mathbf{C}$ and $I_2 = \frac{1}{2}(\text{tr}(\mathbf{C})^2 - \text{tr}(\mathbf{C}^2))$. Moreover, we denote the isochoric invariants as $\bar{I}_1 = J^{-2/3} I_1$ and $\bar{I}_2 = J^{-4/3} I_2$.

We consider a hyperelastic material, characterized by a strain energy density functional $\mathcal{W}: \text{GL}^+(d) \rightarrow \mathbb{R} \cup \{+\infty\}$, such that the Piola-Kirchhoff stress tensors is given by $\mathbf{P}(\mathbf{F}) = \frac{\partial \mathcal{W}}{\partial \mathbf{F}}$. In such a way, the strain energy density \mathcal{W} characterizes the material constitutive law. The knowledge of \mathcal{W} allows to predict the material response to deformations. The goal of this work is to reverse this process, that is to discover the function \mathcal{W} by observing the deformation of specimens of the considered materials.

The following differential problem models the configuration assumed by a body when subject to a distributed load \mathbf{f} , to a boundary load \mathbf{h} on $\Gamma_N \subset \partial\Omega$ and assigned displacement \mathbf{g} on $\Gamma_D \subset \partial\Omega$ (where Γ_N and Γ_D form a disjoint partition of $\partial\Omega$):

$$\begin{cases} -\nabla \cdot \mathbf{P}(\mathbf{F}) = \mathbf{f} & \text{in } \Omega, \\ \mathbf{P}(\mathbf{F})\mathbf{n} = \mathbf{h} & \text{on } \Gamma_N, \\ \mathbf{d} = \mathbf{g} & \text{on } \Gamma_D, \end{cases} \quad (1)$$

where \mathbf{n} denotes the outward normal vector to Γ_N . Notice that method proposed in this work extends to different kind of boundary conditions (such as follower loads, spring-like boundary conditions, sliding conditions, symmetric conditions), but, for the sake of brevity, we will carry on the presentation restricting ourselves to the case of (1).

We assume that N_{exp} experiments are conducted, each corresponding to a potentially different specimen geometry Ω^i , boundary conditions type, and loads $\mathbf{f}^i, \mathbf{h}^i, \mathbf{g}^i$. Let $\mathbf{d}^i: \Omega^i \rightarrow \mathbb{R}^3$ denote the displacement field resulting from the i -th experiment, for $i = 1, \dots, N_{\text{exp}}$. For each experiment, we dispose of measurements of the displacement at a collection of N_{pts}^i points. Specifically, we denote by $\mathbf{d}_{\text{obs}}^{ij} \simeq \mathbf{d}^i(\mathbf{x}^{ij})$ a (typically noisy) measurement of the displacement of the i -th experiment at the j -th point (denoted by $\mathbf{x}^{ij} \in \Omega^i$), for $j = 1, \dots, N_{\text{pts}}^i$.

Furthermore, typical experimental setups allow to measure the global reaction forces acting on the boundaries where prescribed displacements are assigned, that is correspondence of Dirichlet boundary conditions. Suppose that the i -th experiment has N_{reac}^i Dirichlet boundary segments, denoted by $\Gamma_D^{ik} \subset \partial\Omega^i$,

for $k = 1, \dots, N_{\text{reac}}^i$. Then, we denote by R^{ik} the function that maps a displacement field into the associated reaction force:

$$R^{ik}(\mathbf{d}) := \int_{\Gamma_D^{ik}} \mathbf{P}(\mathbf{I} + \nabla \mathbf{d}) \mathbf{n} \cdot \mathbf{n} \, d\sigma.$$

Using this notation, we denote by $R_{\text{obs}}^{ik} \simeq R^{ik}(\mathbf{d}^i)$ the (typically noisy) measured reaction force.

To summarize, the full set of training data is given by:

$$\{\mathbf{d}_{\text{obs}}^{ij}\}_{i=1, \dots, N_{\text{exp}}^i}^{j=1, \dots, N_{\text{pts}}^i}, \quad \{R_{\text{obs}}^{ik}\}_{i=1, \dots, N_{\text{exp}}^i}^{k=1, \dots, N_{\text{reac}}^i}.$$

2.2.1 Finite element approximation

Considering problem (1), let us introduce the space of kinematically admissible displacements V , and the space of admissible variations (test function space) V_0 :

$$\begin{aligned} V &:= \{\mathbf{d} \in [H^1(\Omega)]^3 \text{ s.t. } \mathbf{d} = \mathbf{g} \text{ on } \Gamma_D\}, \\ V_0 &:= \{\mathbf{d} \in [H^1(\Omega)]^3 \text{ s.t. } \mathbf{d} = \mathbf{0} \text{ on } \Gamma_D\}. \end{aligned}$$

The equilibrium residual $r: V \times V_0 \rightarrow \mathbb{R}$ is given by:

$$r(\mathbf{d}, \mathbf{v}) = \int_{\Omega} \mathbf{P}(\mathbf{F}) : \nabla \mathbf{v} \, dV - \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, dV - \int_{\Gamma_N} \mathbf{h} \cdot \mathbf{v} \, d\sigma.$$

The weak formulation of problem (1) reads as finding $\mathbf{d} \in V$ such that $r(\mathbf{d}, \mathbf{v}) = 0$ for all $\mathbf{v} \in V_0$.

Let us consider a finite element space $V_h \subset V$, defined on a triangulation of the domain Ω , and let us define $V_{h,0} = V_h \cap V_0$. Let $\{\varphi_n\}_{n=1}^{N_h}$ be a basis for $V_{h,0}$. Then, each function of V_h can be expressed as $\mathbf{d}_h(\mathbf{x}) = \sum_{n=1}^{N_h} d_n \varphi_n(\mathbf{x}) + \mathbf{g}_h$, where $\mathbf{g}_h \in V_h$ is the lifting of the boundary datum \mathbf{g} . By using this notation, the finite element approximation of problem (1) consists of finding $\mathbf{d}_h \in V_h$ such that $r(\mathbf{d}_h, \mathbf{v}_h) = 0$ for all $\mathbf{v}_h \in V_{h,0}$, or, equivalently, such that $r(\mathbf{d}_h, \varphi_n) = 0$ for $n = 1, \dots, N_h$. In what follows, we will employ the superscript i to denote the variables associated with the i -th experiment (e.g., V_h^i , \mathbf{d}_h^i , N_h^i , φ_n^i).

2.2.2 Material law discovery

We introduce an hypothesis space $\mathcal{H} \subset \{\mathcal{W}: \text{GL}^+(d) \rightarrow \mathbb{R} \cup \{+\infty\}\}$, that is a set of candidate material laws. The issue of designing a convenient hypothesis space, sufficiently rich but at the same time numerically tractable, will be addressed below. For the moment, let us assume this space to be given. Unsupervised material discovery consists in finding the material law $\mathcal{W} \in \mathcal{H}$ that is, in a suitable sense, most consistent with the observations $\mathbf{d}_{\text{obs}}^{ij}$ and R_{obs}^{ik} . Such material discovery problem can be formulated in different manners.

A first possibility is to rely on the VFM formulation (such in [12]), which consists in looking for the material law such that the equilibrium residual is minimized:

$$\mathcal{W}^* = \underset{\mathcal{W} \in \mathcal{H}}{\text{argmin}} \left(\sum_{i=1}^{N_{\text{exp}}} \sum_{n=1}^{N_h^i} \|r(\mathbf{d}_{h,\text{obs}}^i, \varphi_n^i)\|^2 + \alpha_R \sum_{i=1}^{N_{\text{exp}}} \sum_{k=1}^{N_{\text{reac}}^i} \|R_{\text{obs}}^{ik} - R^{ik}(\mathbf{d}_{h,\text{obs}}^i)\|^2 \right),$$

where $\mathbf{d}_{h,\text{obs}}^i \in V_h^i$ is a suitably denoised projection of the observed displacement $\{\mathbf{d}_{\text{obs}}^{ij}\}_{j=1, \dots, N_{\text{pts}}^i}$ onto the finite element space V_h^i , and where $\alpha_R > 0$ is a weight constant. Clearly, this approach is valid only when the observed displacement $\{\mathbf{d}_{\text{obs}}^{ij}\}_{j=1, \dots, N_{\text{pts}}^i}$ covers the whole domain Ω^i , otherwise the equilibrium residual cannot be computed.

Instead, we propose to change the role of the equilibrium condition from a penalization term to a constraint, and using the observed displacement as a penalization. Our proposed formulation thus reads:

$$\mathcal{W}^* = \underset{\mathcal{W} \in \mathcal{H}}{\text{argmin}} \left(\sum_{i=1}^{N_{\text{exp}}} \sum_{j=1}^{N_{\text{pts}}^i} \|\mathbf{d}_{\text{obs}}^{ij} - \mathbf{d}_h^i(\mathbf{x}^{ij})\|^2 + \alpha_R \sum_{i=1}^{N_{\text{exp}}} \sum_{k=1}^{N_{\text{reac}}^i} \|R_{\text{obs}}^{ik} - R^{ik}(\mathbf{d}_h^i)\|^2 \right), \quad (2)$$

subject to the constraints $r(\mathbf{d}_h^i, \mathbf{v}_h^i) = 0$ for all $\mathbf{v}_h^i \in V_{h,0}^i$. In this work, the normalization constant is defined as

$$\alpha_R = \frac{\sum_{i=1}^{N_{\text{exp}}} \sum_{j=1}^{N_{\text{pts}}^i} \|\mathbf{d}_{\text{obs}}^{ij}\|^2}{\sum_{i=1}^{N_{\text{exp}}} \sum_{k=1}^{N_{\text{reac}}^i} \|R_{\text{obs}}^{ik}\|^2}$$

so as to balance the relative contributions of the displacement and reaction terms in the objective function.

A third alternative is to treat both the equilibrium condition and the displacement observations as penalization terms. In this case, both the constitutive law \mathcal{W} and the displacement fields \mathbf{d}^i are variables of the optimization problem. This formulation, when neural networks are used as ansatz for the displacement fields and the equilibrium condition is expressed in strong form, takes the form of a Physics-Informed Neural Network (PINN) [45]. As preliminary investigations with this techniques highlighted very little stability of the training process and serious difficulties in convergence, we decided not to proceed further, and focus on the second approach only.

2.3 Designing the hypothesis space

In this section we address the issue of designing an hypothesis space \mathcal{H} of candidate constitutive laws that is sufficiently rich to encompass a wide range of possible materials, while being computationally tractable.

2.3.1 Constitutive modeling requirements

To define a suitable hypothesis space \mathcal{H} , let us first review the requirements that the strain energy density functional $\mathcal{W}: \text{GL}^+(d) \rightarrow \mathbb{R} \cup \{+\infty\}$ must satisfy for physical meaningfulness and mathematical well-posedness of the equilibrium equation (1) [54, 2, 21].

R1 Residual energy condition: $\mathcal{W}(\mathbf{I}) = 0$.

This condition fixes the zero of the energy and ensures uniqueness of the energy reference. Although the energy always appears under differentiation in the stress, having a reference value is convenient for consistency.

R2 Residual stress free condition: $\mathbf{P}(\mathbf{I}) = \mathbf{0}$.

This ensures that, in the rest configuration ($\mathbf{d} = \mathbf{0}$), no residual stress is present.

R3 Frame indifference: $\mathcal{W}(\mathbf{R}\mathbf{F}) = \mathcal{W}(\mathbf{F}) \quad \forall \mathbf{F} \in \text{GL}^+(d), \mathbf{R} \in \text{SO}(d)$.

This expresses the requirement, also known as objectivity, that the material response is independent of rigid-body rotations of the current configuration. Physically, the internal energy of the material does not change if the body is rotated as a whole.

R4 Material symmetry: $\mathcal{W}(\mathbf{F}\mathbf{R}) = \mathcal{W}(\mathbf{F}) \quad \forall \mathbf{F} \in \text{GL}^+(d), \mathbf{R} \in \mathcal{G}$,

where $\mathcal{G} \subseteq \text{SO}(d)$ is the material symmetry group, i.e., the set of all rotations that leave the material response invariant. For isotropic materials, $\mathcal{G} = \text{SO}(d)$.

R5 Polyconvexity: $\mathcal{W}(\mathbf{F}) = g(\mathbf{F}, \text{cof } \mathbf{F}, \det \mathbf{F})$ for some convex $g: \mathbb{R}^{3 \times 3} \times \mathbb{R}^{3 \times 3} \times \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$.

This condition, together with suitable growth and coercivity conditions (see next points), ensures existence of weak solutions to the equilibrium equation (1), as proved by John Ball [3]. Notice that polyconvexity can be relaxed to the weaker notion of quasiconvexity (a necessary condition for the existence of solutions) [3], but the latter is untractable and difficult to verify in practice, as it is a global property which cannot be reformulated as a local one [35].

R6 Extreme compression growth: $\mathcal{W}(\mathbf{F}) \rightarrow +\infty$ as $J \rightarrow 0^+$.

This condition ensures physically realistic behavior under extreme compression and prevents interpenetration of matter, reflecting that infinite energy would be required to collapse a volume element to zero.

R7 Coercivity: $\mathcal{W}(\mathbf{F}) \geq c_0(|\mathbf{F}|^2 + |\operatorname{cof} \mathbf{F}|^{3/2}) + c_1 \quad \forall \mathbf{F} \in \operatorname{GL}^+(d) \quad \text{for some } c_0 > 0, c_1 \in \mathbb{R}.$

Similarly to the previous condition, this ensures that the extreme strains should be maintained by infinite stresses. From the mathematical viewpoint, this condition is key for the application of the direct method of the calculus of variations to the proof of existence, as it guarantees that minimizing sequences remain bounded and that a weakly convergent subsequence exists, allowing the attainment of a minimizer. The coercivity condition considered in this work is a weaker version of the original one used by [3], and is due to Müller [42].

The frame indifference requirement R3 is equivalent to asking that \mathcal{W} can be expressed as a function of the right Cauchy-Green tensor $\mathbf{C} = \mathbf{F}^T \mathbf{F}$. Furthermore, for isotropic material ($\mathcal{G} = \operatorname{SO}(d)$ in R4), \mathcal{W} can be expressed as a function of the invariants $I_1 = \operatorname{tr} \mathbf{C}$, $I_2 = \frac{1}{2}(\operatorname{tr}(\mathbf{C})^2 - \operatorname{tr}(\mathbf{C}^2))$ and J , that is there exists a function $\mathcal{W}_1: [0, +\infty) \times [0, +\infty) \times [0, +\infty) \rightarrow \mathbb{R} \cup \{+\infty\}$ such that $\mathcal{W}(\mathbf{F}) = \mathcal{W}_1(I_1, I_2, J)$.

2.3.2 Structure preserving general form

Several neural network architectures have been proposed in the literature to satisfy subsets of the requirements outlined above; however, to the best of our knowledge, no practically usable architecture has yet been shown to satisfy all of them simultaneously. For instance, the approach in [11] guarantees frame indifference and material symmetry, but polyconvexity may be violated. In [56], some of the constraints are enforced only weakly through penalty terms in the loss function and are therefore not guaranteed throughout the training process. The work of [34] represents the first neural-network-based constitutive model in which polyconvexity is enforced by design; however, this is achieved at the expense of exactly satisfying either frame indifference or the residual-stress-free condition. In [50], Neural ODEs are employed to guarantee convexity with respect to individual invariants, but under the assumption of an additive decomposition of the energy into terms depending on a single invariant, which precludes the representation of energies involving interactions among multiple invariants. In [52], polyconvexity and the residual-stress-free condition are jointly enforced; nevertheless, coercivity is not guaranteed by construction. Moreover, a convex and monotonically increasing dependence on the Jacobian J is imposed, whereas convexity alone would suffice, thereby unnecessarily restricting the admissible hypothesis space. Finally, the isochoric invariant \bar{I}_2 is used as a network input; since this invariant is not polyconvex [22, Lemma 2.4], the overall polyconvexity of the strain energy density is not strictly guaranteed.

To define a general strain energy formulation that fully preserves the structure imposed by the requirements listed in the previous section, we observe that, when the strain energy density is expressed as a function of the invariants, the Piola–Kirchhoff stress tensor admits the following representation:

$$\begin{aligned} \mathbf{P}(\mathbf{F}) &= \frac{\partial \mathcal{W}(\mathbf{F})}{\partial \mathbf{F}} = \frac{\partial \mathcal{W}_1}{\partial I_1} \frac{\partial I_1}{\partial \mathbf{F}} + \frac{\partial \mathcal{W}_1}{\partial I_2} \frac{\partial I_2}{\partial \mathbf{F}} + \frac{\partial \mathcal{W}_1}{\partial J} \frac{\partial J}{\partial \mathbf{F}} \\ &= \frac{\partial \mathcal{W}_1}{\partial I_1} 2\mathbf{F} + \frac{\partial \mathcal{W}_1}{\partial I_2} 2(I_1 \mathbf{F} - \mathbf{F} \mathbf{F}^T \mathbf{F}) + \frac{\partial \mathcal{W}_1}{\partial J} \operatorname{cof} \mathbf{F} \end{aligned}$$

It follows that, in the reference configuration $\mathbf{F} = \mathbf{I}$:

$$\mathbf{P}(\mathbf{I}) = \left(2 \frac{\partial \mathcal{W}_1}{\partial I_1} + 4 \frac{\partial \mathcal{W}_1}{\partial I_2} + \frac{\partial \mathcal{W}_1}{\partial J} \right) \Big|_{\mathbf{F}=\mathbf{I}} \mathbf{I} \quad (3)$$

Hence a practical way of guarantee the residual stress free condition R2 is to let the term inside the parenthesis in (3) vanish.

Based on the above observations, we express the candidate strain energy density functionals in the general form of the next theorem.

Theorem 1. *Consider the strain energy density functional:*

$$\mathcal{W}_1(I_1, I_2, J) = \mathcal{W}_{\text{base}}(I_1, I_2, J) + \mathcal{W}_{\text{vol}}(J) + \omega(J - 1) - \mathcal{W}_0, \quad (4)$$

where:

- $\mathcal{W}_{\text{base}}$ is a polyconvex energy density;
- $\mathcal{W}_{\text{vol}}: [0, +\infty) \rightarrow [0, +\infty)$ is a convex, \mathcal{C}^1 function such that $\mathcal{W}_{\text{vol}}(1) = \mathcal{W}'_{\text{vol}}(1) = 0$.
- $\omega \in \mathbb{R}$ is a constant defined as: $\omega = - \left(2 \frac{\partial \mathcal{W}_{\text{base}}}{\partial I_1} + 4 \frac{\partial \mathcal{W}_{\text{base}}}{\partial I_2} + \frac{\partial \mathcal{W}_{\text{base}}}{\partial J} \right) \Big|_{\mathbf{F}=\mathbf{I}}$;
- $\mathcal{W}_0 \in \mathbb{R}$ is a constant defined as: $\mathcal{W}_0 = \mathcal{W}_{\text{base}}|_{\mathbf{F}=\mathbf{I}}$.

Then, the material model (4) satisfies the requirements R1–R5.

Proof. Requirements R1–R2–R3–R4 are easy to verify. Polyconvexity follows from the fact that the sum of polyconvex functions is polyconvex, and that a convex function of J is trivially polyconvex. \square

Theorem 1 provides a systematic mechanism to enforce most of the requirements listed in Sec. 2.3.1, while retaining substantial flexibility in the choice of $\mathcal{W}_{\text{base}}$. In particular, the construction reduces the design of the constitutive model to ensuring polyconvexity of $\mathcal{W}_{\text{base}}$ and the satisfaction of the associated growth and coercivity conditions for the resulting strain energy density functional.

Remark 1. To ensure the residual stress free requirement R2, the term $\omega(J - 1)$ could be equivalently replaced either by $\frac{1}{2}\omega(I_1 - 3)$ or by $\frac{1}{4}\omega(I_2 - 3)$, or by linear combinations of these three terms. However, this would render more tricky to ensure polyconvexity. Indeed, polyconvexity requires that the strain energy be a convex function of \mathbf{F} , $\text{cof } \mathbf{F}$, and $\det \mathbf{F}$. While J is directly one of the polyconvex arguments and thus adding a term proportional to $(J - 1)$ preserves polyconvexity regardless of the sign of the coefficient, the invariants I_1 and I_2 are convex functions of \mathbf{F} and $\text{cof } \mathbf{F}$, respectively. Hence, polyconvexity is preserved provided that I_1 and I_2 are composed with a convex non-decreasing function (see Proposition 1). However, ensuring that the corresponding coefficient ω is non-negative is not straightforward if $\mathcal{W}_{\text{base}}$ is allowed to take general expressions.

2.3.3 Polyconvex Neural Networks

We introduce here a neural network architecture that satisfies polyconvexity by design, termed Polyconvex Neural Network. Similarly to other architectures proposed in the literature, PNNs are inspired by Input-Convex Neural Networks (ICNNs), introduced in [1], which are built upon two fundamental principles: (i) non-negative weighted sums of convex functions are themselves convex, and (ii) the composition of a convex function with a convex, non-decreasing function preserves convexity.

Definition 1. A Polyconvex Neural Network (PNN) with $L \geq 1$ hidden neuron layers and n_1, \dots, n_L neurons per layer is a function $\mathcal{W}_{\text{PNN}}: [0, +\infty) \times [0, +\infty) \times [0, +\infty) \rightarrow \mathbb{R}$, mapping $(I_1, I_2, J) \mapsto z_{\text{out}}$ as recursively defined below:

$$\begin{cases} \mathbf{z}_1 = \rho(\mathbf{w}_1^1(I_1 - 3) + \mathbf{w}_1^2(I_2 - 3) + \mathbf{w}_1^J(J - 1) + \mathbf{b}_1), \\ \mathbf{z}_i = \rho(\mathbf{w}_i^1(I_1 - 3) + \mathbf{w}_i^2(I_2 - 3) + \mathbf{w}_i^J(J - 1) + \mathbf{W}_{i-1}\mathbf{z}_{i-1} + \mathbf{b}_i), & \text{for } i = 2, \dots, L \\ z_{\text{out}} = \overline{\mathcal{W}} \mathbf{w}_{\text{out}} \cdot \mathbf{z}_L. \end{cases} \quad (5)$$

Here, $\mathbf{z}_i \in \mathbb{R}^{n_i}$ denotes the vector of neuron activations of the i -th layer, for $i = 1, \dots, L$. The activation function $\rho: \mathbb{R} \rightarrow \mathbb{R}$ is a convex, non-decreasing, non-constant and \mathcal{C}^1 function, while $\overline{\mathcal{W}} > 0$ is a fixed scaling constant. The trainable parameters of the network are $\mathbf{w}_i^1, \mathbf{w}_i^2, \mathbf{w}_i^J, \mathbf{b}_i \in \mathbb{R}^{n_i}$, for $i = 1, \dots, L$; $\mathbf{W}_i \in \mathbb{R}^{n_{i+1} \times n_i}$, for $i = 1, \dots, L - 1$; $\mathbf{w}_{\text{out}} \in \mathbb{R}^{n_L}$. To ensure polyconvexity, the parameters must satisfy the positivity constraints $\mathbf{w}_i^1 \succeq 0$, $\mathbf{w}_i^2 \succeq 0$, $\mathbf{W}_i \succeq 0$, $\mathbf{w}_{\text{out}} \succeq 0$ for any i .

Following the design of ICNNs [1], two architectural choices are made: first, positivity is required for most of the weights; second, the activation function ρ is chosen to be a convex and non-decreasing. As we prove below (see Theorem 3), these two choices combined together ensure that \mathcal{W}_{PNN} is polyconvex. We notice that the weight that directly multiplies J is not required to be positive. In fact, as observed in Remark 1, while I_1 and I_2 are convex functions of the triplet $(\mathbf{F}, \text{cof } \mathbf{F}, J)$, and hence can be multiplied only by

nonnegative weights to preserve polyconvexity, J is part of the triplet itself, and so can safely be multiplied by negative weights.

The constant $\overline{\mathcal{W}} > 0$ is a dimensional hyperparameter necessary to recover the physical dimension of an energy density (e.g. Pa). Its value could be set based on a prior estimate of the material overall stiffness, to provide the model with the suitable order of magnitude.

We notice that, for the sake of generality, we have included skip connections in (5), allowing each layer to directly depend on the invariants I_1 , I_2 and J , through the weights \mathbf{w}_i^1 , \mathbf{w}_i^2 and \mathbf{w}_i^J for $i \geq 2$. However, these terms can be removed without compromising the validity of the theoretical results stated below.

Theorem 2. *Let us consider the strain energy density functional \mathcal{W}_{PNN} of Definition 1. Then, for any choice of the neural network hyperparameters and parameters satisfying the positivity constraints listed in the definition, \mathcal{W}_{PNN} is polyconvex.*

Proof. First, we observe that

$$\begin{aligned} I_1 &= \text{tr } \mathbf{C} = \text{tr}(\mathbf{F}^T \mathbf{F}) = |\mathbf{F}|^2, \\ I_2 &= \text{tr}(\text{cof } \mathbf{C}) = \text{tr}((\text{cof } \mathbf{F})^T \text{cof } \mathbf{F}) = |\text{cof } \mathbf{F}|^2, \end{aligned} \tag{6}$$

where we have exploited the identity $\text{cof } \mathbf{C} = J^2 \mathbf{C}^{-T} = J^2 \mathbf{F}^{-1} \mathbf{F}^{-T} = (\text{cof } \mathbf{F})^T \text{cof } \mathbf{F}$. This shows that I_1 and I_2 are convex functions of \mathbf{F} and $\text{cof } \mathbf{F}$, respectively.

Each entry of the argument of ρ in the first layer is polyconvex, being the sum of polyconvex functions. Hence, each neuron of the first layer in the network is also polyconvex, being the composition between a polyconvex function with a convex non-decreasing function. By similar arguments, also neurons in successive layers are polyconvex. Finally, \mathcal{W}_{PNN} is polyconvex being the positive-weighted sum of polyconvex functions. \square

2.3.4 Hyperelastic Neural Networks

We are ready to introduce the hypothesis space \mathcal{H} used within this work.

Definition 2. A Hyperelastic Neural Network (HNN) is a function $\mathcal{W}_{\text{HNN}}: [0, +\infty) \times [0, +\infty) \times [0, +\infty) \rightarrow \mathbb{R}$, defined as in (4), by setting $\mathcal{W}_{\text{base}} = \mathcal{W}_{\text{PNN}}$, with \mathcal{W}_{PNN} being a PNN, and where ω and \mathcal{W}_0 are defined as in Theorem 1.

The following results shows that, for an HNN, all the requirements R1–R7 are satisfied by design, provided that the volumetric term \mathcal{W}_{vol} satisfies two additional hypothesis (growth at $J \rightarrow 0^+$, and superlinear growth at $J \rightarrow +\infty$).

Theorem 3. *Let us consider a HNN, where \mathcal{W}_{vol} satisfies the hypotheses of Theorem 1 with the additional hypotheses:*

$$\lim_{J \rightarrow 0^+} \mathcal{W}_{\text{vol}}(J) = +\infty, \quad \lim_{J \rightarrow +\infty} \frac{\mathcal{W}_{\text{vol}}(J)}{J} = +\infty. \tag{7}$$

Then, for any choice of the neural network hyperparameters and parameters, the strain energy density functional satisfies all the requirements R1–R7.

Proof. Requirements R1–R5 follow from Theorem 1 and Theorem 2. To prove requirement R6, it suffices to observe that \mathcal{W}_{PNN} is non-decreasing in both I_1 and I_2 , being the composition of non-decreasing functions. Therefore $\mathcal{W}_{\text{PNN}}(I_1, I_2, J) \geq \mathcal{W}_{\text{PNN}}(0, 0, J)$, which has clearly a finite limit when $J \rightarrow 0^+$. Hence, thanks to (7), requirement R6 holds true.

To prove coercivity (requirement R7), we start by observing that, as ρ is \mathcal{C}^1 convex and non-decreasing but non-constant, we have that $\rho(y) \geq ay + b$ for some $a > 0$ and $b \in \mathbb{R}$. It follows that the first layer of \mathcal{W}_{PNN} satisfies:

$$\exists a > 0, b \in \mathbb{R}, c \in \mathbb{R} \text{ s.t. } \quad \mathbf{z}_1 \succeq a(I_1 + I_2) + bJ + c.$$

By the same argument, all successive layers \mathbf{z}_i satisfy the same inequality. Therefore, we have:

$$\exists a > 0, b \in \mathbb{R}, c \in \mathbb{R} \text{ s.t.} \quad \mathcal{W}_{\text{PNN}}(I_1, I_2, J) \geq a(I_1 + I_2) + bJ + c.$$

Then:

$$\exists a > 0, b \in \mathbb{R}, c \in \mathbb{R} \text{ s.t.} \quad \mathcal{W}_{\text{HNN}}(I_1, I_2, J) \geq a(I_1 + I_2) + bJ + \mathcal{W}_{\text{vol}}(J) + c.$$

The function $J \mapsto bJ + \mathcal{W}_{\text{vol}}(J)$ is continuous and tends to $+\infty$ both for $J \rightarrow 0^+$ and $J \rightarrow +\infty$ (thanks to the super-linear growth of \mathcal{W}_{vol}), and hence it is bounded from below. It follows that

$$\exists a > 0, b \in \mathbb{R}, c \in \mathbb{R} \text{ s.t.} \quad \mathcal{W}_{\text{HNN}}(I_1, I_2, J) \geq a(I_1 + I_2) + c = a(|\mathbf{F}|^2 + |\text{cof } \mathbf{F}|^2) + c,$$

where we have exploited (6). This entails the coercivity of the strain energy density functional. \square

The term $\mathcal{W}_{\text{vol}}(J)$ determines the behavior of the energy for $J \rightarrow 0^+$ and $J \rightarrow +\infty$. In principle, we could think of learning $\mathcal{W}_{\text{vol}}(J)$, by designing an architecture that satisfies all the associated hypotheses. However, this would be poorly constrained by the training data, as both extreme regimes of volumetric compression and expansion are not observed in experiments. Indeed, the role of $\mathcal{W}_{\text{vol}}(J)$ is mainly to ensure the mathematical well-posedness of the problem and the finite element solvability along the whole optimization process. Therefore, we prefer to chose a priori the shape of $\mathcal{W}_{\text{vol}}(J)$, and limit ourselves to learn a multiplicative parameter. Specifically, we set $\mathcal{W}_{\text{vol}}(J) = \frac{1}{2}w_{\text{vol}}(J - 1) \log J$, where $w_{\text{vol}} > 0$ is a trainable parameter. Notice that this choice is very conservative, as it has a very mild growth both for $J \rightarrow 0^+$ (logarithmic) and for $J \rightarrow +\infty$ ($\sim J \log J$, considering that superlinear growth is required by Theorem 3).

Remark 2. An alternative to the architecture described in Definition 1 consists in replacing the invariants I_1 and I_2 with their isochoric counterparts. More precisely, we can replace I_1 by \bar{I}_1 . However, it is not convenient to use \bar{I}_2 , since it is not polyconvex (see Lemma 2.4 in [22]), and therefore the proof of Theorem 2 would no longer apply. As an alternative, one may use $\bar{I}_2^{3/2}$ as an input of the neural network, since this quantity is proven to be polyconvex (Corollary 2.3 in [22]). With this choice, the result of Theorem 2 remains valid. On the other hand, the corresponding HNN is no longer strictly coercive in the sense of requirement R7, but only coercive on sublevel sets of the form ($J < J_{\text{max}}$), as can be readily shown by revisiting the proof of Theorem 3. Nevertheless, the volumetric term $\mathcal{W}_{\text{vol}}(J)$ prevents excessive growth of J in practice, and therefore this formulation can still be safely employed in applications.

2.4 Initialization and parametrization of HNNs

The initialization of trainable parameters plays a crucial role in the successful training of any neural network architecture. Poor initialization may result in slow convergence or cause the optimizer to become trapped in undesirable regions of the parameter space [19, 36]. Whenever a novel neural architecture is introduced, a carefully designed initialization strategy is therefore essential. Typically, initialization schemes aim to control the magnitude of activations and gradients, avoiding both explosion and collapse, while preventing saturation of the activation functions, typically by keeping pre-activations in a neighborhood of zero [19, 24].

Moreover, in the HNN architecture several weights are constrained to assume positive values in order to preserve polyconvexity. Since most training algorithms are designed for working in unconstrained parameter spaces, a reparametrization is required to convert the constrained optimization problem into an unconstrained one. In this section, we describe the proposed strategy that jointly addresses the parametrization and initialization of the HNN trainable parameters.

First, we select an activation function satisfying $\rho(0) = 0$. Specifically, through this work, we employ the shifted softplus function $\rho(z) = \log(1 + e^z) - \log(2)$, which is convex, non-decreasing, \mathcal{C}^1 , non-constant, and therefore satisfies the assumptions of Definition 1. Moreover, we initialize all biases \mathbf{b}_i to zero. These conditions together ensure that, in the rest configuration (i.e. $\mathbf{F} = \mathbf{I}$), all the activations in Eq. (5) vanish. Consequently, around the rest configuration, the activation function operates in the vicinity of $z \approx 0$, where its response is well-conditioned.

Controlling the activations alone, however, does not guarantee that the initialized constitutive model exhibits physically meaningful properties. Indeed, since the Piola-Kirchhoff stress tensor is obtained by differentiating the strain energy density \mathcal{W} with respect to the deformation gradient \mathbf{F} , it is essential to ensure that the gradients of the network with respect to its inputs neither vanish – leading to an excessively compliant material – nor blow up – resulting in an unrealistically stiff response. This issue is particularly delicate for the derivatives with respect to the invariants I_1 and I_2 , whose associated coefficients are constrained to be non-negative: because the linear layers also involve non-negative weights, successive layers may accumulate positive contributions, potentially amplifying gradients as the number of neurons increases. To counteract this effect, we scale each linear layer by the inverse of the number of incoming neurons, as we detail below.

Moreover, to enforce positivity of the constrained weights, we adopt a standard reparametrization in which each constrained weight is expressed as the composition of an unconstrained trainable variable with a smooth positivity-enforcing function $\tau: \mathbb{R} \rightarrow (0, +\infty)$. Specifically, we use the softplus function $\tau(w) = \log(1 + e^w)$.

In conclusion, the resulting parametrization (for $i = 1, \dots, L$) reads:

$$\mathbf{w}_i^1 = \tau(\tilde{\mathbf{w}}_i^1), \quad \mathbf{w}_i^2 = \tau(\tilde{\mathbf{w}}_i^2), \quad \mathbf{W}_i = \frac{1}{n_i} \tau(\tilde{\mathbf{W}}_i), \quad \mathbf{w}_{\text{out}} = \frac{1}{n_L} \tau(\tilde{\mathbf{w}}_{\text{out}}),$$

where $\tilde{\mathbf{w}}_i^1$, $\tilde{\mathbf{w}}_i^2$, \mathbf{w}_i^J , $\tilde{\mathbf{W}}_i$, $\tilde{\mathbf{w}}_{\text{out}}$ are initialized independently according to Gaussian distributions with zero mean and standard deviation $\sigma_{\text{init}} > 0$. Similarly, we parametrize w_{vol} as $\tau(\tilde{w}_{\text{vol}})$, with \tilde{w}_{vol} initialized to zero.

In Appendix C, we show analytically that, thanks to the proposed scaling and parametrization, the expected value of the derivatives of the hyperelastic energy with respect to the invariants, evaluated at the reference configuration, is kept under control (i.e., it does not vanish nor it blows up) as the number of neurons increases.

2.5 Neural-DFEM

Our proposed material discovery method is based on the formulation (2) wherein the HNN architecture described in Sec. 2.3.4 is employed to define the hypothesis space \mathcal{H} . The HNN parameters are learned by minimizing the loss function (2), whose evaluation involves the solution of a set of finite element equilibrium problems, one for each training experiment. Crucially, the training procedure exploits a differentiable finite element solver, allowing gradients to be propagated through the equilibrium solution itself. As a result, the parameter updates explicitly account for the effect of the constitutive model on the mechanical response of the training specimens. We refer to this approach as Neural Differentiable Finite Element Method (Neural-DFEM).

The practical realization of this framework entails a number of nontrivial challenges that must be addressed to ensure scalability and robustness. In the following, we outline these challenges and describe how they are resolved within the Neural-DFEM framework.

- The first challenge concerns the differentiation of the finite element solver. Since the governing equilibrium problem is nonlinear and solved via an iterative Newton scheme, gradients cannot be conveniently obtained by directly differentiating the solver iterations. Instead, we rely on the solution of the corresponding algebraic adjoint system associated with the discretized equilibrium equations [10]. This approach requires solving one sparse linear system per training experiment to compute the gradients with respect to the constitutive parameters.
- A second challenge arises from the computational cost associated with the evaluation of the loss function, which requires solving multiple finite element problems – one for each training experiment – at every training epoch. This may potentially lead to a significant per-epoch cost. To mitigate this effect, we reduce the total number of training epochs by adopting a quasi-Newton optimization strategy, specifically the BFGS method [43]. This choice is particularly effective in the present setting, as the neural constitutive model operates in a low-dimensional space (e.g., the space of strain invariants), resulting in a relatively small number of trainable parameters. Consequently, the linear systems involved in the BFGS updates do not constitute a computational bottleneck. As a stopping criterion,

the optimization is terminated when the relative improvement of the loss over the last 50 iterations falls below 10^{-4} .

In addition, the cost of the initial training phase is controlled through a load continuation strategy, which is used to compute a suitable initial equilibrium solution. During subsequent training epochs, the finite element solver is warm-started using the equilibrium solution from the previous epoch. Since the constitutive response evolves smoothly over the course of training, the equilibrium solution changes only marginally between consecutive epochs, leading to a modest cost for each finite element problem.

- A further challenge stems from the fact that the loss function depends on the solution of the equilibrium problem, which must remain well posed throughout the entire optimization process, from the initial parameter guess to the final learned model. This requires mathematical well-posedness to be ensured over the whole hypothesis space explored during training. In Neural-DFEM, this requirement is satisfied by the HNN architecture introduced in Sec. 2.3.4, which is constructed to inherently enforce the fundamental principles of hyperelasticity. As proved therein, the HNN guarantees physical meaningfulness and mathematical well-posedness of the equilibrium problem for all admissible parameter values.
- Despite the mathematical well-posedness of the equilibrium problem, convergence of the finite element solver cannot be guaranteed in all cases, as the Newton method guarantees only local convergence. Two strategies can, in principle, be adopted to address this issue. A first option consists in introducing a continuation procedure to gradually transition between successive equilibrium solutions. While effective, this approach increases the computational complexity of the training process. In Neural-DFEM, we instead adopt a more conservative strategy, to prevent parameter updates from moving the system outside the region in the parameter space of approximate linearity, where gradient-based directions remain reliable. This is achieved by leveraging the backtracking mechanism embedded in the line search of the BFGS algorithm, so that trial steps leading to non-convergence of the Newton solver are automatically rejected. As a result, the optimization proceeds while remaining within the basin of attraction of the equilibrium solution, without requiring additional continuation steps during training.

2.6 Hyperparameters tuning

In this work, the following hyperparameters of the HNNs are tuned via a grid search over finite sets of candidate values, listed below:

- the number of layers, $L \in \{1, 2, 3\}$;
- the number of neurons in each layer, $n_i \in \{5, 10, 20\}$;
- the presence or absence of skip connections;
- the choice of input invariants, either the native set (I_1, I_2, J) or the isochoric set $(\bar{I}_1, \bar{I}_2^{3/2}, J)$, where the exponent $3/2$ ensures polyconvexity, as discussed in Remark 2;
- the initialization standard deviation, $\sigma_{\text{init}} \in \{0.05, 0.1, 0.2, 0.5, 0.8\}$;
- the dimensional scaling constant, $\bar{W} \in \{1, 5, 10, 20\}$.

The optimal configuration is selected as the one achieving, for $\sigma_{\text{noise}} = 10^{-3}$, the lowest training loss. Owing to the strong structural priors embedded in the HNN architecture, which substantially restrict the admissible hypothesis space, we do not employ cross-validation for hyperparameters tuning, as discussed in the Results. The selected hyperparameters values are reported in Appendix D.

2.7 Test cases

In this section, we provide details regarding the test cases implemented to test the capabilities of the proposed method.

2.7.1 Material models

We consider the following ground-truth constitutive laws.

- **Ishihara (IH)** [27], with $C_1 = 0.5$, $C_2 = 1$, $C_3 = 3$, $K = 1.5$:

$$\mathcal{W}(\mathbf{F}) = C_1(\bar{I}_1 - 3) + C_2(\bar{I}_2 - 3) + C_3(\bar{I}_1 - 3)^2 + K(J - 1)^2;$$

- **Mooney-Rivlin (MR)** [41, 47], with $C_1 = 1.0$, $C_2 = 0.8$, $K = 1$:

$$\mathcal{W}(\mathbf{F}) = C_1(I_1 - 3 - 2 \log J) + C_2(I_2 - 3 - 4 \log J) + \frac{1}{2}K(J - 1) \log J;$$

- **Fung (FU)** [17], with $C = 1$, $b = 3$, $K = 1.5$:

$$\mathcal{W}(\mathbf{F}) = \frac{C}{2b} \left(e^{b(\bar{I}_1 - 3)} - 1 \right) + \frac{K}{4} ((J - 1)^2 + \log(J)^2).$$

2.7.2 Mechanical setups

Setup 1. Setup 1, taken from [12], consists of a square plate with edge length 2 containing a central circular hole of radius 0.1, as shown in Fig. 2a. The mesh consists of 1441 nodes. The plate is subjected to asymmetric, displacement-controlled biaxial tension, with prescribed displacements of δ in the vertical direction and $\delta/2$ in the horizontal direction, with $\delta \in \{0.1, 0.2, \dots, 0.8\}$. For computational convenience, only the top-left quarter of the plate is simulated, with symmetry boundary conditions applied to account for the remaining portion of the domain. Specifically, the following boundary conditions are applied:

$$\begin{aligned} \mathbf{d} \cdot \mathbf{n} &= 0, & (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})(\mathbf{P}\mathbf{n}) &= \mathbf{0} & \text{on } \Gamma_{\text{left}} \cup \Gamma_{\text{down}}, \\ \mathbf{d} \cdot \mathbf{n} &= \delta, & (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})(\mathbf{P}\mathbf{n}) &= \mathbf{0} & \text{on } \Gamma_{\text{up}}, \\ \mathbf{d} \cdot \mathbf{n} &= \delta/2, & (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})(\mathbf{P}\mathbf{n}) &= \mathbf{0} & \text{on } \Gamma_{\text{right}}, \\ \mathbf{P}\mathbf{n} &= \mathbf{0} & & & \text{on } \partial\Omega \setminus (\Gamma_{\text{down}} \cup \Gamma_{\text{right}} \cup \Gamma_{\text{up}} \cup \Gamma_{\text{left}}). \end{aligned}$$

Setup 2. Setup 2, also from [12], consists of square plate with two ellipsoidal holes, as shown in Fig. 2b. The mesh has 4334 nodes. The specimen is subjected to displacement-controlled uniaxial tension in the vertical direction, with displacement $\delta \in \{0.1, 0.2, \dots, 1.0\}$, while the remaining portions of the boundary sides (including those of the inner holes) are traction-free. The boundary conditions read as follows:

$$\begin{aligned} \mathbf{d} &= \mathbf{0} & \text{on } \Gamma_{\text{down}}, \\ \mathbf{d} &= \begin{pmatrix} 0 \\ \delta \end{pmatrix} & \text{on } \Gamma_{\text{up}}, \\ \mathbf{P}\mathbf{n} &= \mathbf{0} & \text{on } \partial\Omega \setminus (\Gamma_{\text{down}} \cup \Gamma_{\text{up}}). \end{aligned}$$

Setup 3. Setup 3, adapted from [58], consists of 10 distinct domains, each defined as a square plate of unit edge length containing a random number of circular holes – ranging from 1 to 3 – with randomly sampled sizes and positions. Some samples are displayed in Fig. 2c. The number of mesh nodes per each geometry ranges between 5300 and 6000. The plates are subjected to normal-spring boundary conditions (with constant $k_{\perp} = 0.01$) and loaded on the two vertical sides by a uniformly distributed traction $F \in$

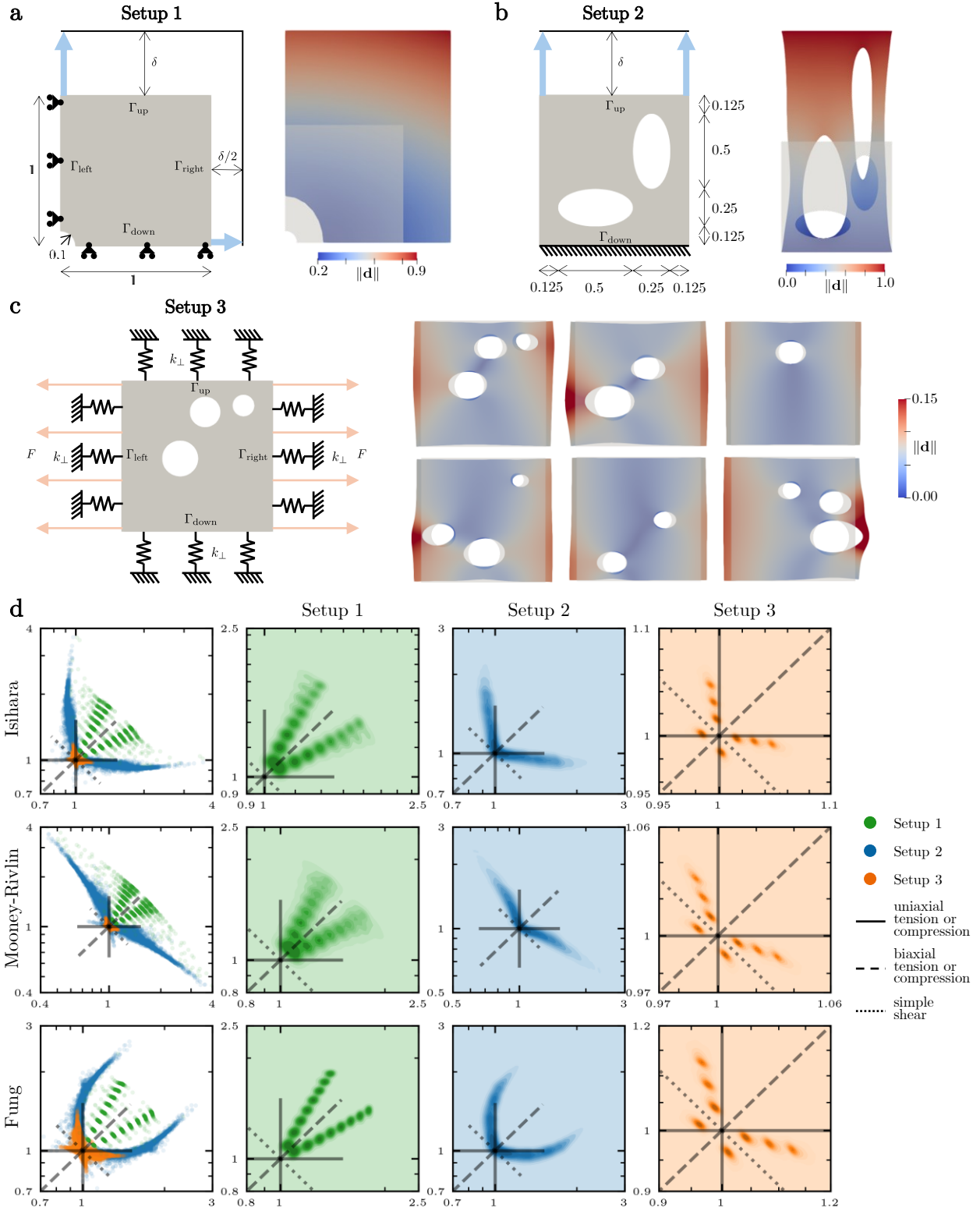


Figure 2: **2D setups.** (a)–(b)–(c): Graphical representation of Setups 1–2–3 (left), and corresponding sample solutions obtained with the IH model (right). For Setup 3, six representative domains are shown. Light blue arrows represent displacement-controlled loads, whereas light orange arrows denote applied forces per unit surface. (d): Statistical distributions of the principal stretches for the three setups considered. The scatter plots in the first column show the relative location, in the principal-stretch plane, of the deformation states obtained for the three setups, while the remaining three columns detail the principal-stretch distributions for each setup, displayed via kernel density estimation. For clarity of visualization, the principal stretches are plotted without enforcing an ordering, i.e. treating the two stretches symmetrically. Black lines correspond to canonical deformations described in Appendix B.

$\{-0.1, 0.1, 0.2, 0.3\}$, where negative values correspond to compressive loading and positive values to tensile loading. More precisely, the boundary conditions are given by:

$$\begin{aligned} \mathbf{Pn} + k_{\perp}(\mathbf{n} \otimes \mathbf{n})\mathbf{d} &= \mathbf{0} && \text{on } \Gamma_{\text{top}} \cup \Gamma_{\text{down}}, \\ \mathbf{Pn} + k_{\perp}(\mathbf{n} \otimes \mathbf{n})\mathbf{d} &= F\mathbf{n} && \text{on } \Gamma_{\text{left}} \cup \Gamma_{\text{right}}, \\ \mathbf{Pn} &= \mathbf{0} && \text{on } \partial\Omega \setminus (\Gamma_{\text{down}} \cup \Gamma_{\text{right}} \cup \Gamma_{\text{up}} \cup \Gamma_{\text{left}}). \end{aligned}$$

Setup 4. Setup 4 (see Fig. 3a) represents a three-dimensional generalization of the bi-dimensional Setup 1. It consists of a cube with an edge length 2 and a circular hole with radius 0.5. The mesh consists of 7093 nodes. Similarly to Setup 1, for symmetry reasons we simulate only 1/8 of the domain. The specimen is subjected to asymmetric, displacement-controlled triaxial tension, with prescribed displacements of δ , $\delta/2$ and $\delta/4$ in the three orthogonal directions, with $\delta \in \{0.1, 0.2, \dots, 0.5\}$. The boundary conditions hence read as follows:

$$\begin{aligned} \mathbf{d} \cdot \mathbf{n} &= 0, & (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})(\mathbf{Pn}) &= \mathbf{0} && \text{on } \Gamma_{\text{left}} \cup \Gamma_{\text{down}} \cup \Gamma_{\text{front}}, \\ \mathbf{d} \cdot \mathbf{n} &= \delta, & (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})(\mathbf{Pn}) &= \mathbf{0} && \text{on } \Gamma_{\text{right}}, \\ \mathbf{d} \cdot \mathbf{n} &= \delta/2, & (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})(\mathbf{Pn}) &= \mathbf{0} && \text{on } \Gamma_{\text{back}}, \\ \mathbf{d} \cdot \mathbf{n} &= \delta/4, & (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})(\mathbf{Pn}) &= \mathbf{0} && \text{on } \Gamma_{\text{up}}, \\ \mathbf{Pn} &= \mathbf{0} && && \text{on } \partial\Omega \setminus (\Gamma_{\text{up}} \cup \Gamma_{\text{down}} \cup \Gamma_{\text{left}} \cup \Gamma_{\text{right}} \cup \Gamma_{\text{front}} \cup \Gamma_{\text{back}}). \end{aligned}$$

Setup 5. Setup 5 consists of a cubic specimen featuring an oblique through-hole, as illustrated in Fig. 3b, based on a 2585-node mesh. One face of the specimen is fully constrained by imposing zero displacement, while the opposite face is subjected to a displacement-controlled load combining two components: a uniaxial traction of magnitude δ and a torsional deformation with rotation angle $\theta = \frac{2\pi}{5}\delta$, where $\delta \in \{0.1, 0.2, \dots, 0.5\}$. Specifically, the applied boundary conditions are defined as follows, where $\mathbf{x} = (x, y, z)^{\top}$ denotes the spatial coordinates, with the z -axis oriented along the vertical direction:

$$\begin{aligned} \mathbf{d} &= \mathbf{0} && \text{on } \Gamma_{\text{down}}, \\ \mathbf{d} &= \begin{pmatrix} (\cos \theta - 1)x - \sin \theta y \\ \sin \theta x + (\cos \theta - 1)y \\ \delta \end{pmatrix} && \text{on } \Gamma_{\text{up}}, \\ \mathbf{Pn} &= \mathbf{0} && \text{on } \partial\Omega \setminus (\Gamma_{\text{down}} \cup \Gamma_{\text{up}}). \end{aligned}$$

Setup 6. Setup 6 involves a more complex geometry than the previous cases, based on a three-dimensional mounting clip from a publicly available open-source CAD model [53]. This setup is designed to assess the generalization capability of the proposed method to markedly different geometries and mixed boundary conditions, representative of nontrivial, real-world-inspired configurations. The mesh features 12729 nodes. The geometry, shown in Fig. 3c, is constrained by imposing zero vertical displacement on the bottom surface, corresponding to a roller-type (sliding) boundary condition. The two arms of the clip are actuated by a follower load of magnitude $F \in \{0.01, 0.02, \dots, 0.06\}$, which tends to open them, as illustrated in the figure, while a uniformly distributed forward load (i.e. in the direction $\mathbf{e}_{\text{front}}$) of magnitude $F/5$ is simultaneously applied. In addition, the body is subjected to normal-spring boundary conditions on both the front surface, with spring constant $k_{\text{front}} = 0.01$, and the surface of the central hole, with spring constant $k_{\text{hole}} = 0.1$. These elastic supports are introduced to induce nontrivial internal stress states when the clip is pushed forward. To summarize, the boundary conditions are as follows:

$$\begin{aligned} \mathbf{d} \cdot \mathbf{n} &= 0, & (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})(\mathbf{Pn}) &= \mathbf{0} && \text{on } \Gamma_{\text{down}}, \\ \mathbf{Pn} + k_{\text{front}}(\mathbf{n} \otimes \mathbf{n})\mathbf{d} &= \mathbf{0} && \text{on } \Gamma_{\text{front}}, \\ \mathbf{Pn} + k_{\text{hole}}(\mathbf{n} \otimes \mathbf{n})\mathbf{d} &= \mathbf{0} && \text{on } \Gamma_{\text{hole}}, \\ \mathbf{Pn} &= -F\mathbf{J}\mathbf{F}^{-T}\mathbf{n} - F/5\mathbf{e}_{\text{front}} && \text{on } \Gamma_{\text{back}}, \\ \mathbf{Pn} &= \mathbf{0} && \text{on } \partial\Omega \setminus (\Gamma_{\text{up}} \cup \Gamma_{\text{down}} \cup \Gamma_{\text{front}} \cup \Gamma_{\text{hole}} \cup \Gamma_{\text{back}}). \end{aligned}$$

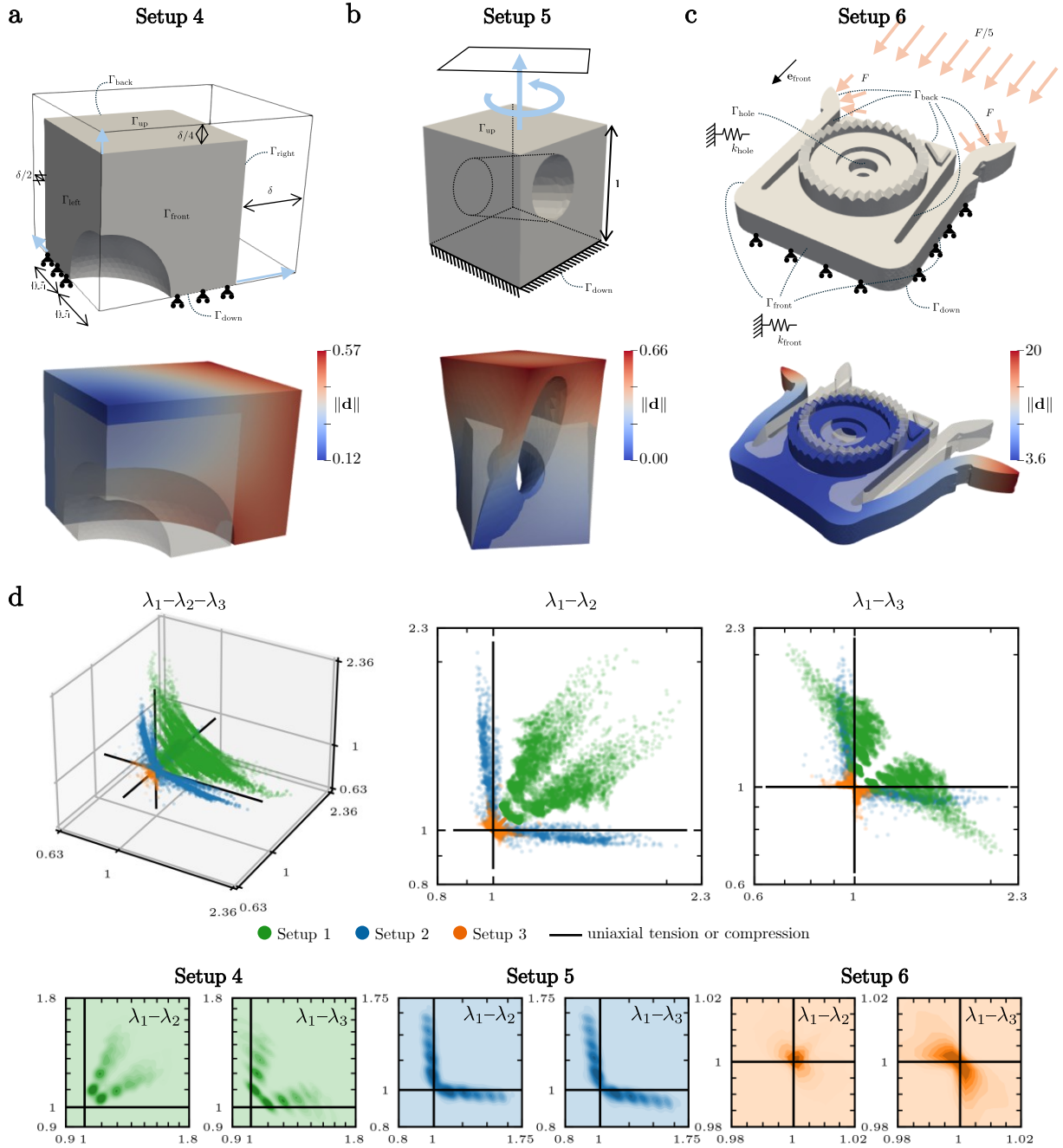


Figure 3: **3D setups.** (a)–(b)–(c): Graphical representation of Setups 4–5–6 (top), and corresponding sample solutions obtained with the IH model (bottom). Light blue arrows represent displacement-controlled loads, whereas light orange arrows denote applied forces per unit surface. (d): Statistical distributions of the principal stretches for the three setups considered. The three scatter plots in the first row show the relative location, in principal-stretch space, of the deformation states obtained for the three setups, for the principal stretch triplet $(\lambda_1, \lambda_2, \lambda_3)$, and for the (λ_1, λ_2) and (λ_1, λ_3) pairs, respectively. The second row shows the corresponding principal-stretch distributions for each setup, for the (λ_1, λ_2) and (λ_1, λ_3) pairs, displayed via kernel density estimation. For clarity of visualization, the principal stretches are plotted without enforcing an ordering, i.e. treating the stretches symmetrically. Black lines correspond to the canonical uniaxial tension or compression deformations.

3 Results

To investigate the capabilities of the proposed methods, we consider several synthetic test cases, starting from two-dimensional domains deformed under plane strain conditions, then moving to three-dimensional domains. To generate training and testing data, we employ the finite element method, and we compute numerically the equilibrium configuration resulting from known hyperelastic models. Specifically, we consider three ground truth material models, namely the Ishihara (IH) [27], the Mooney-Rivlin (MR) [41, 47], and the Fung (FU) [17] hyperelastic constitutive laws. To take into account the measurement error that unavoidably affects any source of experimental data, we add an artificial independent Gaussian noise with prescribed standard deviation σ_{noise} , that we will refer to as the noise level. To quantify the discrepancy between model predictions and ground-truth data, we consider the root mean square error normalized by the displacement variance square root (vRMSE): by construction, a naïve predictor that outputs the mean value at each point attains $\text{vRMSE} = 1$, whereas a perfect predictor with zero error yields $\text{vRMSE} = 0$ (see Appendix A for precise definitions).

3.1 Plane strain deformations with full displacement field observation

We first consider the case of two-dimensional domains with deformations taking place under plane strain conditions. Here we assume that the displacement field is measured over the whole domain (full displacement field observation). We take as benchmark cases the mechanical setups proposed in [12], and later considered in subsequent works [13, 14, 52]. Specifically, the models are trained based on the displacement field recorded from a square plate with a circular hole, subjected to a asymmetrical displacement-controlled biaxial tension with varying loading, hereafter referred to as Setup 1 (Fig. 2a). The discovered models are then tested on a different geometry, namely a square plate with two ellipsoidal holes, and subjected to displacement-controlled uniaxial tension in the vertical direction, while horizontal sides are stress-free (Setup 2, see Fig. 2b).

First, we consider the IH model with a noise magnitude $\sigma_{\text{noise}} = 10^{-3}$, representative of modern DIC setups [52]. For the proposed method, we tune the hyperparameters using a simple grid search, selecting the configuration with the lowest training loss (more details in the Section 2.6). Notably, the presence of strong structural priors and the consequent robustness to overfitting allows us to rely on this straightforward selection criterion, eliminating the need for cross-validation, which is typically required to control overfitting. For the selected architecture, we train 5 models by varying the random seed used to initialize the trainable parameters, and we select the model with lowest training loss. We benchmark our method with the only available method in the literature allowing to discover a material model from these data (unsupervised model discovery), namely the VFM. In particular, we rely on the EUCLID approach [52], based on the VFM. To provide a further reference, we also consider a traditional identification approach in which the functional form of the hyperelastic law is fixed *a priori* – here chosen as Neo-Hookean – and only its material parameters are calibrated on the training data. To provide a fair comparison, we consider the same training data and the same optimization algorithm as for Neural-DFEM. In Fig. 4a, we report boxplots of the error, obtained on the test specimen, as a function of the applied load. The results show that the model discovery methods (i.e. VFM and Neural-DFEM) achieve much better results than the calibrated Neo-Hookean model. Notably, Neural-DFEM consistently provides the most accurate results across all load values, with an average vRMSE of 2.16×10^{-3} against 1.36×10^{-2} of the VFM method (6.3 times smaller).

3.2 Noise robustness

Next, we investigate the noise robustness of Neural-DFEM, by considering noise levels up to two orders of magnitude larger than the one considered in the previous section (i.e. up to $\sigma_{\text{noise}} = 10^{-1}$), for the three ground truth material models, namely IH, MR and FU.

In Fig. 5a we show the load-reaction curves of Setup 2, obtained for different noise levels and compared with the ground truth. Except for the highest noise level in the FU case, the curves is practically indistinguishable, from a visual standpoint, from that obtained with the ground-truth model.

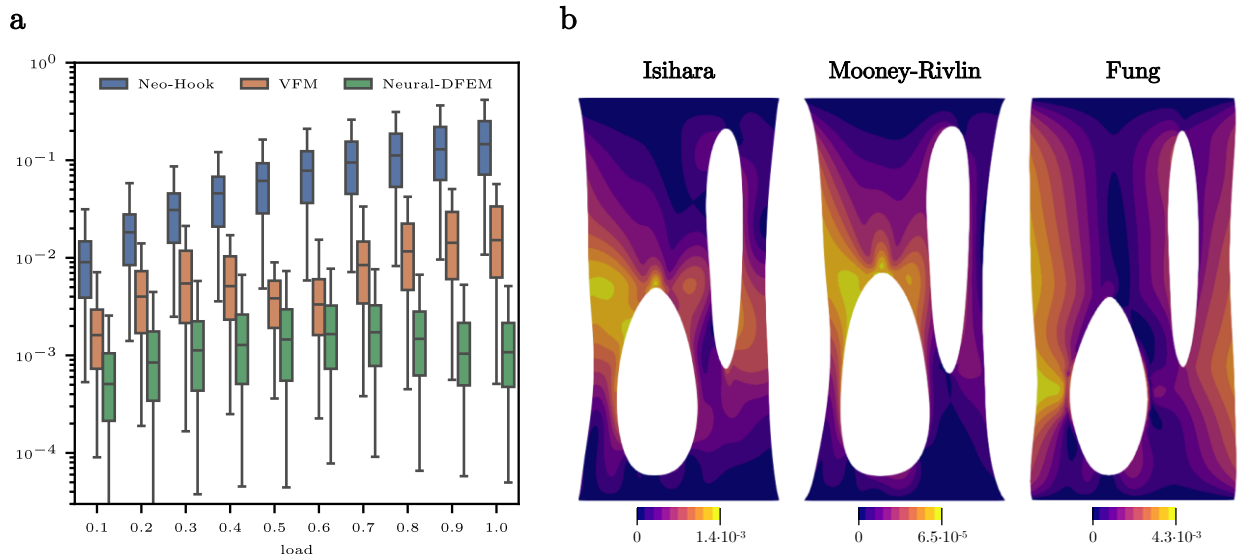


Figure 4: **Results of 2D test cases (training on Setup 1, testing on Setup 2).** (a) Boxplots of the displacement errors, normalized by the standard deviation of the ground-truth displacement, shown as a function of the applied load. The training data are generated using Setup 1 with $\sigma_{\text{noise}} = 10^{-3}$, while the test data correspond to Setup 2. Three approaches are compared: classical parameter fitting of a Neo-Hookean material; EUCLID [52], based on the VFM; and the proposed Neural-DFEM. (b) Spatial distribution of the local displacement error for Setup 2 obtained with Neural-DFEM at the largest applied load (i.e., $\delta = 1$) and with $\sigma_{\text{noise}} = 10^{-3}$, for the three material models considered. Note the different color scales used in the three cases.

Besides Setup 2, we challenge the discovered models to predict the displacement under conditions that are even farther to the training ones. To this aim, we introduce Setup 3 (see Fig. 2c), consisting of 10 different perforated plates with a random number of holes – ranging from 1 to 3 – with random sizes and positions, as in [58]. On the sides of the plates, normal spring boundary conditions are applied, and two opposite sides are loaded with normal traction with varying magnitude, taking both positive (traction) and negative (compression) sign.

In Fig. 5b, we report boxplots of the errors in both the displacement and the reaction force, as a function of the noise level, for the three materials considered. The figure shows the errors obtained not only for the training setup (Setup 1), but also for the two testing mechanical setups (Setups 2 and 3). The results indicate that the constitutive laws discovered from Setup 1 generalize well to Setup 2 and Setup 3. Indeed, for sufficiently low noise levels, a normalized error on the order of 10^{-2} or smaller is achieved in all cases. As expected, the lower the noise in the training data, the more accurate is the model learned by Neural-DFEM. Furthermore, Neural-DFEM exhibits excellent robustness to noise even at relatively high noise levels, which here reach up to two orders of magnitude higher than those considered reasonable in previous works and in typical applications [52].

Among the three materials, the highest accuracy is achieved for the MR model, suggesting that certain material behaviors may be more easily represented within the considered hypothesis space. Moreover, generalization from Setup 1 to Setup 3 appears more challenging than to Setup 2, as indicated by the larger errors. This aspect is further investigated in the next section.

3.3 Distributional shift in strain-state space and generalization

To shed light on the generalization capabilities of models discovered from a setup to a different setup, it is instructive to consider the strain state of the deformation involved. Indeed, because the domain shape differs from a setup to another, to provide a portrait of the strain state that is comparable from one setup to

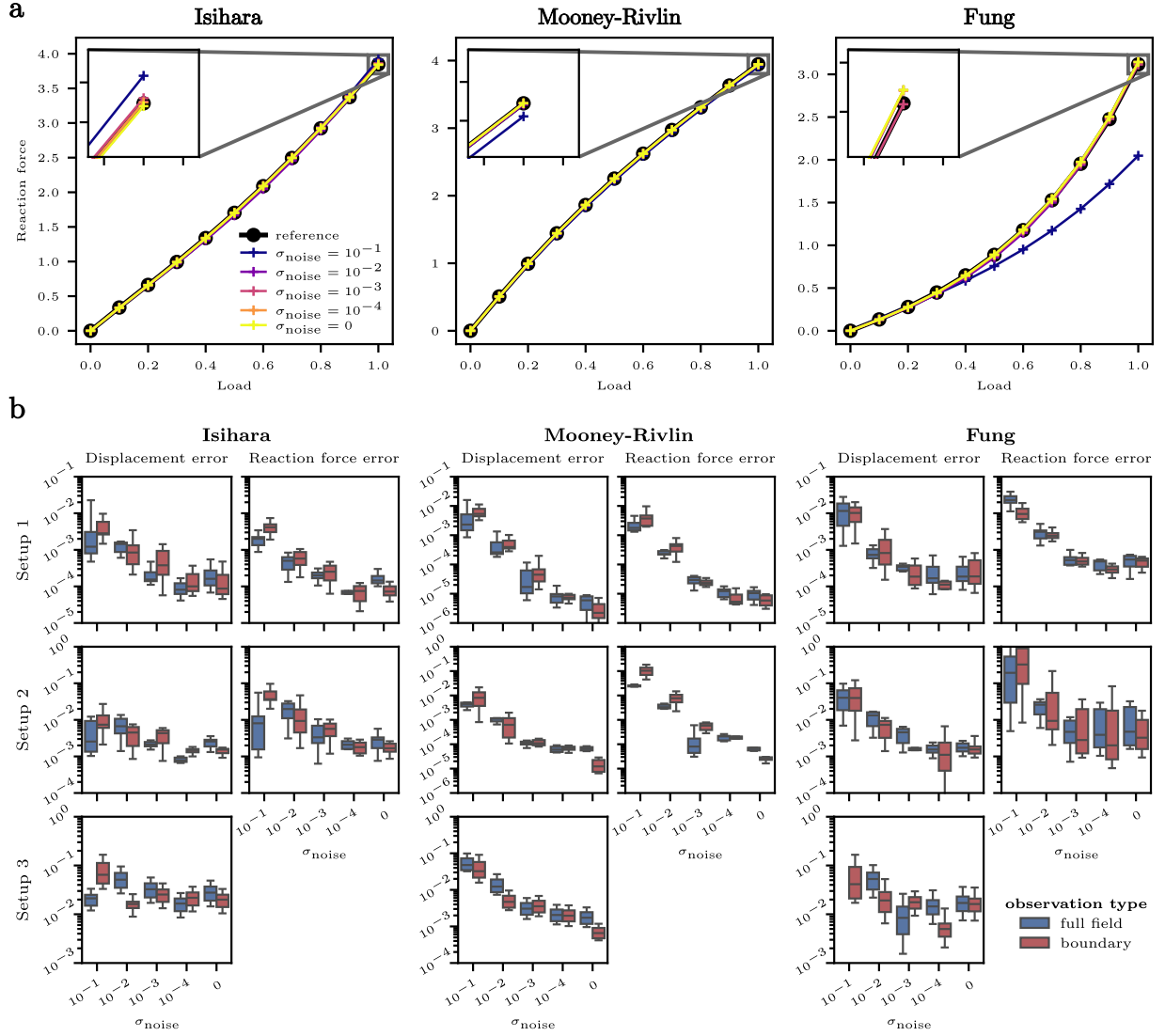


Figure 5: **Results of 2D test cases (training on Setup 1, testing on Setup 2 and Setup 3).** (a) Predicted reaction force on the top boundary of Setup 2 as a function of the applied load and of the training data noise σ_{noise} , compared with the reference reaction force obtained using the ground-truth material model. The insets show a magnified view of the force peak. (b) Boxplots of displacement and reaction force errors, both normalized by the standard deviation of the corresponding ground-truth, shown as a function of the training noise level $\sigma_{\text{noise}} \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 0\}$. Results are reported for the three material models considered (indicated at the top of each column) and for both the full-field and boundary observation cases (see the legend in the bottom-right panel). The three rows correspond to Setups 1–2–3, respectively. Hence, while the first row shows the performance in the training setting (fitting regime), the second and third rows assess the ability of Neural-DFEM to generalize to mechanical setups different from those observed at training time. For Setup 3, no reaction force is defined, since no Dirichlet boundary conditions are prescribed.

another, we propose to consider the statistical distribution of the principal stretches (λ_1, λ_2) . As shown in Fig. 2d, the training data coming from Setup 1 mainly lay in a region comprised within the state of perfect biaxial tension ($\lambda_1 = \lambda_2 > 1$) and uniaxial tension ($\lambda_1 > 1, \lambda_2 = 1$). This is not surprising, considering that the macroscopic tension state is an asymmetrical biaxial tension of the specimen. In Setup 2, the macroscopic uniaxial tension translates at the microscopic level in a strain distribution that lies in proximity of the perfect uniaxial tension state. As for Setup 3, the presence of compression load, not present in the first two setups, introduces a significant presence of microscopic strain states close to the perfect uniaxial compression state ($\lambda_1 = 1, \lambda_2 < 1$). Moreover, while Setup 1 and 2 employ a displacement-controlled load, Setup 3 is based on a tension-controlled load, with the loaded sides free to compress in the tangent direction due to the volumetric changes penalization of the materials, thus favouring the introduction microscopic strain states close to the simple shear state ($\lambda_1 \cdot \lambda_2 = 1$). As a consequence, the overall strain distribution in the specimens of the training setup (Setup 1) is closer to Setup 2 than to Setup 3. This explains the higher accuracy achieved in predicting the displacement for Setup 2 compared to Setup 3.

To quantify the distributional shift between training and test deformation regimes, we compute the Sinkhorn divergence between the empirical distributions of the principal stretches. The Sinkhorn divergence provides a measure of distributional mismatch based on entropy-regularized optimal transport [9]. The results confirm that larger divergence values consistently correspond to test cases exhibiting stronger generalization gaps, thus supporting the interpretation that generalization performance is influenced by the coverage of the strain-state manifold during training. In particular, comparing Setup 2 with Setup 3, the divergence increases from 0.17 to 0.32 for IH, from 0.24 to 0.39 for MR, and from 0.15 to 0.27 for the FU model.

3.4 Plane strain deformations with boundary observation

We consider now the case when the training specimen displacement field is measured only at the boundary of the domain. Remarkably, this case cannot be handled with VFM-based approaches, as the full displacement field should be available to evaluate the loss function. Instead, the proposed method allows – with a minor modification of the loss function with respect to the case with full-field observation – to handle this case too. We run again the same tests above, for the same noise levels. The results, reported in Fig. 5b, show that a virtually identical accuracy as for the full-field observation condition is obtained in all the cases considered, both for Setup 2 and Setup 3. More precisely, the vRMSE ratio between full-field and boundary observation cases is 1.03 ± 0.67 (mean \pm standard deviation). This remarkable result provides evidence that the full-field deformation is not necessary to learn the material response, but a partial observation, such as the boundary displacement, can be sufficient.

3.5 3D specimens with boundary observations

Finally, we consider three-dimensional specimens, which allow for a full range of deformation modes that were partially constrained in the previous plane-strain configurations.

For training, we consider Setup 4 (see Fig. 3a), a three-dimensional generalization of Setup 1, consisting of a cube containing a spherical inclusion and subjected to asymmetric triaxial displacement-controlled loads. The learned models are then tested on two additional setups designed to challenge them under markedly different deformation regimes. In Setup 5 (see Fig. 3b), the specimen is a cube with an oblique through-hole, subjected to a displacement-controlled load combining tension and torsion, while the remaining faces are stress-free. In Setup 6 (see Fig. 3c), we consider a more complex, real-world-inspired geometry based on a three-dimensional mounting clip from an open-source CAD model [53]. The clip is constrained by a roller-type condition on the bottom surface and subjected to combined follower and distributed forward loads. Additionally, normal-spring boundary conditions are imposed on selected surfaces to generate nontrivial internal stress states during loading. This setup is designed to assess generalization to markedly different geometries and mixed boundary conditions.

In Fig. 6a, we report the load-reaction curves obtained for Setup 5 at different noise levels with a boundary-only observation, while Figure 6b shows the boxplots of the displacement reconstruction error as a function of the noise level, both for the training setup and for the two testing setups. The results show

that Neural-DFEM successfully learns constitutive laws that generalize well across different configurations, also in the case of three-dimensional specimens. Remarkably, and consistently with the two-dimensional plane-strain experiments, the accuracy obtained with boundary-only observations is nearly identical to that achieved with full-field data (vRMSE ratio 1.03 ± 0.41). These results show that boundary-only measurements can be sufficient for effective material discovery even in fully three-dimensional settings. Taking $\sigma_{\text{noise}} = 10^{-3}$ and the boundary-only observation case as a reference, the vRMSE in Setups 5 and 6 is equal to 6.79×10^{-3} and 1.54×10^{-2} for IH, 9.25×10^{-4} and 4.21×10^{-4} for MR, and 1.22×10^{-3} and 1.41×10^{-3} for FU, respectively. In all cases, the method exhibits strong robustness to noise.

For the IH material model, reducing the noise level in the training set leads to a saturation of the error, indicating that Neural-DFEM is unable to extract additional information from cleaner data. This behavior is explained by the fact that the IH model is not polyconvex (as can be readily shown using [22, Lemma 2.4]) and therefore cannot be approximated with arbitrary accuracy by a HNN, which is polyconvex by construction. This behavior was not apparent in the plane-strain case, where the kinematics are restricted to a subset of deformations and the three invariants are no longer independent, effectively masking the non-polyconvex character of the IH model. Consequently, Neural-DFEM identifies the closest polyconvex, thermodynamically admissible surrogate consistent with the observed boundary responses. Importantly, despite this representational restriction, the identified surrogate still reproduces the macroscopic mechanical response with good accuracy (error of the order of 10^{-2}). In contrast, for the MR and FU material models, which are polyconvex, the approximation error decreases consistently as the noise level is reduced, confirming that Neural-DFEM can recover the underlying constitutive behavior with increasing accuracy when it is not far from the considered hypothesis class.

Similarly to the two-dimensional case, the slightly larger error observed in Setup 6 compared to Setup 5 is explained by the fact that, as shown in Fig. 3d, the microscopic strain states in Setup 5 are closer to those of the training setup (Setup 4) than those in Setup 6. Quantitatively, the Sinkhorn divergence from the empirical distribution of the principal stretches observed during training increases from Setup 5 to Setup 6: from 0.08 to 0.16 for IH, from 0.10 to 0.16 for MR, and from 0.08 to 0.15 for the FU model. This supports that generalization can be analyzed through the lens of the distributional shift in the strain distribution from the training to the test set.

4 Discussion

This work shows, for the first time in the literature, that material model discovery can be successfully performed in a fully unsupervised setting even under partial observation regimes, relying solely on measured specimen boundary deformations and without paired strain–stress data. These results provide evidence that experimentally viable material discovery under partial observability may be feasible, although validation on real experimental data remains an important direction for future work.

The proposed computational framework is built upon a differentiable finite element solver embedded within the learning process. This design choice allows us to establish a direct link between candidate constitutive laws and the resulting deformation fields, enabling a high flexibility in the formulation of the loss function, including, for instance, mean squared error terms evaluated on arbitrary subsets of the domain. Such flexibility is not attainable neither with supervised material discovery methods, which require paired strain–stress data, nor with standard VFM, which requires full-field displacement data to be known *a priori*. Moreover, because VFM relies on differentiating measured displacement fields – an operation that amplifies measurement noise – it typically necessitates a dedicated denoising stage. In contrast, the proposed approach avoids this requirement entirely, resulting in a learning process that is inherently robust to noisy observations.

Neural-DFEM relies on a novel neural-network-based constitutive ansatz, represented by the class of HNNs, which provides mathematically provable guarantees of physical and thermodynamic consistency, as well as mathematical well-posedness of the associated equilibrium problem. These properties are essential for Neural-DFEM, as convergence of the embedded solver must be ensured throughout the entire training trajectory, rather than only at the final learned solution. Finally, the proposed framework incorporates a set of carefully designed numerical strategies – including quasi-Newton optimization schemes, continuation meth-

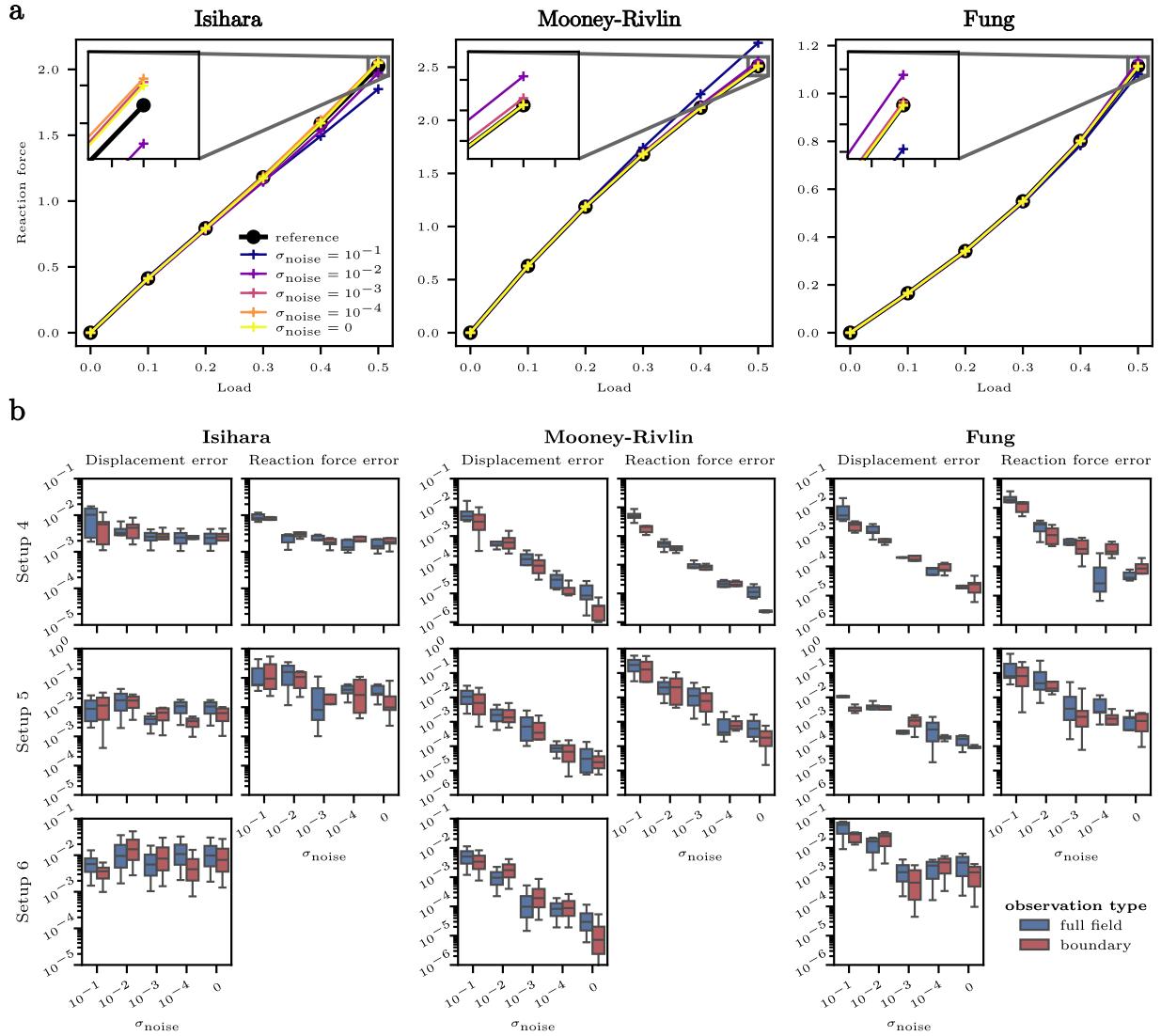


Figure 6: **Results of 3D test cases (training on Setup 4, testing on Setup 5 and Setup 6).** (a) Predicted reaction force on the top boundary of Setup 5 as a function of the applied load and of the training data noise σ_{noise} , compared with the reference reaction force obtained using the ground-truth material model. The insets show a magnified view of the force peak. (b) Boxplots of displacement and reaction force errors, both normalized by the standard deviation of the corresponding ground-truth, shown as a function of the training noise level $\sigma_{\text{noise}} \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 0\}$, and of the observation type (full-fields vs boundary only, see legend). Results are reported for the three material models considered (indicated at the top of each column) and for both the full-field and boundary observation cases (see the legend in the bottom-right panel). The three rows correspond to Setups 4–5–6, respectively. Hence, while the first row shows the performance in the training setting (fitting regime), the second and third rows assess the ability of Neural-DFEM to generalize to mechanical setups different from those observed at training time. For Setup 6, no reaction force is defined, since no Dirichlet boundary conditions are prescribed.

ods, a backtracking line search that enforces convergence of the nonlinear solver, and a purposely designed initialization strategy for the neural network parameters – which together ensure robustness and computational efficiency of the overall approach. As a result, training can be carried with limited computational resources. More precisely, all experiments were executed on a single CPU core of an AMD processor (up to 3.63 GHz), and training took on average 7.38 h (IH), 1.84 h (MR) and 1.38 h (FU) in the two-dimensional case; 20.16 h (IH), 8.26 h (MR) and 24.79 h (FU) in the three-dimensional case.

An alternative paradigm that could, in principle, enable material discovery from partial observations is that of Physics-Informed Neural Networks (PINNs). However, our initial investigations (not reported here) revealed severe convergence difficulties for PINNs in this class of problems, even for comparatively simple configurations. While further studies may be warranted, these observations suggest that approaches based on differentiable solvers are more suitable for constitutive model discovery in mechanics.

In parallel with advances in material discovery, a growing body of work has focused on operator learning methods aimed at predicting the deformation of elastic bodies directly from data. Examples include approaches based on autoencoders [31] graph neural networks [51], PointNet architectures [30], Universal Solution Manifold Networks [58], and DeepONets [23]. While these methods can deliver extremely rapid predictions at inference time, they typically require large training datasets and often struggle to generalize to geometries and boundary conditions that differ substantially from those encountered during training. As such, these methods primarily function as surrogate models, acting as accelerators of numerical solvers trained on extensive sets of precomputed simulations that provide the dominant learning bias. By contrast, Neural-DFEM relies on strong inductive biases rooted in fundamental physical principles: the momentum balance equation is enforced explicitly at every training epoch through the finite element solver, and the constitutive model is constrained by requirements such as frame indifference, polyconvexity, and coercivity, thanks to the newly introduced architecture. These ingredients reduce the dependence on purely data-driven learning bias and promote generalization across geometries and boundary conditions well beyond those used during training, provided that they induce strain states sufficiently represented during training, as illustrated by the numerical experiments. From this perspective, the learning task shifts from approximating the global, macroscopic response of a body to identifying its local, microscopic material behavior.

Consequently, within this framework, concepts such as generalization and extrapolation are most naturally interpreted at the level of local strain states. In particular, generalization across experimental setups can be understood as a form of distributional shift in the space of strain invariants. Our results indicate that predictive accuracy deteriorates as the strain-state distribution encountered at test time departs from that sampled during training, consistently with the broader machine-learning literature on out-of-distribution generalization. This behavior is not specific to the present method but reflects a fundamental limitation of data-driven material discovery: when local deformation states are insufficiently represented in the training data, the inferred surrogate is constrained only by the inductive biases embedded in the hypothesis class, but too strong structural priors typically sacrifice expressivity. Therefore, analyzing the statistical distribution of local strain states and quantifying their distributional mismatch relative to the training regime provides a physically grounded lens through which generalization performance can be interpreted. Such analysis may serve as an *a posteriori* diagnostic of prediction reliability and suggests a potential guideline for designing informative training configurations.

A limitation of the present work is its restriction to isotropic materials. Nevertheless, the proposed HNN framework can be extended in a straightforward manner to anisotropic settings by incorporating anisotropic invariants into the model input. If the directions of anisotropy are known *a priori*, the extension is immediate; if they are unknown, they may be introduced as additional variables to be inferred within the optimization process.

Ultimately, this study highlights the importance of incorporating fundamental physical principles as inductive biases directly into computational frameworks for scientific discovery. Constraining the hypothesis space to thermodynamically consistent constitutive laws and enforcing equilibrium throughout training structurally regularizes the inverse problem, enabling reliable and robust generalization even under partial observability.

Acknowledgements

The present research has received support from the project FIS, MUR, Italy 2025-2028, Project code: FIS-2023-02228, CUP: D53C24005440001, “SYNERGIZE: Synergizing Numerical Methods and Machine Learning for a new generation of computational models”. The author acknowledges MUR, grant Dipartimento di Eccellenza 2023-2027. The author is member of GNCS, “Gruppo Nazionale per il Calcolo Scientifico” (National Group for Scientific Computing) of INdAM (Istituto Nazionale di Alta Matematica).

Appendix

A Error metrics

Given a collection of N points, and associated displacement vectors \mathbf{d}_i , for $i = 1, \dots, N$, we define the average displacement as

$$\bar{\mathbf{d}} = \frac{1}{N} \sum_{i=1}^N \mathbf{d}_i,$$

and the displacement variance as

$$\text{Var}(\mathbf{d}) = \frac{1}{N} \sum_{i=1}^N \|\mathbf{d}_i - \bar{\mathbf{d}}\|^2.$$

Now, let us consider a collection of reference displacements $\hat{\mathbf{d}}_i$, for $i = 1, \dots, N$. We define the *root mean square error* (RMSE) between $\{\mathbf{d}_i\}_i$ and $\{\hat{\mathbf{d}}_i\}_i$ as:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N \|\mathbf{d}_i - \hat{\mathbf{d}}_i\|^2}.$$

Consequently, we define the *variance-normalized root mean square error* (vRMSE) as:

$$\text{vRMSE} = \frac{\text{RMSE}}{\sqrt{\text{Var}(\hat{\mathbf{d}})}}.$$

This definition is such that a naïve predictor, predicting the average displacement in each point, would score $\text{vRMSE} = 1$, while a perfect predictor achieving zero error satisfies $\text{vRMSE} = 0$.

B Canonical deformations

In Fig. 2c, in addition to the distribution of principal stretches corresponding to the considered mechanical setups, the principal stretches associated with the following canonical deformations are shown for reference:

- Uniaxial tension: $\mathbf{F} = \begin{pmatrix} 1 + \delta & 0 \\ 0 & 1 \end{pmatrix}$;
- Uniaxial compression: $\mathbf{F} = \begin{pmatrix} \frac{1}{1 + \delta} & 0 \\ 0 & 1 \end{pmatrix}$;
- Biaxial tension: $\mathbf{F} = \begin{pmatrix} 1 + \delta & 0 \\ 0 & 1 + \delta \end{pmatrix}$;

- Biaxial compression: $\mathbf{F} = \begin{pmatrix} \frac{1}{1+\delta} & 0 \\ 0 & \frac{1}{1-\delta} \end{pmatrix}$;
- Simple shear: $\mathbf{F} = \begin{pmatrix} 1 & \delta \\ 0 & 1 \end{pmatrix}$.

In all cases, the range $\delta \in [0, 0.5]$ is represented in the figure.

C HNN initialization

In this section, we analytically prove that, under the proposed parametrization and initialization strategy, the derivative of a HNN with respect to the invariants I_1 and I_2 at the reference configuration remains bounded as the number of neurons increases. For brevity, we present the derivation for I_1 only, as the same arguments apply to I_2 .

Case with skip connections. Consider first the architecture including skip connections. From (5), we obtain

$$\begin{cases} \frac{\partial \mathbf{z}_1}{\partial I_1} = \mathbf{w}_1^1 \odot \rho'(\mathbf{w}_1^1(I_1 - 3) + \mathbf{w}_1^2(I_2 - 3) + \mathbf{w}_1^J(J - 1) + \mathbf{b}_1), \\ \frac{\partial \mathbf{z}_i}{\partial I_1} = \left(\mathbf{w}_i^1 + \mathbf{W}_{i-1} \frac{\partial \mathbf{z}_{i-1}}{\partial I_1} \right) \odot \rho'(\mathbf{w}_i^1(I_1 - 3) + \mathbf{w}_i^2(I_2 - 3) + \mathbf{w}_i^J(J - 1) + \mathbf{W}_{i-1} \mathbf{z}_{i-1} + \mathbf{b}_i), & \text{for } i = 2, \dots, L \\ \frac{\partial z_{\text{out}}}{\partial I_1} = \overline{\mathcal{W}} \mathbf{w}_{\text{out}} \cdot \frac{\partial \mathbf{z}_L}{\partial I_1}, \end{cases}$$

where \odot denotes the Hadamard (i.e., componentwise) product. In the reference configuration ($\mathbf{F} = \mathbf{I}$), all pre-activations vanish, hence $\rho'(\cdot) = \rho'(0)$. Therefore,

$$\begin{cases} \frac{\partial \mathbf{z}_1}{\partial I_1} = \mathbf{w}_1^1 \rho'(0), \\ \frac{\partial \mathbf{z}_i}{\partial I_1} = \left(\mathbf{w}_i^1 + \mathbf{W}_{i-1} \frac{\partial \mathbf{z}_{i-1}}{\partial I_1} \right) \rho'(0), & \text{for } i = 2, \dots, L \\ \frac{\partial z_{\text{out}}}{\partial I_1} = \overline{\mathcal{W}} \mathbf{w}_{\text{out}} \cdot \frac{\partial \mathbf{z}_L}{\partial I_1}. \end{cases}$$

Taking expectations with respect to the random initialization (leveraging independence and identical distributions of the trainable parameters), and denoting $\mathbb{E}[\tau(\sigma_{\text{init}} Z)] = \overline{w}$, where $Z \sim \mathcal{N}(0, 1)$ is a standard Gaussian random variable, we obtain the recurrence:

$$\begin{cases} \mathbb{E} \left[\frac{\partial \mathbf{z}_1}{\partial I_1} \right] = \overline{w} \rho'(0) \mathbf{1}_{n_1}, \\ \mathbb{E} \left[\frac{\partial \mathbf{z}_i}{\partial I_1} \right] = \overline{w} \rho'(0) \left(\mathbf{1}_{n_i} + \frac{1}{n_{i-1}} \mathbf{1}_{n_i \times n_{i-1}} \mathbb{E} \left[\frac{\partial \mathbf{z}_{i-1}}{\partial I_1} \right] \right), & \text{for } i = 2, \dots, L \\ \mathbb{E} \left[\frac{\partial z_{\text{out}}}{\partial I_1} \right] = \overline{\mathcal{W}} \overline{w} \frac{1}{n_L} \mathbf{1}_{n_L} \cdot \mathbb{E} \left[\frac{\partial \mathbf{z}_L}{\partial I_1} \right]. \end{cases}$$

By defining the constant $\gamma = \bar{w}\rho'(0)$, we have by induction:

$$\begin{aligned}\mathbb{E}\left[\frac{\partial \mathbf{z}_2}{\partial I_1}\right] &= \gamma(1+\gamma)\mathbf{1}_{n_2} \\ \mathbb{E}\left[\frac{\partial \mathbf{z}_3}{\partial I_1}\right] &= \gamma(1+\gamma(1+\gamma))\mathbf{1}_{n_3} \\ &\vdots \\ \mathbb{E}\left[\frac{\partial \mathbf{z}_i}{\partial I_1}\right] &= \sum_{k=1}^i (\gamma)^k \mathbf{1}_{n_i} = \frac{\gamma}{1-\gamma} (1-\gamma^i) \mathbf{1}_{n_i}\end{aligned}$$

where the last equation holds for any $i = 1, \dots, L$. In the above calculation, we have exploited the identity $\frac{1}{n_{i-1}}\mathbf{1}_{n_i \times n_{i-1}}\mathbf{1}_{n_{i-1}} = \mathbf{1}_{n_i}$. Finally, the derivative of the output neuron with respect to I_1 reads:

$$\mathbb{E}\left[\frac{\partial z_{\text{out}}}{\partial I_1}\right] = \bar{W}\bar{w} \frac{\gamma}{1-\gamma} (1-\gamma^L) \quad (8)$$

This means that the expected value of the derivative with respect to I_1 is independent of the number of neurons of each layer. It instead depends on the number of layers. However, Eq. (8) provides a guideline to avoid blows up of this quantity when deep architecture are employed: indeed, σ_{init} and ρ must be chosen so that the constant $\gamma = \bar{w}\rho'(0)$ is smaller than one. In this case, indeed, we have a finite limit:

$$\mathbb{E}\left[\frac{\partial z_{\text{out}}}{\partial I_1}\right] \xrightarrow{L \rightarrow +\infty} \bar{W}\bar{w} \frac{\gamma}{1-\gamma}$$

Case without skip connections. Let us now consider the case when skip connections are not employed, that is when the terms \mathbf{w}_i^1 , \mathbf{w}_i^2 and \mathbf{w}_i^J for $i \geq 2$ are removed. By proceeding similarly to above, we have:

$$\begin{cases} \mathbb{E}\left[\frac{\partial \mathbf{z}_1}{\partial I_1}\right] = \gamma \mathbf{1}_{n_1}, \\ \mathbb{E}\left[\frac{\partial \mathbf{z}_i}{\partial I_1}\right] = \gamma \frac{1}{n_{i-1}} \mathbf{1}_{n_i \times n_{i-1}} \mathbb{E}\left[\frac{\partial \mathbf{z}_{i-1}}{\partial I_1}\right], & \text{for } i = 2, \dots, L \\ \mathbb{E}\left[\frac{\partial z_{\text{out}}}{\partial I_1}\right] = \bar{W}\bar{w} \frac{1}{n_L} \mathbf{1}_{n_L} \cdot \mathbb{E}\left[\frac{\partial \mathbf{z}_L}{\partial I_1}\right]. \end{cases}$$

This entails that

$$\mathbb{E}\left[\frac{\partial \mathbf{z}_i}{\partial I_1}\right] = \gamma^i \mathbf{1}_{n_i},$$

and, in conclusion

$$\mathbb{E}\left[\frac{\partial z_{\text{out}}}{\partial I_1}\right] = \bar{W}\bar{w} \gamma^L.$$

In this case, the expected derivative is again independent of the number of neurons per layer. For deep architectures, choosing γ close to one prevents both vanishing and blow-up of the gradient.

D Hyperparameters values

The values of hyperparameters selected as described in Sec. 2.6 are reported in Tab. 1.

Hyperparameter	2D specimens			3D specimens		
	IH	MR	FU	IH	MR	FU
Number of layers L	2	1	1	1	1	2
Number of neurons per layer n_i	5	5	5	5	5	5
Skip connections	\times	-	-	-	-	\times
Isochoric input invariants	\times	\times	\times	\times	\times	\checkmark
Initialization std. σ_{init}	0.6	0.1	0.1	0.1	0.1	0.6
Scaling constant \overline{W}	1	10	10	10	10	1

Table 1: Selected hyperparameters for the HNN models.

E Basic lemmas on convex functions

In this section we collect standard definitions and results on convex functions, employed in this work.

Definition 3. A function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ is convex if for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and $\lambda \in [0, 1]$:

$$f(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) \leq \lambda f(\mathbf{x}) + (1 - \lambda) f(\mathbf{y}). \quad (9)$$

Proposition 1. Consider a convex function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ and a convex, non-decreasing function $g: \mathbb{R} \rightarrow \mathbb{R}$. Then the composition $g \circ f: \mathbb{R}^n \rightarrow \mathbb{R}$ is also convex.

Proof. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and $\lambda \in [0, 1]$. Since g is non-decreasing, we can apply it to both sides of (9):

$$\begin{aligned} g(f(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y})) &\leq g(\lambda f(\mathbf{x}) + (1 - \lambda) f(\mathbf{y})) \\ &\leq \lambda g(f(\mathbf{x})) + (1 - \lambda) g(f(\mathbf{y})), \end{aligned}$$

where in the last line we have used the convexity of g . This proves that $g \circ f$ is convex. \square

Lemma 1. Let $f: \mathbb{R} \rightarrow \mathbb{R}$ be a C^1 function which is convex, non-decreasing, and non-constant. Then there exist constants $a > 0$ and $b \in \mathbb{R}$ such that

$$f(x) \geq ax - b, \quad \forall x \in \mathbb{R}.$$

Proof. Since f is C^1 , non-decreasing but non-constant, then there exists $x_0 \in \mathbb{R}$ such that $f'(x_0) > 0$. Moreover, as f is convex, we have

$$f(x) \geq f(x_0) + f'(x_0)(x - x_0), \quad \forall x \in \mathbb{R},$$

which proves the thesis. \square

References

- [1] Brandon Amos, Lei Xu, and J Zico Kolter. Input convex neural networks. In *International Conference on Machine Learning*, pages 146–155. PMLR, 2017.
- [2] Stuart S Antman. *Nonlinear Problems of Elasticity*. Springer, 2005.
- [3] John M Ball. Convexity conditions and existence theorems in nonlinear elasticity. *Archive for Rational Mechanics and Analysis*, 63(4):337–403, 1976.
- [4] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016.

- [5] Kathleen Champion, Bethany Lusch, J Nathan Kutz, and Steven L Brunton. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116(45):22445–22451, 2019.
- [6] Sergio Conti, Stefan Müller, and Michael Ortiz. Data-driven problems in elasticity. *Archive for Rational Mechanics and Analysis*, 229(1):79–123, 2018.
- [7] Miles Cranmer, Sam Greydanus, Stephan Hoyer, Peter Battaglia, David Spergel, and Shirley Ho. Lagrangian neural networks. *arXiv preprint arXiv:2003.04630*, 2020.
- [8] Miles Cranmer, Alvaro Sanchez Gonzalez, Peter Battaglia, Rui Xu, Kyle Cranmer, David Spergel, and Shirley Ho. Discovering symbolic models from deep learning with inductive biases. *34th Conference on Neural Information Processing Systems (NeurIPS)*, 33:17429–17442, 2020.
- [9] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in Neural Information Processing Systems*, 26, 2013.
- [10] Patrick E Farrell, David A Ham, Simon W Funke, and Marie E Rognes. Automated derivation of the adjoint of high-level transient finite element programs. *SIAM Journal on Scientific Computing*, 35(4):C369–C393, 2013.
- [11] Mauricio Fernández, Mostafa Jamshidian, Thomas Böhlke, Kristian Kersting, and Oliver Weeger. Anisotropic hyperelastic constitutive models for finite deformations combining material theory and data-driven approaches with application to cubic lattice metamaterials. *Computational Mechanics*, 67(2):653–677, 2021.
- [12] Moritz Flaschel, Siddhant Kumar, and Laura De Lorenzis. Unsupervised discovery of interpretable hyperelastic constitutive laws. *Computer Methods in Applied Mechanics and Engineering*, 381:113852, 2021.
- [13] Moritz Flaschel, Siddhant Kumar, and Laura De Lorenzis. Discovering plasticity models without stress data. *npj Computational Materials*, 8(1):91, 2022.
- [14] Moritz Flaschel, Siddhant Kumar, and Laura De Lorenzis. Automated discovery of generalized standard material models with euclid. *Computer Methods in Applied Mechanics and Engineering*, 405:115867, 2023.
- [15] Moritz Flaschel, Huitian Yu, Nina Reiter, Jan Hinrichsen, Silvia Budday, Paul Steinmann, Siddhant Kumar, and Laura De Lorenzis. Automated discovery of interpretable hyperelastic material models for human brain tissue with euclid. *Journal of the Mechanics and Physics of Solids*, 180:105404, 2023.
- [16] Ari L Frankel, Reese E Jones, and Laura P Swiler. Tensor basis gaussian process models of hyperelastic materials. *Journal of Machine Learning for Modeling and Computing*, 1(1), 2020.
- [17] Yuan-cheng Fung. *Biomechanics: Mechanical Properties of Living Tissues*. Springer Science & Business Media, 2013.
- [18] Emmanuel E. Gdoutos. *Digital Image Correlation (DIC)*, pages 251–259. Springer International Publishing, Cham, 2022.
- [19] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.
- [20] Michel Grediac, Fabrice Pierron, Stéphane Avril, and Evelyne Toussaint. The virtual fields method for extracting constitutive parameters from full-field measurements: a review. *Strain*, 42(4):233–253, 2006.

- [21] Morton E Gurtin, Eliot Fried, and Lallit Anand. *The Mechanics and Thermodynamics of Continua*. Cambridge University Press, 2010.
- [22] Stefan Hartmann and Patrizio Neff. Polyconvexity of generalized polynomial-type hyperelastic strain energy functions for near-incompressibility. *International Journal of Solids and Structures*, 40(11):2767–2791, 2003.
- [23] Junyan He, Seid Koric, Shashank Kushwaha, Jaewan Park, Diab Abueidda, and Iwona Jasiuk. Novel deepnet architecture to predict stresses in elastoplastic structures with variable complex geometries and loads. *Computer Methods in Applied Mechanics and Engineering*, 415:116277, 2023.
- [24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.
- [25] Hagen Holthusen, Kevin Linka, Ellen Kuhl, and Tim Brepols. A generalized dual potential for inelastic constitutive artificial neural networks: A jax implementation at finite strains. *Journal of the Mechanics and Physics of Solids*, page 106337, 2025.
- [26] Daniel Z Huang, Kailai Xu, Charbel Farhat, and Eric Darve. Learning constitutive relations from indirect observations using deep neural networks. *Journal of Computational Physics*, 416:109491, 2020.
- [27] Akira Ishihara. Statistical theory of rubber-like elasticity (the network structure). *Journal of the Physical Society of Japan*, 3(5-6):289–291, 1948.
- [28] Raban Iten, Tony Metger, Henrik Wilming, Lída Del Rio, and Renato Renner. Discovering physical concepts with neural networks. *Physical Review Letters*, 124(1):010508, 2020.
- [29] George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.
- [30] Ali Kashefi, Leonidas J Guibas, and Tapan Mukerji. Physics-informed pointnet: On how many irregular geometries can it solve an inverse problem simultaneously? application to linear elasticity. *Journal of Machine Learning for Modeling and Computing*, 4(4), 2023.
- [31] Jannick Kehls, Ellen Kuhl, Tim Brepols, Kevin Linka, and Hagen Holthusen. Autoencoder-based non-intrusive model order reduction in continuum mechanics. *arXiv preprint arXiv:2509.02237*, 2025.
- [32] Trenton Kirchdoerfer and Michael Ortiz. Data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 304:81–101, 2016.
- [33] Trenton Kirchdoerfer and Michael Ortiz. Data-driven computing in dynamics. *International Journal for Numerical Methods in Engineering*, 113(11):1697–1710, 2018.
- [34] Dominik K Klein, Mauricio Fernández, Robert J Martin, Patrizio Neff, and Oliver Weeger. Polyconvex anisotropic hyperelasticity with neural networks. *Journal of the Mechanics and Physics of Solids*, 159:104703, 2022.
- [35] Jan Kristensen. On the non-locality of quasiconvexity. *Annales de l’Institut Henri Poincaré C, Analyse non linéaire*, 16(1):1–13, 1999.
- [36] Yann LeCun, Léon Bottou, Genevieve B Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–50. Springer, 2002.
- [37] Guanghui Liang and K Chandrashekhara. Neural network based constitutive model for elastomeric foams. *Engineering Structures*, 30(7):2002–2011, 2008.

- [38] Kevin Linka, Markus Hillgärtner, Kian P Abdolazizi, Roland C Aydin, Mikhail Itskov, and Christian J Cyron. Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning. *Journal of Computational Physics*, 429:110010, 2021.
- [39] Kevin Linka and Ellen Kuhl. A new family of constitutive artificial neural networks towards automated model discovery. *Computer Methods in Applied Mechanics and Engineering*, 403:115731, 2023.
- [40] Mohammadamin Mahmoudabadbozchelou, Krutarth M Kamani, Simon A Rogers, and Safa Jamali. Unbiased construction of constitutive relations for soft materials from experiments via rheology-informed neural networks. *Proceedings of the National Academy of Sciences*, 121(2):e2313658121, 2024.
- [41] Melvin Mooney. A theory of large elastic deformation. *Journal of Applied Physics*, 11(9):582–592, 1940.
- [42] Stefan Müller, Tang Qi, and BS Yan. On a new class of elastic deformations not allowing for cavitation. *Annales de l’Institut Henri Poincaré C, Analyse non linéaire*, 11(2):217–243, 1994.
- [43] Jorge Nocedal and Stephen J Wright. *Numerical Optimization*. Springer, 2006.
- [44] Gregory B Olson. Designing a new material world. *Science*, 288(5468):993–998, 2000.
- [45] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.
- [46] Maziar Raissi, Alireza Yazdani, and George Em Karniadakis. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science*, 367(6481):1026–1030, 2020.
- [47] RSI Rivlin. Large elastic deformations of isotropic materials. i. fundamental concepts. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 240(822):459–490, 1948.
- [48] Yuelin Shen, K Chandrashekhara, WF Breig, and LR Oliver. Neural network based constitutive model for rubber material. *Rubber Chemistry and Technology*, 77(2):257–277, 2004.
- [49] Laurent Stainier, Adrien Leygue, and Michael Ortiz. Model-free data-driven methods in mechanics: material data identification and solvers. *Computational Mechanics*, 64(2):381–393, 2019.
- [50] Vahidullah Tac, Francisco Sahli Costabal, and Adrian B Tepole. Data-driven tissue mechanics with polyconvex neural ordinary differential equations. *Computer Methods in Applied Mechanics and Engineering*, 398:115248, 2022.
- [51] Mehdi Taghizadeh, Mohammad Amin Nabian, and Negin Alemazkoo. Multifidelity graph neural networks for efficient and accurate mesh-based partial differential equations surrogate modeling. *Computer-Aided Civil and Infrastructure Engineering*, 40(7):841–858, 2025.
- [52] Prakash Thakolkaran, Akshay Joshi, Yiwen Zheng, Moritz Flaschel, Laura De Lorenzis, and Siddhant Kumar. Nn-euclid: Deep-learning hyperelasticity without stress data. *Journal of the Mechanics and Physics of Solids*, 169:105076, 2022.
- [53] Thingiverse. <https://www.thingiverse.com/thing:379894>, 2019. Accessed: 2026-02-23. Licensed under Creative Commons.
- [54] Clifford Truesdell and Walter Noll. *The Non-Linear Field Theories of Mechanics*. Springer, 2004.
- [55] Silviu-Marian Udrescu and Max Tegmark. Ai feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631, 2020.

- [56] Nikolaos N Vlassis, Ran Ma, and WaiChing Sun. Geometric deep learning for computational mechanics part I: Anisotropic hyperelasticity. *Computer Methods in Applied Mechanics and Engineering*, 371:113299, 2020.
- [57] Sidney Yip and Michael P Short. Multiscale materials modelling at the mesoscale. *Nature Materials*, 12(9):774–777, 2013.
- [58] Linying Zhang, Stefano Pagani, Jun Zhang, and Francesco Regazzoni. Shape-informed surrogate models based on signed distance function domain encoding. *Journal of Computational Physics*, page 114178, 2025.
- [59] Quan Zhou, Peizhe Tang, Shenxiu Liu, Jinbo Pan, Qimin Yan, and Shou-Cheng Zhang. Learning atoms for materials discovery. *Proceedings of the National Academy of Sciences*, 115(28):E6411–E6417, 2018.

MOX Technical Reports, last issues

Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

- 29/2026** Regazzoni, F.
The internal law of a material can be discovered from its boundary
- 28/2026** Daniele, F.; Leimer Saglio, C. B.; Pagani, S.; Antonietti, P. F.
Mathematical and numerical modeling of coupled oxygen dynamics and neuronal electrophysiology
- 27/2026** Antonietti, P. F.; Abdalla, O. M. O.; Garroni, M. G.; Mazzieri, I.; Parolini, N.
A hybrid reduced-order and high-fidelity discontinuous Galerkin Spectral Element framework for large-scale PMUT array simulations
- 23/2026** Ballini, E.; Muscarnera, L.; Fumagalli, A.; Scotti, A.; Regazzoni, F.
Elimination-compensation pruning for fully-connected neural networks
- 25/2026** Carrara, D.; Hirschvogel, M.; Bonizzoni, F.; Pagani, S.; Pezzuto, S.; Regazzoni, F.
Shape-informed cardiac mechanics surrogates in data-scarce regimes via geometric encoding and generative augmentation
- 26/2026** Dokuchaev, A.; Bonizzoni, F.; Pagani, S.; Regazzoni, F.; Pezzuto, S.
Learning geometry-dependent lead-field operators for forward ECG modeling
- 21/2026** Bottacini, G.; Torzoni, M.; Manzoni, A.
Neural Markov chain Monte Carlo: Bayesian inversion via normalizing flows and variational autoencoders
- 20/2026** Caldera, L.; Cavinato, L.; Cirone, A.; Cama, I.; Garbarino, S.; Lodi, R.; Tagliavini, F.; Nigri, A.; De Francesco, S.; Cappozzo, A.; Piana, M.; Ieva, F.;
DISARM++: Beyond scanner-free harmonization
- 19/2026** Caldera, L.; Cavinato, L.; Ieva, F.
Scanner-agnostic MRI harmonization via SSIM-guided disentanglement
- 17/2026** Caldera, L.; Bottacini, G.; Cavinato, L.
MAGIC-Flow: multiscale adaptive conditional flows for generation and interpretable classification