



---

**Review**

## **Constitutive modeling of heterogeneous materials by interpretable neural networks: A review**

**Antonio Bilotta<sup>1,\*</sup> and Emilio Turco<sup>2</sup>**

<sup>1</sup> Department of Informatics, Modelling, Electronics and System Engineering (DIMES), University of Calabria, Via P. Bucci, Cubo 42/C, Rende (CS) 87036, Italy

<sup>2</sup> Department of Architecture, Design and Urban Planning (DADU), University of Sassari, Palazzo del Pou Salit, Piazza Duomo 6, Alghero (SS) 07041, Italy

\* **Correspondence:** Email: antonio.bilotta@unical.it; Tel: +390984496910.

**Abstract:** Is it possible to interpret the modeling decisions made by a neural network trained to simulate the constitutive behavior of simple or complex materials? The problem of the interpretability of a neural network is a crucial aspect that has been studied since the first appearance of this type of modeling tool and it is certainly not specific to applications related to constitutive modeling of heterogeneous materials. All areas of application, such as computer vision, biomedicine, and speech, suffer from this fuzziness, and for this reason, neural networks are often referred to as “black-box models”. The present work highlighted the efforts dedicated to this aspect in the constitutive modeling of the behavior of path independent materials, reviewing both more standard neural networks and those adopting, more or less strongly, the specific point of view of interpretability.

**Keywords:** constitutive modeling; hyperelasticity; data-driven; machine learning; artificial neural networks

---

### **1. Introduction**

Constitutive modeling of a complex material that exhibits a high degree of nonlinear behavior can be a challenging task. There are two main approaches to obtaining strain-stress relationships: Phenomenological modeling based on empirical observations and mechanistic models derived from first principles based on the underlying structure of the material. Both approaches are usually formulated within the widely accepted general framework of continuum mechanics [1, 2], but retain their specific features.

When working with highly heterogeneous materials that are irregular or characterized by ambiguous microstructures, the use of phenomenological models is particularly valuable and often unavoidable.

However, phenomenological models have inherent limitations because they rely on empirical relationships, i.e., models that must be selected heuristically. This approach often requires the tuning of various parameters to fit experimental or observational data. A process that easily allows the development of ready-to-use formulations, but often the correct description of the underlying physics of the material can be lost, sometimes violating essential principles or constraints. In addition, phenomenological models can be more sensitive to the quality and quantity of available data; they can lose their effectiveness if the data are sparse, noisy, or biased, with a high risk of overfitting and then a poor ability to describe the general behavior of the material.

Many of the problems that can plague phenomenological models can be avoided by using mechanistic models that are capable of providing a theoretically sound description of material behavior. These models are formulated through strict adherence to known physical principles, incorporating requirements such as objectivity, respect for material symmetries, and stability constraints [3–5]. However, despite their rigor, mechanistic models become increasingly difficult to formulate as the complexity of material structure and behavior increases. Even for materials that are not overly complex, developing a mechanistic model easily becomes an unaffordable task because of the complexity of defining a single framework that captures the full range of behavior.

The general picture quickly sketched above also becomes more challenging when we consider *metamaterials*. It is not easy to come up with a generally accepted definition of what a *metamaterial* is, because the contributions to this very new subject are numerous and constantly evolving. Then we recall the textual definition given in [6]: “*A material which has been designed to meet a specific purpose, by combining more elementary materials (characterized by a smaller micro length scale) and by shaping them with geometrical structures and mechanical interactions (what we call a microstructure) characterized by the same micro length scale*”. The term *metamaterial* is advocated in several contexts and for very different applications. For example we have pantographic lattices which are architected materials with many interesting mechanical properties, see [7–10]. In the field of deployable structures the use of origami-like structures, see [11, 12], is emerging as a very promising approach to solving this kind of problem. The optimization of the storage modulus and damping capacity of multilayer nanocomposites [13] and nanocomposite beams which show highly tunable nonlinear stiffness and damping capacity [14] represent interesting applications in the field of nanocomposites. Moreover *metamaterials* constitute the leading trend emerging in several other application fields such as electromagnetism, optics, and acoustics [15]. In all cases, however, the apparent mechanical behavior is by no means approachable through a phenomenological model and can hardly be described through a direct mechanistic formulation.

The above considerations are leading many researchers to consider new approaches that we can label as *data-driven* [16]. Among them, those based on the use of *artificial neural networks* (ANNs) [17–19] are gaining a lot of attention. ANNs are not a very recent invention – the first pioneering work dates back to the 1940s [20] but their potentialities become clear only in more recent years when the availability of powerful computational resources made it possible to exploit very deep ANN architectures in the solution of computer vision problems, see, for example, [21], thanks to their excellent nonlinear function fitting capabilities, see [22]. Thanks to this strong impulse ANNs and, more generally, machine learning approaches have found application in several engineering fields [23]. For example, in [24] convolutional neural networks were proposed to predict the damage in steel-concrete beams and in [25] variational autoencoders were trained to generate the description of rubble masonry geometries. The proposal of a

symbolic regression pipeline for constitutive law discovery, capable of promoting low complexity, the ability to systematically embed constraints stemming from domain knowledge, and with a generalization capability, can be found in [26]. In [27], a hybrid model combining the lattice Boltzmann method and various machine learning algorithms was proposed to predict the intrinsic permeability of porous media. In [28], a long short-term memory (LSTM) approach is compared to LSTM-based modeling with Monte Carlo dropout in modeling the stress-strain response of frozen soils. Again, an LSTM model combined with proper orthogonal decomposition is proposed in [29] to predict the elasto-plastic response of structures. Finally, k-nearest neighbors, multilayer perceptron, support vector regression, decision trees, random forest, and gradient boosting decision trees were compared in predicting the thermal diffusivity of soils, see [30].

In the present work the capabilities of neural networks specifically proposed to formulate general strain–stress modeling frameworks are reviewed. In particular the constitutive modeling of path independent materials is considered. The reviewed proposals are also discussed in terms of their interpretability [31] which constitutes a long-debated aspect of this kind of numerical models. The organization of the paper is as follows. Section 2 gives a quick resumé regarding the basics of the constitutive modeling of materials and of ANNs. Section 3 presents the data-driven modeling approaches here reviewed, collecting them on the basis of the following classification: interpolations; neural networks, first attempts; standard neural networks; physically constrained neural networks; and mathematically constrained neural networks. The closing Section 5 draws the conclusions of the work.

## 2. Background

### 2.1. Kinematics and constitutive equations

We introduce the deformation map  $\varphi$  as the mapping of material points  $X$  in the undeformed configuration to points  $x = \varphi(X)$  in the deformed configuration [1, 2]. The gradient of the deformation map  $\varphi$  with respect to the undeformed coordinates  $X$  defines the deformation gradient  $\mathbf{F}$  with its determinant  $J$ ,

$$\mathbf{F} = \nabla_X \varphi \quad \text{with} \quad J = \det(\mathbf{F}) > 0. \quad (2.1)$$

As deformation measures, we introduce the right and left Cauchy-Green deformation tensors:

$$\mathbf{C} = \mathbf{F}^t \mathbf{F} \quad \text{and} \quad \mathbf{b} = \mathbf{F} \mathbf{F}^t. \quad (2.2)$$

In the undeformed state, all three tensors are identical to the unit tensor,  $\mathbf{F} = \mathbf{I}$ ,  $\mathbf{C} = \mathbf{I}$ , and  $\mathbf{b} = \mathbf{I}$ , and the Jacobian is one,  $J = 1$ . A Jacobian smaller than one,  $0 < J < 1$ , denotes compression and a Jacobian larger than one,  $1 < J$ , denotes extension. In order to characterize *isotropic* materials, the three principal  $I_1$ ,  $I_2$ , and  $I_3$  invariants are introduced. They can be expressed in terms of the deformation gradient  $\mathbf{F}$ :

$$\begin{aligned} I_1 &= \mathbf{F} : \mathbf{F}, \\ I_2 &= \frac{1}{2} \left( I_1^2 - (\mathbf{F}^t \mathbf{F}) : (\mathbf{F}^t \mathbf{F}) \right), \\ I_3 &= \det(\mathbf{F}^t \mathbf{F}) = J^2, \end{aligned} \quad (2.3)$$

or, equivalently, in terms of the right or left Cauchy-Green deformation tensors:

$$\begin{aligned} I_1 &= \text{tr}(\mathbf{C}) = \mathbf{C} : \mathbf{I}, & I_1 &= \text{tr}(\mathbf{b}) = \mathbf{b} : \mathbf{I}, \\ I_2 &= \frac{1}{2} (I_1^2 - \mathbf{C} : \mathbf{C}), & I_2 &= \frac{1}{2} (I_1^2 - \mathbf{b} : \mathbf{b}), \\ I_3 &= \det(\mathbf{C}) = J^2, & I_3 &= \det(\mathbf{b}) = J^2. \end{aligned} \quad (2.4)$$

It is possible to extend the use of the invariants also to materials more complex than the isotropic materials by introducing the *generalized invariants* as shown in [5]. Finally from right or left Cauchy-Green deformation tensors the Green-Lagrange or the Almansi strain measure, respectively, can be defined as

$$\mathbf{E} = \frac{1}{2} (\mathbf{C} - \mathbf{I}), \quad \mathbf{e} = \frac{1}{2} (\mathbf{I} - \mathbf{b}^{-1}). \quad (2.5)$$

Materials whose constitutive behavior depends solely on the current state of the deformation are typically classified as elastic. In these cases, any stress on a particle  $X$  is determined by the current deformation gradient  $\mathbf{F}$  associated with that point. When  $\mathbf{E}$  is used as the strain measure, its conjugate stress measure,  $\mathbf{S}$  (the second Piola-Kirchhoff stress tensor), will be used to define the fundamental material relations. Consequently, elasticity can be generally expressed as

$$\mathbf{S} = \mathbf{G}(\mathbf{E}(X), X), \quad (2.6)$$

where the direct dependency upon  $X$  allows for the possible inhomogeneity of the material. When the work done by stresses during a deformation process depends solely on the initial state and the final configuration, the material behavior is described as path-independent, and the material is referred to as *hyperelastic*. In this case a stored *energy density function* or *elastic potential*

$$w \equiv w(\mathbf{E}(X), X) \quad \text{or} \quad w \equiv w(\mathbf{C}(X), X), \quad (2.7)$$

exists such that

$$\mathbf{S} = \frac{\partial w}{\partial \mathbf{E}} = 2 \frac{\partial w}{\partial \mathbf{C}}. \quad (2.8)$$

The latter equation is often used as the definition of a hyperelastic material.

### 2.1.1. Physical constraints and other desirable properties

Definition (2.7) automatically ensures *material objectivity* or *frame indifference*, meaning that that constitutive laws do not depend on the external frame. Meanwhile, condition (2.8) ensures *thermodynamic consistency* (in absence of dissipating phenomena), guaranteeing that the stress  $\mathbf{S}$  inherently complies with the second law of thermodynamics.

Another recurrent physical constraint is material symmetry which implies that the material response does not change if the reference configuration is rotated. This is a well-known characteristic of *isotropic* material, allowing the strain energy density function to be expressed in terms of invariants:

$$w \equiv w(I_1, I_2, I_3). \quad (2.9)$$

Additional physically reasonable constraints include the non-negativity of the energy density function,  $w$ , for all non-zero deformation states. Moreover,  $w$  approaches zero in the reference configuration and tends toward infinity under extreme conditions of infinite compression and expansion.

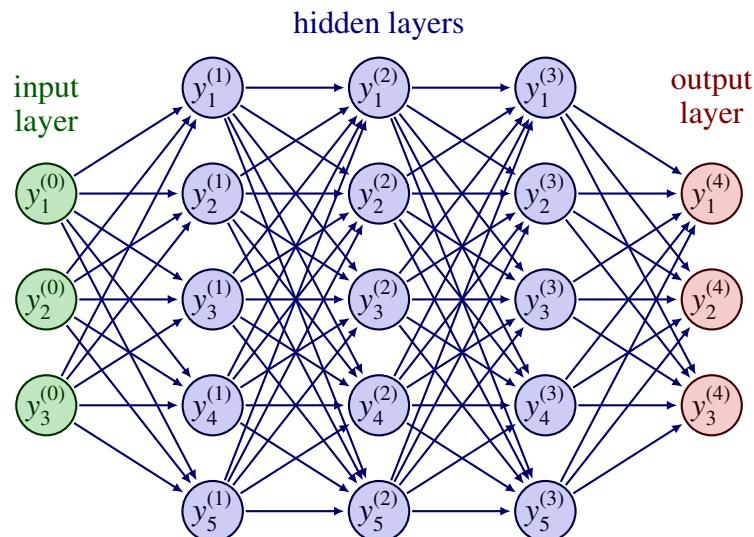
Material stability or *ellipticity* [3], i.e., the condition which ensures that only real wave speeds are permissible in the material and, more importantly, the well-posedness of the underlying boundary value problem, deserves the last comment. For strain energy density functions that can be differentiated twice, the concept of ellipticity is equivalent to convexity in the directions related to rank-one tensors, which is often called rank-one convexity. However, establishing that a model is elliptic from the very start can be quite difficult in practice. As a result, many researchers in continuum mechanics choose to apply a stronger mathematical condition, *polyconvexity*, that inherently guarantees ellipticity [4].

## 2.2. Artificial neural networks

The scope of the review is to explore how nonlinear elastic constitutive laws has been modeled on the basis of mappings extracted from silicon-based or experimentally-based data. Such mappings are formulated by assuming the existence of a one-to-one correspondence between states of strain and states of stress, a relationship that can be expressed as

$$S = \mathbf{G}(E), \quad (2.10)$$

and approximating  $\mathbf{G}$  in some way. In particular we will consider mainly the case in which the approximator is a neural network  $\text{NN}_\theta$  trained on the basis of the available data.  $\theta$  denotes the set of trainable parameters defined in the following. Figure 1 shows a generic architecture of fully connected neural networks whose main components are the *neurons*, shown as circles, representing the basic information unit. Neurons are also grouped to form a sequence of *layers* connected by several arrows representing the *weights*, as will be better explained in what follows.



**Figure 1.** Fully connected neural network architecture. Assuming three strain components as input and their corresponding three stress components as output, the network can be considered a basic example of a constitutive model.

### 2.2.1. Layers, weights, biases, and neurons

Neural networks are versatile function approximators that are capable of learning any nonlinear function [22]. This result is obtained by applying a series of successive transformations to the input data which constitute the *input layer* of the neural network. The result obtained by means of each transformation constitutes a *hidden layer* of data or the *output layer* depending on the architecture chosen for the neural networks. The neural network's architecture, i.e., the number of layers and their lengths, determine the number of transformations that are applied to the input data and how its size shrinks or grows going through each transformation. By assuming only two hidden layers, beyond the input and output layer, the most basic sequence of cascading transformations associated to a neural network can be formulated as follows:

$$\begin{aligned} \mathbf{y}^{(0)} &= \mathbf{E}, \\ \mathbf{y}^{(1)} &= \mathbf{W}^{(1)} \mathbf{y}^{(0)} + \mathbf{b}^{(1)}, \\ \mathbf{y}^{(2)} &= \mathbf{W}^{(2)} \mathbf{y}^{(1)} + \mathbf{b}^{(2)}, \\ \mathbf{S} &= \mathbf{W}^{(3)} \mathbf{y}^{(2)} + \mathbf{b}^{(3)}, \end{aligned} \quad (2.11)$$

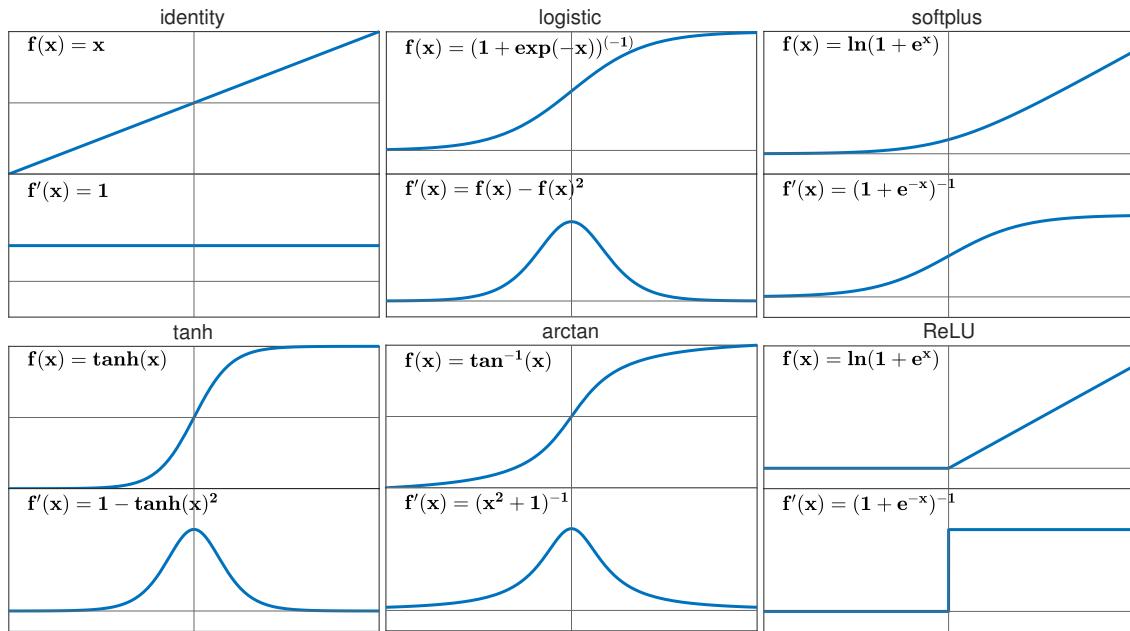
where the matrix  $\mathbf{W}^{(l)}$  relative to the  $l$ -th layer contains its weights and  $\mathbf{b}^{(l)}$  is its bias vector. The input layer has a number of *neurons* equal to the size of  $\mathbf{E}$ , and the latter determines also the number of columns of  $\mathbf{W}^{(1)}$ . On the contrary the number of rows of the matrices  $\mathbf{W}^{(1)}$  and  $\mathbf{W}^{(2)}$  can be freely chosen, determining in this way the number of neurons of the two hidden layers. The output layer has a number of neurons equal to the size of  $\mathbf{S}$ . All the coefficients of the weights' matrices and bias vectors  $\mathbf{W}^{(l)}$  and  $\mathbf{b}^{(l)}$  constitute  $\boldsymbol{\theta}$ , i.e., the set of trainable parameters.

### 2.2.2. Activation functions

The preceding sequence of transformations, see Eq (2.11), is not absolutely capable of defining a generic nonlinear function. Suitably defined *activation functions* are applied to the results of each transformation before feeding them to the next layer. Similarly to the brain that processes input signals and decides whether a neuron should fire or not, activation functions decide whether the nodal input is important or not in the process of approximating the final function. On this basis sequence (2.11) can be written as

$$\begin{aligned} \mathbf{y}^{(0)} &= \mathbf{E}, \\ \mathbf{y}^{(1)} &= f^{(1)}(\mathbf{W}^{(1)} \mathbf{y}^{(0)} + \mathbf{b}^{(1)}), \\ \mathbf{y}^{(2)} &= f^{(2)}(\mathbf{W}^{(2)} \mathbf{y}^{(1)} + \mathbf{b}^{(2)}), \\ \mathbf{S} &= f^{(3)}(\mathbf{W}^{(3)} \mathbf{y}^{(2)} + \mathbf{b}^{(3)}), \end{aligned} \quad (2.12)$$

where  $f^{(l)}$  is the activation function of the  $l$ -th layer. Some of the most commonly used activation functions are shown in Figure 2.



**Figure 2.** Popular activation functions along with their derivatives include the identity, logistic or soft step, soft plus, hyperbolic tangent, inverse tangent, rectified linear unit or ReLU functions. Activation functions can be continuous or discontinuous, linear or nonlinear, and bounded or unbounded.

### 2.2.3. Neural network function approximator

In order to make evident the function approximator produced by the sequence described in Eq (2.12), it can be condensed as follows:

$$\begin{aligned}
 S &= f^{(3)} \left( \mathbf{W}^{(3)} \mathbf{y}^{(2)} + \mathbf{b}^{(3)} \right) \\
 &= f^{(3)} \left( \mathbf{W}^{(3)} \left( f^{(2)} \left( \mathbf{W}^{(2)} \mathbf{y}^{(1)} + \mathbf{b}^{(2)} \right) \right) + \mathbf{b}^{(3)} \right) \\
 &= f^{(3)} \left( \mathbf{W}^{(3)} \left( f^{(2)} \left( \mathbf{W}^{(2)} \left( f^{(1)} \left( \mathbf{W}^{(1)} \mathbf{y}^{(0)} + \mathbf{b}^{(1)} \right) \right) + \mathbf{b}^{(2)} \right) \right) + \mathbf{b}^{(3)} \right) \\
 &= f^{(3)} \left( \mathbf{W}^{(3)} \left( f^{(2)} \left( \mathbf{W}^{(2)} \left( f^{(1)} \left( \mathbf{W}^{(1)} \mathbf{E} + \mathbf{b}^{(1)} \right) \right) + \mathbf{b}^{(2)} \right) \right) + \mathbf{b}^{(3)} \right) \\
 &= \text{NN}_\theta(\mathbf{E}).
 \end{aligned} \tag{2.13}$$

### 2.2.4. Loss function and neural network's training

In Eq (2.13) the unknown parameters are the weights  $\mathbf{W}^{(l)}$  and the biases  $\mathbf{b}^{(l)}$ , collectively denoted by the symbol  $\theta$ . Assuming that a set of stress-strain data pairs  $(\widehat{\mathbf{E}}^{(m)}, \widehat{\mathbf{S}}^{(m)})$ , with  $m = 1 \dots N$ , is available, the unknown parameters  $\theta$  are determined by introducing the *loss function*

$$\mathcal{L}_\theta \equiv \sum_{m=1}^N \left\| \text{NN}_\theta(\widehat{\mathbf{E}}^{(m)}) - \widehat{\mathbf{S}}^{(m)} \right\|^2, \tag{2.14}$$

which measures the distance between predicted and true values.

---

**Remark 1.** *The definition of the loss function given in Eq (2.14) is not the only one possible. As it will be shown in the following, it can be changed on the basis of specific design choices or it can be amended and/or enriched through the imposition of additional constraints.*

The set of parameters  $\theta$  are determined by imposing the minimization of the loss function, i.e.,

$$\theta = \underset{\beta \in \mathbb{R}^{N_\theta}}{\operatorname{argmin}} \mathcal{L}_\beta. \quad (2.15)$$

Equation (2.15) defines an optimization problem whose solution, i.e., the evaluation of all the coefficients defining the weight matrices  $W^{(l)}$  and the bias vectors  $b^{(l)}$ , constitutes the training of the neural network. Several training algorithms are available in any deep learning library, see, for example, [32, 33]; choosing one of them is an important step that is usually improved by experimenting with more training algorithms.

### 3. Data-driven modeling patterns

This section is dedicated to a certainly non-exhaustive presentation of data-driven constitutive models proposed in recent papers for the case of path-independent materials. The contributions are ordered according to an ascending level of interpretability, which researchers often have instilled in their models by imposing physical or mathematical constraints. In addition to these types of proposals, the papers reviewed also include models based on the more familiar interpolation techniques, standard neural networks, and some of the pioneering works using neural networks.

Table 1 gives a summary of the formulations considered in the present work, reporting for each formulation: the input and output variables; the context in which the formulation is applied; the type of model; and the level of interpretability that, in our opinion, can be attributed to the formulation. With respect to this latter aspect, the attention is focused on the three fundamental ingredients reported in Table 1: Input, Output, and Model. A 0 level of interpretability, labeled as absent, is assigned to those approaches that use the strain components as input, use the stress components as output, and define the model's architecture by using the standard patterns available in deep learning libraries as quickly described in Section 2.2, see also Figure 1 for a typical representation of this kind of network. Those models that have a different strategy for 1, 2, or 3 of these basic ingredients were assigned a level of interpretability equal to 1, labeled low, 2, labeled medium, or 3, labeled high. When this happens, the reviewed paper typically makes specific choices in order to improve the physical soundness of the proposed formulations and/or to impose specific mathematical constraints.

#### 3.1. Interpolations

In reference [34] in order to efficiently evaluate the mechanical response of nonlinearly elastic heterogeneous materials, the effective strain-energy potential is evaluated by finite element analysis of a given representative volume element (RVE) at various points within an assigned macroscopic strain space. By interpolating these discrete results, the effective strain-energy potential for the entire macroscopic strain space can be constructed. The effective stress-strain relation and tangent tensor are then derived in a direct and explicit manner. In particular two interpolation techniques are proposed. One utilizes multidimensional cubic spline interpolation, allowing for easy determination of coefficients in 2D problems without significant computational demands. However, for 3D problems, this process

requires considerable numerical resources. The other one uses an outer product decomposition of the hypermatrix, addressing the limitations of the previous approach. Both methods produce very similar accuracy. In both cases the evaluation of the mechanical response of the RVE by finite element analysis is the most computationally intensive part of the proposed procedure.

The approach proposed in [35] is similar to the previous one, but with an additional step before the interpolation construction. It is based on Kriging, which: (1) sets up a structure for the input design variables, and (2) interpolates the response obtained for each sample of the input design variables. Nonparametric regression methods, such as polynomial interpolation and spline interpolation, do not include the first step, resulting in a loss of accuracy compared to Kriging. In fact, it can be shown that spline interpolation is equivalent to Kriging with fixed covariance and a polynomial degree of tendency.

In [36] a construction of the constitutive manifold directly from the available data is proposed. In particular, assuming to observe an assigned structure subjected to  $N$  randomly applied external loads, the  $N$  pairs of strain and stress are collected in a single vector forming a generic point of a space with dimension equal to  $D$ . It is assumed that all these  $N$  points belong to a certain low-dimensional manifold embedded in the high-dimensional space, and a manifold learning (or nonlinear dimensionality reduction) technique is applied. In particular, the local linear embedding (LLE) technique is employed, a method that proceeds in two steps: (1) each point is linearly interpolated from its nearest neighbors; (2) each previously generated linear patch is mapped onto a lower dimensional embedding space of dimension  $d \ll D$ . For example, in linear elastic behavior, the application of the technique just described results, as expected, in a flat manifold of dimension two, i.e., a linear behavior depending on Young's modulus and Poisson's ratio.

### 3.2. Neural networks, first attempts

The first attempt to model the mechanical response of materials by using neural networks can be traced back to the 1990s with a paper by Ghaboussi and Garret [37]. In the cited paper a path-dependent material behavior is the subject of the work but we decided to cite it in the present review dedicated to path-independent material behaviors, because [37] is the first paper about the use of a neural network for modeling the mechanical behavior of materials. In particular the neural network is trained, using experimental results, in order to represent the biaxial behavior of plain concrete. The input layer is fed with the current values of the stress and strain state plus the applied increments of stress (load-controlled case) or strain (displacement-controlled case). The output layer is designed in order to give the corresponding strain (load-controlled case) or stress (displacement-controlled case) increments. Two hidden layers, both with 40 neurons activated by the sigmoid function, complete the architecture of the neural network. The described architecture is used in the case of proportional loading tests. In the case of cyclic loading the input layer is enriched with stress and strain states relative to the previous two experimental steps. This design choice is noteworthy because it is far ahead of its time, considering that the concept of a recurrent neural network (RNN) was brought up in the 1980s and the famous long short-term memory (LSTM) network architecture [38] was proposed in 1997.

Also one of the first attempts to introduce the adoption of a constitutive neural network in a finite element code is due to Ghaboussi et al. [39]. In the cited work the neural network is used to update the stress state given the current stress-strain state and strain increment, and to calculate the material stiffness matrix needed to correct the nodal displacement relative to nodes of the finite element mesh. This is an important ingredient to assure an efficient convergence of the Newton iterations. In particular

the paper assumes a layered network architecture, specifically the nested adaptive neural network (NANN) proposed in [40], which is a variation of the standard multi-layer neural network and is characterized by design choices that remember the architecture of LSTM networks, see [38]. The finite element analyses presented are in regard to a beam bending problem and a deep excavation problem.

The work presented in [41] largely anticipated, in an embryonal form, what happened with constitutive artificial neural networks (CANNs) almost 20 years after, see [42]. The aim of the proposal was to solve the problems posed by phenomenological approaches which are biased by an arbitrary selection of the strain energy form and a subsequent fitting process for materials that hardly follow the relationship dictated by the chosen strain energy function. The proposed solution is a simple neural network with one hidden layer. The input layer receives the first and second strain invariants and the volume change ratio, and the output layer provides the corresponding strain energy function. The sigmoid function is used as the activation function which, together with the simple architecture of the network, allows an easy evaluation of the derivatives used to calculate the stress. Hyperparameter tuning is performed in order to select the best fitting model. The context of the work is the modeling of rubber and the data used to train the networks are experimental test data.

Based on our knowledge, we can assume that after the works of the 1990s and early 2000s (most likely the ones already cited and the references therein), the interest in using neural networks almost disappeared, only to regain attractiveness when neural networks and, in general, machine learning received a strong impulse in recent years [17]. This new impetus also led to a very large production of work on what we can call constitutive neural networks. This effort has produced a lot of proposals, even if we limit the investigation to materials that have a path-independent mechanical response. It is almost impossible to try to consider all the works proposed on the subject, but we will present a selection of some works and, for each proposal, we will highlight the features capable of making the model more interpretable.

### 3.3. Standard neural networks

A decoupled computational homogenization method for nonlinear elastic materials was proposed by Yvonnet and He, et al. in [43]. The aim was to make more efficient the  $FE^2$  analysis by substituting the FEM computations at the RVE level with a neural network trained for representing the effective strain energy density function parameterized by the macroscopic strains and some microstructural parameters. The architecture of the neural networks tested in the numerical experimentation was based on a variable number of layers, no more than 6, and of neurons per layer, no more than 6. As an activation function the sigmoid function was adopted. This showed how the proposed models are capable of describing the effective strain energy density in spaces of order 10 on the basis of a set of points also randomly distributed. This constitutes an important improvement with respect to models based on interpolation as those proposed by the same author in [34].

In the work proposed by Bessa et al., see [35], the main focus was a rational definition of the design of experiments starting from the selection of input variables, for example those describing material geometry (microstructure), phase properties, and external conditions and their sampling through carefully defined procedures aiming to extract the fundamental information and to avoid the *curse of dimensionality*, i.e., the exponential increase of sampling points. In particular, the Sobol sequence and variants of Latin hypercube sampling were tested. The successive step is the high-fidelity analysis of the RVE, i.e., the finite element analysis of the boundary value problem defined on the

assigned RVE. The data collected as previously described are then used to train a neural network giving in the output layer a strain energy density function or the stress components.

A neural network used in combination with proper orthogonal decomposition (POD) was proposed in [44] to improve the efficiency of numerical analysis of solids by the  $FE^2$  approach. For this purpose, the offline evaluation phase was used to train a neural network to predict the micro stress field over the RVE at each macroscopic integration point. In particular, the input of the trained network is constituted by the macro strain components and the output, in order to reduce the computational cost, are the coefficients of the POD representation of the micro stress field of the RVE domain. This strategy simplifies the offline computation of the data-driven  $FE^2$ , making the multiscale simulation of heterogeneous materials more affordable.

Always within the data-driven  $FE^2$  analysis of hyperelastic solids, [45] proposed an approach which uses a deep neural network to represent the constitutive behavior of the material through a standard map between strain and stress components. However the main novelty is the adaptive sampling points without prior knowledge of the specific mechanical problem. The data augmentation strategy updates the sampling points gradually using the distance minimization algorithm with mechanistic constraints, including the equilibrium and compatibility equations.

In [46] a neural network was used to model the hyperelastic response of PS polymers. The data used to train the network were obtained by massive molecular dynamics (MD) simulations considering monotonic and combined loading data for PS polymers. The massive data set from the MD simulations was divided into means and their standard deviations, and then used separately to train two NNs that receive strain components as input and provide stress components as output. These models were embedded in a nonlinear finite element analysis and finally used to predict the stochastic finite element model (FEM) results.

### 3.4. Physically constrained neural networks

The physically informed neural networks (PINNs) approach aims to integrate machine learning with fundamental physics [47]. Unlike traditional neural networks, which rely solely on large amounts of labeled data, PINNs embed known physical laws, often expressed as partial differential equations (PDEs), directly into their structure. This allows them to incorporate prior knowledge about the system being modeled, ensuring that the results are not only data-driven, but also physically meaningful. The amount of data required for training is generally less than for traditional neural networks. This makes PINNs suitable for problems where data is sparse or noisy. Another notable feature of PINNs is their ability to tackle inverse problems. In the following, we review some proposals that are clearly inspired by this way of thinking.

The neural network proposed in [48] aimed to simulate the constitutive behavior of isotropic hyperelastic solids. The required set of data was generated numerically by performing several finite element simulations on samples constituted by an Ogden type material and collecting strain-stress data pairs at the quadrature points. However in order to reduce the dimensionality of the data space, three deformation type invariants were used in the input layer and three stress coefficients were extracted from the output layer of the neural network. Moreover in order to ensure thermodynamic consistency, the previously trained network was modified by constructing a pseudo-potential used to correct the network's weights in order to satisfy Eq (2.8). Recently, in [49], the same authors also proposed an approach for the multiscale modeling of anisotropic solids under finite strain elasticity, by using

---

physics-augmented neural networks (PANNs).

The neural network proposed in [50] was strictly inspired by PINNs approaches. The two fundamental laws of thermodynamics are encoded in the architecture of the neural networks, hence the name thermodynamics-based artificial neural networks (TANNs). This ensures a model that is a priori thermodynamically consistent and then easier to train due to smaller data set requirements. The neural networks are formulated to model plastic materials, but are also suitable for hyperelastic materials. The input layer requires the strain increment and the previous material state identified by stress, temperature, and internal state variables. A last additional input is the time increment. The output layer provides the internal variables increment, temperature increment, and the energy potential at the time step. By differentiating the latter with respect to the inputs, it is possible to obtain the stress increment and the dissipation rate. An important component is represented by the adopted activation functions, which are chosen in such a way as to avoid the problem of both first- and second-order vanishing gradients [17]. The architecture of the network is based on three different sub-ANNs: one for predicting the internal variable increment, another one for predicting the temperature increment, and the third one for predicting the Helmholtz free energy.

### 3.5. Mathematically constrained neural networks

This section considers proposals that introduce constraints directly into the mathematical architecture of the neural network. This is a recent approach that has been introduced to correctly describe some expected or desirable features of the constitutive model used to describe the material behavior. The requirements considered are: To ensure material objectivity; to guarantee the satisfaction of the second law of thermodynamics or, equivalently, the convexity of the strain energy function; to respect material symmetries; and material stability and other physically reasonable constraints. All these aspects allow us to obtain a well-posed constitutive model and to make the resulting ANN description of the material behavior more interpretable.

The scope of the proposal presented in [51] was the definition of the constitutive neural network to be used as a surrogate in nonlinear computational homogenization frameworks. The data chosen for training the neural networks were the strain-stress data pairs generated by analyzing a triangular coupon whose macro-level is described by a single, geometrically nonlinear, triangular membrane element with only one integration point. At the micro-level, the representative volume element (RVE) with its finite element mesh describes the behavior of a material unit of woven fabric. Three kinds of neural networks were tested and compared: a standard NN, and hyperelastic NN, and a convex hyperelastic NN. The standard NN used as the loss function the same loss function defined in Eq (2.14). In the case of the hyperelastic NN the loss function was defined with respect to the distance between the true values of the stress and the gradient of the neural network, the latter one being computed by exploiting reverse mode automatic differentiation. The convex hyperelastic NN is characterized by additional features. In particular, all weights, except those connecting directly to the input, are enforced to be non-negative. This condition is obtained by defining the weights by non-negative functions of the real trainable parameters of the network. Moreover the chosen activation functions are convex and non-decreasing and, in order to perform all derivations required, are at least twice differentiable. These features are obtained by using the squared version of Softplus as an activation function. Finally the experimentation performed highlighted the absence of spurious material instability in the system when the third kind of neural network was used.

In [52, 53] the design criterion assumed in the formulation of the neural networks was to obtain constitutive models capable of capturing the effective response of complex *metamaterials* but also complying with suitable mathematical requirements. To this end the first design choice was to use invariants to feed the input layer of the neural networks, allowing them to automatically satisfy objectivity and material symmetries. The output layer was used to describe the elastic potential from which was derived the approximated stress field, satisfying thermodynamic consistency. Finally a different kinds of neural networks were used that could be considered input convex feed-forward neural networks (ICNN) [54] by construction. In particular the architecture of ICNNs imposes convexity by using the Softplus activation function in the input layer and in the hidden layers, where non-negative weights were also adopted.

A new neural network architecture, called the Cholesky-factored symmetric positive definite neural network (SPD-NN), was presented in [55]. The main feature was that the output layer of the network was trained to predict the Cholesky factor of the tangent stiffness matrix instead of directly predicting the stress components. Then the Cholesky factor was used to calculate the stress increments relative to the current step of the nonlinear analysis. This made it possible to weakly impose convexity on the strain energy function, which improved numerical stability in finite element simulations. Two types of training strategies were tested. The first strategy, called the direct training method, trains the neural networks using the usual strain-stress pairs, and the use of strain increments was also considered. The second strategy uses indirect full-field data, such as displacement and external load data, by coupling the neural network with a dynamic structural equation solver, in which case the demanding requirement for strain-stress data is relaxed. The approach was tested not only for hyperelastic materials but also for elasto-plastic applications.

A new neural network architecture, called constitutive artificial neural networks (CANNs), was proposed in [42]. The formulation was based on the generalized invariant theory which allows it to express strain energy functions also for generic anisotropic materials [5]. Thanks to this kind of approach, the information about the material anisotropy is captured by the generalized structure tensors entering in the evaluation of the generalized invariants, for more details, see [42], which become the main information used to feed the network architecture. To this end a first step is defined during which the basic input information, i.e., the kinematic measure of strain  $\mathbf{C}$  and the optional feature vector describing the material anisotropy, are first transformed into generalized invariants. In the second step, instead of using a single neural network the different invariants are individually mapped to sub-nets, in order to promote the physical interpretability of the network and to quantify the precise role of the different invariants. The sub-nets' outputs are then combined in the output layer which is responsible for the evaluation of the strain energy function.

The neural network proposed in [56] can be considered an enrichment of the CANNs discussed in the previous paragraph. The approach is based on the preliminary definition of all desirable properties for constitutive relations in terms of kinematic, thermodynamic, and physical constraints. These constraints are then used to design neural networks capable of satisfying all of them. In particular, the architecture of the neural networks is defined as follows. To satisfy objectivity and material symmetries, the input layer is designed to accept strain invariants in a number chosen on the basis of the material to be modeled, and the possible presence of the incompressibility constraint is also dealt with at this level. The output layer is used to describe the free energy function from which the stress components are derived, which allows it to satisfy the thermodynamic consistency. The activation functions are chosen

to satisfy physically reasonable constraints, see Section 2.1.1, and convexity. To this end, powers and squared powers are used in the first layer, and identity and exponential functions are applied to these powers in the second layer. Finally, the architecture of the network is not fully connected a priori to satisfy the condition of polyconvexity.

The special features of ICNNs, i.e., material stability, material objectivity, and stress-free reference configuration, were exploited in [57] in order to represent generic, isotropic or anisotropic, and hyperelastic constitutive behaviors. However, unlike the previously discussed approaches, it takes as input only experimentally measurable data in the form of full-field displacements, such as can be obtained from digital image correlation (DIC) techniques, and global force data provided by mechanical testing machines. The approach is called NN-EUCLID (neural network and efficient unsupervised constitutive law identification and discovery [58, 59]) which, in the absence of energy density/stress labels, uses the conservation of linear momentum to guide the estimation of the learnable parameters of the ICNN, i.e., the minimization problem is based on a loss function weighting the force balance residuals. Several benchmarks for isotropic and anisotropic hyperelasticity have been used to validate the proposed NN-EUCLID framework, which is capable of identifying the underlying material behavior from the data of a single experiment.

In [60] the TBNN (tensor basis neural network) approach [61] was considered. This approach simplifies the maps that need to be discovered thanks to the direct adoption of the tensor basis elements in the network architecture. This constitutes a significant part of the functional complexity of the representation that does not need to be learned, as happens for more traditional NN-based approaches that learn component-based input-output maps. The proposed TBNN methodology can provide a surrogate for the constitutive response but also discover the type and the orientation of the symmetry of an anisotropic material. This kind of result is obtained thanks to the adopted TBNN architecture: the input layer includes the stretch tensor and a set of structure tensors spanning the possible symmetry groups, which are used to compute the isotropic invariants and the anisotropic invariants; a deep densely connected, feed-forward neural network connects the input layer with the output layer; and the output layer provides a representation of the elastic potential through the isotropic and anisotropic tensor basis coefficients. The set of learnable parameters consists of the weights, the biases, and six additional parameters, 4 of which are used to describe the orientation of the material anisotropy and 2 to control its degree. Finally it is worth mentioning that this approach has been further extended in [62] to tackle thermo-hyperelasticity on the basis of a polyconvex network.

**Table 1.** Main features of the reviewed models and interpretability level, **I**: absent ; low ; medium ; high .

Input	Output	Context	Model	I
strain [34] components	strain potential	FE <sup>2</sup> analysis of heterogeneous materials	spline and hypermatrix interpolation	

*Continued on next page*

Input	Output	Context	Model	I
[35] strain components	stress components or strain potential	FE <sup>2</sup> analysis of heterogeneous materials	Kriging	●
[36] strain and stress components	constitutive manifold	constitutive equations	local linear embedding	●
[37] strain or stress states (or increments)	strain or stress increments	biaxial behavior of plain concrete	NN	●
[41] strain invariants	strain energy	rubbers' constitutive equations	NN	●
[39] strain or stress states (or increments)	strain or stress increments and FE stiffness matrix	FEM analysis of beam bending and deep excavation	NN	●
[43] strain components	strain potential	FE <sup>2</sup> analysis of heterogeneous materials	NN	●
[35] strain components	strain potential	FE <sup>2</sup> analysis of heterogeneous materials	NN	●
[44] macro-strain components	POD coefficients of the micro stress field	FE <sup>2</sup> analysis of heterogeneous materials	NN	●
[45] macro strain components	macro stress components	FE <sup>2</sup> analysis of heterogeneous materials	NN	●
[46] strain components	stress components	stochastic FE analysis of hyperelastic materials	NN	●
[48] strain invariants	stress coefficients	constitutive modeling of hyperelastic materials	NN plus pseudo-potential	●

*Continued on next page*

	<b>Input</b>	<b>Output</b>	<b>Context</b>	<b>Model</b>	<b>I</b>
[50]	strain increment and previous material state	new material state	constitutive equations	TANN	●
[51]	strain components	gradient of the neural network	FE <sup>2</sup> analysis of heterogeneous materials	constrained NN	●
[52]	strain invariants	elastic potential	RVE analysis of heterogeneous materials and coupled problems	ICNN	●
[55]	strain components	Cholesky factor of the tangent stiffness matrix	FEM analysis of hyperelastic materials	NN	●
[42]	generalized invariants	free energy function	constitutive equations	CANN	●
[56]	strain invariants	free energy function	constitutive equations	polyconvex CANN	●
[57]	observed displacement and reaction force data	elastic potential	constitutive equations	ICNN plus EUCLID	●
[60]	strain invariants	stress components derived from elastic potential	constitutive equations	TBNN	●

#### 4. Discussion

For the models discussed here, the interpretability of the ANN model is enforced by imposing physical or mathematical constraints, both of which allow the obtained models to exhibit the physical soundness of mechanistic models. These modeling choices, which affect one or more of the three fundamental ingredients of the neural network, i.e., input layer, output layer, and the architecture of the hidden layers, not only allow a greater control over the formulation of the model, but also provide several advantages that are clearly described in the reviewed papers and are summarized here as follows.

---

- The training of the model can be done even if the amount of available data is small. This condition can easily occur when the data to be used comes from laboratory experiments or in-situ measurements. In contrast, standard neural network models usually require a large amount of data.
- The ability to describe a general behavior is improved, allowing the underlying physics of the material to be captured without violating essential principles or constraints. This behavior is generally not expected from generic ANN models, which tend to strictly reproduce only the data on which they are trained.
- The models are less sensitive to the quality and quantity of available data and do not lose effectiveness when the data is sparse, noisy, or biased. The risk of overfitting is very small.
- The ability to extrapolate, i.e., when the model is exposed to input that it has never seen before, is able to provide an output that maintains physical soundness. Also this feature is not generally expected by generic ANN models.

The above considerations underline the importance of using more interpretable constitutive neural networks, not only to remove the “black box” label often attached to this type of model, but also to have a very important positive effect on the cost of building the model, in terms of the amount of data required, and on its predictive capabilities.

## 5. Conclusions

The modeling of the path independent response of heterogeneous materials has been discussed by reviewing some of the more recent proposals based on the use of artificial neural networks. We tried to highlight those aspects of the formulations that, in our opinion, allowing for improvement of the interpretability of the models, and then to remove the “black-box” label often used with neural network-based models. This result is obtained by adopting specific choices for three important ingredients of the neural network: the input layer, the output layer, and the architecture of the hidden layers. Modifying one or more of these ingredients allows for a higher level of interpretability and, as a consequence, an improvement of the effectiveness of the resulting model.

### Authors contribution

Both authors contributed equally to the literature review, discussion, writing, and revision of the paper.

### Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

### Acknowledgments

Emilio Turco gratefully acknowledges the support of the University of Sassari (Fondo di Ateneo per la Ricerca 2020). The research reported in the present contribution was carried out as part of the project “Metamaterials design and synthesis with applications to infrastructure engineering” funded by the MUR Progetti di Ricerca di Rilevante Interesse Nazionale (PRIN) Bando 2022-grant 20228CPHN5.

## Conflict of interest

The authors declare there are no conflicts of interest.

## References

1. J. Bonet, A. J. Gil, R. D. Wood, Nonlinear solid mechanics for finite element analysis: Statics. Cambridge University Press, 2016. <https://doi.org/10.1017/CBO9781316336144>
2. G. A. Holzapfel, *Nonlinear Solid Mechanics*, John Wiley & Sons Ltd, 2000.
3. J. K. Knowles, E. Sternberg, On the ellipticity of the equations of nonlinear elastostatics for a special material, *J. Elasticity*, **5** (1975), 341–361. <https://doi.org/10.1007/BF00126996>
4. J. Bonet, A. J. Gil, R. Ortigosa, A computational framework for polyconvex large strain elasticity, *Comput. Methods Appl. Mech. Engrg.*, **283** (2015), 1061–1094. <https://doi.org/10.1016/j.cma.2014.10.002>
5. M. Itskov, Tensor algebra and tensor analysis for engineers, in *Mathematical Engineering*, Springer Cham, (2019). <https://doi.org/10.1007/978-3-319-98806-1>
6. F. dell’Isola, D. Steigmann, A. Della Corte, E. Barchiesi, M. Laudato, F. Di Cosmo, et al., *Discrete and Continuum Models for Complex Metamaterials*, Cambridge University Press, 2020. <https://doi.org/10.1017/9781316104262>
7. E. Turco, A. Misra, M. Pawlikowski, F. dell’Isola, F. Hild, Enhanced Piola-Hencky discrete models for pantographic sheets with pivots without deformation energy: Numerics and experiments, *Int. J. Solids Struct.*, **147** (2018), 94–109. <https://doi.org/10.1016/j.ijsolstr.2018.05.015>
8. E. Turco, Stepwise analysis of pantographic beams subjected to impulsive loads, *Math. Mech. Solids*, **26** (2021), 62–79. <https://doi.org/10.1177/1081286520938841>
9. F. dell’Isola, P. Seppecher, M. Spagnuolo, E. Barchiesi, F. Hild, T. Lekszycki, et al., Advances in pantographic structures: Design, manufacturing, models, experiments and image analyses, *Continuum Mech. Thermodyn.*, **31** (2019), 1231–1282. <https://doi.org/10.1007/s00161-019-00806-x>
10. F. dell’Isola, P. Seppecher, J. J. Alibert, T. Lekszycki, R. Grygoruk, M. Pawlikowski, et al., Pantographic metamaterials: An example of mathematically driven design and of its technological challenges, *Continuum Mech. Thermodyn.*, **31** (2019), 851–884. <https://doi.org/10.1007/s00161-018-0689-8>
11. E. Turco, E. Barchiesi, A. Causin, F. dell’Isola, M. Solci, Kresling tube metamaterial exhibits extreme large-displacement buckling behavior, *Mech. Res. Commun.*, **134** (2023), 1–7. <https://doi.org/10.1016/j.mechrescom.2023.104202>
12. E. Turco, E. Barchiesi, A. Causin, F. dell’Isola, M. Solci, Harnessing unconventional buckling of tube origami metamaterials based on Kresling pattern, *Int. J. Solids Struct.*, **300** (2024), 1–18. <https://doi.org/10.1016/j.ijsolstr.2024.112925>
13. G. Formica, F. Milicchio, W. Lacarbonara, Storage and damping optimization in hysteretic multilayer nanocomposites, *Int. J. Multiscale Comput. Engrg.*, **18** (2020), 141–157. <https://doi.org/10.1615/IntJMultCompEng.2020032669>

14. W. Lacarbonara, S. K. Guruva, B. Carboni, B. Krause, A. Janke, G. Formica, et al., Unusual nonlinear switching in branched carbon nanotube nanocomposites, *Sci. Rep.*, **13** (2023), 5185. <https://doi.org/10.1038/s41598-023-32331-y>
15. M. Kadic, G. W. Milton, M. van Hecke, M. Wegener, 3D metamaterials, *Nat. Rev. Phys.*, **1** (2019), 198–210. <https://doi.org/10.1038/s42254-018-0018-y>
16. T. Kirchdoerfer, M. Ortiz, Data-driven computational mechanics, *Comput. Methods Appl. Mech. Engrg.*, **304** (2016), 81–101. <https://doi.org/10.1016/j.cma.2016.02.001>
17. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016.
18. Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE*, **86** (1998), 2278–2324. <https://doi.org/10.1109/5.726791>
19. G. E. Hinton, S. Osindero, Y. Teh, A fast learning algorithm for deep belief nets. *Neural Comput.*, **18** (2006), 1527–1554. <https://doi.org/10.1162/neco.2006.18.7.1527>
20. W. S. McCulloch, W. Pitts, A logical calculus of ideas immanent in nervous activity, *Bull. Math. Biophys.*, **5** (1943), 115–133. <https://doi.org/10.1007/BF02478259>
21. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, et al., Imagenet large scale visual recognition challenge, *Int. J. Comput. Vision*, **115** (2015), 211–252. <https://doi.org/10.1007/s11263-015-0816-y>
22. K. Hornik, Approximation capabilities of multilayer feedforward networks, *Neural Netw.*, **4** (1991), 251–257. [https://doi.org/10.1016/0893-6080\(91\)90009-T](https://doi.org/10.1016/0893-6080(91)90009-T)
23. W. Pedrycz, S. M. Chen, *Deep Learning: Algorithms and Applications*, Springer Cham, 2020. <https://doi.org/10.1007/978-3-030-31760-7>
24. A. Bilotta, M. Morassi, E. Turco, Damage identification for steel-concrete composite beams through convolutional neural networks, *J. Vib. Control*, **30** (2024), 876–889. <https://doi.org/10.1177/10775463231152926>
25. A. Bilotta, A. Causin, M. Solci, E. Turco, Automatic description of rubble masonry geometries by machine learning based approach, in *Mathematical Modeling in Cultural Heritage*, (G. Bretti, C. Cavaterra, M. Solci, M. Spagnuolo (eds)), Springer Singapore, **55** (2023), 51–67. [https://doi.org/10.1007/978-981-99-3679-3\\_4](https://doi.org/10.1007/978-981-99-3679-3_4)
26. G. Kissas, S. Mishra, E. Chatzi, L. De Lorenzis, The language of hyperelastic materials, *Comput. Methods Appl. Mech. Engrg.*, **428** (2024), 117053. <https://doi.org/10.1016/j.cma.2024.117053>
27. Q. Kang, K. Q. Li, J. L. Fu, Y. Liu, Hybrid lbm and machine learning algorithms for permeability prediction of porous media: A comparative study, *Comput. Geotech.*, **168** (2024), 106163. <https://doi.org/10.1016/j.compgeo.2024.106163>
28. K. Q. Li, Z. Y. Yin, N. Zhang, Y. Liu, A data-driven method to model stress-strain behaviour of frozen soil considering uncertainty, *Cold Reg. Sci. Technol.*, **213** (2023), 103906. <https://doi.org/10.1016/j.coldregions.2023.103906>
29. S. Im, J. Lee, M. Cho, Surrogate modeling of elasto-plastic problems via long short-term memory neural networks and proper orthogonal decomposition, *Comput. Methods Appl. Mech. Engrg.*, **385** (2021), 114030. <https://doi.org/10.1016/j.cma.2021.114030>

30. K. Li, R. Horton, H. He, Application of machine learning algorithms to model soil thermal diffusivity, *Int. Commun. Heat Mass Transfer*, **149** (2023), 107092. <https://doi.org/10.1016/j.icheatmasstransfer.2023.107092>

31. F. L. Fan, J. Xiong, M. Li, G. Wang, On interpretability of artificial neural networks: A survey, *IEEE Trans. Radiat. Plasma Med. Sci.*, **5** (2021), 741–760. <https://doi.org/10.1109/trpms.2021.3066428>

32. M. H. Beale, M. T. Hagan, H. B. Demuth, Deep Learning toolbox user's guide, The MathWorks, Inc, 2021.

33. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, et al., Pytorch: An imperative style, high-performance deep learning library, in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Curran Associates Inc., Red Hook, NY, USA, 2019, 8026–8037.

34. J. Yvonnet, D. Gonzalez, Q. C. He, Numerically explicit potentials for the homogenization of nonlinear elastic heterogeneous materials, *Comput. Methods Appl. Mech. Engrg.*, **198** (2009), 2723–2737. <https://doi.org/10.1016/j.cma.2009.03.017>

35. M. A. Bessa, R. Bostanabad, Z. Liu, A. Hu, D. W. Apley, C. Brinson, et al., A framework for data-driven analysis of materials under uncertainty: Countering the curse of dimensionality, *Comput. Methods Appl. Mech. Engrg.*, **320** (2017), 633–667. <https://doi.org/10.1016/j.cma.2017.03.037>

36. R. Ibañez, D. Borzacchiello, J. V. Aguado, E. Abisset-Chavanne, E. Cueto, P. Ladeveze, et al., Data-driven non-linear elasticity: Constitutive manifold construction and problem discretization, *Comput. Mech.*, **60** (2017), 813–826. <https://doi.org/10.1007/s00466-017-1440-1>

37. J. Ghaboussi, J. H. Garrett, X. Wu, Knowledge-based modeling of material behavior with neural networks, *J. Eng. Mech.*, **117** (1991), 132–153. [https://doi.org/10.1061/\(ASCE\)0733-9399\(1991\)117:1\(132\)](https://doi.org/10.1061/(ASCE)0733-9399(1991)117:1(132))

38. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.*, **9** (1997), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>

39. Y. M. A. Hashash, S. Jung, J. Ghaboussi, Numerical implementation of a neural network based material model in finite element analysis, *Int. J. Numer. Meth. Engrg.*, **59** (2004), 989–1005. <https://doi.org/10.1002/nme.905>

40. J. Ghaboussi, D. E. Sidarta, New nested adaptive neural networks (nann) for constitutive modeling, *Comput. Geotech.*, **22** (1998), 29–52. [https://doi.org/10.1016/S0266-352X\(97\)00034-7](https://doi.org/10.1016/S0266-352X(97)00034-7)

41. Y. Shen, K. Chandrashekara, W. F. Breig, L. R. Oliver, Neural network based constitutive model for rubber material, *Rubber Chem. Technol.*, **77** (2004), 257–277. <https://doi.org/10.5254/1.3547822>

42. K. Linka, M. Hillgärtner, K. P. Abdolazizi, R. C. Aydin, M. Itskov, C. J. Cyron, Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning, *J. Comput. Phys.*, **429** (2021), 110010. <https://doi.org/10.1016/j.jcp.2020.110010>

43. B. A. Le, J. Yvonnet, Q. C. He, Computational homogenization of nonlinear elastic materials using neural networks, *Int. J. Numer. Meth. Engrg.*, **104** (2015), 1061–1084. <https://doi.org/10.1002/nme.4953>

44. S. Kim, H. Shin, Data-driven multiscale finite-element method using deep neural network combined with proper orthogonal decomposition, *Engrg. Comput.*, **40** (2023), 661–675. <https://doi.org/10.1007/s00366-023-01813-y>

45. S. Kim, H. Shin, Deep learning framework for multiscale finite element analysis based on data-driven mechanics and data augmentation, *Comput. Methods Appl. Mech. Engrg.*, **414** (2023), 116131. <https://doi.org/10.1016/j.cma.2023.116131>

46. I. Chung, S. Im, M. Cho, A neural network constitutive model for hyperelasticity based on molecular dynamics simulations, *Int. J. Numer. Meth. Engrg.*, **122** (2021), 5–24. <https://doi.org/10.1002/nme.6459>

47. G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, *Nat. Rev. Phys.*, **3** (2021), 422–440. <https://doi.org/10.1038/s42254-021-00314-5>

48. K. A. Kalina, L. Linden, J. Brummund, P. Metz, M. Kästner, Automated constitutive modeling of isotropic hyperelasticity based on artificial neural networks, *Comput. Mech.*, **69** (2022), 213–232. <https://doi.org/10.1007/s00466-021-02090-6>

49. K. A. Kalina, J. Brummund, W. Sun, M. Kästner, Neural networks meet anisotropic hyperelasticity: A framework based on generalized structure tensors and isotropic tensor functions, preprint, arXiv:2410.03378, 2024. <https://doi.org/10.48550/arXiv.2410.03378>

50. F. Masi, I. Stefanou, P. Vannucci, V. Maffi-Berthier, Thermodynamics-based artificial neural networks for constitutive modeling, *J. Mech. Phys. Solids*, **147** (2021), 104277. <https://doi.org/10.1016/j.jmps.2020.104277>

51. F. As'ad, P. Avery, C. Farhat, A mechanics-informed artificial neural network approach in data-driven constitutive modeling, *Int. J. Numer. Meth. Engrg.*, **123** (2022), 2738–2759. <https://doi.org/10.1002/nme.6957>

52. D. K. Klein, R. Ortigosa, J. Martínez-Frutos, O. Weeger, Finite electro-elasticity with physics-augmented neural networks, *Comput. Methods Appl. Mech. Engrg.*, **400** (2022), 115501. <https://doi.org/10.1016/j.cma.2022.115501>

53. D. K. Klein, M. Fernández, R. J. Martin, P. Neff, O. Weeger, Polyconvex anisotropic hyperelasticity with neural networks, *J. Mech. Phys. Solids*, **159** (2022), 104703. <https://doi.org/10.1016/j.jmps.2021.104703>

54. B. Amos, L. Xu, J. Zico Kolter, Input convex neural networks, *Proc. Machine Learning Res.*, **70** (2017), 146–155. Available from: <https://proceedings.mlr.press/v70/amos17b.html>.

55. K. Xu, D. Z. Huang, E. Darve, Learning constitutive relations using symmetric positive definite neural networks, *J. Comput. Phys.*, **428** (2021), 110072. <https://doi.org/10.1016/j.jcp.2020.110072>

56. K. Linka, E. Kuhl, A new family of constitutive artificial neural networks towards automated model discovery, *Comput. Methods Appl. Mech. Engrg.*, **403** (2023), 115731. <https://doi.org/10.1016/j.cma.2022.115731>

57. P. Thakolkaran, A. Joshi, Y. Zheng, M. Flaschel, L. De Lorenzis, S. Kumar, NN-EUCLID: Deep-learning hyperelasticity without stress data, *J. Mech. Phys. Solids*, **169** (2022), 105076. <https://doi.org/10.1016/j.jmps.2022.105076>

---

- 58. M. Flaschel, S. Kumar, L. De Lorenzis, Unsupervised discovery of interpretable hyperelastic constitutive laws, *Comput. Methods Appl. Mech. Engrg.*, **381** (2021), 113852. <https://doi.org/10.1016/j.cma.2021.113852>
- 59. M. Flaschel, H. Yu, N. Reiter, J. Hinrichsen, S. Budday, P. Steinmann, et al., Automated discovery of interpretable hyperelastic material models for human brain tissue with euclid, *J. Mech. Phys. Solids*, **180** (2023), 105404. <https://doi.org/10.1016/j.jmps.2023.105404>
- 60. J. N. Fuhr, N. Bouklas, R. E. Jones, Learning hyperelastic anisotropy from data via a tensor basis neural network, *J. Mech. Phys. Solids*, **168** (2022), 105022. <https://doi.org/10.1016/j.jmps.2022.105022>
- 61. J. Ling, R. Jones, J. Templeton, Machine learning strategies for systems with invariance properties, *J. Comput. Phys.*, **318** (2016), 22–35. <https://doi.org/10.1016/j.jcp.2016.05.003>
- 62. J. N. Fuhr, A. Jadoon, O. Weeger, D. T. Seidl, R. E. Jones, Polyconvex neural network models of thermoelasticity, *J. Mech. Phys. Solids*, **192** (2024), 105837. <https://doi.org/10.1016/j.jmps.2024.105837>



AIMS Press

© 2025 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>)