



A Review on Data-Driven Constitutive Laws for Solids

Jan N. Fuhr¹ · Govinda Anantha Padmanabha² · Nikolaos Bouklas^{2,3} · Bahador Bahmani⁴ · WaiChing Sun⁴ · Nikolaos N. Vlassis⁵ · Moritz Flaschel⁶ · Pietro Carrara⁶ · Laura De Lorenzis⁶

Received: 7 May 2024 / Accepted: 9 October 2024

© The Author(s) under exclusive licence to International Center for Numerical Methods in Engineering (CIMNE) 2024

Abstract

This review article highlights state-of-the-art data-driven techniques to discover, encode, surrogate, or emulate constitutive laws that describe the path-independent and path-dependent response of solids. Our objective is to provide an organized taxonomy to a large spectrum of methodologies developed in the past decades and to discuss the benefits and drawbacks of the various techniques for interpreting and forecasting mechanics behavior across different scales. Distinguishing between machine-learning-based and model-free methods, we further categorize approaches based on their interpretability and on their learning process/type of required data, while discussing the key problems of generalization and trustworthiness. We attempt to provide a road map of how these can be reconciled in a data-availability-aware context. We also touch upon relevant aspects such as data sampling techniques, design of experiment, verification, and validation.

1 Introduction

Problems in solid mechanics are formulated in terms of three sets of equations: the first encodes basic conservation principles (e.g., balance of linear momentum) and governs the equilibrium of deformable bodies following the definition of a stress tensor; the second describes the kinematics of motion in terms of displacements, strains and strain rates. The third set, denoted as *constitutive law* (CL) (or *material model*), describes the response of a material to external stimuli by establishing a relation between kinematic and static quantities (e.g., between strains and stresses) eventually

mediated through other measurable and/or internal (i.e., non-measurable) field variables, whose *evolution laws* are part and parcel of the description. The material behavior and, therefore, the CL can be classified either as *path-independent* (also, *history-independent*) if the current state of a point does not depend on its previous states, or *path-dependent* (also, *history-dependent*) when the current state depends on the history of states experienced by the point. While conservation principles and kinematics are considered axiomatic and epistemic, material modeling is empirical in nature; it is in continuous evolution and it constitutes one of the most important research fields in mechanics.

Since the introduction of Young's modulus in the 19th century [1], engineers have predominantly defined CLs using the so-called *phenomenological* approach, where experimental observations and physical requirements are distilled into a priori selected analytical ansatz relationships whose parameters are meant to be characteristic of the material [2–5]. Thus, constitutive modeling is traditionally based on data and, due to the limitations of traditional experimental setups, hinged on limited observations on a restricted set of load cases (e.g., inducing uniaxial, biaxial and hydrostatic stress states). On the other hand, CLs were expected to generalize to significantly more complex conditions, a task for which the development of continuum thermodynamics theories and computational techniques has been crucial.

The development of full-field experimental methods such as *digital image correlation* (DIC) [6, 7], X-ray

✉ Nikolaos Bouklas
nbouklas@cornell.edu

¹ Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX, USA

² Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY, USA

³ Center for Applied Mathematics, Cornell University, Ithaca, NY, USA

⁴ Department of Civil Engineering and Engineering Mechanics, Columbia University, New York, NY, USA

⁵ Department of Mechanical and Aerospace Engineering, Rutgers University, New Brunswick, NJ, USA

⁶ Department of Mechanical and Process Engineering, ETH, Zurich, Switzerland

computed tomography [8–10] and *digital volume correlation* (DVC) [11, 12] and advances in the related computational approaches [13, 14] have shifted the constitutive modeling paradigm from a limited- to a large-data regime. In particular, the computational multiscale techniques available to date deliver in-silico observations of arbitrary complex implicit macroscopic behaviors from simple CLs defined at the micro-scale. However, along with the opportunity to obtain better CL comes also the challenge of analyzing the increased amount of data available. In this context, the recent breakthrough in big data analysis and information mining gives a unique opportunity to improve constitutive modeling for applications in mechanics and material science. Although the preliminary results obtained in this area bring great promise of efficient and highly accurate predictions, the fast development of this discipline calls for a timely identification of the most promising directions to pursue in order to boost the progress of mechanics and that can introduce a diversity of solution techniques.

1.1 Classification and Nomenclature

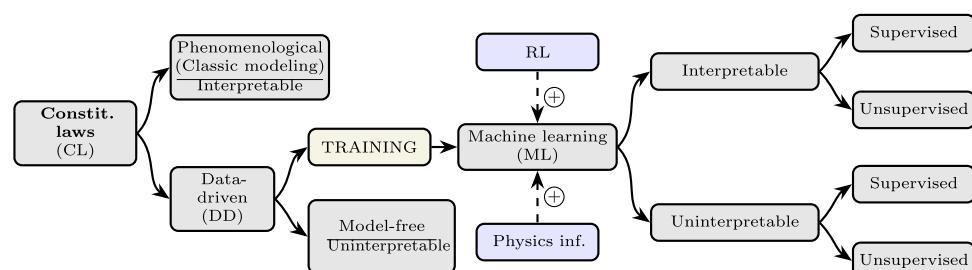
Available techniques that deal with large data are often referred to as *data-driven* (DD) approaches, due to their strict dependence upon a set of experimental or numerical observations of the material behavior (Fig. 1). In general, each method deploys a set of algorithms, assumptions and procedures, whose purpose is to analyze the available data and deduct useful information to describe the behavior of a certain material or class of materials. Since the recent breakthroughs in deep learning, machine learning (ML) techniques have received renewed interest in applications in mechanics and materials. Particularly the rise of sparse identification and discovery [15, 16], Physics Informed neural networks (PINNs) and operator learning approaches [17, 18] for the solution of forward and inverse problems involving partial differential equations (PDEs) [19, 20] have had a significant impact on the computational engineering field, shifting the attention to combining data with physics. The great promise of fast and highly accurate predictions for mechanistic simulations as well as cost reduction in an industrial setting has led to numerous attempts to integrate a wide spectrum of these techniques in the simulation workflow

and also to facilitate material innovation. Combining physical constraints and data in the context of ML approaches has shown potential towards moderating the need for large training datasets [21].

For the purpose of this review on DD constitutive modeling in solid mechanics, with the intent of providing a structured presentation, we propose for the available DD approaches the classification outlined in Fig. 1. We distinguish among two broad categories with corresponding sub-categories, as follows:

- methods based on *machine learning* (ML) techniques. Within this category, we distinguish among
 - *uninterpretable* (black-box or grey-box) approaches. This subcategory includes methods that obtain CLs in which the relation existing between inputs and outputs cannot be physically explained. These methods are also sometimes referred to as *encoding* or *learning* of CLs. Prominent ML techniques used in uninterpretable approaches are those based on many different types of neural networks (NNs) (see Sect. 2.1.1). The difference between black-box and grey-box approaches is that the latter encodes some known information about the physical system in the learning framework; this augments the clarity of the learned model but still does not enable interpretability. Recently, sparsification approaches for NN models aim to balance expressivity and interpretability.
 - *interpretable* approaches. Within this category fall the techniques that aim at defining an *interpretable* analytical expression for the CL, namely a relation between input and output quantities (e.g., strains and stresses) through mathematical operators and parameters whose role can be physically explained. In the literature, some of these methods are referred to as methods for *automated discovery* of CLs from data, whereby the term *discovery* is used to refer to simultaneous *model selection* and *parameter identification*. The special case of parameter identification, in which the functional form of the model is known a priori and only the unknown parameters are identified, collapses in terms of scope with the

Fig. 1 Characteristic features of the approaches in the constitutive modeling context



parameter identification of traditional constitutive modeling, but is now sometimes also carried out with ML. Prominent ML techniques used in interpretable approaches are symbolic regression and sparse regression (see Sect. 2.1.1).

In ML-based approaches, the definition of the operators and parameters takes the name of *training* or *learning* process, whereas their structure and configuration is also known as *architecture*.

- *model-free* approaches. These aim at integrating the material observations into the solution stream of a solid mechanics problem. In other words, they identify the solution of the system of PDEs governing a solid mechanics problem within the collected material observation pairs, thus bypassing an explicit analytical link between them. Note that, while ML-based methods deliver a CL mapping the input to the output which, once obtained, does not need any interaction with the original data, model-free methods cannot be set apart from the dataset they try to represent, since the dataset becomes part of the solution.

The above classification places a strong emphasis on interpretability. Interpretable approaches often lead to parsimonious (i.e. simple) CL representations, and parsimony is known to counteract the issue of overfitting which comes as a side effect of over-parametrization and limited training data. Additionally, parsimonious representations can enable generalization capabilities beyond the range of input data, given that the physics remains consistent (follows the same discovered law) out of that range.

An important aspect not contained in the above classification is the distinction between “*supervised*” and “*unsupervised*” methods. In ML, training is performed in a *supervised* fashion when matching pairs of labeled input and output data are used, or in an *unsupervised* manner if no labeled output data are available or in the absence of a bijective mapping between input and output quantities. Cases where the learning process objective is to minimize the deviations between the model predictions and the labeled data are part of the first category, while the methods whose goal is to detect hidden patterns or relationships within the data fall in the second. In the context of constitutive modeling, input and output quantities are, in the simplest case, strains and stresses (the situation becomes more complex for path-dependent material models, see Sect. 5). For this reason, we denote as supervised the methods that require training data in the form of stress–strain pairs, and as unsupervised those that do not require such pairs. The distinction is important because, while strains can be measured (almost) directly, stresses can only be computed from measured forces under very simple loading conditions, such as uniaxial tension, and

are therefore not available in general cases. Unsupervised methods formulate the learning procedure in such as way as not to rely on the availability of stress–strain pairs; to compensate for the lack of such pairs, they typically rely on the enforcement of physics constraints which can be formulated in terms of realistically measurable data, such as displacements and forces.

Physics constraints are not exclusive of unsupervised methods; approaches of all types can augment learning by imposing the fulfillment of such constraints (e.g., global or local equilibrium, governing PDEs), in which case they are sometimes also denoted as *physics-informed* (Fig. 1). To enforce these constraints, mainly three procedures can be deployed, either alone or in synergy: i.) the selection of an architecture that a priori satisfies them, ii.) penalizing the selection of parameters leading to the violation of the constraints, iii.) the a posteriori rejection of optimized models which are incompatible with the constraints following predefined rejection/acceptance policies. Another ML paradigm is the so-called *reinforcement learning* (RL), whose aim is not limited to the optimization of a predefined architecture but extends to learning a set of rules that allows for the improvement of the architecture itself by maximizing its predictive capability following a given metric (Fig. 1). This is normally done in a trial-and-error fashion by casting architectures whose performances are tested against the available data; a reward or a penalty is then given following a set of predefined policies. Note that the described methodologies are not mutually exclusive, but can be coupled in order to achieve a better description of the material behavior. The main characteristic features of the approaches to define a CL are summarized in Fig. 1, while their detailed description will be provided later.

1.2 Objective and Organization of the Paper

The primary goal of this review is not to provide an exhaustive list or a ranking of DD attempts in the field of solid mechanics but, rather, a map of how DD techniques can be used to advance the constitutive modeling field. We provide an overview of how DD approaches can exploit the large amount of available observations in order to obtain richer CLs able to accurately predict the behavior of different materials. The available literature is reviewed and classified following the main aspects summarized in Fig. 1 and whether the behavior they try to represent (e.g., elasticity, plasticity, damage) is path-independent or path-dependent. Whenever possible and relevant, we also overview the mechanics/physics knowledge that the various approaches embed, the capability to generalize to situations different than those represented by the data they are exposed to, and the amount of data they need to provide reliable results (namely, the *data hunger*). In the overview, we attempt to pinpoint the

specific advantages and drawbacks of some of the current state-of-the-art techniques and our perspectives on the future of DD constitutive modeling for solid mechanics problems. Although not treated extensively, the strict relation between modern experimental techniques and DD approaches is also hinted at, especially regarding the need for integrated frameworks where the experiments are designed to support the definition of accurate and realistic CL.

The paper is structured as follows: in Sect. 2 we summarize some earlier reviews in the context of DD approaches in solid mechanics and we provide a short overview of DD methods to establish a common ground for the remainder of the paper. Section 3 focuses on data sampling approaches, a central aspect of every approach that focuses on data. Section 4 reviews DD CLs for path-independent problems with a focus on small strain elasticity and finite strain hyperelasticity. Section 5 reviews DD CLs for a range of path-dependent problems, from plasticity, viscoelasticity, damage, fracture, and fatigue, to problems in multiphysics. Section 6 presents some thoughts on guidelines for validation and verification and proposes a set of possible performance metrics, while Sect. 7 discusses the current limitations of the reviewed approaches and draws the main conclusions.

2 DD Methods Outline and Earlier Reviews

2.1 Overview of the Available DD Methods

In the following, a non-exhaustive list of important DD methods is provided, as they are used either alone or in combination for the different constitutive models discussed in the following sections. For most of these models, different variations exist that allow for either classification of data or fitting regression curves. Since in constitutive modeling, we are usually concerned with the prediction of quantities instead of labels, the following summary mainly focuses on regression tasks.

2.1.1 ML Approaches

Interpretable approaches. Symbolic regression and sparse regression are probably the closest ML techniques to traditional constitutive modeling (based on parameter identification of assumed laws) and have the advantage that functional dependencies between inputs and outputs are highly interpretable, and constitutive modeling constraints can be incorporated, which typically leads to high extrapolation power. Both approaches are based on finding an analytical function that best fits a given dataset. In general, approaches that derive mathematical expressions from targeted experimental data are commonly referred to as symbolic regression methods; hence, also sparse regression can be interpreted

as a specific instance of symbolic regression (see also the extensive review in [22], where symbolic regression methods, including sparse regression, are viewed through the lens of operations on graphs). However, in the literature very often symbolic regression is used to denote methods based on genetic algorithms. In what follows, we adopt this terminology; hence, we refer to methods that are based on genetic algorithms as symbolic regression and treat sparse regression separately.

In symbolic regression, building blocks such as mathematical operators and constants are combined with commonly applied analytical functions such as sine or logarithm using genetic algorithms [23, 24]. In the physical sciences, symbolic regression was popularized by [15], and an overview with applications to material science is provided in [25]. Different software packages can be found in the literature, e.g., GPTIPS [26], Eureqa [27], gplearn [28], AIFeynman [29]. A recent comparison between common open-source implementations can be found in [30]. The disadvantage of these methods is the long training time of the genetic algorithms without guaranteed convergence to a good solution. Furthermore, in comparison to other methods, which are more automatized, a high degree of user knowledge is required since the space of mathematical expressions has to be *a priori* provided by the user. Additionally, due to the commonly low number of trainable parameters, the method is not as expressive and might struggle with highly complex functional dependencies.

In contrast to symbolic regression, sparse regression [31–33] discovers a symbolic model expression from a predefined catalog (often denoted as a library) of candidate models. The strength of sparse regression is its computational efficiency and, in the frequent case of a convex objective function, guaranteed convergence. The idea of using sparse regression in the physical sciences was initiated by [16]. Recently, sparse regression has gained increasing popularity for material model discovery [34–36]. Also, in this case, user knowledge is required since the catalog of material models is chosen upfront by the user. However, recent research is trying to automatize this task towards a self-generated library [22].

Uninterpretable approaches. In recent years, artificial NNs and especially deep NNs have been widely used for many complex regression tasks and NNs are probably the most encountered type of ML method in the context of DD constitutive modeling. Their historically first and simplest form, known as feed-forward neural NN or fully connected NN, takes vector-valued inputs and maps them to the vector-valued output using only forward-directed connections between the hidden layers of the network, that consist of several hidden nodes. Other forms include sequential modeling architectures such as recurrent (or sequential) NNs (RNNs) [37], Gated Recurrent Unit (GRU) [38], Long

Short-Term Memory (LSTM) [37], and Transformers [39], which are based on internal states that allow for processing time- and history-dependent datasets. Lastly, convolutional NNs (CNNs) [40], attention networks [41], and graph NNs (GNNs)[42] are used to analyze visual imagery and graph dependencies, respectively. For more information on these types of NNs, we refer to [43]. One important reason for the popularity of NNs is that all the major current software implementations are based on Automatic Differentiation [44]. The most commonly employed open-source libraries are Tensorflow [45], Pytorch [46] and JAX [47]. Other than their flexibility, the advantages of NNs include their ease of usage and their ability to deal with very large datasets. An important positive aspect of NNs is that they allow to weakly incorporate constitutive model constraints. On the other hand, even though NNs with arbitrary widths or depths are theoretically able to approximate any function [48, 49], from an engineering point of view with naturally limited computational resources, convergence cannot be guaranteed. Furthermore, NNs do not offer exact inference properties, their analytical function is hard to interpret due to the number of trainable weights, and they are reliant on a large amount of user-chosen parameters (hyperparameters). Also, performing Bayesian inference using NNs, known as Bayesian NNs (BNNs)[50], is popular in the ML community for uncertainty quantification (UQ). Here, additional constraints in constitutive modeling can be enforced in various ways [51]. However, the process typically requires a higher understanding of statistical learning theory in comparison to enforcing constraints with NNs.

NNs and Bayesian NNs are widely used for many computationally expensive scientific problems for performing tasks such as inverse modeling or UQ. However, these models are confined to the mapping between a predefined set of input and output data. They cannot be directly applied to control or experimental design problems due to the difficulty in learning optimal decision-making strategies in dynamic and uncertain environments. To address this issue, an RL technique that hinges on the concept of dynamic programming is employed [52]. RL is a type of ML model in which the agent or an NN model learns the best actions to make decisions by interacting with the environment. The decision-making process in RL is executed by defining states, actions, and rewards. During training, the data from the environment known as the state is provided to the agent to assess the reward. The main objective of RL is to enhance the model's policy or value function, enabling it to take the best actions over the iterations known as episodes. There are mainly two types of RL models, namely, model-free and model-based RL. Model-free RL algorithms, such as Q-learning [53] and Policy Gradient algorithms [54], focus on learning the optimal solution by directly interacting with the environment. On the other hand, model-based RL algorithms such as Dyna

[55] and AlphaGo Zero [56] depend on planning based on the learned model.

Support vector regression (SVR) is well established in the ML community for real-valued function estimation [57]. Rooted in statistical learning theory, SVR is based on its well-known classification-based counterpart known as support vector machines (SVM), which try to systematically find a linear hyperplane able to cluster input and related output data in classes. However, the latter assumption fails to represent the classification when the mapping between input and output parameters is non-linear. In such cases, the 'kernel trick' [58] is employed to convert the original data space into a higher dimensional one where a linear hyperplane is again a suitable ansatz. Considering the domain and codomain of an unknown function composed of classes representing a restricted range of values, it is possible to endow the SVM with regression capabilities leading thus to the SVR. To avoid overfitting, SVR penalizes predictions farther away than a specific value from the desired output in a convex loss function. For constitutive modeling in solid mechanics, SVR has traditionally been far less commonly employed compared to other ML methods although different libraries offer the possibility to use SVR. The main open-source software library used in the literature is libSVM [59]. Alternatively, the Scikit-learn [60] also has an implementation of SVR, which is, however, also based on libSVM. The advantages of SVR methods are that they are robust against outliers in the data, are easy to implement, their computational complexity does not depend on the dimensionality of the input space, and, when well fitted, they tend to have good generalization capabilities [61, 62]. Furthermore, due to the convexity of the loss function, SVR training delivers a unique solution that makes training in comparison to other advanced ML methods easier. Constraints generally encountered in constitutive modeling can be enforced by adjusting the loss function [63]. However, the method has problems in the big data domain and with datasets that contain a significant amount of noise. Furthermore, in comparison to Gaussian process regression (GPR), there is no probabilistic explanation for the fitting and no exact inference.

Apart from the above-mentioned models for regression tasks, feature extraction models and probabilistic generative models are other popular ML areas that are widely used in the scientific domain. Feature extraction is used to reduce datasets into their informative and non-redundant parts. For more details, we refer to [64]. They usually involve some form of dimensionality reduction, and common techniques of this kind are principal component analysis (PCA) [65], autoencoders [66], and clustering techniques such as k-means clustering. Other popular ML domains are probabilistic generative models such as restricted Boltzmann machine (RBM) [67], variational autoencoder (VAE) [68], generative adversarial network (GAN) [69], flow-based

generative models (or Normalizing Flows) [70] or diffusion models (DM) [71], which have been successfully implemented for various tasks such as dimensionality reduction, inverse surrogate modeling [72], image generation [69], anomaly detection [73], and many others. Due to the relatively small datasets generally encountered in constitutive modeling, these methods have not yet seen a lot of interest in the community. However, with the ever-increasing availability of data, they may soon become more important.

Bayesian inference and statistical learning. Another important class of ML methods is models based on Bayesian inference. They revolve around using Bayes' theorem to find the probability distribution of parameters or functions given observed data and prior distributions. Since often no closed-form solutions for the posterior distributions exist, efficient simulation algorithms like Markov Chain Monte Carlo (MCMC), Metropolis-Hastings (MH), and Hamiltonian (or hybrid) Monte Carlo (HMC) are applied to find approximations. For more information, we refer to [74]. Analytical posterior distributions can be found by using Gaussian processes as priors. The resulting interpolation method is typically known as GPR or Kriging [75]. In the context of ML CL modeling, GPR is the most encountered form of Bayesian regression due to its simplicity. Different open-source software packages for GPR are available, e.g., DACE [76], DiceKriging [77] or GPyTorch [78]. GPR methods have significant advantages compared to other ML models: they have rigorous convergence guarantees, yield exact inference in the absence of noise, allow for probabilistic error estimation, and are fully trainable with only a limited amount of user-chosen parameters [75]. The major issue of GPR is that it scales cubically with the size of the dataset; this has historically prevented it from being used in the big data context [79]. Overall, utilizing GPR for constitutive modeling has been a relatively recent phenomenon in comparison to other ML models. Finally, classical but proven techniques of statistical learning theory should not be neglected when deciding which predictive tool to use. Methods like k-nearest neighbor, random forests, or spline interpolation might, depending on the complexity of the data, even be the best-performing methods for certain tasks. For a textbook on these traditional techniques, we refer to [80].

Closing remarks. As pointed out, all of the presented methods have their own strengths and weaknesses, and similar to most application areas in engineering there is also no single best method to apply for constitutive modeling. The choice of which method to use is typically dependent on the size of the dataset, the type of the data source, the required level of interpretability, the degree of dimensionality of the inputs and outputs, and the complexity and amount of noise in the data. Moreover, in the end, the choice of an ML method usually boils down to user knowledge of specific methods and algorithms and the availability of codes;

facilitating this process could be beneficial for the broader community. In this regard, due to the number of historical papers and available tutorials, NNs can certainly be seen as the easiest and most flexible method for inexperienced users. Approaches that initially show a steeper learning curve might however have some significant advantages for specific applications and goals.

2.1.2 Model-Free Approaches

As mentioned earlier, model-free approaches first proposed by [81] do not explicitly create material models or their surrogates to be integrated into computations but, instead, directly inform the forward problem with a set of discrete material behavior observations (the so-called *material data set*). The main idea is to relax the requirement of the classical solution procedure of a mechanical boundary value problem that seeks the solution as the intersection between a set of basic governing equations (e.g. equilibrium, compatibility) and the CL. Instead, the model-free approach identifies for each point of the domain the state within the available material observations that is closest to the subset of points satisfying compatibility and equilibrium. Hence, the model-free solution strategy relies on the definition of a discrete quantity, representing a distance induced by a given metric in the state space, which attains its minimum in correspondence with the material data point that best represents the solution under the given boundary conditions [82]. The aim of this approach is to minimize assumptions on the modeling part by relying only on discrete observations of the material behavior. This makes the model-free approach data-hungry [83, 84] since it is unable to generalize the observed behavior. For the same reason, it poses limitations to the description of dissipative CLs, where the major challenge is to ensure the representability of the material behavior without the introduction of a priori-defined dependencies from postulated internal/history variables or oversized datasets.

2.2 Earlier Reviews

There are several recent reviews in the context of DD approaches for a wide array of solid mechanics problems. We summarize here the main contributions towards three main fields: i. single or multiscale approaches to represent the material behavior, ii. design of new (meta-)materials with prescribed behavior (namely, material behavior optimization) and, iii., synergistic approaches integrating experimental mechanics and ML methods.

Material behavior representation. In [85] a review of DD modeling approaches in engineering is presented that aims to introduce the reader to a variety of approaches and applications. With a focus on *virtual twins*, namely computational frameworks replicating the material behavior that support

and extend the experimental practice, [86] discusses efficient approaches for the exploitation of data in computations beyond classical parameters calibration. More focused on multiscale approaches, surrogate models, and design optimization of composite materials and structures, [87] revises the available approaches employing NNs, while [88] focuses more on physics-informed ML and tries to identify applications and opportunities in the general context of computational mechanics. Also, [89] extends the review to the whole ecosystem of ML approaches and tries to outline the future perspectives of the field. Focusing on mechanical properties, [90] discusses the importance of data collection, generation, and preprocessing towards applications in multiscale materials design.

Design of meta-materials. [91] reviews the advancements driven by ML tools towards material design innovation, while in [92] a classification among descriptive, predictive, or prescriptive is utilized to map ML approaches to problems of parameter calibration, material characterization, and material design and optimization. Focusing on material discovery and the materials genome, [93] discusses how ML approaches can take advantage of mechanical and chemical datasets to propel the design of new meta-materials with improved properties, while [94] and [95] review respectively the methods available and the challenges and opportunities involved.

Synergy between experimental and DD methods. Various hierarchical identification procedures and related data-reduction methods to accelerate the exploitation of the experimental information are accounted for in [96], while [97] reviews the ML approaches involving data from acoustic emission and resonant ultrasound spectroscopy and points out that including known physical and mechanistic relationships in the ML approaches increases the reliability of the trained models. The integration of experimental and ML methods is exploited extensively in the area of biomechanics. With this focus, [98] reviews aspects of the solution of biomechanics problems using DD methods based on patient-specific data and highlights how this enables an automatic consideration of the case-wise variability of the material parameters and of the uncertainty propagation.

Despite the number of review articles published, to the best of the authors' knowledge a review that solely targets constitutive modeling approaches in solid mechanics is still lacking. With this paper, we aim to fill this gap.

3 Data Sampling and Design of Experiments

One of the open problems in DD material modeling is the dependence of the performance on the amount of available data. One possible remedy to the unavailability of sufficient data is to train the ML tools on less but qualitatively more

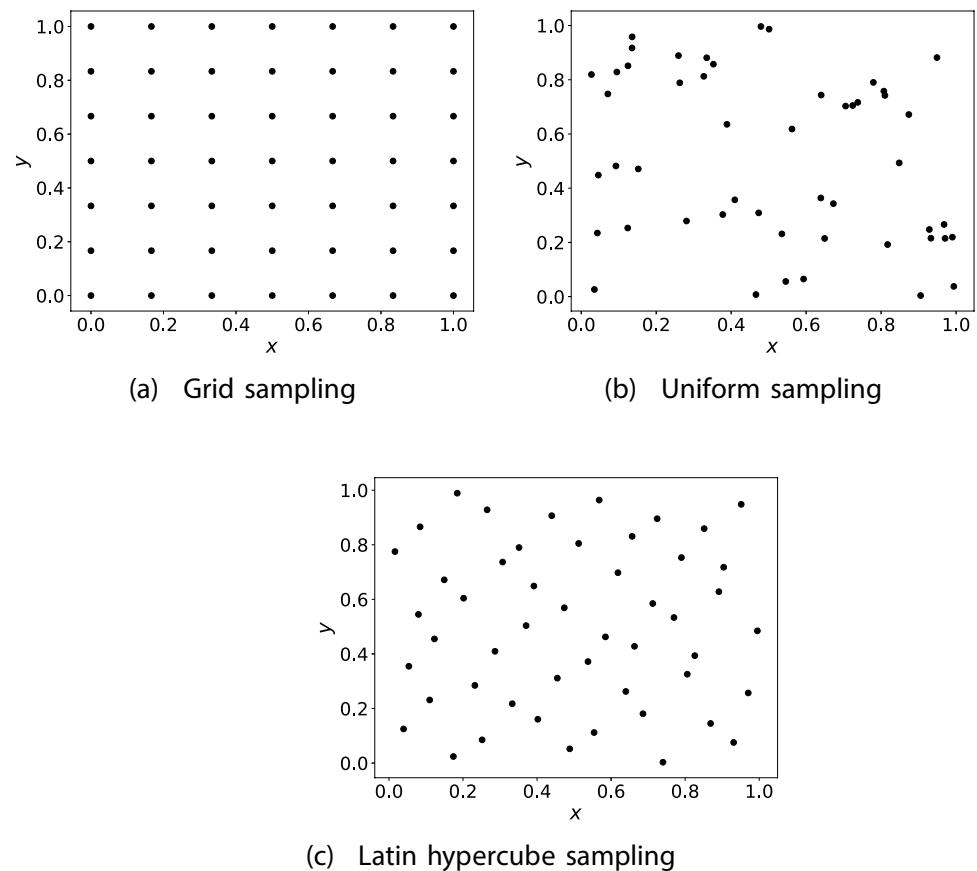
relevant data, i.e. data that capture the major complexities of the mapping. This can be achieved by relying on pertinent data sampling strategies. In this section, we briefly review some of these strategies that have been employed in the literature for DD material modeling. We differentiate between one-shot and sequential sampling approaches.

The design of sampling strategies can be paralleled to experimental design. Experimental design denotes the systematic planning and structuring of lab experiments (e.g. loading paths, types of experiments) to gather valuable and relevant data from lab tests to calibrate constitutive models. While both sampling strategies and experimental design aim to efficiently extract the most meaningful data, experimental design brings an additional layer of complexity. It must not only consider the theoretical significance of the data but also the feasibility and efficiency of conducting the experiment in a real-world lab setting. Practical limitations, such as equipment availability, time constraints, and material properties, often dictate what can be accomplished in the lab. After discussing one-shot and sequential sampling approaches, we briefly overview the works that utilize deep RL for experimental design, showcasing its potential in optimizing and informing experimental setups.

One-shot sampling. One-shot sampling is characterized by the determination of the sample size and points in a single stage. To this end, the input domain for both time-dependent and time-independent inputs has to be known. Non-temporal inputs are characterized by a fixed range of interest, i.e. a sampling interval, or a probability distribution function. Since sampling from a probability distribution is straightforward, in the following we focus on deterministic input domains. The simplest way of generating data for non-temporal inputs is grid-based sampling (Fig. 2a) where a few equally spaced values are selected for each parameter. Apart from its simplicity, this type of strategy has several other advantages, including easy setup for parameter sensitivity studies and numerical integration, and of course its space-filling properties. Grid-based sampling strategies have been applied in a variety of publications [99–101].

A major limitation of grid-based sampling is their "collapsing" property [102] which essentially means that sample points may have the same coordinate value when projected onto a parameter axis. This has the undesirable effect that, when one of the design parameters has (almost) no influence on the mapping, then two points whose only difference is this property can essentially be seen as the same point. This means that for the purposes of generating a surrogate for the mapping the same point will be evaluated twice. Hence, the two crucial criteria for one-shot sampling designs are space-filling and non-collapsing properties. An obvious way to avoid a collapsing design would be to use uniform random sampling. However, especially if a small quantity of sample points is involved, random sampling approaches tend to be

Fig. 2 Different one-shot sampling strategies. **a** Grid sampling. The input domain is discretized using equidistant samples. **b** Uniform sampling. The sample points are generated from a uniform distribution in the input domain. **c** Latin hypercube sampling. The points are sampled such that there is only one point at each section of a divided input space



not reliably space-filling, i.e. they show clustering behavior and fail to provide points in large portions of the region, as seen in Fig. 2b. For this reason quasi-random techniques such as Latin hypercube sampling [103] are often preferred (Fig. 2c). This method aims to generate a space-filling design that is non-collapsing. This strategy is employed in various works in the context of DD material modeling [104–106]. In some applications, due to indirect access to the input space (i.e. the input domain cannot be explicitly sampled), different strategies have to be employed. E.g., [21] proposes a sampling approach that generates space-filling points in the invariant space corresponding to a bounded domain of the deformation gradient tensor. The approach builds a space-filling sampling strategy based on simulated annealing, which provides more efficient and reliable physics-informed constitutive models than quasi-random sampling in the deformation gradient space.

Sampling from temporal inputs is more complicated as now a sequence of points has to be generated. This is e.g. required when the response depends on a specific three-dimensional loading path. Some aspects of effective temporal sampling are still an open issue in the literature, e.g. how to sample temporal curves in a space-filling manner. Different approaches have been proposed. In strain-controlled loading, [107] defines an upper and lower bound for each

individual strain component and a fixed number of time steps. The authors then propose to generate equally spaced points in the chosen bounds over the time frame using random sampling. These points are used as control points of an interpolator in order to generate smooth loading curves. Similar approaches are used in [108–110]. [111] samples the loading path as a random walk which changes its direction and step size using realizations of independent uniform random variables. The same method is employed in [110]. Other approaches [112] define the load paths directly as realizations of a multivariate Gaussian process.

Even though these methods have proven to be successful, overall, reliable and goal-oriented sampling of sequentially-based material responses still remains an open issue. Since non-temporal data-generation techniques are in general more reliable, DD modeling frameworks that transform a many-to-one mapping (as seen in plasticity) into one-to-one mappings can be considered more reliable.

Sequential/adaptive sampling. The problem with one-shot sampling is that, without a target function, it is difficult to predetermine an optimal or appropriate sample size as well as the ideal sample placement. This is especially true for temporal input parameters. For these reasons sequential or adaptive sampling techniques have been proposed in the broader surrogate modeling community [113–115]. These techniques

have been only sparingly applied in the context of DD material modeling. However, some approaches have been proposed in this field that could pave the way forward. To the best of the authors' knowledge, [116] is the first contribution that employs an adaptive sampling scheme for multiscale problems. The authors use multivariate kriging as a fit for a path-dependent DD material model. The proposed method is based on using the variance estimation of the kriging interpolant as an error estimate in the local material point routine of a finite element model. In essence, the surrogate takes the stress, pressure, temperature, and hardening variable of the current load step as input and returns the approximate values of the lower scale model, such as the directionality of the plastic velocity gradient, as well as an error estimate. If this error estimate is larger than a tolerance value, then a new data point is added at this input point and the surrogate model is retrained. This framework is refined in [117]. Similar approaches are proposed in [118, 119], but instead of relying on the error estimate of the surrogate, the model is adaptively refined when convergence in the local integration loop at a material point of the macroscopic structural problem cannot be achieved. In the context of model-free plasticity modeling [120] proposes a related idea. Overall, especially concerning path- and history-dependent material modeling, adaptive sampling strategies are largely underused. They have the potential to offer a reliable framework to train more robust DD material models.

Deep RL for experimental design. Deep RL is a subdomain of RL that combines the concepts of RL and deep NNs for tackling complex tasks involving continuous state and action spaces in an uncertain and dynamic environment. In the ML community, there are various deep RL models, such as Deep Q-Network (DQN) [121], Deep Deterministic Policy Gradient (DDPG) [122], and others that have been used for a wide range of problems. These include learning acquisition functions for Bayesian optimization [123], trading and finance [124], robotics manipulation [125], controls [126], experimental design [127], and many more. Recently, [128] developed a meta-modeling game framework based on RL, where two ML agents compete against each other to generate experimental data for calibrating a constitutive model. Out of all possible design options for laboratory experiments, the developed deep RL framework can determine sophisticated experimental designs that capture the relevant information to train reliable constitutive models. With a focus on using deep RL for expensive experimental data, [127] develops a framework that combines deep RL and Kalman filters for calibrating materials models.

4 DD Modeling for Path-Independent CLs

This section reviews the main DD modeling approaches proposed for path-independent CLs, namely for conservative and thus fully reversible material behavior. A key feature

of this type of materials is that the current state of a point is uniquely defined by the current value of the strain in that point. In the following, starting with linear elasticity and continuing with the finite-strain case, we review the main related contributions in the context of DD modeling.

4.1 Small-Strain Elasticity

The simplest assumptions in phenomenological constitutive modeling of solid materials are the assumption of small strains and the often accompanying assumption that the stress state depends linearly on the strain state $\sigma = \mathbb{C} : \epsilon$, where σ , ϵ and \mathbb{C} denote the Cauchy stress tensor, the infinitesimal strain tensor, and the elasticity tensor, respectively. Under these assumptions, the material characterization objective boils down to identifying the parameters that govern the linear stress-strain relation, i.e., the components of the elasticity tensor \mathbb{C} . Although not denoted as such, this parameter identification problem has often been faced from a DD perspective since its inception. Despite the simplicity of the problem, ML opens new doors for elastic property prediction.

While the response of homogeneous materials appears straightforward to model, mesoscopically heterogeneous materials such as multiphase materials and composites exhibit more complex stress-strain (or energy-strain) relations at the macroscopic scale. Hierarchical modeling is a powerful tool to describe the macroscopic material responses of such materials; the idea is to conduct lower-scale simulations of representative volume elements (RVEs), which capture the characteristic topology of the heterogeneous material, to predict the effective material behavior at the macroscopic scale. Prerequisites to such lower-scale simulations are, first, that the material properties of the different phases at the micro- or meso-scale are known and, second, extensive computational effort, especially in the three-dimensional case. Hierarchical methods such as FE² require the execution of a lower-scale simulation whenever the unknown stress state needs to be calculated from a known strain state, i.e., the stress-strain relation is implicit and costly to evaluate. ML-based methods promise a significant speed-up; given that a number of lower-scale simulations have been executed, i.e., a set of stress-strain data pairs is known, ML models can be used to interpolate the stress-strain data to arrive at an explicit model for the macroscopic stress-strain relation

$$\mathcal{M} : \epsilon \mapsto \sigma, \quad (1)$$

which can then be used efficiently in forward simulations. Following the same idea, the data can alternatively be used to train a differentiable ML model to arrive at an explicit form of the strain energy density W

$$\mathcal{M} : \varepsilon \mapsto W. \quad (2)$$

In this case, the stress-strain relation can be found by differentiating the above mapping with respect to the strain. A clear advantage of (2) over (1) is that the output of the mapping is scalar, which simplifies the training process. Expressions in the form of (2) further simplify the enforcement of physical constraints such as thermodynamic consistency, as discussed in later sections.

Uninterpretable ML approaches. One of the first methods for the development of models in the form of (2), termed Numerically Explicit Potentials [129], gathers data of RVE responses in a discretized loading space and interpolates the effective response with cubic splines [129] and later NNs [130]. To reduce the computational cost associated with the micro-scale simulations for data generation, [131] uses reduced-order RVE simulations to train a path-independent NN in the form of (1). The NN is then used as a surrogate for the macroscopic material behavior in forward finite element simulations, however, as the accuracy of the NN cannot be guaranteed outside the training domain, the authors propose to use reduced-order micro-scale simulations to predict the effective material response whenever predictions outside the training domain are required.

[132] proposes an ML-based multiscale material modeling method called Deep Material Network (DMN), which is constructed as a hierarchical network of layered composite building blocks with known analytical homogenization that maps the microscopic material properties $\mathbb{C}_{\text{micro}}$ to the effective material properties $\mathbb{C}_{\text{effective}}$

$$\mathcal{M} : \mathbb{C}_{\text{micro}} \mapsto \mathbb{C}_{\text{effective}}. \quad (3)$$

Each building block takes as input the compliance matrices of the individual phases and outputs the effective material properties in a rotated coordinate system. By chaining the building blocks in a binary tree structure in such a way that consequent blocks take as input the compliance matrices of preceding blocks, the emerging network is able to encode the homogenization process of materials with complex heterogeneities. The network parameters are trained based on two- or three-dimensional [133] linear elastic RVE simulations. Afterwards, in order to reduce the network complexity while retaining its predictive abilities, the authors suggest different model compression methods based on network node deletion and subtree merging. In [134], the same authors show that a pre-trained DMN can be used as initial input for the training process for new similar materials, e.g., the same type of composite but with different volume fraction f of the material phases. In this way, the training process can be accelerated and it can be ensured that different networks share the same architecture. This in turn enables interpolation between trained networks: if e.g. two networks for volume fractions f_1 and f_2 are trained, the material behavior of a structure with volume fraction f , $f_1 < f < f_2$, can be obtained from interpolation. In [135] the properties of DMN are studied on a theoretical basis. To this end, the authors interpret the network as a hierarchy of homogenization functions of generalized standard materials. In this way, they show that the networks are thermodynamically consistent and exhibit stress-strain monotonicity. They further propose new types of building blocks by allowing the composite building blocks proposed by [132] to rotate, and discuss their implementation in forward finite element simulations in [136]. Further, [137] and [138] show applications of the DMN in the context of porous and woven materials, respectively.

Realizing that supervised training of NNs requires a high amount of stress-strain data pairs, which are hardly available from experimental measurements or too costly to generate from RVE simulations, [139] proposes to use the energy-based characterization method (see [140]) to train NN-based CLs in an unsupervised manner, i.e., by leveraging full-field displacement measurements and global reaction forces, but no stress data. [141, 142] and [19] encounter the same problem and train NNs using indirectly measured data for the constitutive modeling of a fiber-reinforced plate [141] and two-dimensional in-plane shear material behavior [19, 142].

Another context in which ML finds its application is that of reduced-order modeling. By solving reduced-order models at the micro-scale, the computational cost of two-scale simulations can be drastically decreased while retaining high computational accuracy. [143] proposes to use ML, and in particular k -means cluster analysis, to divide the RVE geometry in the offline stage into a finite set of clusters with similar strain concentration. Assuming constant local variables over these clusters greatly simplifies the solution process of the Lippmann-Schwinger equation during the online stage. The method is non-parametric, naturally thermodynamically consistent, and is known as self-consistent clustering analysis. A great advantage of the method is that the offline stage can be computed under the assumption of small-strain elasticity, and the results of the offline stage can be utilized afterwards to compute material behavior beyond elasticity in the online stage. An equivalent approach is proposed independently by [144], which derives the method from the perspective of the Hashin-Shtrikman variational principle. [145] discusses the analogy between the methods proposed by [143] and [144] from a theoretical perspective and further points out the connection between the self-consistent clustering analysis and a related reduced-order modeling method, i.e., the transformation field analysis. Applications and extensions of the self-consistent clustering analysis are provided by [146–153]. [154] employ a denoising diffusion probabilistic model (DDPM) to map the reduced-order representation of small-strain elasticity to an embedded representation of the corresponding complex microstructures.

Thus, they resolve the optimization in a reduced-order space to facilitate the inverse design and generation of microstructures with specific targeted non-linear properties.

Finally, we mention ML-based approaches which allow to construct black-box models \mathcal{M} taking a set of microstructure or process parameters as input \mathcal{I} and outputting a set of targeted effective material properties \mathcal{O} , e.g., the elastic stiffness

$$\mathcal{M} : \mathcal{I} \mapsto \mathcal{O}, \quad (4)$$

The model \mathcal{M} may be any ML-based mapping, such as e.g. an NN or other function approximator. In this way, material properties may be predicted without experimental or numerical testing, but solely by knowing characteristic features of the production process and/or the microstructure. Interested readers are referred to the (non-comprehensive) list of references [139, 155–165]. The same idea can be extended to predicting mechanical properties as spatially varying fields. The ML-based mapping \mathcal{M} may take spatial measurements $\mathcal{I}(\mathbf{x})$ as input to predict a spatial property field $\mathcal{O}(\mathbf{x})$

$$\mathcal{M} : \mathcal{I}(\mathbf{x}) \mapsto \mathcal{O}(\mathbf{x}), \quad (5)$$

where \mathbf{x} denotes the spatial coordinate, see e.g. [163, 164].

Model-free approaches. A model-free approach is proposed in [81] with a first application on truss systems. The data set is composed of stress and strain tensor pairs and the solution is obtained by minimizing a distance defined in an energetic sense between simulated and observed stresses and strains subject to kinematic and equilibrium constraints, the latter enforced through Lagrange multipliers. [166] follows this idea by interpreting the given stress and strain data as a low-dimensional manifold that is embedded in the high-dimensional phase space (dimensionality is twelve, due to six independent components of the stress and strain tensors). [167, 168] propose the inverse problem of the problem posed by [81, 166], see also [168, 169]. The objective of their proposed method, denoted as DD identification, is to calculate the stress field given full-field displacement measurements, e.g., generated through DIC, without postulating any material model.

4.2 Finite-Strain Elasticity

Finite-strain elasticity, while sharing similarities with its small-strain counterpart, especially in the area of identifying elasticity parameters, introduces unique complexities. A primary challenge resides in incorporating geometric nonlinearity and material frame considerations, which complicate the task of parameter identification [170]. Despite these complexities, constitutive modeling work has turned to DD models to facilitate and automate calibration of novel materials. These models do not merely aim for an accurate

fit to the training data, but also for trustworthy predictions on unseen data points. The robustness of these predictions can be validated via thorough sensitivity checks of the input parameters, solidifying the reliability of the models for broader applications. In this section, we discuss the evolution of these models and the key considerations they embody, with a particular focus on parametrization and adherence to physical constraints.

Parametrization and the purposeful design of the input and output spaces are important aspects of DD finite-strain problems. The concern lies in the optimal determination of the input and output strain and stress states for the DD material laws. Some of the strain measures that are used in the finite-strain description are the deformation gradient (\mathbf{F}), the Green-Lagrange strain tensor (\mathbf{E}), and the right Cauchy-Green deformation tensor (\mathbf{C}), paired with their respective conjugate stress measures: the first Piola-Kirchhoff stress (\mathbf{P}) or the second Piola-Kirchhoff stress (\mathbf{S}). In the case of hyperelasticity, the formulation involves learning energy-density functions with respect to these variables, as illustrated by

$$\mathbf{P}(\mathbf{F}) = \frac{\partial \bar{W}(\mathbf{F})}{\partial \mathbf{F}} \quad \text{or} \quad \mathbf{S}(\mathbf{E}) = \frac{\partial \hat{W}(\mathbf{E})}{\partial \mathbf{E}} = 2 \frac{\partial \tilde{W}(\mathbf{C})}{\partial \mathbf{C}}. \quad (6)$$

The frame of reference chosen for parametrization has a substantial impact on the size and complexity of the learning problem. For instance, the deformation gradient \mathbf{F} and the stress tensor \mathbf{P} consist of nine components as two-point tensors, while the tensors \mathbf{E} , \mathbf{C} , and \mathbf{S} are symmetric with six components each. This difference calls for close consideration as it involves learning a non-symmetric fourth-order elasticity tensor $\mathbb{C}^{PF}(\mathbf{F})$ (with 81 coefficients in general) as opposed to symmetric $\mathbb{C}^{SE}(\mathbf{E})$, $\mathbb{C}^{SC}(\mathbf{C})$ elasticity tensors, both of which possess major and minor symmetries and thus require calibration for fewer elasticity coefficients. The CL can be formulated in terms of strain energy density as

$$\mathcal{M} : \mathbf{F} \mapsto \bar{W}, \quad \mathcal{M} : \mathbf{E} \mapsto \hat{W} \quad \text{or} \quad \mathcal{M} : \mathbf{C} \mapsto \tilde{W}, \quad (7)$$

or relying on the corresponding strain–stress mappings, i.e.

$$\mathcal{M} : \mathbf{F} \mapsto \mathbf{P}, \quad \mathcal{M} : \mathbf{C} \mapsto \mathbf{S}, \quad \text{or} \quad \mathcal{M} : \mathbf{E} \mapsto \mathbf{S}. \quad (8)$$

Formulations also often involve mappings from invariants of strain tensors to energy and stress measures, as addressed in numerous research works discussed below. Either of the aforementioned finite-strain elastic tensors depend on the current state of deformation, making their definition more complex than in the small-strain framework where only one constant tensor is needed. An additional important property of hyperelastic material models is polyconvexity, which along with coercivity is a sufficient condition for the solvability of boundary value problems under general boundary conditions and body forces.

The selection of strain and stress inputs and outputs in DD laws has consequences on an important required property of finite-strain material laws, namely, material frame indifference/objectivity. This property ensures that both the energy and stress response of the material remain constant upon rigid body motion and rotation. In the context of hyperelasticity, it is defined as follows

$$\bar{W}(\mathbf{F}) = \bar{W}(\mathbf{Q}\mathbf{F}), \quad \bar{\mathbf{P}}(\mathbf{F}) = \bar{\mathbf{P}}(\mathbf{Q}\mathbf{F}), \quad \forall \mathbf{Q} \in SO(3), \quad (9)$$

where \mathbf{Q} represents a rotation tensor and $SO(3)$ denotes the 3D rotation group. DD material laws expressed in terms of $\mathbf{P} - \mathbf{F}$ relations do not necessarily satisfy objectivity and may require additional enforcement of the related constraint, whereas material laws expressed as $\mathbf{S} - \mathbf{E}$ or $\mathbf{S} - \mathbf{C}$ relations are known to be automatically objective [170]. Frame invariance can also be inherently achieved by selecting an appropriate strain invariant formulation.

Uninterpretable ML approaches. One of the earliest attempts to model finite-strain elasticity using NNs goes back to [171, 172]. These authors mainly focus on rubber materials, utilizing a small NN to model a hyperelastic energy function. Their approach resembles the finite-strain model proposed by [173], which also serves as a benchmark. The authors adopt a strain invariant formulation of the input, ensuring the concise representation of the strain characteristics. Moreover, their approach also emphasizes the need for stress and higher-order derivatives from the learned function, particularly to facilitate the successful implementation of the trained NN within the finite element solver. This early exploration of finite-strain elasticity via NNs showcases the potential and challenges inherent to the task.

More recent work in finite-strain elasticity further explores the question of training a hyperelastic energy function and subsequently deriving the stress information through differentiation. This preference is rooted not only in its potential to simplify the learning problem - mapping from the strain space to a single, scalar-valued function W - but also in its utility in enforcing and validating necessary properties in the learned material law. [174] leverages both feed forward NNs and CNNs for predicting a homogenized energy function of a single microstructure for data from multiple numerical simulations of evolving crystal microstructures. The trained NNs then facilitate the calculation of the $\mathbf{P}(\mathbf{F})$ values through differentiation of the learned $W(\mathbf{F})$ in the multiscale approach. Meanwhile, [175] introduces the Sobolev training technique by [176] in the hyperelasticity training procedure for polycrystals. By doing so, the authors are able to constrain the optimized architecture to accurately predict derivatives of the learned $W(\mathbf{C})$ to calculate \mathbf{S} . Furthermore, by adopting a graph representation of the polycrystals as material descriptors of anisotropy, they also generalize the law to a family of microstructures.

[177] encodes hyperelastic energy functions using input convex NNs, addressing the issues of polyconvexity and thus material stability. They explore two methods for obtaining polyconvexity: firstly, they employ polyconvex, anisotropic, and objective invariants as inputs; secondly, they use the deformation gradient, its cofactor, and determinant as inputs while performing additional data augmentation to satisfy the objectivity condition. [178] also discusses an input convex NN-based energy function regression model framework to enforce mechanics principles such as objectivity, consistency, and dynamic material stability through proper parameterization of the energy mapping in the $\mathbf{S} - \mathbf{E}$ frame, augmenting the loss function, and modeling the NN weights after a softplus function respectively. Along similar lines, [179] trains input convex NNs as hyperelastic strain energy density functions and satisfies key physical constraints through a specifically designed NN architecture. In this work, training is carried out in an unsupervised manner, i.e. using a loss function which enforces balance of linear momentum based only on full-field displacement and global force data, and no stress data. Hence, the approach is directly applicable to experimental measurements (see also the discussion on interpretable approaches based on sparse regression). [180] employs neural ordinary differential equations (ODEs), a form of polyconvex NNs, to develop DD material models that inherently satisfy the polyconvexity condition in elasticity, using the properties of ODEs to create monotonic functions that approximate the strain energy derivatives, thus effectively modeling highly nonlinear, anisotropic materials. [181] trains anisotropic NN hyperelastic models for monoclinic crystals in terms of both $\mathbf{P} - \mathbf{F}$ and $\mathbf{S} - \mathbf{E}$ relations. This study also incorporates energy and stress frame invariance terms into the loss function during training and discusses post hoc validation tests to test for material stability. [182] deploys NNs to learn the homogenized three-dimensional constitutive behavior of anisotropic hyperelastic cubic lattice metamaterials while ensuring objectivity and material symmetry in the problem formulation. Expanding on this approach, [183] embeds a parametric dependence in the NN formulation, which allows to capture intricate topological and material variations, thus refining the learned energy functions.

Many ML studies in the literature establish a direct mapping from the strain to the stress material response. This strain–stress approach is sometimes preferred over an energy formulation due to its more straightforward implementation and the common lack of energy data samples. [184] utilizes principal stress-stretch data in the training of elastic NN laws, which are found to approximately satisfy objectivity conditions. In contrast, [185] utilizes NNs to formulate CLs in terms of $\mathbf{S} - \mathbf{E}$ using a dataset derived from molecular dynamics, later validating the learned constitutive tangent modulus through numerical differentiation. In another study,

[186] NNs are trained to establish both the stress-strain $S - E$ and stiffness-strain $C^{SE} - E$ nonlinear relations for crystal structures under various symmetry conditions. [187] conducts a comparative analysis of NNs, with and without Sobolev constraints, and local approximate Gaussian process regression in mapping stress-strain $S - C$ and employ Latin hypercube sampling, c.f. section 3, to uniformly sample the space spanned by the deformation gradient.

Interpretable ML approaches. [188] models hyperelastic behavior for soft materials by identifying the energy function using spline regression and smoothing penalization, therefore producing an interpretable ML model. Notably, the authors use stability conditions in order to handle the noisy data often present in such materials. Also, [189] and [21] use the representation theorem of tensor-valued tensor functions, c.f. [190], which is used to write the stress as a linear combination of invariant-dependent coefficient functions and basis generators. This allows learning the mapping from the invariants of the deformation to the values of the coefficients and guarantees that the material response is frame indifferent. In this context, [191] studies and compares different formulations of the representation. In order to lower the number of required training samples [21] proposes a space-filling sampling algorithm in invariant space.

Recently, several efforts have been made to deduce analytical expressions for material models from data using either symbolic regression or sparse regression (see the discussion on terminology in Sect. 2.1.1). In the context of hyperelasticity, symbolic regression has been used since the early work by [192], see [193] for a more recent application. Even more recently, [22] proposes a symbolic regression method in which potential expressions for physically valid hyperelastic CLs are generated using regular tree grammars and model discovery is carried out by combining variational autoencoders and a covariance matrix adaptation evolutionary strategy.

Sparse regression in the context of material modeling on the other hand is a very recent research field. The first works in which sparse regression from a potentially large library of candidate material models is used to discover hyperelastic strain energy functions as symbolic mathematical expressions are [34], that develops a method denoted as EUCLID standing for Efficient Unsupervised Constitutive Law Identification and Discovery (see [35] for an overview), and [36], that develops a method coined Variational System Identification. Beside being interpretable approaches, these methods are also unsupervised. Instead of relying on labeled stress-strain data pairs, which are in reality only available under very simple loading conditions like uniaxial tension or simple torsion, the material model discovery is informed by experimentally measurable displacement and global reaction force data. The lack of stress labels is compensated for by applying a physics-motivated loss function based on the

conservation of linear momentum. This has the advantage that the learning process can be informed by real data instead of data generated from micro-scale simulations, which are only feasible if the microstructure of the material and models of all its material constituents are known. It is shown by [194] that the EUCLID framework can also be considered from a Bayesian perspective for discovering symbolic hyperelastic models with quantifiable uncertainties. A first experimental validation of a supervised version of EUCLID is provided in [195], where sparse regression is leveraged to discover hyperelastic strain energy functions for human brain tissue using data stemming from uniaxial tension and torsion tests of human brain tissue. In [22], the hand-crafting of the material model library required in sparse regression approaches is replaced by its automatic generation as the "language of hyperelastic material models", i.e. as the set of expressions generated by a context free grammar while accounting for physics constraints. [196] discusses the discovery and interpolation of hyperelastic models for the human brain tissue in a framework that selects model inputs and forms from a family of constitutive building blocks of classical models from the hyperelasticity literature through the optimization of an NN. Another approach for replacing the hand-crafting of libraries is suggested in [197] by enabling extreme sparsification of physics-augmented neural networks utilizing a smoothed L_0 regularization approach showcasing a combination of expressivity and interpretability in a range of experimental and synthetic test cases.

Model-free approaches. The strain-stress formulation is particularly appealing for model-free applications. For instance, in [198], a model-free approach is extended from [81] to finite-strain nonlinear elasticity. The physical constraints (the principle of virtual work) are enforced with Lagrange multipliers and the search problem is formulated in terms of $S - E$. This choice is made to automatically satisfy the objectivity conditions and ensure the symmetry of the stiffness matrix. Also, [199] studies well-posedness and existence of minimizers of model-free finite strain formulations and proposes a formulation of finite elasticity in terms of $P - F$. It provides the necessary equilibrium and compatibility constraints, defines conditions for material frame indifference in this approach, and delves into the concepts of convexity within the finite-strain model-free framework. Moreover, [83] employs a DD approach to model anisotropic nonlinear elasticity. The solution search is conducted in both strain-stress data and anisotropy orientation spaces based on manifold embedding and a convexity-preserving reconstruction scheme, called local convexity DD computing.

The model-free approach based on distance minimization [81] is extended to account for finite deformations by introducing appropriate metrics in (F, P) or (E, S) phase spaces, as discussed by [198] and [200]. To enhance the performance of the model-free solver in scenarios with

limited or noisy data, [83] and [201] incorporate an online locally linear embedding scheme, leveraging the advantages of local convex linear interpolation. Additionally, [202] develop an autoencoder framework to enhance local search efficiency and mitigate noise sensitivity within the model-free approach by discovering lower-dimensional embedding spaces. [84] introduces a global manifold learning approach employing invertible NNs, allowing for direct interpolation on the manifold and eliminating the need for local discrete searches. This concept is extended to incorporate an isometric (distance-preservation) constraint to maintain metric structures between ambient and embedding phase spaces [203]. This method introduces a geometrically inspired regularization technique within the classical autoencoder framework, facilitating noise reduction and interpolation on the data manifold.

5 DD Modeling for Path-Dependent CLs

This section is devoted to review the DD approaches to describe path-dependent materials. The characteristic feature of this class of materials is that the current stress depends on both the current strain and on the entire deformation history [204]. Therefore, the bijectivity of the CL is not fulfilled since a single state of deformation can pair with a possibly infinite number of stress states, leading to the so-called *one-to-many* strain–stress mapping. This is in contrast with the path-independent models, where each strain state corresponds to only one stress state, leading to a *one-to-one* mapping. The description of the path-dependent constitutive behavior is also often complicated by an irreversibility constraint of the (dissipative) processes characterizing the material response. One potential way of addressing the one-to-many mapping issue is to introduce in the CL so-called history variables, also known as internal variables (since they are not directly observable), whose definition should ensure a one-to-one mapping, i.e. each strain–stress pair corresponds to a unique set of internal variables.

In the following we specifically focus our attention on hypo- and elastoplasticity, viscoelasticity, damage, fracture, fatigue and, finally, on their coupling with other physical processes, i.e. within a multiphysics representation of the constitutive behavior. As in Sect. 4, we begin each section with a brief summary of the main features of the specific material behavior and then we review the most significant contributions in the context of DD approaches.

5.1 Plasticity

The perhaps most widely used history-dependent material modeling framework is plasticity. Over the years, different theories of plasticity have been developed, such as

hypoplasticity [205], elastoplasticity [206], hyperplasticity [207] or generalized plasticity [208]. Small-strain elastoplasticity makes a distinction between elastic and inelastic components of the constitutive response by an additive decomposition of the strain tensor into elastic and plastic strains and introduces a notion of yield surface $f(\sigma, \bullet)$, which is dependent on a stress measure σ and on a number of thermodynamic forces used to define the hardening behavior. The yield function defines the plastically admissible domain as the set of stresses for which $f(\sigma, \bullet) \leq 0$. The yield function is classically decomposed as $f = \bar{\sigma}(\sigma, \bullet) - k(\bullet)$ into a scalar *equivalent stress* $\bar{\sigma}$, that can be e.g. dependent on a kinematic hardening tensor, and a resistance k , often a function of a *drag stress* which is used to describe isotropic hardening. On the other hand hypoplastic models, which are commonly applied in geomechanics, often do not separate the strain into reversible and irreversible parts and do not employ yield functions to characterize the onset of yielding. Here, the information about the past is all concentrated in the current stress. Mirroring this distinction, as follows we classify DD constitutive models for plasticity into models utilizing hypoplastic and elastoplastic ideas.

5.1.1 DD Plasticity Modeled After Hypoplasticity

We start with the hypoplastic case which, for isotropic materials, aims to predict the current stress or strain values from a time-discrete version of the CL

$$\mathcal{M} : \{\sigma, \dot{\epsilon}, \bullet\} \mapsto \dot{\sigma} \quad (10)$$

where $\dot{\sigma}$ is the Jaumann stress rate and \mathcal{M} is a tensorial function [209].

Uninterpretable ML approaches. Since the early works on ML plasticity modeling were performed in the geomechanics community, they have close similarities to the hypoplastic theory. These include the first and perhaps most influential ML constitutive models for history-dependent problems published in the early 1990s by the group of J. Ghaboussi. The first study employs a standard NN with sigmoidal activation functions to fit monotonic loading data from plain concrete under a biaxial state of stress at small strains [210]. By treating the mapping problem in a quasi-sequential manner, this initial study already introduces most of the major ideas that will be found in papers of the following years. Both stress- and strain-controlled models are developed. Both approaches in this setting require a network with six inputs and two outputs; e.g. in the stress-controlled approach, the six inputs are the two principal stresses, the two principal strains and two stress increments ($\Delta\sigma_1, \Delta\sigma_2$), whereas two strain increments ($\Delta\epsilon_1, \Delta\epsilon_2$) are given as outputs. The utilized NN architecture is visualized in Fig. 3a. The authors only offer visual comparisons between experimental and approximated

responses on selected test loading paths, and the developed DD model (strain-controlled) is not tested in a finite element framework. The approach is extended to uniaxial cyclic loading in a follow-up paper [212]. Here the architecture of the network is (heuristically) changed by the authors to be able to distinguish loading and unloading curves by not only using the current point on the stress-strain curve as input but also utilizing the previous two points on the loading path, see Fig. 3b. More information on the networks, as well as the modeling ideas, can be found in [213]. The findings are reiterated in [214, 215].

A similar approach as in [210] (training history-dependence in a hypoplastic framework) for DD material modeling of sand is tested in [216], where an additional material parameter is considered as input. In the 1990 s, various authors reapply the same ideas to laminates [217], clays [218, 219], sand [220, 221], rocks [222, 223], soft soils [224] or concrete [225]. This approach is also studied and tested in a more rigorous way on different material behaviors in [226]. In contrast to these works, the study in [227] defines two NNs of the same architecture to model rock joint material behavior; one is used to deal with small-displacement behavior and the other one with large displacements (the value of 1.5 mm separating the two ranges is arbitrarily chosen by the authors). This approach aims to distribute the nonlinearities of the material model between the two networks (since the networks at this point cannot be too deep due to the available computational resources). Other than NNs, SVR-based plastic modeling of geomaterials is investigated in [228, 229] while polynomial regression models are studied in [230–232]. All of these follow the same idea of using time-discrete current strain and stress values as inputs and the predicted stress as an output to their formulations. An adaptive approach to determine the ideal network architecture of the NNs for path-dependent material modeling is proposed in [213, 233–235]. The number of nodes and the necessary number of history inputs, i.e. $([\epsilon_j, \sigma_j], [\epsilon_{j-1}, \sigma_{j-1}], \dots)$, can be trained adaptively. The authors use the term "nested" NN for this process, which is later used for modeling the

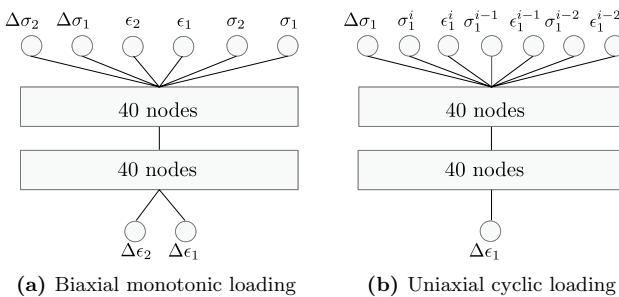


Fig. 3 First NN architectures based on quasi-sequential data used for stress-controlled path-dependent modeling of concrete inspired by [211]

behavior of soils [236]. This technique is also employed to obtain path-dependent ML CLs from indirect data, such as global load-deflection responses or displacement fields. Here, the training algorithm of the ML tool is built around a numerical framework such as the finite element method (FEM)[237]. Other works employ similar approaches in later years [238–243]. However, approaches that adaptively change the architecture of NNs seem to have been no longer actively pursued in the context of material modeling.

The approaches reviewed so far are based on using the history of observable quantities such as stress and strain as inputs to a ML model in order to predict the next stress state. However, due to the path-dependence of the stress evolution, these methods likely fail when studying longer loading curves with complex loading patterns or when extrapolating outside of the domain of the training data. To address this issue, more advanced ML approaches introduce internal variables as additional inputs. In particular, [244, 245] include for the first time internal variables in a ML stress evolution law. They focus on a viscoplastic model with kinematic and isotropic hardening and utilize an NN, with the current strain, the internal variables ξ and the current stress as input, and the current rates of the viscoplastic strain $\dot{\epsilon}_{vp}$ and of the internal variables as outputs. For this purpose, the authors define an implicit constitutive model in the state space of the form

$$\mathcal{M} : \{\epsilon_{vp}, \xi, \sigma\} \mapsto \{\dot{\epsilon}_{vp}, \dot{\xi}\} \quad (11)$$

Assuming that the initial conditions are known, a forward Euler scheme is employed to update the variables from time step n to $n + 1$ with

$$\begin{aligned} \epsilon_{vp,n+1} &= \epsilon_{vp,n} + \dot{\epsilon}_{vp,n} \Delta t \\ \xi_{n+1} &= \xi_n + \dot{\xi} \Delta t. \end{aligned} \quad (12)$$

The stress update is obtained in a similar manner. In this approach, the internal variables have to be explicitly known. If this is not the case, the authors describe a procedure to obtain information about the internal variables from cyclic loading curves and apply this methodology in an example.

Other work dealing with the explicit definition of the internal variables includes [246, 247] which rely on data from repeating unit cells. In [248], an energy-based internal variable and an internal variable that implies the direction for the next time- or load-step along the equilibrium path are used as additional inputs to an NN. Similar approaches that make use of known or implicitly obtained internal variables as inputs to NNs that describe the stress evolution law can be found later in [249], where the accumulated absolute strain is used as an additional input. Nowadays, the use of internal variables as additional inputs to guide the training process is becoming more and more established as DD modeling of path-dependent materials shifts more and more away from

experimental data towards data obtained from numerical simulations. One of the problems associated with using plain NN approaches that adopt previous time-discrete strains and stresses as input and current stress as output is their dependence on the strain increment. In order to avoid this problem, [250] proposes a two-stage algorithm based on the total strains for modeling the homogenized response of a lower-scale problem. In the first step, a SVM is used to classify whether the current loading step (described by total strain and last equilibrium value of the history variable) results in an update of the history variable, which is comparable to a return mapping procedure. Based on this information, in the second step, the stresses are obtained from a NN.

The aforementioned approaches suffer of reliability and robustness issues when the behavior need to be extrapolated outside of the range of data used during the training. A possibility to relieve this drawback is to introduce physical consistency into the training process, a technique that can also reduce the data hunger of the method. To the best of the authors' knowledge, [251] were the first to include a physical concept in the training process of an NN used for DD path-dependent material modeling; they enforce isotropy of their CL in a soft manner by augmenting the training data-set by rotated counterparts of existing data. The concept of physics-informed ML in the incremental modeling of path-dependent materials using NNs is recently receiving increasing attention, with two recent works leading the way. First of all, [252] propose using the Cholesky factor L of the tangent stiffness matrix as the NN output instead of the stress. The stress update is obtained by

$$\sigma_{n+1} = L(\epsilon_{n+1}, \epsilon_n, \sigma_n) L(\epsilon_{n+1}, \epsilon_n, \sigma_n)^T (\epsilon_{n+1} - \epsilon_n) + \sigma_n. \quad (13)$$

The positive aspect of this formulation is that it implicitly enforces the tangent stiffness matrix to be symmetric semidefinite. Secondly, thermodynamics-based NNs (TANNs) encode the underlying thermodynamic principles directly into the networks [253, 254]. This is achieved by relying on

a dual potential formulation, consisting of the Helmholtz free energy density F and the dissipation rate potential D , from which the variables of interest can be derived in a thermodynamically consistent form. In the isothermal case, the framework consists of two NNs, denoted as $s\text{NN}_\xi$ and $s\text{NN}_F$. The former is used to predict the increment of the internal variables

$$\Delta\xi = s\text{NN}_\xi(\epsilon_{n+1}, \Delta\epsilon_n, \sigma_n, \xi_n) \quad (14)$$

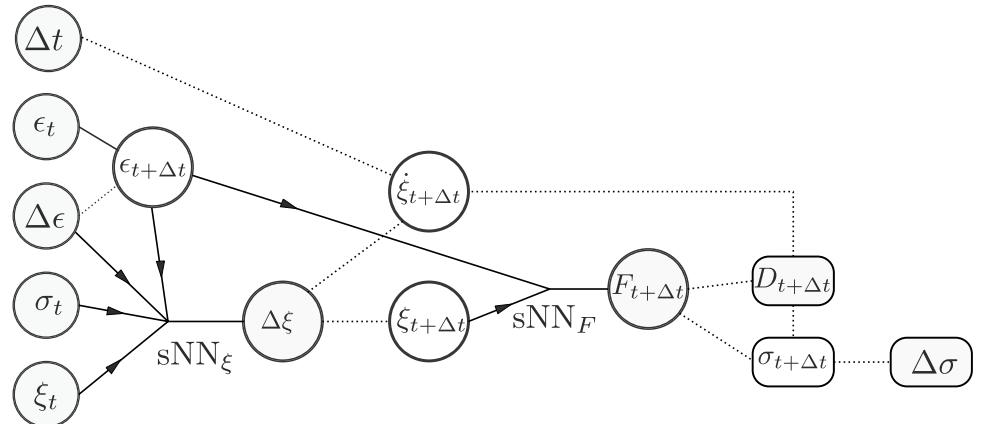
The other network predicts the value of the Helmholtz free energy density of the next time step

$$F_{n+1} = s\text{NN}_F(\epsilon_{n+1}, \xi_{n+1}) \quad (15)$$

which can be used to obtain the stress update with $\sigma_{n+1} = \frac{\partial F_{n+1}}{\partial \epsilon_{n+1}}$. The TANN architecture is schematized in Fig. 4. As a result of the (implicit) thermodynamic consistency, the task of identifying the underlying pattern of thermodynamic laws no longer needs to be performed by the ML tool.

A possibility to deal with path-dependence without explicitly introducing internal variables is relying on ML architectures that embed and describe internal states. In this context, RNNs have been investigated in the literature for plastic material modeling. The study in [255] is the first to apply sequential NNs, specifically the Jordan network [256], in order to learn the path-dependent stress-strain curve of sand under consideration of a constant strain increment in a triaxial state of stress. In contrast to the previous approaches, the internal variables are here implicitly stored in the NN. Based on investigations of [255] this type of NN offers better results than simple NNs for the application at hand. For this reason, they utilize a network with seven inputs, including the current stress and strain rates, while the stress state resulting from the next strain increment defines two network outputs. Some critical points of this paper are further discussed in [257], including the issue that the network is only applicable to a specific strain rate. The same network

Fig. 4 TANNs proposed by [253]



architecture is later used by [258] to model clay. The same authors [259, 260] later improve upon this approach to build a ML model for sand and gravel by (similarly to [227]) dividing the input space of a feature into two parts, in order to train two different sequential NNs for the same dataset. Instead of using Jordan networks, the studies in [261, 262] employ Elman networks [263], i.e. another simple RNN, for simulating and predicting the shear behavior of two different soils. Simple RNN architectures are also used to model unsaturated soils [264], clays [265] or sand [266]. In [267] a sequential NN is employed for the three-dimensional stress-strain relationship of sand under monotonic loading. Different mapping techniques including quasi-sequential mapping, sequential mapping, function fragmentation, and function labeling for history-dependent material datasets are compared in [268] and [269], where a hybrid quasi-sequential approach (relying on function labeling) is found to perform most proficiently. Later on, the author also tests time-delay NNs for history-dependent datasets [270].

Recently, other types of RNNs such as LSTMs and GRUs have received more attention [99, 108–111, 271–275] because they are able to be trained more reliably by avoiding the vanishing/exploding gradients problems that are associated with the earlier version of RNNs. Even though these general DD frameworks for path-dependent materials are conceptually similar to the early approach proposed by [255], they are now predominantly used for three-dimensional applications and also include other parametrizations as input. E.g. the study in [107] describes the sequential nature of plasticity using a GRU which takes the history of spatially averaged strains, material properties for each microstructural phase and the current time as input and maps them to the predicted spatially averaged stress. Other work includes [276], where an LSTM takes the strain history over a sequence of times as a vectorial input as well as the latent space data from three-dimensional crystallographic orientation images using a CNN. The LSTM is then used to predict the stress evolution. This idea is later refined by the same authors [277] by relying on convolutional LSTMs to resolve the stress evolution also spatially. In order to reduce the reliance on user-chosen hyperparameters, [278] proposes an approach based on RL that finds the optimal hyperparameter settings (including the network architecture) of RNNs for modeling path-dependent materials. Lastly, [279] defines an adversarial training scheme based on input perturbations that increases the prediction robustness of models trained with RNNs (GRU specifically). A general problem associated with RNNs (such as LSTMs or GRU) in the training of path-dependent plastic material behavior is the dependence of these methods on the size of the increment. To overcome this challenge, [280] proposes a new RNN architecture that enforces self-consistency, i.e. their predictions converge when the increment size is decreased. A different approach

is proposed in [281], which treats the material model prediction as a mapping between two function spaces without the need for time discretization.

As a remark, we could not find any ML path-dependent material modeling approach based on RNNs that includes physical concepts in the training procedure. This is definitely an area where more work and development is necessary. The work that comes the closest is the one in [282], where a neural ODE framework is employed to model the stress evolution. In contrast to RNNs, a neural ODE incorporates time step scaling of the dynamics. Using this architecture as an internal state variable model and relying on representation theory, the authors are able to build a ML model that obeys physical principles, such as frame invariance and the second law of thermodynamics.

Model-free approaches. The framework based on the model-free paradigm [81] is more complicated when dealing with path-dependent material behavior. In [283], the material dataset is enlarged with internal variables. In [120] an energy-based parametrization is proposed that augments the phase space with the free energy and the dissipation, thereby enforcing thermodynamic consistency at all times. Similar approaches include [166, 284]. A different model-free idea is studied in [285, 286], where the three-dimensional stress/strain state is projected onto uniaxial tension/compression data.

5.1.2 DD Plasticity Modeled After Elastoplasticity

Elastoplasticity, in contrast to hypoplasticity, is characterized by modularity due to its clear distinction between elastic and plastic constitutive responses, with historically established assumptions for each of these components. In classical modeling, elastoplasticity is based on a formulation of the elastic response, a yield function description, an assumption on the direction of the plastic flow, and phenomenological models for the hardening behavior. DD elastoplastic modeling exploits this modularity by choosing each of these components (or subcomponents) to be either represented by a DD formulation or a classical phenomenological model. The general framework is summarized in Fig. 5. In contrast to the DD hypoplastic models, described earlier, these models have both advantages and disadvantages. Some of the advantages include

- More interpretability. The input–output mapping of DD models is potentially easier to understand since each model is used as a representation scheme for a specific subproblem of the elastoplastic formulation.
- Less data is needed. DD submodels can be chosen in areas where more data are available, while phenomenological models can represent the remaining components. This allows to reduce the reliance on big data.

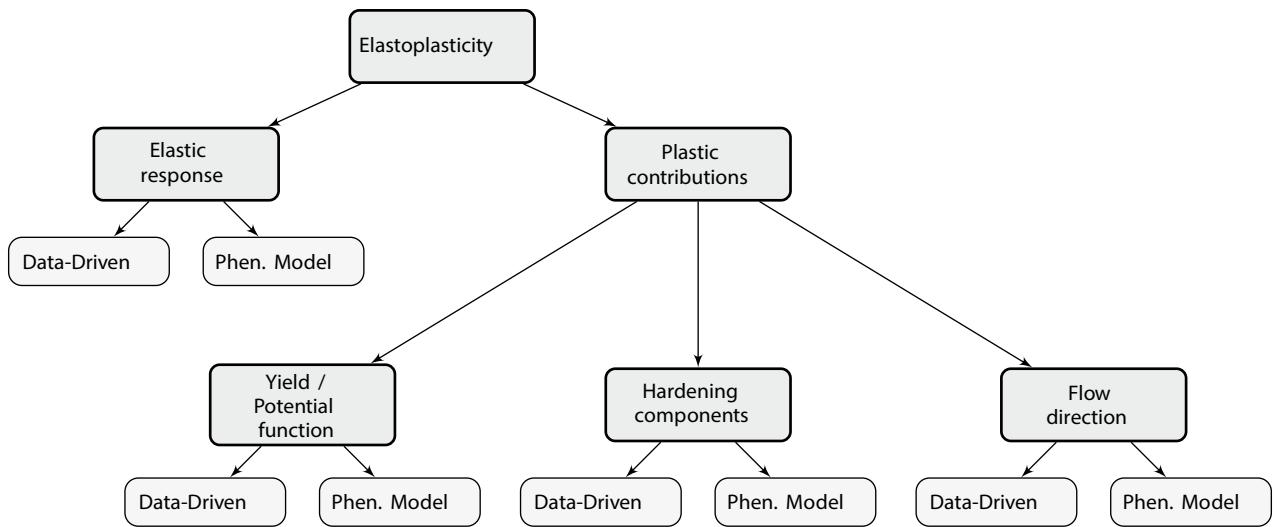


Fig. 5 Modular elastoplastic material modeling. The initial yielding can be fully DD or can separately include the equivalent stress measure and the yield stress. Hardening components can e.g. include the deformation resistance or some form of hardening moduli. See [287] and [288]

- Shift to one-to-one mappings. DD hypoplastic models intrinsically aim to find a consistent mapping for the whole path-dependent stress evolution. This leads to a one-to-many mapping problem which is solved e.g. with RNNs. On the other hand, DD elastoplastic models can only use ML models for components that can be modeled by simple one-to-one mappings, like yield function representations which can be decoupled from the time integration. Of course, this requires the access to data for these specific subproblems.
- Simpler enforcement of thermodynamic consistency. Since the majority of classical elastoplastic models were designed to be thermodynamically consistent, replacing single components of these models with specialized DD solutions allows for easier enforcement of physical constraints compared to ML models for plasticity modeled after hypoplasticity.

On the other hand, this modeling framework also has a disadvantage:

- More constraints and assumptions. DD hypoplastic models as presented before are based on a very limited range of assumptions (such as the choice of the internal variables) and are therefore models that can discover the material behavior that underlies the training data. On the other hand, elastoplastic modeling needs to make more assumptions (e.g. the split between elastic and plastic components) and is therefore more restrictive in its ability to discover unknown physical processes.

Uninterpretable ML approaches. A number of ML approaches have been proposed that are modeled after

elastoplasticity. In [289] both back stress and drag stress are incrementally updated using independent NNs to represent complex combined hardening laws under uniaxial cyclic loading, while the elastic law and the yield function are represented by traditional formulations. [290] proposes an ML plasticity model that relies on the representation theorem of scalar-valued functions. By exploiting the modularity assumption, the authors define an NN for the elastic response and a separate NN for the flow rule, both of which can be treated as one-to-one maps. This allows for physical concepts such as material frame indifference and symmetry conditions to be enforced implicitly. Another example is given by [291], where established phenomenological models of the elastic response, the yield function, the plastic flow direction, and the hardening modulus are combined to find the best fit of a dataset through RL. In [292], a hybrid formulation is proposed where the plastic flow and the back stress evolution are represented by one NN, while the yield function is given by a phenomenological model. [293] assumes that the elastic response is given by a linear law, whereas a NN takes the equivalent stress as an input to obtain the updated stress of the plastic region. This allows for a smooth transition between elastic and plastic behaviors. In [294], the linear elastic law and equivalent stress formulation of the yield function are phenomenologically assumed, while the deformation resistance and the dilation angle between the deviatoric and spherical parts of the normalized direction of flow are trained by two separate NNs. [295] also splits the ML model into elastic and plastic components; the elastic trial stress is obtained from an NN which, if non-admissible, is mapped back using an RNN formulation. An objective

loss function that includes frame indifference concepts is used for training. [296] replaces the general return mapping framework with an NN that updates the predicted trial stresses. In a series of recent papers [100, 287, 297], the elastic law and the yield function evolution are treated as separately trainable ML models. An NN-based yield function is trained using a level-set hardening framework whose evolution is dependent on an internal variable. Other work concentrates on just finding a ML representation of (parameterized) initial yield functions [298–302], whereas the remaining components are given by phenomenological models. [303] uses input convex NNs to enforce the ML yield function representation to be convex. Hybrid frameworks that locally improve phenomenological yield surfaces with ML models are investigated in [304, 305], with the latter enforcing physical constraints. Different subproblems of elastoplasticity may use NN representations, such as the deformation resistance [306–316] or the isotropic hardening modulus [317]. Other works employ SVR [318, 319] in the same context.

Interpretable ML approaches Lately, interest is growing in deducing symbolic material models from data using symbolic regression and sparse regression. The benefit of symbolic regression and sparse regression is that the models are expressed by short mathematical expressions that only involve a limited number of terms. This increases their computational efficiency, physical interpretability, communicability, and calibratability. In the context of elastoplasticity, this idea is followed by [299, 320, 321] who use symbolic regression and by [322] who apply the sparse-regression-based EUCLID framework to discover symbolic expressions for the yield function and the hardening behavior from reaction force and full-field displacement data, thus avoiding the use of labeled stress–strain data pairs. In a similar fashion, but bypassing the development of hand-crafted libraries, [197] sparsify physics-augmented neural networks that are developed in a modular fashion, to obtain compact expressions related to yield function but also to isotropic and kinematic hardening. In [323], EUCLID is extended to pressure-sensitive and non-associated plasticity. In [324], the method is further generalized to the framework of generalized standard materials, i.e., a general framework which encompasses elasticity, viscoelasticity, plasticity and viscoplasticity.

Model-free approaches In section 5.1.1, we have introduced approaches that rely on the so-called model-free approach to model plastic behavior directly on data of a given dataset more akin to the hypoplastic modeling approach. A different model-free elastoplastic-like technique is proposed in [325], where transition rules between elastic and plastic responses are introduced which map to different subsets of the data.

5.1.3 Summary

We conclude this section with the summary table in Fig. 6, which attempts to categorize some of the reviewed approaches based on the amount of data they need in input and on the number and type of constraints they are designed to satisfy. In general, a decreasing resort to constitutive modeling constraints is expected to correlate to an increasing need for data.

5.2 Viscoelasticity

One of the traditionally most challenging problems for constitutive modeling is that of viscoelasticity. Several mechanisms can be involved in the micromechanical processes that govern the macroscopic viscoelastic response, and correspondingly many characteristic timescales may have to be considered. Linear viscoelastic models are used in the regime of small deformations and are often represented through a combination of linear springs and dashpots, such as in the Maxwell, Kelvin-Voigt, Zener, and Generalized Maxwell models. Their nonlinear counterparts can account for nonlinear elastic contributions (such as those encountered in the context of hyperelasticity) and also nonlinear evolution equations. In general, and similar to what was discussed in the context of elastoplasticity, viscoelasticity can be cast in term of external (or observable) and internal variables. We can express the update of the stress σ and the internal variables ξ at time t_{n+1} as

$$\mathcal{M} : \{\varepsilon_n, \xi_n\} \mapsto \{\sigma_{n+1}, \xi_{n+1}\} \quad (16)$$

which, in contrast to plasticity, are not subject to any additional constraints. This makes the form of the equations at hand similar in structure to the problem that an RNN is designed to replicate. Namely, as the RNN works with sequential data, the hidden state vector a_{n+1} at time t_{n+1} depends on the value of the hidden state vector a_n at time t_n , whereas the output vector y_{n+1} also depends on a_n , indicative of the history dependence that is being captured. This natural similarity has led to a lot of ML-enabled constitutive modeling approaches in the context of viscoelasticity to follow this general approach. The bulk of the literature can be separated in two main categories, those that focus on parameter estimation for known viscoelastic constitutive models, and others that utilize different variants of NN architectures for problems of different complexity. We begin with the latter.

Uninterpretable ML approaches. In a rather simple setting, but aiming to obtain CLs that generalize, the authors in [327] utilize a simple NN architecture to obtain what they refer to as a neural constitutive model for nonlinear

Fig. 6 Quantitative categorization of DD modeling of (hypo- and elasto-) plastic material behavior. More available data correlates to less need for constitutive modeling constraints

The table illustrates the quantitative categorization of DD modeling of material behavior, based on the availability of data (Y-axis) and the number of constraints (X-axis).

Y-axis (Data Availability):

- Top row: Phenomenological modeling, NLK models [326]
- Second row: EUCLID [34, 322], MAP123 [285]
- Third row: TANN [253], Vlassis et al. [100], Jones et al. [290]
- Fourth row: EUCLID [323]
- Fifth row: Vlassis et al. [297]
- Sixth row: Jones et al. [282]
- Bottom row: Furukawa et al. [245], PODFNN [249]
- Bottom row: Mozaffar et al. [107], Abueidda et al. [109]

X-axis (Constraints):

- Column 1: Model parameters (e.g., explicit internal variables, fixed model functional form, fixed flow direction, fixed hardening types)
- Column 2: Functional forms (e.g., explicit internal variables, fixed flow direction, fixed hardening types)
- Column 3: Functional forms and Hardening types (e.g., explicit internal variables, fixed flow direction)
- Column 4: Functional forms and Stress evolution law (e.g., explicit internal variables)
- Column 5: Functional forms and Hardening types (e.g., latent space internal variables, fixed flow direction)
- Column 6: Stress evolution law (e.g., thermodynamically constrained flow, latent space internal variables)
- Column 7: Stress evolution law (e.g., no thermodynamic constraints, explicit internal variables)
- Column 8: Stress evolution law (e.g., no thermodynamic constraints, latent space/implicit internal variables)

viscoelasticity. The framework is also designed around the question of data availability and it allows for learning of rather simplified scalar-valued CLs from 1D creep test data, focusing on the non-zero component of the stress tensor for

the particular test. The network takes temperature and initial stress level as additional inputs, aiming for further flexibility of the predictions. In a similar fashion, [328] consider elastic properties as additional inputs to the NN and aim to learn

nanoindentation creep responses under fixed loading conditions; however, these creep responses cannot be considered as full CLs that could e.g. be used in a FEM simulation. In the same year of [327] another paper [329], leveraging expertise in NN-based rate-dependent and rate-independent CLs, constructs a complete framework for a rate-dependent NN-based CL for viscoelasticity. Here, the stress and the strain are decomposed in volumetric and deviatoric components and the NN takes as inputs the current value of the strain and the values of stress and strain at the previous increment. The approach is shown to work satisfactorily and is also implemented in an Abaqus UMAT so that structural problems can be tested. In their training data generation, the authors discuss data augmentation based on symmetry to train tensor-valued maps, and also obtain training data from structural simulations.

Several papers focus on learning the response of 1D experiments with ML approaches in the context of viscoelasticity, but in their essence, they are simple extensions of [327]. E.g., in [330] radial-basis-function NNs are utilized to learn the 1D dynamic response of thermoplastic elastomers, where the specific network architecture is chosen to accelerate the training process. More recently, [331] focuses on the rate-dependent response of polypropylene; it utilizes simple NNs that take strain and temperature as input to learn the 1D large deformation response as a function of the loading strain rate. A robot-assisted testing system is deployed to generate large datasets, and Bayesian regularization is used to identify the network parameters. [332] also focuses on learning the creep response, in this case, optical glasses, through an array of ML tools. The authors find these tools beneficial in terms of extrapolation to extreme temperatures compared to traditional phenomenological approaches.

From a different starting point, but still focusing on the 1D case, [333] addresses the fundamentals of history-dependent responses. Extending previous work where they had developed a model with associative (short-term) and hereditary (long-term) memory, inspired by a combination of the Kelvin-Voigt and the Maxwell models, the authors develop a new constitutive model with associative and hereditary memory as a system of integrodifferential equations. They then show that this system of equations can be approximated by a simple NN. They train based on 1D experimental data and test on non-monotonic 1D loading paths. The most interesting contribution of this work lies in the inherent interpretability of the approach since a direct connection is established between the NN architecture and the system of integrodifferential equations at hand. An early model proposed in [271] focuses on learning the time response of rheological material models stemming from a fractional differential equation using an RNN. A partial RNN is chosen, where the signal flow occurs in the forward direction and the fading memory is realized by internal feedback connections.

More recently, but in a similar fashion, [163] learns the solution of a specific boundary value problem with two non-zero stress components and encodes their response with an RNN. The work in [334] learns a fuzzy representation of the stress-strain response with an RNN recovering a fractional viscoelastic model, and [335] utilizes the RNN-based fuzzy CLs in a fuzzy-FEM setting to solve structural problems. A more recent work in the context of computational homogenization [336] establishes that the homogenized constitutive response may be approximated by a RNN. As an additional feature compared to previous approaches, a set of internal variables discovered in the learning of the homogenization procedure is tracked as a function of the history of the strain. [337] extends the work of [21] to consider viscoelasticity (focusing on materials with limited memory) by employing tensor representation theorems for the deformation tensors and for the rate of these tensors. By performing the training based on limited experiments corresponding to specific deformation modes and by utilizing constrained GPR, the trained material laws are shown to proficiently generalize in the strain space and also with respect to the strain rate.

A multitude of works focus on ML or physics-informed (in the style of PINNs) solutions of PDEs involving the viscoelastic response of solids [338, 339] but this is not the focus of the review here. We only mention [340], which deploys PINNs to train viscoelastic NN-based CLs based on limited sensor data by formulating a PDE-constrained optimization problem. The NN-based viscoelastic CL embedded in the structural optimization problem is not fundamentally different from the one in [242].

Interpretable ML approaches. For applications of symbolic regression in the context of viscoelasticity we refer to [341] and more recently [342]. Further, the sparse-regression-based EUCLID framework, which utilizes a large library of phenomenological models and unlabeled data, is successfully used by [343] to identify linear viscoelastic CLs. The same concept is extended in [324] to the theory of generalized standard materials, which naturally includes viscoelastic material behavior.

The approaches that focus on parameter estimation are not covered in depth in this review. In [344], from thin circular plate bubble experiments with temperature effects, and reducing the response to 1D, the material constants are learned for the Christensen viscoelastic model. [345] aims to optimize the viscoelastic response through additives; based on experimental measurements, it calibrates the values of the loss and storage moduli utilizing several ML approaches. [346] fits the same parameters to compare to experimental data using a Kelvin-Voigt model. In the context of magnetorheological elastomers, [347] suggests the use of NNs and Extreme Learning Machines (feed-forward NNs with a single hidden layer) to map shear strain and magnetic field to storage and loss moduli. [348] uses GPR to fit the constants

of a complex viscoelastic model based on experimental data; even though the task is seemingly straightforward, uniqueness is not guaranteed as the test data are only uniaxial and monotonic.

Model-free approaches. In [283], the model-free approach proposed for plasticity is also adapted to viscoelasticity by relying on a differential representation of the material evolution history. The model is tested within a monodimensional state space and applied to the analysis of truss structures.

5.3 Damage and Fracture

The loss of integrity of a structural component can be modeled using different approaches depending on the process responsible for the material degradation. In the following, we distinguish between (i) diffuse damage, (ii) fracture, and (iii) fatigue. Damage describes a gradual deterioration of the material stiffness and/or strength without significant residual deformations. Fracture entails the formation and propagation of a crack, i.e. a discontinuity, which evolves following Irwin's or Griffith's criteria or extensions thereof. Note that the evolution of a crack in a continuous body can also be described using continuum damage models that allow for the localization of the damage parameter in bands with a limited but not vanishing thickness. Fatigue consists of the nucleation and propagation of a crack at subcritical load levels under repeated loading.

5.3.1 Diffuse Damage

The continuum damage mechanics approach pioneered by Kachanov [349] aims at describing the gradual deterioration of the structural integrity of a material point when subjected to some type of action (e.g., displacements, forces, temperature changes, or aging). The main idea behind this theory is that, at the macroscopic scale, the reduction of stiffness and strength related to the material deterioration can be condensed in a scalar or tensorial internal damage variable. Often the onset of damage takes place after an initial elastic regime, which is followed by a stress-softening phase leading to an elasto-damage constitutive behavior. Since it cannot be directly measured, the damage variable belongs to the category of internal variables and, in the most common case, is assumed to be a scalar. Therefore, the related CL can be written as

$$\mathcal{M}_\sigma : \{\boldsymbol{\varepsilon}, d\} \mapsto \boldsymbol{\sigma}, \quad (17)$$

where the damage variable d is governed by an often a priori postulated evolution law of the type

$$\mathcal{M}_d : \{\boldsymbol{\varepsilon}, \boldsymbol{q}\} \mapsto \dot{d}, \quad \text{subjected to } \dot{d} \geq 0, \quad (18)$$

Here \boldsymbol{q} is a vector collecting the set of mechanical and internal quantities governing damage evolution, while the non-negativity constraint is assumed to fulfill the second law of thermodynamics and takes the name of *irreversibility condition*. This condition is the major responsible for the history dependence of the constitutive models accounting for material damage and it constitutes one of the main difficulties in the definition of a proper DD approach. On the other hand, it is of primary importance since it allows for a physically sound description of the unloading/reloading branches in the material response.

In the following, we first provide an overview of the most relevant DD approaches including identification problems and constitutive modeling using NNs, and model-free approaches. Most studies deal with NNs where damage is often implicitly accounted for while defining the CL and not explicitly introduced as an internal variable. No interpretable approaches are yet available.

Uninterpretable ML approaches. The earliest ML approaches in continuum damage modeling are devoted to parameter identification of available models. Although this review does not focus specifically on them, we propose here a brief overview due to their relevance in developing DD approaches for damage. [350] identifies the parameters of the Gurson-Tvergaard-Needelman (GTN) model using a NN, which is trained using a set of load–displacement curves from FEM computations. The trained NN receives as input the experimental load–displacement curves and outputs the material parameters, thus surrogating the solution of an inverse problem. Aware of the poor extrapolation capabilities of the NN, the authors include in the cost function a penalty term to avoid extrapolation outside the range of the training data. Also, they point out that a reliable identification of the full set of GTN model parameters is not possible, hence, they restrict identification to a subset of the total parameters that need to be selected a priori depending on the material at hand. To extend the number of identified parameters, in [351] the same authors propose a different identification method involving an NN to surrogate the solution of the forward boundary value problem for a small punch test. The NN is trained using FEM computations; it receives as input the displacement and the GTN parameters and it outputs the applied force. The parameter identification for a new set of experimental load–displacement data is then performed by adopting a successive quadratic programming algorithm. Similar identification approaches are adopted with minor modifications in other studies, e.g. [352].

The second class of approaches deals with the definition of elasto-damaging CLs using different NN architectures. [353] surrogates the cohesive traction-separation law at the interface between concrete and steel using a NN trained through a set of FEM simulations. The NN takes as input the components of the interface separation vector and outputs

the traction components. Hence, damage is not explicitly introduced but only implicitly accounted for. Since the NN is not informed by any physical requirement, the authors enforce some basic features of the CL by partly substituting ad-hoc calibrated linear relationships to the NN predictions. These include a vanishing traction value for vanishing or very large separation values, i.e. close to the unloaded state and to complete decohesion. Also, the obtained model is not able to distinguish unloading/reloading states and is thus limited to locally monotonic loading histories. In [250] the same authors propose a ML constitutive model for the concrete bulk material to be used in a multiscale FEM framework. The approach involves the definition of a NN trained with micro-scale FEM analyses to surrogate the stress-strain law of the material, while an SVR algorithm with an exponential kernel is used to detect the unloading/reloading branches. The algorithm is formulated according to a criterion similar to a limit surface whose extension depends on a history variable defined as the maximum strain reached during the loading history of a point. To exclude any mesh dependence related to the underlying (implicit) local damage approach, the size of the finite elements at the macro-scale is used as an input in the NN. In turn, this calls for a training set that includes data for different sizes of the domain at the micro-scale. Although the formulation of the approach is general, the authors illustrate its performance only for 1D cases.

An ML-enhanced multiscale framework is also proposed by [293, 294] to model elastoplastic damaging foams. In [293], the authors surrogate the macroscopic stress-strain relationship for proportional loading with a NN trained using lower-scale FEM computations. The homogenized local tangent stiffness tensor is computed directly by derivation of the NN mapping and the cost function is complemented with a penalty term to prevent large differences in the order of magnitude of the neural weights. A different approach is explored in [294], where the definitions of homogenized limit surfaces, flow directions and stiffness deterioration due to the (implicit) damage are ascribed to three different NNs. In particular, the definition of a limit surface allows to distinguish between the dissipative and elastic unloading/reloading stages.

A multiscale framework to define the behavior of a damaging poroelastic material is presented in [273], where three scales (i.e., micro-, meso- and macro-) are accounted for. The authors use LSTM-type RNNs to upscale the material parameters between micro- and mesoscale and between meso- and macro-scale. The NNs are trained using discrete element and finite element computations for the micro- and mesoscale, respectively, while the analyses at the macro-scale are performed using the FEM. The material models at each scale are defined using directed graphs where the relations between the various parameters involved are known

as physical or empirical relationships. Also in this case the damage parameter is not explicitly defined but is implicitly considered in the CL. The adoption of RNNs makes the model intrinsically history-dependent, thus allowing for easy discrimination between different states, such as unloading/reloading branches. Another addressed point is the objectivity of the material response given by the RNN, which is not satisfied a priori. The authors propose a method to achieve material objectivity based on a spectral representation of the training data, which effectively reduces the deviations in stresses and energy between different observer frames when evaluating the same system.

Two different NNs are adopted in [354] to define the macroscopic CL of a fiber/matrix composite material. The first NN is used to surrogate the stress-strain relationship of the material; also in this case the damage variable is not explicitly defined and the model does not include any unloading/reloading criterion. The second network is used to identify if the damage takes place in the fibers or in the matrix without explicitly accounting for them in the simulation.

[355] uses NNs to surrogate the traction-separation law between adjacent grain boundaries at different temperatures. In particular, they adopt a standard NN to define the interface secant stiffness, which is then multiplied by the separation vector giving a ResNet-like CNN [356]. Differently from an RNN, the proposed architecture is unable to process temporal information and, hence, it can be used only for monotonic loading history. On the other hand, the major advantage of this approach compared to an RNN is that it drastically limits the amount of data needed for training. The authors illustrate the approach in 2D training of the NN by using a dataset of molecular dynamics simulations. Also, a procedure to optimize the NN architecture is proposed.

PINNs are adopted by Haghigat et al. [357] to surrogate a coupled damage/plasticity model. The introduction of physical and modeling constraints allows us to include in the PINN conditions such as damage irreversibility, vanishing stress for vanishing strain, and complete failure after a critical damage value, and to automatically detect elastic loading/unloading/reloading states. However, the approach requires a very large amount of training data to achieve reasonable accuracy.

Among the approaches that do not belong to the aforementioned categories, [358] proposes a supervised CL discovery approach exploiting directed graph theory to automatically generate different models. The authors test the proposed approach to obtain the traction-separation law of a cohesive interface with implicit damage for 2D problems. The approach requires the definition of the parameters that possibly govern the CL and a set of *rules* that the relationships between the various parameters must satisfy to create a valid CL. The generation of a model takes the name of *game of model generation* and each *move* during a game

connects two parameters with a relationship until the input parameters are connected with the outputs. The completed model is then evaluated by assigning a *score* based on the comparison between the model predictions and a set of material observations. Unlike in [273], here the relationships between the parameters are encoded by an RNN, which is trained using a set of discrete element analyses. A deep RL algorithm is used to iteratively improve the models game after game, based on the evaluation of the probability that a certain move (i.e., a connection between two parameters) is selected by a *gamer* at a certain point of the game along with its expected contribution to the final score. Probabilities of moves and expected score contributions are predicted using an NN enhanced by a Monte Carlo tree search algorithm. A new move is then selected so as to maximize the final model score. The major feature of this supervised approach is that, once the game and a dataset of material observations are defined, no operator intervention is needed since the algorithm is able to learn how to improve its predictive capabilities from the previous games played. Also, the adoption of an RNN ensures that the model history dependence is included.

Model-free approaches. A model-free approach is adopted in [359] to define the macroscopic behavior of granular materials subjected to different loading conditions. The material dataset, generated with the level-set discrete element method, includes also a parametrization that allows to describe the history dependence of the material behavior. The definition of the latter is critical to obtain meaningful results and the authors propose to use either the dissipated energy or a set of internal variables known to satisfactorily describe the microstructural arrangement of a granular material. Both approaches give similar results but the method to obtain the material dataset must be selected with particular care since it must allow for the definition of the desired quantities.

5.3.2 Fracture

The propagation of a crack inside a solid can be described by means of different fracture mechanics models depending on whether the material exhibits a linear or nonlinear elastic or elastoplastic behavior. Here we will focus only on linear-elastic fracture mechanics approaches since, to date, the available literature about DD methods deals only with this class of behavior. Hence, the available theories revolve around Irwin's or Griffith's criteria and extensions thereof. Hence, they call for the definition of a stress intensity factor (SIF) or of an energy release rate that are functions of the current size of the crack a . Also in this case the fulfillment of the second law of thermodynamics calls for the introduction of an irreversibility condition on the crack size, namely $\dot{a} \geq 0$.

Uninterpretable ML approaches. One of the earliest approaches is reported in [360, 361]. Here the authors use an NN to solve problems involving a crack contained in a linear elastic domain, considering or not the unilateral constraint given by crack face contact. The definition of the weights and of the activation functions is inspired by the variational principle of energy minimization; the outputs are taken as the displacements, the weights carry the information related to the elastic properties of the material and the cost function is the elastic energy of the system. Thus, this can be considered a prototype of the physics-informed DD approach. The approach does not account for any propagation criterion and is used to solve both direct and inverse problems. In forward problems, the kinematic and static fields are computed for a given external load, while the weights of the NN are defined using FEM computations. The authors observe a faster convergence of the constrained problem compared to its unconstrained counterpart. For the constrained case, the computational cost is lower than with FEM with singular elements, while for the unconstrained case, the opposite is true. In inverse (identification) problems, a set of displacements is supplied to the algorithm and the elastic properties of the material (i.e. the weights) are obtained following a backpropagation solution scheme.

Different contributions deal with the computation of the SIF for different geometries and boundary conditions (see, e.g., [348, 362]). In particular, [362] proposes a NN to compute the SIF for a micro-cantilever beam. Training is performed using a set of FEM computations that is adaptively enriched through the targeted addition of sampling points within the portion of the state space where less accuracy is expected. The accuracy is estimated pointwise as the maximum deviation within thousands of trained NNs and their average prediction. Although the method exploits the linearity between load and SIF to achieve a more general result, the results depend on the geometry considered during training. The same applies to [348], where the critical SIF is obtained for a Brazilian test. Here the authors also explore the possibility of using ML approaches different than NNs, such as decision trees, random forest regression, and extra regression trees, concluding that the latter two approaches are more efficient than NNs.

An effective method to describe crack nucleation and propagation using a damage parameter that localizes in crack-like narrow bands is the phase-field approach to fracture [363]. In this context, ML-related work has concentrated on the solution of the governing PDEs obtained from the minimization of the phase-field energy functional. [364] uses NNs to surrogate the solution of these PDEs. Two different NNs are trained, one for the linear elastic bulk material and the second for the evolution of the damage variable, and they are embedded into a FEM code either together or alone. The solution of phase-field fracture problems with an

approach known as the deep Ritz method (DRM) [365] is pursued in some recent studies [366–369]. Unlike PINNs, for problems in which the governing PDEs stem from the minimization of an energy functional, the DRM directly minimizes the energy functional instead of the PDE residual. The approach in [369] solves examples of crack initiation, propagation, kinking, branching, and coalescence within one single numerical setup. An operator learning approach [20], namely variational DeepONets, has also been applied to predict the crack path in a problem involving crack propagation in quasi-brittle materials [370]. The potential of all these approaches is not to be applied to the solution of a single boundary value problem, but rather to learn solutions to parametric phase-field fracture models. In this setting, NNs can be trained on a few realizations of the parameter space and results can be inferred online for all other realizations, leveraging the true potential of the DRM and of neural operators.

An SVR approach based on a novel kernel function is proposed in [371] to surrogate the solution of a phase-field problem accounting for the stochastic distribution of the material and geometry parameters. The pool of input parameters includes the fracture and elastic properties of the material and the point of application of the load, while the geometry is considered deterministic. The SVR training is performed through Monte Carlo sampling, using FEM results where the material and geometry parameters are sampled from a given statistical distribution and is enhanced through a clustering technique.

Model-free approaches. A model-free approach based on variational principles is proposed in [372], where the fracture-related constitutive behavior is encoded into a dataset of material observations including crack position a and dissipated energy per unit surface (i.e., the fracture toughness). The constraint set for the admissible state space involves conditions encoding either a global or a local minimization of the total energy of the system subjected to a crack irreversibility condition. The authors propose and discuss different metrics defining the generalized distance and investigate the effect of noise in the material dataset on the final results. In [373] the approach is extended to rate-dependent fracture mechanics. The authors show that some classical constitutive assumptions, such as irreversibility or monotonicity, are redundant in a model-free setting since they are implicitly encoded in the material dataset. Note that both cases are defined and tested for setups for which an analytical expression of the energy release rate is available.

5.3.3 Fatigue

Classic linear elastic fracture mechanics states that crack propagation is triggered if a relevant quantity (either the SIF or the energy release rate) reaches a certain critical value. However, the experimental evidence demonstrates that, even

below the critical value, a crack can still propagate but to an extent that becomes measurable only after the application of several load cycles, giving rise to fatigue crack propagation.

To date, fatigue is still typically described using empirical laws calibrated on wide experimental datasets, i.e. in an inherently DD fashion. The most prominent mathematical description of the fatigue behavior that can be interpreted as a CL is due to Paris [374] and reads

$$\frac{da}{dN} = C\Delta K^m, \quad (19)$$

where N is the cycle number, da/dN is the fatigue crack growth rate, ΔK is the SIF range spanned by the crack tip in a single load cycle, and C and m are two material parameters.

Uninterpretable ML approaches. In [375] the authors use an NN to obtain the fatigue crack growth rate and the remaining fatigue life given the applied load, the crack size, and additional information commonly used to estimate the fatigue life. This includes a pre-calibrated empirical function that should mimic the role of ΔK and some easily accessible experimental parameters, making this method more useful for diagnostics than for modeling.

[376] adopts a Bayesian network approach to predict the probability of activation of a certain slip plane and the associated fatigue crack growth rate in body-centric cubic polycrystal materials. The network is trained using crystal plasticity computations and experimental data.

Model-free approaches. [373] illustrates that the model-free approach for rate-dependent fracture mechanics can be conveniently adapted to fatigue if the time variable is substituted by the number of cycles N . In this case, the dataset of material observations is composed of crack growth rate-driving force pairs, which replace (19). This approach is able to automatically account for characteristic features of the fatigue behavior, such as the presence of a threshold for ΔK below which no fatigue effects are triggered.

5.4 Multiphysics

Many processes of engineering interest involve multiple physical mechanisms coupled with the mechanical behavior. In multiphysics CLs, mechanical quantities such as strains and stresses are insufficient to describe the material state; additional quantities related to the e.g. thermal, hydraulic, or chemical responses may need to be additionally considered. The complexity of the experiments involving multiphysics processes makes the identification of the related CLs even more challenging than in solid mechanics. More specifically, identifying appropriate material descriptors and establishing their functional dependence on the material state in the multiphysics process while satisfying the laws of thermodynamics and accurately modeling experimental data is a complex endeavor. ML algorithms may offer a promising approach to

accelerate the discovery or surrogation of material models in such scenarios. To limit the scope of the following review to a manageable extent, we focus here on multiphysics problems relevant to geomechanics and do not cover multiphysics CLs arising e.g. in thermo-, electro-, or magnetomechanics. Examples of using DD models for the latter can be found in [377, 378].

The strain–stress state of the material in a multiphysics setting may be influenced by a variety of material descriptors, such as the void ratio in geomaterials. The behavior of these descriptors, as well as the strain and stress states themselves, may be influenced by other unknown fields, such as fluid pressure. Moreover, these material descriptors may interact with each other, and their causal relationships are not always clear. To review recent developments in multiphysics constitutive modeling, we can classify problems into three types: (1) problems in which the material descriptors and their causal relationships are known, but the functional form of the CL is unknown; (2) problems in which the material descriptors are known, but their causal relationships and functional form are unknown; and (3) problems in which the material descriptors, their causal relationships, and their functional form are all unknown.

In the first category, traditionally, models are handcrafted based on experimental or simulation data. Advances in computational multiscale methods such as FE² [379, 380] do not need any functional forms for the constitutive model, and they can be built based on first principles of thermodynamics to automatically avoid physical violations. Although these computational approaches are accurate, they usually suffer from high computational cost already in the mechanical context, and even more so in a multiphysics setting. This is because not only the iterations between two scales are required but also different physics dictate distinct temporal and/or spatial resolutions. In this regard, finding a functional form may seem to be a more computationally efficient approach, which additionally enjoys the advantages connected to interpretability.

5.4.1 Methods to Define the Functional Form of the CL

Uninterpretable ML approaches. [381] employs fully connected NNs to model macro-scale parameters for an isotropic poroelastic medium, without enforcing physics constraints in their surrogate modeling approach. In a related work [382], a similar framework is employed for the anisotropic finite-strain regime, where microstructure descriptors such as porosity and Poisson's ratio are used to learn macroscopic homogenized values. This approach bypasses the need for computationally expensive fine-scale simulations in concurrent multiscale frameworks. [109] develops a surrogate model using GRU and temporal CNNs, which predicts the homogenized stress tensor and the temperature based on

micro-scale specimen information and loading conditions. Their approach does not take into account thermodynamics rules. [383] utilizes surrogate constitutive models to simulate mass transfer in dual-porosity materials at the macro-scale, reducing the computational cost in classical multiscale simulations. They demonstrate the effectiveness of autoregressive NNs, which have the advantage of being easier to implement and train compared to RNNs. [384] investigates the use of surrogate modeling for unsaturated soils; complex retention curves with different wet and dry branches and anisotropic permeability models are determined in a purely DD manner using fine-scale simulations. Additionally, the study proposes an automated framework based on RL to identify the best set of hyperparameters.

Overall, surrogate modeling approaches based on NNs or other types of differentiable approximators (e.g., Gaussian processes) have the advantage of being scalable for high-dimensional problems with different types of algebraic and ODE constraints.

Interpretable ML approaches. In multiphysics CLs, the material behavior can be expressed by algebraic equations, in simple cases, or in general, by a system of coupled high-dimensional ODEs or PDEs. Although symbolic regression methods find the explicit form of these equations, they become more and more computationally inefficient when the number of material descriptors grows, or the desired functional form is not algebraic, i.e., it is given by ODEs or PDEs [385]. Moreover, incorporating physics (e.g. thermodynamic) constraints in their discrete search algorithm may further increase the computational cost. Recently, methods based on RL have been developed to improve the computational burden of the symbolic regression search space [386]. Based on the current challenges regarding symbolic regression, sparse regression appears as a promising option for interpretable ML approaches; surprisingly, to the best of our knowledge, it has received very little attention so far. A first approach in this direction has been suggested by [387], who propose a polyconvex neural network-based thermoelastic framework that achieves interpretability by sparsifying the number of parameters in large neural networks down to only a few remaining ones.

Model-free approaches. [388] extends the non-parametric distance-minimization paradigm (i.e. the model-free approach) to poroelasticity problems. It is assumed that the material descriptors for a coupled poroelastic problem are known (based on the effective stress principle and Darcy's law), but no functional forms are assumed for the relations between strain, effective stress, pore pressure, and Darcy's velocity. This approach minimizes assumptions on the modeling part. However, the model-free approach is known to be data-hungry [83, 84], and this feature becomes even more prominent in a multiphysics setting. To address this issue, the authors introduce variationally consistent multi-fidelity

formulations where a model-based method for one field is hybridized by an entirely DD assumption for the other field based on data availability and quality. Due to the discrete nature of the distance-minimization problem, the brute-force approach for searching in the data (at inference) becomes inefficient for a multiphysics problem. They develop an efficient data structure based on kd-trees to reduce the computational burden exponentially.

5.4.2 Methods to Define the Causal Relationship Between Descriptors and Functional form of the CL

In this category, the problem is more challenging since not only the functional form is unknown, but also it is not known how the material descriptors impact each other. The causality between material descriptors may help to find more generalizable functional forms. More importantly, it helps explain the underlying physical process, which is crucial for a mechanics problem involving e.g. failure or fracture. [358] formulates such a problem as a directed acyclic graph (DAG) problem where the nodes are material descriptors, and the goal is to find edges that lead to an information flow best describing the input–output relationships. In this work, each edge uses an RNN model to incorporate any possible path-dependence which is common in plasticity and viscoelasticity problems. The approach does not include thermodynamics constraints which could be addressed by recent ideas from the PINN community [18, 100, 253]. Moreover, it is assumed that the DAG structure remains constant, which may not apply to other path-dependent problems.

5.4.3 Methods to Define the Material Descriptors, Their Causal Relationships, and the Functional form of the CL

The complexity further increases in the third category since the set of plausible material descriptors is not known *a priori*. Historically, mechanicians have encountered several such issues; e.g., the existence of the fabric tensor as an additional material descriptor for describing the plasticity of granular materials was not apparent based on classical theories and assumptions [389]. Therefore, it could be beneficial to design an intelligent system to find the optimal material descriptors based on the material state information. [390] proposes a probabilistic framework to find the plausible set of material descriptors with their associated confidence interval. They provide a systematic way to incorporate uncertainty in causal discovery and predictions. They study a granular system and show that their framework can recover some classical theories regarding the fabric tensor. One limitation of their work is the static assumption on the DAG structure which could be subjected to change at different loading states.

6 Evaluation, Verification, Validation and Their Challenges

6.1 Performance Metrics

To evaluate the performance of different DD approaches, a set of unbiased performance metrics is necessary. A major problem in DD for mechanics compared to other more established applications is the lack of standardized tests to evaluate performance and even an agreement on metrics to be used for that evaluation. Here, even though our goal is not to establish or solidify such consent, we try to guide the development of a framework around this task. To this end, we propose the following metrics:

- **Accuracy** – the quality of the predictions, within the training data, defined by a metric that measures discrepancy. This error is generally established by sampling errors between ground-truth data and predictions.
- **Precision** – the sensitivity of the predictions with respect to the amount of available data and their noise, the sampling method adopted and the distance between data points in the training set.
- **Physical/mechanical consistency** – the ability to fulfill basic physical/mechanical laws or principles with a general validity, namely not related to specific constitutive modeling choices. Examples are objectivity, material stability, energy balance, vanishing energy and stress associated with an undeformed state, and positive variation of the energy for increasing external loads.
- **Interpretability** – the amount of information that can be understood by modelers and decision makers from a calibrated DD constitutive model.
- **Generalization ability** – the ability of the approach to provide accurate predictions outside of the training set. This has as a prerequisite the efficient interpolation on unseen training data, but more importantly, it evaluates the accuracy of the model in extrapolation outside the training domain.
- **Cost/performance trade-off** – the time and computational cost required to complete the workflow for predictions. This includes the time and computational cost required to train, validate, test, and verify the models, the time required to run the code as well as the demand for data to ensure the required level of accuracy, precision, and generalization.
- **Robustness** – the sensitivity of the above list of metrics with respect to the amount of available data, noise in the data, the distance between the predicted loading path and the calibration data, and the repeatability and reproducibility of the performance metrics.

- **Stability** – the sensitivity of the above list of metrics to perturbations of the free parameters of the DD approach.

The information provided by the proposed metrics also helps to evaluate some specific characteristics of the obtained DD CL in relation to the required employment. A non-comprehensive list is provided as follows

- **Fidelity** – the level of accuracy, precision, and generalization achieved in performing a given task (e.g., optimize a component, predict the behavior of a material under given conditions, better understand the physics behind a studied phenomenon). Depending on the applications it can encompass or not the evaluation of the physical/mechanical consistency.
- **Confidence** – the precision that can be attained in reproducing the phenomenon of interest with a certain accuracy given a set of calibration data. This metric essentially quantifies the uncertainty related to the obtained CL.
- **Data hunger** – the amount of training data needed to reach the desired level of fidelity.
- **Numerical behavior** – possibility to integrate the obtained model in a given numerical framework, its associated cost/performance ratio, and convergence with respect to the spatial and, if relevant, temporal discretization.
- **Reliability/reproducibility** – the sensitivity of the metrics with respect to the source of the calibration data (e.g., different codes for the simulations or laboratories for the physical tests) and of the training code adopted.
- **Trustworthiness** – a combination of all the metrics relevant to the task at hand that give a synthetic global score of the performances of the DD CL.

Whenever adopting a DD approach to reproduce a CL it should be good practice to define which important metrics apply to the case/application at hand and how the operator intends to objectively measure them. Moreover, the definition of the performance metrics should assist in comparing either different approaches to model the same or similar phenomena (e.g., uninterpretable vs. interpretable approaches or comparison between different NNs) or the same approach but on different set of training data (e.g., same material but different experiments, or different materials but same phenomenon of interest). Another field of application of the performance metrics is the validation and verification task. In particular, the definition of the metrics should provide an unbiased and objective way to assess whether the obtained CL is able to reproduce the behavior under investigation and how much the obtained results can be trusted. This topic is better analyzed in the following section.

6.2 Verification and Validation

Verification and validation (V & V) is necessary for a confident quantification of engineering predictions, especially in the context of making decisions in high-consequence scenarios. Generally, in the computational solid mechanics community verification addresses the process of confirming that a mathematical model is correctly represented by its governing equations, appropriately discretized in a numerical setting, and that the solution complies with relevant accuracy requirements [391]. Conversely, validation refers to quantifying the degree of accuracy to which a model represents the investigated phenomenon [392]. In the following, we broadly ignore V & V from a software development perspective (we refer to [393, 394] for more information) and assume that the code is error-free and well implemented. Also, we refrain from proposing or summarizing guidelines about how to verify numerical convergence and stability of a code, for which we refer to [395, 396].

When compared to traditional phenomenological modeling, where the constitutive parameters of the material models are calibrated using experimental data, DD constitutive modeling involves an additional layer of difficulty. In particular, the decision logic of a DD approach is oftentimes opaque even to their designers, therefore a careful, separate, and independent V & V process to ensure they are reliable and safe to use must be performed.

In the context of the present work, the V & V process of DD constitutive modeling involves the same steps as those of traditional phenomenological constitutive models (c.f. [397–399]), however, in light of big data availability, the possible black-box character and presence of many parameters of some ML approaches, different additional caveats have to be accounted for. Past work [400, 401] has also considered the simplicity of the model as a selection criterion through the development of OPAL, the Occam-Plausibility Algorithm, which is a Bayesian framework developed to address uncertainties in parameters, data, and model selection. The general workflow of the process is inspired by [394] and is schematized in Fig. 7. The validation tests and their outcome including the UQ analysis constitute the ground truth obtained from the experimental side (which could come from a simulation or a physical experiment), while the right branch starting from the definition of the calibration data highlights the development and employment of the computational model. The general V & V process starts with the definition of the process or phenomenon of interest and the identification of the relevant or available variables and parameters (which possibly constitute the *inputs* of the CL) and the *Quantities of Interest* (QoI) which we aim to predict (namely, the *output* of the CL). Then, a *conceptual model* has to be developed that includes the general objective, a target accuracy between modeled and true QoI, as

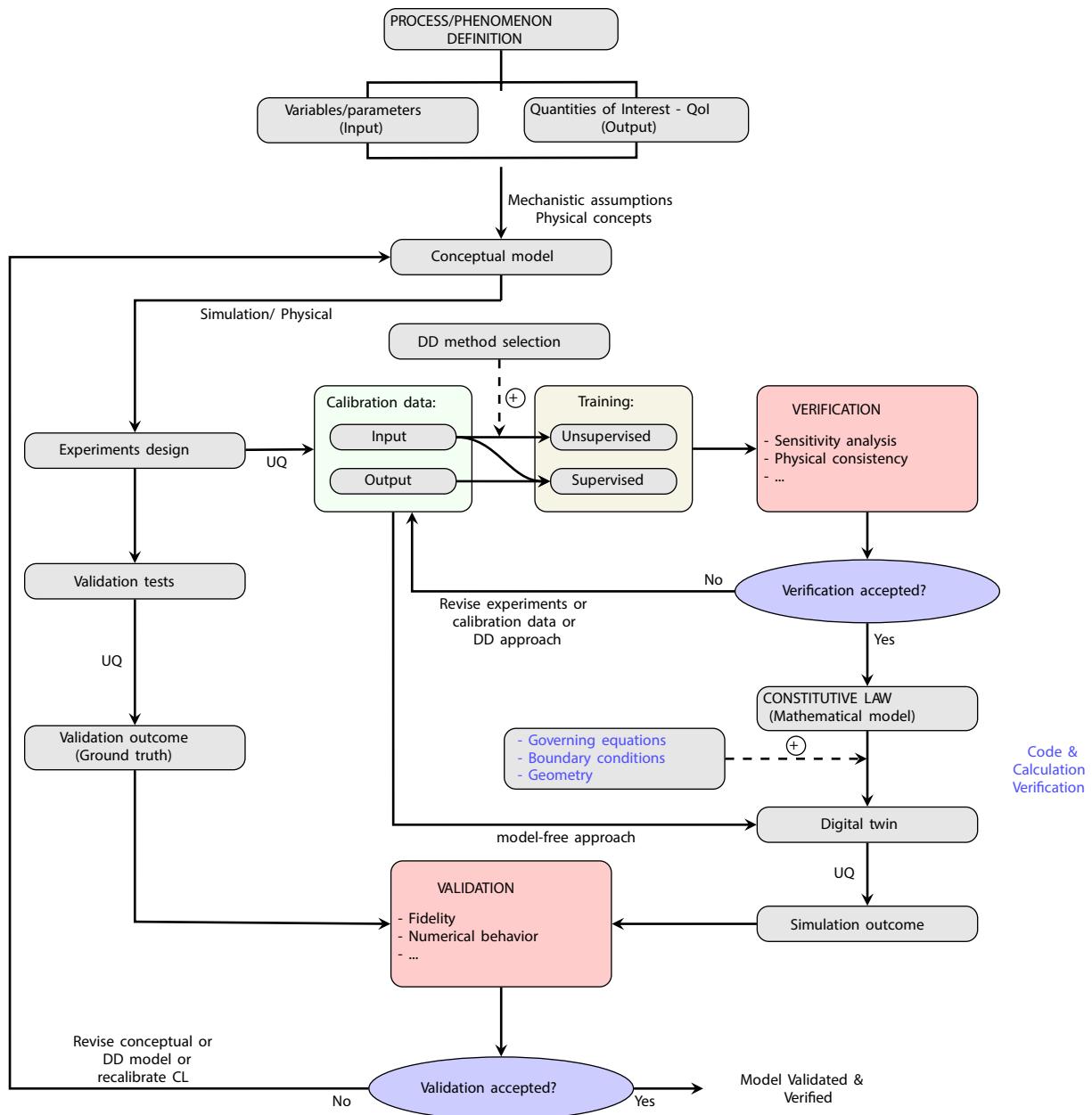


Fig. 7 Flow chart of model development, verification and validation of a computational model involving DD constitutive models

well as necessary physical and mechanical assumptions of the model. In this phase, the independent governing variables whose effect should be included and investigated must be defined (e.g., which environmental parameters should be included as variables). Based on the conceptual model, an *experiments design* should be carried out so as to allow for, on the one hand, the collection of the *calibration data*, and on the other hand the definition of the validation tests.

Once the input and output quantities (variables/parameters and QoIs) are collected and their UQ is performed, they can be adopted for the training of the selected DD approach, following a supervised (labeled inputs and outputs from the

calibration data are used) or unsupervised approach (only the inputs are used). At this point, the trained DD approach becomes a candidate CL and it must be verified in order to assess if and how well it is able to represent the calibration data and the physics behind it given a set of desirable properties measured by a selection of performance metrics. In this context, the accuracy and sensitivity to the training data are of utmost importance as well as the physical consistency checks, the fidelity (given the task for which the CL is needed), and the reliability. Therefore, if the verification is not satisfied the operator must revise and improve all or some of the steps from the definition of the calibration data

to the selection and training of the DD approach until the verification test is passed. At that moment, the approach (intended as a mathematical description of a set of data) can be considered representative of a material behavior and, hence, a CL.

Once the verified CL is complemented with governing equations, boundary (and, possibly, initial) conditions, and the geometry a *digital twin* of the investigated process is created. Note that, in the case of a model-free approach the calibration data are directly injected into the digital twin while avoiding their manipulation. This also implies the absence of any CL in terms of a mathematical description. The digital twin can be used for the validation, which is assessed with respect to (at least) two aspects: (i) the capability of the digital twin to reproduce the behavior of real components (namely, the fidelity) and (ii) the numerical behavior of the model. In the former case, the digital twin is adapted to the validation experiments by prescribing the proper geometry and boundary conditions and then the outcomes of the validation tests and of the digital twin simulations are quantitatively compared to determine if the trained CL is able to reproduce the phenomenon of interest. We remark that it is here particularly important to adopt independent validation tests, namely tests whose results cannot be represented by simple combinations of the calibration data. In other words, the validation data should also test the capacity of the CL to reproduce loading paths not included in the calibration dataset. Particular attention should be paid here in testing the possible overfitting of the data [402–404], an issue that can lead to inconsistent predictions of the material behavior. In this phase, it is also important to perform a UQ to estimate the confidence of the approach to replicate real behaviors and compare it to the one of the experiments. Concerning the numerical behavior it is important to evaluate how the DD approach integrates into the numerical solver (e.g., possibility of obtaining a consistent tangent matrix, smoothness of the operators) and the overall convergence behavior of the model. For both aspects, a set of metrics should be defined to assess the success or rejection of the validation procedure. In the negative case, the whole process should be revised, including the definition of the digital twin and the numerical framework thereof.

Even though less developed than in computational mechanics, V & V in the context of DD model development has been pushed more into the focus lately [405–407] due to the rising popularity of DD even in high-consequence applications. In the following, we briefly revise the available contributions in the context of the V & V of DD approaches. As a general comment, we remark that the available literature deals mostly with NNs, while other approaches are less investigated.

DD verification. In principle verifying a DD model requires checking that the model is able to broadly represent

its intended use case and does not behave unexpectedly (e.g., blow-up, undefined output for certain inputs) under a broad range of circumstances. Also, the operator should test the sensitivity of the model to both free parameters of the training process as well as calibration data quantity and quality. Due to the general opaqueness (black-box nature) of many DD models, corner cases are generally more unpredictable than for traditional phenomenological modeling. Therefore, for practitioners, it is crucial to answer the question as to which specific range of input values the model should be verified against. In principle, the latter should be sufficient to explore all the possible practical cases of interest, however, if this is not possible, the range of input tested should always be documented and made available.

Another common type of ML verification is the check for local robustness of a model, see e.g. [408] for a definition. Essentially a small perturbation in the input should only lead to a small change in the output. Adversarial attacks [409] are typically employed in the literature to provide evidence that the model is robust when no provable guarantee is available. Other critical verification properties include the output reachability property, see e.g. [410], i.e. if we know that a given stress component value can reach a certain magnitude we need to ensure that our ML output is able to represent this number. For more information on and general definitions of verification techniques for ML models (specifically, fully connected NNs), we refer to [411]. In the context of model-free approaches, verification usually takes the form of a convergence study with respect to the number of points available in the material dataset and to the noise amplitude [81, 372, 412].

Verification of ML models has seen less attention in the literature compared to their validation.

DD validation. Validation describes the process of assessing if an approach reproduces the phenomenon of interest in a broad range of applications and its predictive capabilities. Compared to classical ML tasks, this is especially important for constitutive modeling of solids due to the low/limited data availability often encountered in this area and the frequent need to extrapolate outside of the training domain. In the following we review only the aspects of validation that are specific to DD approaches for CLs; for the classical aspects (e.g., comparison with validation tests and convergence) we refer to the available literature [393–396], which deals particularly with the issue of over-parametrization for the ML approaches involving many fitting parameters.

In contrast to phenomenological modeling, over-parametrization is a typical problem in ML constitutive modeling, hence, validating an ML model often requires checks for overfitting because it is a major sign of subpar generalization behavior. Various strategies have been proposed in the literature to either detect overfitting or prevent it, see

[402, 403] for more information. The modification of an ML algorithm so as to prevent overfitting is often referred to as regularization [404]. In DD constitutive modeling, the following explicit or implicit regularization methods have been proposed:

- Early stopping. The most common type of validation is built upon validation datasets that can be used for implicit regularization by early stopping (stopping training when the error on the validation dataset increases, as this is a sign of overfitting to the training dataset) [413]. Splitting the dataset into training and validation components is not always straightforward especially if time- or history-dependent material behavior is studied [414].
- L^p regularization. L^p regularization in general adds a term to the training loss function that penalizes the complexity of the trainable parameters of the ML model. Due to the reduction of model complexity, it is hoped to avoid overfitting and reduce the generalization error. Common regularization techniques of this type include *Ridge* regression or L^2 -regularization [415], which is used to make the amplitude of the trainable parameters of a model smaller [416]. In an adjusted form L^2 -regularization can appear as "weight decay" in ML when using stochastic gradient descent schemes [417]. L^p -regression with $0 < p \leq 1$ can also be used to enforce sparsity on a model, which can reduce the number of trainable parameters and the sensitivity to noise while making the model more interpretable [32, 34, 322]. The special case $p = 1$ is known as *Lasso* regression [33]. Other methods such as *elastic net* [418] use a combination of *Ridge* and *Lasso*.
- Dilution and Dropout. When training NNs, Dilution and Dropout are used to randomly drop units and relevant connections from the NNs during training. This prevents weights from co-adapting too much, thereby aiding generalization and avoiding overfitting [419]. This technique can generally be employed to regularize any ML model and was e.g. employed by [175] to obtain DD models for anisotropic hyperelasticity.
- Physics-based regularization. The generalization error of a DD model can be vastly improved by either implicitly or explicitly adding physics constraints that underlie the data to the model. In constitutive modeling, this can take the form of implicitly enforcing polyconvexity [177, 420] of hyperelastic laws, convexity of yield functions [303] or thermodynamic consistency of time- and history-dependent material models [288, 421].
- Adversarial ML. Originally developed to prevent model exploitation against attacks, adversarial training, i.e. training on intentionally misleading or perturbed data, can help the model to generalize better and become more robust [422]. This includes schemes such as weight perturbation [423] or training on perturbed inputs which

are commonly referred to as adversarial examples [424]. [279] uses this technique to improve the generalization performance of an elastoplastic material model trained by an RNN.

- Post-hoc explainability. Black-box models, e.g. those based on NNs, can be validated after the training process has ended by trying to produce useful approximations of the decision logic of the model that correspond to understandable/interpretable representations. This process typically involves the generation of a second (post-hoc) model that is used to explain the behavior of the first black-box model. This can e.g. be achieved by pruning the model [425], e.g. using parameter reduction techniques, or training an interpretable model e.g. with symbolic or polynomial regression. The latter has e.g. been deployed in [426] for modeling plasticity.

Even though many different ML validation techniques have been introduced to counteract overfitting and help improve generalization, one of the open questions in the field is how ML tools can be certified [427]. There is a lack of general metrics that specify at which point an ML model can be trusted, i.e. how interpretable/trustworthy it needs to be. Furthermore, benchmarks have to be established to tell if a model can be deemed safe enough to be used even in high-risk applications.

7 Conclusions

At this point, it is important to distill the main findings of this journey in constitutive modeling and to outline our view of the successes, open issues, and opportunities that lie ahead. Phenomenological constitutive approaches are by definition based on observations, making them in some sense the original DD models, but contrary to some of their more modern counterparts they are also interpretable, able to generalize, and not prone to overfitting. Borrowing some language from the ML community, the archetype in mechanics for phenomenological constitutive modeling approaches has been to use simple experiments that result in homogeneous stress states (e.g. uniaxial, biaxial, simple shear) as "training" data and to use structural responses that include inhomogeneous fields as "validation". Even though these analytical approaches often lack expressivity, the strict enforcement of thermodynamic constraints, and balance laws, as well as the incorporation of mechanistic insight allow these models to extrapolate to unseen stress states and be useful and trustworthy for safety-critical predictive calculations in structural engineering as well as in other disciplines. As downsides of these successes, the development of classical phenomenological models requires specialized domain knowledge and a substantial conceptual effort; already the selection of the

most suitable model among available options, along with its calibration, necessitates tedious and time-consuming trial-and-error procedures, which are a significant obstacle towards material innovation in many engineering fields.

From our standpoint, DD constitutive modeling approaches, through the introduction of new computational tools, have the potential to: 1) accommodate the utilization of larger datasets from modern experimental techniques and assist in the interpretation of the data, 2) enhance the flexibility in the exploration and utilization of the modeling space to best interpret the data; 3) simplify the utilization of multifidelity and multimodal information (e.g. merge experimental and computational data), 4) automate the connection from data to predictive simulations streamlining the necessary infrastructure, 5) enable multiscale calculations, as well as 6) encode information for material variability as a first step towards uncertainty quantification and reliability analysis.

As the second wave of DD constitutive modeling approaches is reaching a stage of maturation beyond initial exploratory attempts, it is becoming clear that a way to maintain the benefits of phenomenological modeling while developing and utilizing expressive and automated DD approaches is to take full advantage of a long and rich history of insightful mechanistic research formalized through what we have denoted throughout the paper as physics constraints. We have discussed several examples, all over different classes of DD constitutive models, where successes in terms of robustness, interpretability, generalization, and overall trustworthiness are gained by enforcing such constraints.

Even though some promising results have been recorded, there are still many open challenges. One of them is the seamless integration of DD approaches in the existing infrastructure of advanced experimental mechanics on the one hand, and of established computational engineering on the other hand. Experimental approaches based on modern imaging such as DIC or DVC rely on complex correlation algorithms that, starting from raw data such as grey level fields, infer full-field displacement information that can be directly utilized in learning tasks (e.g. DD discovery of interpretable CLs, or training of NN-based DD CLs). In principle, an integration between correlation and learning algorithms could reduce the accumulation of errors and streamline the process leading to the establishment of DD CLs from this type of experimental data. At the other end of the process, discovered or trained CLs need to be integrated with traditional computational mechanics tools such as finite element solvers, in order to be readily used by engineering practitioners in the solution of boundary value problems. Automatic differentiation, which is a key tool for this purpose, often leads to a decrease in computational performance over non-linear finite element solvers with

hard-coded algorithms (e.g. in plasticity when using return mapping algorithms). Note also that error measurements which commonly indicate the success of a DD approach are not sufficient to guarantee the convergence of non-linear finite element solvers embedding DD constitutive models. To this end, thorough testing of these models in a FEM setting is necessary.

For path-dependent material behavior, an interesting task (and one that so far has not been sufficiently explored) is the discovery of underlying irreversible mechanisms and corresponding variables (e.g. hardening variables) directly from data. Also, sampling becomes significantly more complex and restricted from experimental protocols, as the learning task focuses on objects that depend on internal variables (e.g. yield function).

Additionally, we should strive for DD constitutive models to be robust in a low-data setting (contrary to the ongoing trends in computer science), as one could envision access to limited experimental samples and also limited computational resources, especially as multiscale problems become more complex and path-dependent behaviors are considered. In principle, already available DD approaches are able to discover CLs on a one-shot basis, i.e. using only one test. Clearly, the success of this approach needs a sufficiently complex test geometry and a sufficiently comprehensive test method (e.g. including loading and unloading stages, possibly at different rates). Thus, the optimal design of test methods and specimen geometries to guarantee identifiability (especially critical for highly expressive models) could deliver important contributions in this respect and is still largely unexplored. In all these endeavors, UQ is expected to play a major role. Fundamental questions related e.g. to the amount and quality of data needed to deliver a DD model (or several competing models) of target uncertainty, and to the uncertainty of the outcomes of downstream tasks such as the solution of non-linear problems embedding DD CLs can only be answered in a probabilistic framework.

Finally, tasks such as the curation of large datasets along with corresponding benchmark problems should be established as a way to evaluate constitutive DD approaches against the criteria that we have defined in this work. Specifically, here we suggest starting with two open datasets for DD constitutive approaches in the context of hyperelasticity. The first focuses on RVE-generated data (in the context of computational homogenization) for a composite microstructure sampled uniformly in a hypersphere of the components of the deformation gradient and also includes some specific loading paths up to larger deformations as a means to test generalization. To connect with experiments that do not produce labeled data pairs, the second dataset and benchmark should correspond to full-field displacement data and reaction forces for a specific geometry and boundary value problem for training (computationally generated

through FEM), and a complementary dataset from another geometry and boundary value problem to validate the DD constitutive model. More work needs to be done on this front and the community can only benefit by establishing similar benchmarks for more complex, path-dependent problems.

Acknowledgements MF, PC and LDL acknowledge support from the Swiss National Science Foundation (SNF), project number 200021_204316 “Unsupervised data-driven discovery of material laws”. JF and NB gratefully acknowledge support by the Air Force Office of Scientific Research under award number FA9550-22-1-0075. GAP and NB were supported by the SciAI Center, and funded by the Office of Naval Research (ONR), under Grant Number N00014-23-1-2729. WS, BB and NNV are supported by the National Science Foundation under grant contracts CMMI-1846875 and the Department of Energy, National Nuclear Security Administration, Predictive Science Academic Alliance Program (PSAAP) under Award Number DE-NA0003962.

Author Contributions The authors confirm contribution to the paper as follows: JF, NB, WS, and LDL work conceptualization; NB, WS, and LDL funding acquisition; JF, NB, GAP, BB, WS, NNV, MF, PC and LDL draft manuscript preparation and editing; all authors reviewed the manuscript and approved submission.

Data Availability No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare no Conflict of interest.

References

- Young T (1807) A course of lectures on natural philosophy and the mechanical arts. Taylor and Walton, Luton
- Truesdell C, Noll W (1965) The non-linear field theories of mechanics. In: The non-linear field theories of mechanics. Springer, Berlin. pp 1–579
- Holzapfel AG (2000) Nonlinear solid mechanics II. Springer, Berlin
- Simo JC, Hughes TJ (2006) Computational inelasticity, vol 7. Springer, Berlin
- Souza Neto EA, Peric D, Owen DR (2011) Computational methods for plasticity: theory and applications. Wiley, New York
- Sutton MA (2013) Computer vision-based, noncontacting deformation measurements in mechanics: a generational transformation. *Appl Mech Rev.* <https://doi.org/10.1115/1.4024984>
- Sutton MA, Hild F (2015) Recent advances and perspectives in digital image correlation. *Exp Mech* 55(1):1–8. <https://doi.org/10.1007/s11340-015-9991-6>
- Withers PJ, Bouman C, Carmignato S, Cnudde V, Grimaldi D, Hagen CK, Maire E, Manley M, Du Plessis A, Stock SR (2021) X-ray computed tomography. *Nat Rev Methods Primers* 1(1):18. <https://doi.org/10.1038/s43586-021-00015-4>
- Yang Z, Ren W, Sharma R, McDonald S, Mostafavi M, Verityagina Y, Marrow TJ (2017) In-situ X-ray computed tomography characterisation of 3d fracture evolution and image-based numerical homogenisation of concrete. *Cement Concr Compos* 75:74–83. <https://doi.org/10.1016/j.cemconcomp.2016.10.001>
- Carrara P, Wu T, Kruse R, Lorenzis LD (2016) Towards multi-scale modeling of the interaction between transport and fracture in concrete. *RILEM Lett* 1:94–101. <https://doi.org/10.21809/rilemtechlett>
- Leclerc H, Périé JN, Roux S, Hild F (2011) Voxel-scale digital volume correlation. *Exp Mech* 51(4):479–490. <https://doi.org/10.1007/s11340-010-9407-6>
- Mendoza A, Neggers J, Hild F, Roux S (2019) Complete mechanical regularization applied to digital image and volume correlation. *Comput Methods Appl Mech Eng* 355:27–43. <https://doi.org/10.1016/j.cma.2019.06.005>
- Pierron F, Grédiac M (2021) Towards material testing 2.0. A review of test design for identification of constitutive parameters from full-field measurements. *Strain* 57(1):1–22. <https://doi.org/10.1111/str.12370>
- Pierron F (2023) Material testing 2.0: a brief review. *Strain* 2022:1–20. <https://doi.org/10.1111/str.12434>
- Schmidt M, Lipson H (2009) Distilling free-form natural laws from experimental data. *Science* 324(5923):81–85. <https://doi.org/10.1126/science.1165893>
- Brunton SL, Proctor JL, Kutz JN (2016) Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc Natl Acad Sci* 113(15):3932–3937
- Lagaris IE, Likas A, Fotiadis DI (1998) Artificial neural networks for solving ordinary and partial differential equations. *IEEE Trans Neural Netw* 9(5):987–1000
- Raissi M, Perdikaris P, Karniadakis GE (2019) Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J Comput Phys* 378:686–707
- Liu X, Tao F, Yu W (2020) A neural network enhanced system for learning nonlinear constitutive law and failure initiation criterion of composites using indirectly measurable data. *Compos Struct* 252:112658. <https://doi.org/10.1016/j.compstruct.2020.112658>
- Lu L, Jin P, Pang G, Zhang Z, Karniadakis GE (2021) Learning nonlinear operators via Deeponet based on the universal approximation theorem of operators. *Nat Mach Intell* 3(3):218–229
- Fuhg JN, Bouklas N (2022) On physics-informed data-driven isotropic and anisotropic constitutive models through probabilistic machine learning and space-filling sampling. *Comput Methods Appl Mech Eng* 394:114915
- Kissas G, Mishra S, Chatzi E, De Lorenzis L (2024) The language of hyperelastic materials. *Comput Methods Appl Mech Eng* 428:117053
- Koza J (1994) Genetic programming as a means for programming computers by natural selection. *Stat Comput*. <https://doi.org/10.1007/BF00175355>
- Billard L, Diday E (2002) Symbolic regression analysis. In: Classification, clustering, and data analysis. Springer, Berlin. pp 281–288
- Wang Y, Wagner N, Rondinelli JM (2019) Symbolic regression in materials science. *MRS Commun* 13:793–805
- Searson DP, Leahy DE, Willis MJ (2010) GPTIPS: an open source genetic programming toolbox for multigene symbolic regression. Citeseer, Hong Kong, p 4
- Dubčáková R (2011) Eureqa: software review. Springer, Berlin
- Stephens T (2016) Genetic programming in python, with a Scikit-learn inspired API: gplearn.
- Udrescu S-M, Tegmark M (2020) Ai Feynman: a physics-inspired method for symbolic regression. *Sci Adv* 6(16):2631
- La Cava W, Orzechowski P, Burlacu B, França FO, Virgolin M, Jin Y, Kommenda M, Moore JH (2021) Contemporary symbolic regression methods and their relative performance. *arXiv:2107.14351*
- Santosa F, Symes WW (1986) Linear inversion of band-limited reflection seismograms. *SIAM J Sci Stat Comput* 7(4):1307–1330. <https://doi.org/10.1137/0907087>

32. Frank LE, Friedman JH (1993) A statistical view of some chemometrics regression tools. *Technometrics* 35(2):109–135

33. Tibshirani R (1996) Regression shrinkage and selection via the lasso. *J R Stat Soc Ser B (Methodol)* 58(1):267–288

34. Flaschel M, Kumar S, De Lorenzis L (2021) Unsupervised discovery of interpretable hyperelastic constitutive laws. *Comput Methods Appl Mech Eng* 381:113852. <https://doi.org/10.1016/j.cma.2021.113852>

35. Flaschel M (2023) Automated discovery of material models in continuum solid mechanics. PhD thesis, ETH Zurich. <https://doi.org/10.3929/ETHZ-B-000602750>

36. Wang Z, Estrada JB, Arruda EM, Garikipati K (2021) Inference of deformation mechanisms and constitutive response of soft material surrogates of biological tissue by full-field characterization and data-driven variational system identification. *J Mech Phys Solids* 153:104474. <https://doi.org/10.1016/j.jmps.2021.104474>

37. Sherstinsky A (2020) Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D* 404:132306

38. Nosouhian S, Nosouhian F, Khoshouei AK (2021) A review of recurrent neural network architecture for sequence learning: comparison between LSTM and GRU

39. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. *Adv Neural Inf Process Syst* 30

40. Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, Santamaría J, Fadhel MA, Al-Amidie M, Farhan L (2021) Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *J Big Data* 8:1–74

41. Wang X, Girshick R, Gupta A, He K (2018) Non-local neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 7794–7803

42. Zhou J, Cui G, Hu S, Zhang Z, Yang C, Liu Z, Wang L, Li C, Sun M (2020) Graph neural networks: a review of methods and applications. *AI Open* 1:57–81

43. Goodfellow I, Bengio Y, Courville A, Bengio Y (2016) Deep learning, vol 1. MIT Press, Cambridge

44. Baydin AG, Pearlmutter BA, Radul AA, Siskind JM (2018) Automatic differentiation in machine learning: a survey. *J Mach Learn Res* 18

45. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M et al (2016) Tensorflow: a system for large-scale machine learning. In: 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16). pp 265–283

46. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L et al (2019) Pytorch: an imperative style, high-performance deep learning library. *Adv Neural Inf Process Syst* 32:8026–8037

47. Frostig R, Johnson MJ, Leary C (2018) Compiling machine learning programs via high-level tracing. *Syst Mach Learn* 4(9)

48. Hornik K, Stinchcombe M, White H (1989) Multilayer feed-forward networks are universal approximators. *Neural Netw* 2(5):359–366

49. Lu Z, Pu H, Wang F, Hu Z, Wang L (2017) The expressive power of neural networks: a view from the width. In: Proceedings of the 31st international conference on neural information processing systems. pp 6232–6240

50. Lampinen J, Vehtari A (2001) Bayesian approach for neural networks-review and case studies. *Neural Netw* 14(3):257–274

51. Swiler LP, Gulian M, Frankel AL, Safta C, Jakeman JD (2020) A survey of constrained gaussian process regression: approaches and implementation challenges. *J Mach Learn Model Comput* 1(2)

52. Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA (2017) Deep reinforcement learning: a brief survey. *IEEE Signal Process Mag* 34(6):26–38

53. Clifton J, Laber E (2020) Q-learning: theory and applications. *Annu Rev Stat Appl* 7:279–301

54. Grondman I, Busoniu L, Lopes GA, Babuska R (2012) A survey of actor-critic reinforcement learning: standard and natural policy gradients. *IEEE Trans Syst Man Cybern Part C* 42(6):1291–1307

55. Sutton RS (1991) Dyna, an integrated architecture for learning, planning, and reacting. *ACM Sigart Bull* 2(4):160–163

56. Silver D, Hubert T, Schrittwieser J, Antonoglou I, Lai M, Guez A, Lanctot M, Sifre L, Kumaran D, Graepel T et al (2018) A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science* 362(6419):1140–1144

57. Hearst MA, Dumais ST, Osuna E, Platt J, Scholkopf B (1998) Support vector machines. *IEEE Intell Syst Appl* 13(4):18–28

58. Hofmann M (2006) Support vector machines-kernels and the kernel trick. *Notes* 26(3):1–16

59. Chang C-C, Lin C-J (2011) LIBSVM: a library for support vector machines. *ACM Trans Intell Syst Technol* 2(3):1–27

60. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: machine learning in Python. *J Mach Learn Res* 12:2825–2830

61. Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. *Stat Comput* 14(3):199–222

62. Awad M, Khanna R (2015) Support vector regression. In: Efficient learning machines. Springer, Berlin. pp 67–80

63. Lauer F, Bloch G (2008) Incorporating prior knowledge in support vector regression. *Mach Learn* 70(1):89–118

64. Guyon I, Gunn S, Nikravesh M, Zadeh LA (2008) Feature extraction: foundations and applications, vol 207. Springer, Berlin

65. Jolliffe IT, Cadima J (2016) Principal component analysis: a review and recent developments. *Philos Trans R Soc A* 374(2065):20150202

66. Bank D, Koenigstein N, Giryres R (2023) Autoencoders. In: Machine learning for data science handbook: data mining and knowledge discovery handbook. pp 353–374

67. Zhang N, Ding S, Zhang J, Xue Y (2018) An overview on restricted boltzmann machines. *Neurocomputing* 275:1186–1199

68. Kingma DP, Welling M (2013) Auto-encoding variational bayes. [arXiv:1312.6114](https://arxiv.org/abs/1312.6114)

69. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2020) Generative adversarial networks. *Commun ACM* 63(11):139–144

70. Dinh L, Sohl-Dickstein J, Bengio S (2016) Density estimation using real NVP. [arXiv:1605.08803](https://arxiv.org/abs/1605.08803)

71. Croitoru F-A, Hondu V, Ionescu RT, Shah M (2023) Diffusion models in vision: a survey. *IEEE Trans Pattern Anal Mach Intell*

72. Padmanabha GA, Zabaras N (2021) Solving inverse problems using conditional invertible neural networks. *J Comput Phys* 433:110194

73. Di Mattia F, Galeone P, De Simoni M, Ghelfi E (2019) A survey on gans for anomaly detection. [arXiv:1906.11632](https://arxiv.org/abs/1906.11632)

74. Korb KB, Nicholson AE (2010) Bayesian artificial intelligence. CRC Press, Boca Raton

75. Rasmussen CE (2003) Gaussian processes in machine learning. In: Summer school on machine learning. Springer, Berlin. pp 63–71

76. Lophaven SN, Nielsen HB, Søndergaard J et al (2002) DACE: a Matlab Kriging toolbox, vol 2. Citeseer

77. Roustant O, Ginsbourger D, Deville Y (2012) Dicekriging, diceoptim: two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization. *J Stat Softw* 51:1–55

78. Gardner JR, Pleiss G, Bindel D, Weinberger KQ, Wilson AG (2018) Gpytorch: blackbox matrix-matrix gaussian process inference with GPU acceleration. [arXiv:1809.11165](https://arxiv.org/abs/1809.11165)

79. Liu H, Ong Y-S, Shen X, Cai J (2020) When gaussian process meets big data: a review of scalable gps. *IEEE Trans Neural Netw Learn Syst* 31(11):4405–4423

80. James G, Witten D, Hastie T, Tibshirani R (2013) An introduction to statistical learning, vol 112. Springer, Berlin

81. Kirchdoerfer T, Ortiz M (2016) Data-driven computational mechanics. *Comput Methods Appl Mech Eng* 304:81–101

82. Carrara P, Ortiz M, De Lorenzis L (2022) Model-free fracture mechanics and fatigue. In: Current trends and open problems in computational mechanics. Springer, Berlin. pp 75–82

83. He X, He Q, Chen J-S, Sinha U, Sinha S (2020) Physics-constrained local convexity data-driven modeling of anisotropic nonlinear elastic solids. *Data-Centric Engineering* 1

84. Bahmani B, Sun W (2022) Manifold embedding data-driven mechanics. *J Mech Phys Solids* 166:104927

85. Montáns FJ, Chinesta F, Gómez-Bombarelli R, Kutz JN (2019) Data-driven modeling and learning in science and engineering. *Comptes Rendus Mécanique* 347(11):845–855

86. Chinesta F, Cueto E, Abisset-Chavanne E, Duval JL, El Khaldi F (2020) Virtual, digital and hybrid twins: a new paradigm in data-based engineering and engineered data. *Arch Comput Methods Eng* 27(1):105–134

87. Liu X, Tian S, Tao F, Du H, Yu W (2021) Machine learning-assisted modeling of composite materials and structures: a review. In: AIAA Scitech 2021 Forum. p 2023

88. Peng GC, Alber M, Tepole AB, Cannon WR, De S, Dura-Bernal S, Garikipati K, Karniadakis G, Lytton WW, Perdikaris P et al (2021) Multiscale modeling meets machine learning: what can we learn? *Arch Comput Methods Eng* 28(3):1017–1037

89. Kumar S, Kochmann DM (2022) What machine learning can do for computational solid mechanics. In: Aldakheel F, Hudobivnik B, Soleimani M, Wessels H, Weißenfels C, Marino M (eds) Current trends and open problems in computational mechanics. Springer, Cham. pp 275–285. https://doi.org/10.1007/978-3-030-87312-7_27

90. Guo K, Yang Z, Yu CH, Buehler MJ (2021) Artificial intelligence and machine learning in design of mechanical materials. *Materials Horizons*

91. Mosavi A, Rabczuk T, Varkonyi-Koczy AR (2018) Reviewing the novel machine learning tools for materials design. In: Luca D, Sirghi L, Costin C (eds) Recent advances in technology research and education, vol 660. Springer, Cham. pp 50–58. https://doi.org/10.1007/978-3-319-67459-9_7. Series Title: Advances in Intelligent Systems and Computing. http://link.springer.com/10.1007/978-3-319-67459-9_7

92. Bock FE, Aydin RC, Cyron CJ, Huber N, Kalidindi SR, Klusemann B (2019) A review of the application of machine learning and data mining approaches in continuum materials mechanics. *Front Mater* 6:110

93. Suh C, Fare C, Warren JA, Pyzer-Knapp EO (2020) Evolving the materials genome: how machine learning is fueling the next generation of materials discovery. *Annu Rev Mater Res* 50:1–25

94. Huang J, Liew J, Ademiloye A, Liew K (2020) Artificial intelligence in materials modeling and design. *Arch Comput Methods Eng* 1–15

95. Morgan D, Jacobs R (2020) Opportunities and challenges for machine learning in materials science. *Annu Rev Mater Res* 50:71–103

96. Neggers J, Allix O, Hild F, Roux S (2018) Big data in experimental mechanics and model order reduction: today's challenges and tomorrow's opportunities. *Arch Comput Methods Eng* 25(1):143–164

97. Brodnik N, Muir C, Tulshibagwale N, Rossin J, Echlin M, Hamel C, Kramer S, Pollock T, Kiser J, Smith C et al (2023) Perspective: machine learning in experimental solid mechanics. *J Mech Phys Solids* 173:105231

98. Tepole AB, Zhang J, Gomez H (2022) Data-driven methods in biomechanics. *J Biomech Eng* 144(12):120301

99. Qu T, Di S, Feng Y, Wang M, Zhao T (2021) Towards data-driven constitutive modelling for granular materials via micro-mechanics-informed deep learning. *Int J Plast* 144:103046

100. Vlassis NN, Sun W (2021) Sobolev training of thermodynamic-informed neural networks for interpretable elasto-plasticity models with level set hardening. *Comput Methods Appl Mech Eng* 377:113695

101. Fuhr JN, Böhm C, Bouklas N, Fau A, Wriggers P, Marino M (2021) Model-data-driven constitutive responses: application to a multiscale computational framework. *Int J Eng Sci* 167:103522. <https://doi.org/10.1016/j.ijengsci.2021.103522>

102. Crombecq K, Laermans E, Dhaene T (2011) Efficient space-filling and non-collapsing sequential design strategies for simulation-based modeling. *Eur J Oper Res* 214(3):683–696

103. Stein M (1987) Large sample properties of simulations using latin hypercube sampling. *Technometrics* 29(2):143–151

104. Lu X, Giovanis DG, Yvonnet J, Papadopoulos V, Detrez F, Bai J (2019) A data-driven computational homogenization method based on neural networks for the nonlinear anisotropic electrical response of graphene/polymer nanocomposites. *Comput Mech* 64(2):307–321

105. Feng N, Zhang G, Khandelwal K (2022) Finite strain Fe_2 analysis with data-driven homogenization using deep neural networks. *Comput Struct* 263:106742

106. Fuhr JN, Bouklas N, Jones RE (2022) Learning hyperelastic anisotropy from data via a tensor basis neural network. *J Mech Phys Solids* 168:105022. <https://doi.org/10.1016/j.jmps.2022.105022>

107. Mozaffar M, Bostanabad R, Chen W, Ehmann K, Cao J, Bessa M (2019) Deep learning predicts path-dependent plasticity. *Proc Natl Acad Sci* 116(52):26414–26420

108. Gorji MB, Mozaffar M, Heidenreich JN, Cao J, Mohr D (2020) On the potential of recurrent neural networks for modeling path dependent plasticity. *J Mech Phys Solids* 103972

109. Abueidda DW, Koric S, Sobh NA, Sehitoglu H (2021) Deep learning for plasticity and thermo-viscoplasticity. *Int J Plast* 136:102852

110. Bonatti C, Berisha B, Mohr D (2022) From CP-FFT to CP-RNN: recurrent neural network surrogate model of crystal plasticity. *Int J Plasticity* 103430

111. Wu L, Kilingar NG, Noels L et al (2020) A recurrent neural network-accelerated multi-scale model for elasto-plastic heterogeneous materials subjected to random cyclic and non-proportional loading paths. *Comput Methods Appl Mech Eng* 369:113234

112. Logarzo HJ, Capuano G, Rimoli JJ (2021) Smart constitutive laws: Inelastic homogenization through machine learning. *Comput Methods Appl Mech Eng* 373:113482

113. Settles B (2009) Active learning literature survey. University of California, Santa Cruz

114. Liu H, Ong Y-S, Cai J (2018) A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design. *Struct Multidiscip Optim* 57(1):393–416

115. Fuhr JN, Fau A, Nackenhorst U (2020) State-of-the-art and comparative review of adaptive sampling methods for kriging. *Arch Comput Methods Eng* 1–59

116. Knap J, Barton N, Hornung R, Arsenlis A, Becker R, Jefferson D (2008) Adaptive sampling in hierarchical simulation. *Int J Numer Methods Eng* 76(4):572–600

117. Leiter KW, Barnes BC, Becker R, Knap J (2018) Accelerated scale-bridging through adaptive surrogate model evaluation. *J Comput Sci* 27:91–106

118. Rocha I, Kerfriden P, Meer F (2021) On-the-fly construction of surrogate constitutive models for concurrent multiscale mechanical analysis through probabilistic machine learning. *J Comput Phys X* 9:100083

119. Kalina KA, Linden L, Brummund J, Kästner M (2022) Fe^{ANN}: an efficient data-driven multiscale approach based on physics-constrained neural networks and automated data mining. [arXiv: 2207.01045](https://arxiv.org/abs/2207.01045)

120. Karapiperis K, Ortiz M, Andrade JE (2021) Data-driven nonlocal mechanics: discovering the internal length scales of materials. *Comput Methods Appl Mech Eng* 386:114039

121. Roderick M, MacGlashan J, Tellex S (2017) Implementing the deep q-network. [arXiv:1711.07478](https://arxiv.org/abs/1711.07478)

122. Tiong T, Saad I, Teo KTK, Lago H (2020) Deep reinforcement learning with robust deep deterministic policy gradient. In: 2020 2nd international conference on electrical, control and instrumentation engineering (ICECIE). IEEE, pp 1–5

123. Volpp M, Fröhlich LP, Fischer K, Doerr A, Falkner S, Hutter F, Daniel C (2019) Meta-learning acquisition functions for transfer learning in bayesian optimization. [arXiv:1904.02642](https://arxiv.org/abs/1904.02642)

124. Zhang Z, Zohren S, Stephen R (2020) Deep reinforcement learning for trading. *J Financ Data Sci*

125. Kalashnikov D, Irpan A, Pastor P, Ibarz J, Herzog A, Jang E, Quillen D, Holly E, Kalakrishnan M, Vanhoucke V, et al. (2018) QT-OPT: scalable deep reinforcement learning for vision-based robotic manipulation. [arXiv:1806.10293](https://arxiv.org/abs/1806.10293)

126. Duan Y, Chen X, Houthooft R, Schulman J, Abbeel P (2016) Benchmarking deep reinforcement learning for continuous control. In: International conference on machine learning. PMLR, pp 1329–1338

127. Villarreal R, Vlassis NN, Phan NN, Catanach TA, Jones RE, Trask NA, Kramer SL, Sun W (2023) Design of experiments for the calibration of history-dependent models via deep reinforcement learning and an enhanced kalman filter. *Comput Mech* 72(1):95–124

128. Wang K, Sun W, Du Q (2021) A non-cooperative meta-modeling game for automated third-party calibrating, validating and falsifying constitutive laws with parallelized adversarial attacks. *Comput Methods Appl Mech Eng* 373:113514

129. Yvonnet J, Gonzalez D, He Q-C (2009) Numerically explicit potentials for the homogenization of nonlinear elastic heterogeneous materials. *Comput Methods Appl Mech Eng* 198(33–36):2723–2737. <https://doi.org/10.1016/j.cma.2009.03.017>

130. Le B, Yvonnet J, He Q-C (2015) Computational homogenization of nonlinear elastic materials using neural networks. *Int J Numer Methods Eng* 104(12):1061–1084

131. Fritzen F, Fernández M, Larsson F (2019) On-the-fly adaptivity for nonlinear two-scale simulations using artificial neural networks and reduced order modeling. *Front Mater* 6:75. <https://doi.org/10.3389/fmats.2019.00075>

132. Liu Z, Wu C, Koishi M (2019) A deep material network for multiscale topology learning and accelerated nonlinear modeling of heterogeneous materials. *Comput Methods Appl Mech Eng* 345:1138–1168

133. Liu Z, Wu C (2019) Exploring the 3d architectures of deep material network in data-driven multiscale mechanics. *J Mech Phys Solids* 127:20–46

134. Liu Z, Wu CT, Koishi M (2019) Transfer learning of deep material network for seamless structure-property predictions. *Comput Mech* 64(2):451–465. <https://doi.org/10.1007/s00466-019-01704-4>

135. Gajek S, Schneider M, Böhlke T (2020) On the micromechanics of deep material networks. *J Mech Phys Solids* 142:103984

136. Gajek S, Schneider M, Böhlke T (2021) An FE-DMN method for the multiscale analysis of short fiber reinforced plastic components. *Comput Methods Appl Mech Eng* 384:113952. <https://doi.org/10.1016/j.cma.2021.113952>

137. Nguyen VD, Noels L (2021) Interaction-based material network: a general framework for (porous) microstructured materials. *Comput Methods Appl Mech Eng*. <https://doi.org/10.1016/j.cma.2021.114300>

138. Nguyen VD, Noels L (2022) Micromechanics-based material networks revisited from the interaction viewpoint; robust and efficient implementation for multi-phase composites. *Eur J Mech A Solids* 91:104384. <https://doi.org/10.1016/j.euromechsol.2021.104384>

139. Man H, Furukawa T (2011) Neural network constitutive modelling for non-linear characterization of anisotropic materials. *Int J Numer Methods Eng* 85(8):939–957

140. Furukawa T, Michopoulos JG, Kelly DW (2008) Elastic characterization of laminated composites based on multiaxial tests. *Compos Struct* 86(1–3):269–278. <https://doi.org/10.1016/j.compositstruct.2008.03.043>

141. Huang DZ, Xu K, Farhat C, Darve E (2020) Learning constitutive relations from indirect observations using deep neural networks. *J Comput Phys* 416:109491. <https://doi.org/10.1016/j.jcp.2020.109491>

142. Liu X, Tao F, Du H, Yu W, Xu K (2020) Learning nonlinear constitutive laws using neural network models based on indirectly measurable data. *J Appl Mech* 87(8):081003. <https://doi.org/10.1115/1.4047036>

143. Liu Z, Bessa M, Liu WK (2016) Self-consistent clustering analysis: an efficient multi-scale scheme for inelastic heterogeneous materials. *Comput Methods Appl Mech Eng* 306:319–341

144. Wulffinghoff S, Cavaliere F, Reese S (2018) Model order reduction of nonlinear homogenization problems using a Hashin-Shtrikman type finite element method. *Comput Methods Appl Mech Eng* 330:149–179. <https://doi.org/10.1016/j.cma.2017.10.019>

145. Schneider M (2019) On the mathematical foundations of the self-consistent clustering analysis for non-linear materials at small strains. *Comput Methods Appl Mech Eng* 354:783–801. <https://doi.org/10.1016/j.cma.2019.06.003>

146. Bessa MA, Bostanabad R, Liu Z, Hu A, Apley DW, Brinson C, Chen W, Liu WK (2017) A framework for data-driven analysis of materials under uncertainty: countering the curse of dimensionality. *Comput Methods Appl Mech Eng* 320:633–667

147. Liu Z, Fleming M, Liu WK (2018) Microstructural material database for self-consistent clustering analysis of elastoplastic strain softening materials. *Comput Methods Appl Mech Eng* 330:547–577

148. Yu C, Kafka OL, Liu WK (2019) Self-consistent clustering analysis for multiscale modeling at finite strains. *Comput Methods Appl Mech Eng* 349:339–359

149. Zhang L, Tang S, Yu C, Zhu X, Liu WK (2019) Fast calculation of interaction tensors in clustering-based homogenization. *Comput Mech* 64(2):351–364

150. Cavaliere F, Reese S, Wulffinghoff S (2020) Efficient two-scale simulations of engineering structures using the Hashin-Shtrikman type finite element method. *Comput Mech* 65(1):159–175. <https://doi.org/10.1007/s00466-019-01758-4>

151. Jaworek D, Waimann J, Gierden C, Wulffinghoff S, Reese S (2020) A Hashin-Shtrikman type semi-analytical homogenization procedure in multiscale modeling to account for coupled problems. *Technische Mechanik*. https://doi.org/10.24352/UB_OVGU-2020-012. Artwork Size: 0,41 MB Medium: application/pdf Publisher: Otto von Guericke University Library, Magdeburg, Germany

152. Castrogiovanni A, Marfia S, Auricchio F, Sacco E (2021) TFA and HS based homogenization techniques for nonlinear

composites. *Int J Solids Struct* 225:111050. <https://doi.org/10.1016/j.ijsolstr.2021.111050>

153. Waimann J, Gierden C, Schmidt A, Svendsen B, Reese S (2021) Microstructure simulation using self-consistent clustering analysis. *PAMM* 20(1). <https://doi.org/10.1002/pamm.202000263>

154. Vlassis NN, Sun W (2023) Denoising diffusion algorithm for inverse design of microstructures with fine-tuned nonlinear material properties. *Comput Methods Appl Mech Eng* 413:116126

155. Tutumluer E, Seyhan U (1998) Neural network modeling of anisotropic aggregate behavior from repeated load triaxial tests. *Transp Res* 1615(1):86–93

156. Shin HS, Pande GN (2003) Identification of elastic constants for orthotropic materials from a structural test. *Comput Geotech* 30(7):571–577. [https://doi.org/10.1016/S0266-352X\(03\)00062-4](https://doi.org/10.1016/S0266-352X(03)00062-4)

157. Asteris P, Roussis P, Douvika M (2017) Feed-forward neural network prediction of the mechanical properties of sandcrete materials. *Sensors* 17(6):1344. <https://doi.org/10.3390/s17061344>

158. Yan W, Lin S, Kafka OL, Lian Y, Yu C, Liu Z, Yan J, Wolff S, Wu H, Ndip-Agbor E, Mozaffar M, Ehmann K, Cao J, Wagner GJ, Liu WK (2018) Data-driven multi-scale multi-physics models to derive process-structure-property relationships for additive manufacturing. *Comput Mech* 61(5):521–541. <https://doi.org/10.1007/s00466-018-1539-z>

159. Huber N (2018) Connections between topology and macroscopic mechanical properties of three-dimensional open-pore materials. *Front Mater* 5:69. <https://doi.org/10.3389/fmats.2018.00069>

160. Yang Z, Yabansu YC, Al-Bahrani R, Liao W-K, Choudhary AN, Kalidindi SR, Agrawal A (2018) Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets. *Comput Mater Sci* 151:278–287

161. Messner MC (2020) Convolutional neural network surrogate models for the mechanical properties of periodic structures. *J Mech Des* 142(2):024503. <https://doi.org/10.1115/1.4045040>

162. Rao C, Liu Y (2020) Three-dimensional convolutional neural network (3d-CNN) for heterogeneous material homogenization. *Comput Mater Sci* 184:109850

163. Chen C-T, Gu GX (2021) Learning hidden elasticity with deep neural networks. *Proc Natl Acad Sci* 118(31):2102721118. <https://doi.org/10.1073/pnas.2102721118>

164. Ni B, Gao H (2021) A deep learning approach to the inverse problem of modulus identification in elasticity. *MRS Bull* 46(1):19–25. <https://doi.org/10.1557/s43577-020-00006-y>

165. Mianroodi JR, Rezaei S, Siboni NH, Xu B-X, Raabe D (2021) Lossless multi-scale constitutive elastic relations with artificial intelligence. [arXiv:2108.02837](https://arxiv.org/abs/2108.02837)[cond-mat.]

166. Ibanez R, Abisset-Chavanne E, Aguado JV, Gonzalez D, Cueto E, Chinesta F (2018) A manifold learning approach to data-driven computational elasticity and inelasticity. *Arch Comput Methods Eng* 25(1):47–57

167. Leygue A, Coret M, Réthoré J, Stainier L, Verron E (2018) Data-based derivation of material response. *Comput Methods Appl Mech Eng* 331:184–196. <https://doi.org/10.1016/j.cma.2017.11.013>

168. Stainier L, Leygue A, Ortiz M (2019) Model-free data-driven methods in mechanics: material data identification and solvers. *Comput Mech* 64(2):381–393

169. Dalémat M, Coret M, Leygue A, Verron E (2019) Measuring stress field without constitutive equation. *Mech Mater* 136:103087

170. Holzapfel GA (2002) Nonlinear solid mechanics: a continuum approach for engineering science. Kluwer Academic Publishers, Dordrecht

171. Shen Y, Chandrashekara K, Breig W, Oliver L (2004) Neural network based constitutive model for rubber material. *Rubber Chem Technol* 77(2):257–277

172. Liang G, Chandrashekara K (2008) Neural network based constitutive model for elastomeric foams. *Eng Struct* 30(7):2002–2011

173. Ogden RW (1997) Non-linear Elastic Deformations. Courier Corporation

174. Sagiyama K, Garikipati K (2019) Machine learning materials physics: Deep neural networks trained on elastic free energy data from martensitic microstructures predict homogenized stress fields with high accuracy. [arXiv:1901.00524](https://arxiv.org/abs/1901.00524)

175. Vlassis NN, Ma R, Sun W (2020) Geometric deep learning for computational mechanics part I: anisotropic hyperelasticity. *Comput Methods Appl Mech Eng* 371:113299

176. Czarnecki WM, Osindero S, Jaderberg M, Swirszcz G, Pascanu R (2017) Sobolev training for neural networks. In: Advances in neural information processing systems. pp 4278–4287

177. Klein DK, Fernández M, Martin RJ, Neff P, Weeger O (2022) Polyconvex anisotropic hyperelasticity with neural networks. *J Mech Phys Solids* 159:104703

178. As'ad F, Avery P, Farhat C (2022) A mechanics-informed artificial neural network approach in data-driven constitutive modeling. *Int J Numer Methods Eng* 123(12):2738–2759

179. Thakolkaran P, Joshi A, Zheng Y, Flaschel M, De Lorenzis L, Kumar S (2022) Nn-euclid: Deep-learning hyperelasticity without stress data. *J Mech Phys Solids* 169:105076.

180. Tac V, Costabal FS, Tepole AB (2022) Data-driven tissue mechanics with polyconvex neural ordinary differential equations. *Comput Methods Appl Mech Eng* 398:115248

181. Vlassis NN, Zhao P, Ma R, Sewell T, Sun W (2021) Md-inferred neural network monoclinic finite-strain hyperelasticity models for β -hmx: sobolev training and validation against physical constraints. [arXiv:2112.02077](https://arxiv.org/abs/2112.02077)

182. Fernández M, Jamshidian M, Böhlke T, Kersting K, Weeger O (2021) Anisotropic hyperelastic constitutive models for finite deformations combining material theory and data-driven approaches with application to cubic lattice metamaterials. *Comput Mech* 67:653–677

183. Fernández M, Fritzen F, Weeger O (2022) Material modeling for parametric, anisotropic finite strain hyperelasticity based on machine learning with application in optimization of metamaterials. *Int J Numer Methods Eng* 123(2):577–609

184. Yang H, Guo X, Tang S, Liu WK (2019) Derivation of heterogeneous material laws via data-driven principal component expansions. *Comput Mech* 64(2):365–379

185. Chung I, Im S, Cho M (2021) A neural network constitutive model for hyperelasticity based on molecular dynamics simulations. *Int J Numer Methods Eng* 122(1):5–24

186. Im S, Kim H, Kim W, Cho M (2021) Neural network constitutive model for crystal structures. *Comput Mech* 67(1):185–206

187. Fuhr JN, Marino M, Bouklas N (2022) Local approximate gaussian process regression for data-driven constitutive models: development and comparison with neural networks. *Comput Methods Appl Mech Eng* 388:114217

188. Latorre M, Montáns FJ (2020) Experimental data reduction for hyperelasticity. *Comput Struct* 232:105919

189. Frankel AL, Jones RE, Swiler LP (2020) Tensor basis gaussian process models of hyperelastic materials. *J Mach Learn Model Comput* 1(1)

190. Haupt P (2002) Continuum mechanics and theory of materials. Springer, Berlin

191. Fuhr J, Bouklas N, Jones R (2024) Stress representations for tensor basis neural networks: alternative formulations to finger-rivlin-ericksen. *J Comput Inf Sci Eng* 1–39

192. Schoenauer M, Sebag M, Jouve F, Lamy B, Maitournam H (1996) Evolutionary identification of macro-mechanical models 22

193. Abdusalamov R, Hillgärtner M, Itskov M (2023) Automatic generation of interpretable hyperelastic material models by symbolic regression. *Int J Numer Methods Eng* 7203. <https://doi.org/10.1002/nme.7203>

194. Joshi A, Thakolkaran P, Zheng Y, Escande M, Flaschel M, De Lorenzis L, Kumar S (2022) Bayesian-euclid: Discovering hyperelastic material laws with uncertainties. *Comput Methods Appl Mech Eng* 398:115225. <https://doi.org/10.1016/j.cma.2022.115225>

195. Flaschel M, Yu H, Reiter N, Hinrichsen J, Budday S, Steinmann P, Kumar S, De Lorenzis L (2023) Automated discovery of interpretable hyperelastic material models for human brain tissue with EUCLID. *J Mech Phys Solids* 180:105404. <https://doi.org/10.1016/j.jmps.2023.105404>

196. Linka K, Pierre SRS, Kuhl E (2023) Automated model discovery for human brain using constitutive artificial neural networks. *Acta Biomater* 160:134–151

197. Fuhr JN, Jones RE, Bouklas N (2024) Extreme sparsification of physics-augmented neural networks for interpretable model discovery in mechanics. *Comput Methods Appl Mech Eng* 426:116973

198. Nguyen LTK, Keip M-A (2018) A data-driven approach to nonlinear elasticity. *Comput Struct* 194:97–115

199. Conti S, Müller S, Ortiz M (2020) Data-driven finite elasticity. *Arch Ration Mech Anal* 1–33

200. Platzer A, Leygue A, Stainier L, Ortiz M (2021) Finite element solver for data-driven finite strain elasticity. *Comput Methods Appl Mech Eng* 379:113756

201. He Q, Laurence DW, Lee C-H, Chen J-S (2021) Manifold learning based data-driven modeling for soft biological tissues. *J Biomed* 117:110124

202. He X, He Q, Chen J-S (2021) Deep autoencoders for physics-constrained data-driven nonlinear materials modeling. *Comput Methods Appl Mech Eng* 385:114034

203. Bahmani B, Sun W (2023) Distance-preserving manifold denoising for data-driven mechanics. *Comput Methods Appl Mech Eng* 405:115857

204. Rivlin RS (1972) Materials with memory. Technical report of the office of naval research (December)

205. Wu W, Kolymbas D (2000) Hypoplasticity then and now. In: *Constitutive modelling of granular materials*. Springer, Berlin, pp 57–105

206. Hill R (1998) The mathematical theory of plasticity, vol 11. Oxford University Press, Oxford

207. Housby GT, Puzrin AM (2007) Principles of hyperplasticity: an approach to plasticity theory based on thermodynamic principles. Springer, Berlin

208. Yu M-H (2006) Generalized plasticity. Springer, Berlin

209. Wu W, Bauer E (1994) A simple hypoplastic constitutive model for sand. *Int J Numer Anal Meth Geomech* 18(12):833–862

210. Ghaboussi J, Garrett JH, Wu X (1990) Material modeling with neural networks. In: *Proceedings of the international conference on numerical methods in engineering: theory and applications*. pp 701–717

211. Ghaboussi J, Garrett J Jr, Wu X (1991) Knowledge-based modeling of material behavior with neural networks. *J Eng Mech* 117(1):132–153

212. Wu X, Ghaboussi J (1990) Representation of material behavior: neural network-based models. In: *1990 IJCNN international joint conference on neural networks*. IEEE, pp 229–234

213. Wu X (1991) Neural network-based material modeling. PhD thesis, University of Illinois at Urbana-Champaign

214. Ghaboussi J (1992) Potential applications of neuro-biological computational models in geotechnical engineering. In: *Numerical models in geotechnics*. pp 543–555

215. Ghaboussi J (1992) Neuro-biological computational models with learning capabilities and their applications in geomechanical modeling. In: *Proceedings, workshop on recent accomplishments and future trends in geomechanics in the 21st century*

216. Ellis GW, Yao C, Zhao R (1992) Neural network modeling of the mechanical behavior of sand. In: *Engineering mechanics*. ASCE, pp 421–424

217. Pidaparti R, Palakal M (1993) Material model for composites using neural networks. *AIAA J* 31(8):1533–1535

218. Penumadu D, Jin-Nan L, Chameau J-L, Arumugam S (1994) Rate dependent behavior of clays using neural networks. In: *International conference on soil mechanics and foundation engineering*. pp 1445–1448

219. Amorosi A, Rampello S, Millar D (1996) On the use of artificial neural networks as generic descriptors of geomaterial mechanical behaviour. In: *ISRM international symposium-EUROCK 96*. OnePetro

220. Ghaboussi J, Sidarta D, Lade P (1994) Neural network based modelling in geomechanics. In: *International conference on computer methods and advances in geomechanics*. pp 153–164

221. Sikora Z, Ossowski R, Ichikawa Y, Tkacz K (1998) Neural networks as a tool for constitutive modelling. *Localization and Bifurcation Theory for Soils and Rocks*, Balkema, Rotterdam

222. Millar D, Clarici E (1994) Investigation of back-propagation artificial neural networks in modelling the stress-strain behaviour of sandstone rock. In: *Proceedings of 1994 IEEE international conference on neural networks (ICNN'94)*, vol 5. IEEE, pp 3326–3331

223. Millar DL, Calderbank PA (1995) On the investigation of a multilayer feedforward neural network model of rock deformability behaviour. In: *8th ISRM congress*. OnePetro

224. Logar J, Turk G (1997) Neural network as a constitutive model of soil. *Zeitschrift Fur Angewandte Mathematik Und Mechanik* 77:195–196

225. Wu X, Ghaboussi J (1993) Modelling unloading mechanism and cyclic behavior of concrete with adaptive neural networks. In: *Proceedings, second Asian-Pacific conference on computational mechanics*. Sydney, Australia

226. Pernot S, Lamarque C-H (1999) Application of neural networks to the modelling of some constitutive laws. *Neural Netw* 12(2):371–392

227. Hadjigeorgiou J, Lessard LS (1995) Predicting joint behavior using artificial neural networks. In: *CAMI'95—computer applications in the mineral industry: proceedings of the third Canadian conference on computer applications in the mineral industry*

228. Zhao H, Huang Z, Zou Z (2014) Simulating the stress-strain relationship of geomaterials by support vector machine. In: *Mathematical problems in engineering 2014*

229. Shen J, Zhou X (2015) Least squares support vector machine for constitutive modeling of clay. *Int J Eng* 28(11):1571–1578

230. Javadi A, Rezania M (2009) Applications of artificial intelligence and data mining techniques in soil modeling. *Geomech Eng* 1(1):53–74

231. Javadi AA, Rezania M (2009) Intelligent finite element method: an evolutionary approach to constitutive modeling. *Adv Eng Inform* 23(4):442–451

232. Faramarzi A, Alani AM, Javadi AA (2014) An epr-based self-learning approach to material modelling. *Comput Struct* 137:63–71

233. Joghataie A (1995) Learning and architecture determination through automatic node generation. In: *Proceedings of international conference on artificial neural networks in engineering*, St Louis, November

234. Ghaboussi J, Zhang M, Wu X, Pecknold D (1997) Nested adaptive neural network: A new architecture. In: *Proceeding*,

international conference on artificial neural networks in engineering. pp 67–72

235. Ghaboussi J, Sidarta D (1998) A new nested adaptive neural network for modeling of constitutive behavior of materials. *Int J Comput Geotech* 22(1):29–51

236. Fu Q, Hashash YM, Jung S, Ghaboussi J (2007) Integration of laboratory testing and constitutive modeling of soils. *Comput Geotech* 34(5):330–345

237. Ghaboussi J, Pecknold DA, Zhang M, Haj-Ali RM (1998) Auto-progressive training of neural network constitutive models. *Int J Numer Methods Eng* 42(1):105–126

238. Sidarta D, Ghaboussi J (1998) Constitutive modeling of geomaterials from non-uniform material tests. *Comput Geotech* 22(1):53–71

239. Shin H, Pande G (2000) On self-learning finite element codes based on monitored response of structures. *Comput Geotech* 27(3):161–178

240. Pande G, Shin H (2002) Finite elements with artificial intelligence. In: Eighth international symposium on numerical models in geomechanics-NUMOG VIII. pp 241–246

241. Hashash Y, Ghaboussi J, Jung S, Marulanda C (2002) Systematic update of a numerical model of a deep excavation using field performance data. In: 8th International symposium on numerical models in geomechanics, NUMOG 2002. CRC Press/Balkema, pp 515–522

242. Jung S, Ghaboussi J (2006) Characterizing rate-dependent material behaviors in self-learning simulation. *Comput Methods Appl Mech Eng* 196(1–3):608–619

243. Yun GJ, Saleeb A, Shang S, Binienda W, Menzemer C (2012) Improved selfsim for inverse extraction of nonuniform, nonlinear, and inelastic material behavior under cyclic loadings. *J Aerosp Eng* 25(2):256–272

244. Furukawa T (1997) A neural constitutive model for viscoplasticity. In: International conference on computational engineering science, Costa Rica. pp 453–458

245. Furukawa T, Yagawa G (1998) Implicit constitutive modelling for viscoplasticity using neural networks. *Int J Numer Methods Eng* 43(2):195–219

246. Haj-Ali R, Pecknold D, Ghaboussi J (1998) Micromechanics-based constitutive damage models for composite materials using artificial neural-networks. Modeling and simulation based engineering. 551–557

247. Haj-Ali R, Pecknold DA, Ghaboussi J, Voyiadjis GZ (2001) Simulated micromechanical models using artificial neural networks. *J Eng Mech* 127(7):730–738

248. Yun GJ, Ghaboussi J, Elnashai AS (2008) A new neural network-based model for hysteretic behavior of materials. *Int J Numer Methods Eng* 73(4):447–469

249. Huang D, Fuhr JN, Weißenfels C, Wriggers P (2020) A machine learning based plasticity model using proper orthogonal decomposition. *Comput Methods Appl Mech Eng* 365:113008

250. Unger JF, Könke C (2008) Coupling of scales in a multiscale simulation using neural networks. *Comput Struct* 86(21–22):1994–2003. <https://doi.org/10.1016/j.compstruc.2008.05.004>

251. Lefik M, Schrefler BA (2003) Artificial neural network as an incremental non-linear constitutive model for a finite element code. *Comput Methods Appl Mech Eng* 192(28–30):3265–3283

252. Xu K, Huang DZ, Darve E (2021) Learning constitutive relations using symmetric positive definite neural networks. *J Comput Phys* 428:110072

253. Masi F, Stefanou I, Vannucci P, Maffi-Berthier V (2021) Thermodynamics-based artificial neural networks for constitutive modeling. *J Mech Phys Solids* 147:104277

254. Masi F, Stefanou I (2022) Multiscale modeling of inelastic materials with thermodynamics-based artificial neural networks (TANN). *Comput Methods Appl Mech Eng* 398:115190

255. Ellis G, Yao C, Zhao R, Penumadu D (1995) Stress-strain modeling of sands using artificial neural networks. *J Geotech Eng* 121(5):429–435

256. Jordan MI (1990) Attractor dynamics and parallelism in a connectionist sequential machine. In: Artificial neural networks: concept learning. pp 112–127

257. Najjar YM, Basheer IA (1996) Discussion: stress-strain modeling of sands using artificial neural networks. *J Geotech Eng* 122(11):949–951

258. Penumadu D, Chameau JL (1997) Geomaterial modeling using neural networks. In: Artificial neural networks for civil engineering: fundamentals and applications. pp 160–184

259. Penumadu D, Zhao R (1999) Triaxial compression behavior of sand and gravel using artificial neural networks (ANN). *Comput Geotech* 24(3):207–230

260. Penumadu D, Zhao R, Frost D (2000) Virtual geotechnical laboratory experiments using a simulator. *Int J Numer Anal Methods Geomech* 24(5):439–451

261. Zhu J-H, Zaman MM, Anderson SA (1998) Modeling of soil behavior with a recurrent neural network. *Can Geotech J* 35(5):858–872

262. Zhu J-H, Zaman MM, Anderson SA (1998) Modelling of shearing behaviour of a residual soil with recurrent neural network. *Int J Numer Anal Methods Geomech* 22(8):671–687

263. Elman JL (1990) Finding structure in time. *Cognit Sci* 14(2):179–211

264. Habibagahi G, Bamdad A (2003) A neural network framework for mechanical behavior of unsaturated soils. *Can Geotech J* 40(3):684–693

265. Najjar YM, Huang C (2007) Simulating the stress-strain behavior of georgia kaolin via recurrent neuronet approach. *Comput Geotech* 34(5):346–361

266. Romo MP, García SR, Mendoza MJ, Taboada-Urtzuástegui V (2001) Recurrent and constructive-algorithm networks for sand behavior modeling. *Int J Geomech* 1(4):371–387

267. Najjar Y, Zhang X (2002) Simulating the 3d stress-strain response of canadian river sand via dynamic neuro-mechanistic approach. In: Numerical models in geomechanics: proceedings of the 8th international symposium NUMOG VIII, Rome, Italy, 10–12 April 2002. CRC Press, p 247

268. Basheer I, Najjar Y (1998) Modeling cyclic constitutive behavior by neural networks: Theoretical and real data. In: Proceedings of the 12th engineering mechanics conference, La Jolla, California. pp 952–955

269. Basheer IA (2000) Selection of methodology for neural network modeling of constitutive hystereses behavior of soils. *Comput Aided Civil Infrastruct Eng* 15(6):445–463

270. Basheer I (2002) Stress-strain behavior of geomaterials in loading reversal simulated by time-delay neural networks. *J Mater Civ Eng* 14(3):270–273

271. Oeser M, Freitag S (2009) Modeling of materials with fading memory using neural networks. *Int J Numer Methods Eng* 78(7):843–862

272. Zopf C, Kaliske M (2017) Numerical characterisation of uncured elastomers by a neural network based approach. *Comput Struct* 182:504–525

273. Wang K, Sun WC (2018) A multiscale multi-permeability poroplasticity model linked by recursive homogenizations and deep learning. *Comput Methods Appl Mech Eng* 334:337–380

274. Ghavamian F, Simone A (2019) Accelerating multiscale finite element simulations of history-dependent materials using a recurrent neural network. *Comput Methods Appl Mech Eng* 357:112594

275. Chen Q, Jia R, Pang S (2021) Deep long short-term memory neural network for accelerated elastoplastic analysis of

heterogeneous materials: an integrated data-driven surrogate approach. *Compos Struct* 264:113688

276. Frankel AL, Jones RE, Alleman C, Templeton JA (2019) Predicting the mechanical response of oligocrystals with deep learning. *Comput Mater Sci* 169:109099

277. Frankel A, Tachida K, Jones R (2020) Prediction of the evolution of the stress field of polycrystals undergoing elastic-plastic deformation with a hybrid neural network model. *Mach Learn* 1(3):035005

278. Fuchs A, Heider Y, Wang K, Sun W, Kaliske M (2021) Dnn2: A hyper-parameter reinforcement learning game for self-design of neural network based elasto-plastic constitutive descriptions. *Comput Struct* 249:106505

279. Stöcker J, Fuchs A, Leichsenring F, Kaliske M (2022) A novel self-adversarial training scheme for enhanced robustness of inelastic constitutive descriptions by neural networks. *Comput Struct* 265:106774

280. Bonatti C, Mohr D (2022) On the importance of self-consistency in recurrent neural network models representing elasto-plastic solids. *J Mech Phys Solids* 158:104697

281. Liu B, Kovachki N, Li Z, Azizzadenesheli K, Anandkumar A, Stuart A, Bhattacharya K (2021) A learning-based multiscale method and its application to inelastic impact problems. [arXiv:2102.07256](https://arxiv.org/abs/2102.07256)

282. Jones R, Frankel A, Johnson K (2021) A neural ordinary differential equation framework for modeling inelastic stress response via internal state variables. [arXiv:2111.14714](https://arxiv.org/abs/2111.14714)

283. Eggersmann R, Kirchdoerfer T, Reese S, Stainier L, Ortiz M (2019) Model-free data-driven inelasticity. *Comput Methods Appl Mech Eng* 350:81–99

284. Ladevèze P, Néron D, Gerbaud P-W (2019) Data-driven computation for history-dependent materials. *Comptes Rendus Mécanique* 347(11):831–844

285. Tang S, Li Y, Qiu H, Yang H, Saha S, Mojumder S, Liu WK, Guo X (2020) Map123-ep: a mechanistic-based data-driven approach for numerical elasto-plastic analysis. *Comput Methods Appl Mech Eng* 364:112955

286. Tang S, Yang H, Qiu H, Fleming M, Liu WK, Guo X (2021) Map123-epf: a mechanistic-based data-driven approach for numerical elasto-plastic modeling at finite strain. *Comput Methods Appl Mech Eng* 373:113484

287. Vlassis NN, Sun W (2022) Component-based machine learning paradigm for discovering rate-dependent and pressure-sensitive level-set plasticity models. *J Appl Mech* 89(2)

288. Fuhg JN, Hamel CM, Johnson K, Jones R, Bouklas N (2023) Modular machine learning-based elastoplasticity: Generalization in the context of limited data. *Comput Methods Appl Mech Eng* 407:115930

289. Furukawa T, Hoffman M (2004) Accurate cyclic plastic analysis using a neural network material model. *Eng Anal Bound Elem* 28(3):195–204

290. Jones RE, Templeton JA, Sanders CM, Ostien JT (2018) Machine learning models of plastic flow based on representation theory. *Computer Model Eng Sci* 117

291. Wang K, Sun W, Du Q (2019) A cooperative game for automated learning of elasto-plasticity knowledge graphs and models with ai-guided experimentation. *Comput Mech* 64(2):467–499

292. Stoffel M, Bamer F, Markert B (2019) Neural network based constitutive modeling of nonlinear viscoplastic structural response. *Mech Res Commun* 95:85–88

293. Settgast C, Abendroth M, Kuna M (2019) Constitutive modeling of plastic deformation behavior of open-cell foam structures using neural networks. *Mech Mater* 131(2018):1–10. <https://doi.org/10.1016/j.mechmat.2019.01.015>

294. Settgast C, Hütter G, Kuna M, Abendroth M (2020) A hybrid approach to simulate the homogenized irreversible elastic-plastic deformations and damage of foams by neural networks. *Int J Plast* 126(2019):102624. <https://doi.org/10.1016/j.ijplas.2019.11.003>. [arXiv:1910.13887](https://arxiv.org/abs/1910.13887)

295. Heider Y, Wang K, Sun W (2020) So (3)-invariance of informed-graph-based deep neural network for anisotropic elastoplastic materials. *Comput Methods Appl Mech Eng* 363:112875

296. Jang DP, Fazily P, Yoon JW (2021) Machine learning-based constitutive model for j2-plasticity. *Int J Plast* 138:102919

297. Vlassis NN, Sun W (2022) Geometric deep learning for computational mechanics part ii: Graph embedding for interpretable multiscale plasticity. [arXiv:2208.00246](https://arxiv.org/abs/2208.00246)

298. Hartmaier A (2020) Data-oriented constitutive modeling of plasticity in metals. *Materials* 13(7):1600

299. Park H, Cho M (2021) Multiscale constitutive model using data-driven yield function. *Composites Part B* 108831

300. Shoghi R, Hartmaier A (2022) Optimal data-generation strategy for machine learning yield functions in anisotropic plasticity. *Virtual Mater Des* 879614154

301. Schmidt J, Biswas A, Vajragupta N, Hartmaier A (2022) Data-oriented description of texture-dependent anisotropic material behavior. *Model Simul Mater Sci Eng*

302. Xiao M, Sun W (2022) Geometric prior of multi-resolution yielding manifolds and the local closest point projection for nearly non-smooth plasticity. *Comput Methods Appl Mech Eng* 400:115469

303. Fuhg JN, Wees L, Obstalecki M, Shade P, Bouklas N, Käsemmer M (2022) Machine-learning convex and texture-dependent macroscopic yield from crystal plasticity simulations. *Materialia* 23:101446

304. Ibáñez R, Abisset-Chavanne E, González D, Duval J-L, Cueto E, Chinesta F (2019) Hybrid constitutive modeling: data-driven learning of corrections to plasticity models. *Int J Mater Form* 12(4):717–725

305. Fuhg JN, Fau A, Bouklas N, Marino M (2023) Enhancing phenomenological yield functions with data: challenges and opportunities. *Eur J Mech A* 104925

306. Tsoi AC (1991) Application of neural network methodology to the modelling of the yield strength in a steel rolling plate mill. In: NIPS. pp 698–705

307. Hwu Y-J, Pan Y-T, Lenard JG (1996) A comparative study of artificial neural networks for the prediction of constitutive behaviour of hsla and carbon steels. *Steel Res* 67(2):59–66

308. Hodgson PD, Kong LX, Davies CH (1998) The prediction of the hot strength in steels with an integrated phenomenological and artificial neural network model. *J Mater Process Technol* 87(1–3):131–138

309. Liu J, Chang H, Hsu T, Ruan X (2000) Prediction of the flow stress of high-speed steel during hot deformation using a bp artificial neural network. *J Mater Process Technol* 103(2):200–205

310. Sun Y, Zeng W, Zhao Y, Qi Y, Ma X, Han Y (2010) Development of constitutive relationship model of ti600 alloy using artificial neural network. *Comput Mater Sci* 48(3):686–691

311. Li H-Y, Wang X-F, Wei D-D, Hu J-D, Li Y-H (2012) A comparative study on modified zerilli-armstrong, arrhenius-type and artificial neural network models to predict high-temperature deformation behavior in t24 steel. *Mater Sci Eng A* 536:216–222

312. Bobbili R, Ramakrishna B, Madhu V, Gogia A (2015) Prediction of flow stress of 7017 aluminium alloy under high strain rate compression at elevated temperatures. *Defence Technol* 11(1):93–98

313. Li X, Roth CC, Mohr D (2019) Machine-learning based temperature-and rate-dependent plasticity model: application to analysis of fracture experiments on dp steel. *Int J Plast* 118:320–344

314. Yang H, Qiu H, Xiang Q, Tang S, Guo X (2020) Exploring elasto-plastic constitutive law of microstructured materials through

artificial neural network-a mechanistic-based data-driven approach. *J Appl Mech* 87(9)

315. Shang H, Wu P, Lou Y, Wang J, Chen Q (2022) Machine learning-based modeling of the coupling effect of strain rate and temperature on strain hardening for 5182-o aluminum alloy. *J Mater Process Technol* 302:117501

316. Li X, Roth CC, Bonatti C, Mohr D (2022) Counterexample-trained neural network model of rate and temperature dependent hardening with dynamic strain aging. *Int J Plast* 151:103218

317. Zhang A, Mohr D (2020) Using neural networks to represent von mises plasticity with isotropic hardening. *Int J Plast* 132:102732

318. Desu RK, Guntuku SC, Aditya B, Gupta AK (2014) Support vector regression based flow stress prediction in austenitic stainless steel 304. *Procedia Mater Sci* 6:368–375

319. Peng J, Yamamoto Y, Hawk JA, Lara-Curzio E, Shin D (2020) Coupling physics in machine learning to predict properties of high-temperatures alloys. *NPJ Comput Mater* 6(1):1–7

320. Versino D, Tonda A, Bronkhorst CA (2017) Data driven modeling of plastic deformation. *Comput Methods Appl Mech Eng* 318:981–1004

321. Bomanito GF, Townsend TS, Stewart KM, Esham KV, Emery JM, Hochhalter JD (2021) Development of interpretable, data-driven plasticity models with symbolic regression. *Comput Struct* 252:106557. <https://doi.org/10.1016/j.compstruc.2021.106557>

322. Flaschel M, Kumar S, De Lorenzis L (2022) Discovering plasticity models without stress data. *NPJ Comput Mater* 8(1):1–10

323. Xu H, Flaschel M, De Lorenzis L (2024) Discovering non-associated pressure-sensitive plasticity models with euclid

324. Flaschel M, Kumar S, De Lorenzis L (2023) Automated discovery of generalized standard material models with euclid. *Comput Methods Appl Mech Eng* 405:115867. <https://doi.org/10.1016/j.cma.2022.115867>

325. Ciftci K, Hackl K (2021) Data-driven simulation of inelastic materials using structured data sets, tangent space information and transition rules. [arXiv:2101.10730](https://arxiv.org/abs/2101.10730)

326. Chaboche J-L (1986) Time-independent constitutive theories for cyclic plasticity. *Int J Plast* 2(2):149–188

327. Al-Haik M, Hussaini M, Garmestani H (2006) Prediction of non-linear viscoelastic behavior of polymeric composites using an artificial neural network. *Int J Plast* 22(7):1367–1392

328. Kim H-K (2008) Multi-scale nonlinear constitutive models using artificial neural networks. Georgia Institute of Technology, Georgia

329. Jung S, Ghaboussi J (2006) Neural network constitutive model for rate-dependent materials. *Comput Struct* 84(15–16):955–963

330. Kopal I, Harničárová M, Válfček J, Kušnerová M (2017) Modeling the temperature dependence of dynamic mechanical properties and visco-elastic behavior of thermoplastic polyurethane using artificial neural network. *Polymers* 9(10):519

331. Jordan B, Gorji MB, Mohr D (2020) Neural network model describing the temperature-and rate-dependent stress-strain response of polypropylene. *Int J Plast* 135:102811

332. Vu AT, Gulati S, Vogel P-A, Grunwald T, Bergs T. Physics-informed data-driven models for predicting time-and temperature-dependent viscoelastic material behaviors of optical glasses. *SSRN* 3822865

333. Basistov YA, Yanovsky YG, Danilin AN, Karnet YN (2018) Dynamic neural network as a model of viscoelastic media. *Composites* 9(4)

334. Graf W, Freitag S, Sickert J-U, Kaliske M (2012) Structural analysis with fuzzy data and neural network based material description. *Comput Aided Civil Infrastruct Eng* 27(9):640–654

335. Freitag S, Graf W, Kaliske M (2013) A material description based on recurrent neural networks for fuzzy data and its application within the finite element method. *Comput Struct* 124:29–37

336. Bhattacharya K, Liu B, Stuart A, Trautner M (2022) Learning markovian homogenized models in viscoelasticity. [arXiv:2205.14139](https://arxiv.org/abs/2205.14139)

337. Upadhyay K, Fuhg JN, Bouklas N, Ramesh K (2023) Physics-informed data-driven discovery of constitutive models with application to strain-rate-sensitive soft materials. [arXiv:2304.13897](https://arxiv.org/abs/2304.13897)

338. DeVries PM, Thompson TB, Meade BJ (2017) Enabling large-scale viscoelastic calculations via neural network acceleration. *Geophys Res Lett* 44(6):2662–2669

339. Abueidda DW, Koric S, Al-Rub RA, Parrott CM, James KA, Sobh NA (2022) A deep learning energy method for hyperelasticity and viscoelasticity. *Eur J Mech A* 95:104639

340. Xu K, Tartakovsky AM, Burghardt J, Darve E (2020) Inverse modeling of viscoelasticity materials using physics constrained learning. [arXiv:2005.04384](https://arxiv.org/abs/2005.04384)

341. Ratle A, Sebag M (2001) Grammar-guided genetic programming and dimensional consistency: application to non-parametric identification in mechanics. *Appl Soft Comput* 1(1):105–118. [https://doi.org/10.1016/S1568-4946\(01\)00009-6](https://doi.org/10.1016/S1568-4946(01)00009-6)

342. Abdusalamov R, Kaplunov J, Itskov M (2023) Discovering asymptotic expansions using symbolic regression. [arXiv:2307.01876\[physics\]](https://arxiv.org/abs/2307.01876). <http://arxiv.org/abs/2307.01876>

343. Marino E, Flaschel M, Kumar S, De Lorenzis L (2023) Automated identification of linear viscoelastic constitutive laws with euclid. *Mech Mater* 181:104643. <https://doi.org/10.1016/j.mechmat.2023.104643>

344. Erchiqui F, Ozdemir Z, Souli M, Ezzaidi H, Dituba-Ngoma G (2011) Neural networks approach for characterisation of viscoelastic polymers. *Can J Chem Eng* 89(5):1303–1310

345. Hosseini AS, Hajikarimi P, Gandomi M, Nejad FM, Gandomi AH (2021) Optimized machine learning approaches for the prediction of viscoelastic behavior of modified asphalt binders. *Constr Build Mater* 299:124264

346. Javidan MM, Kim J (2020) Experimental and numerical sensitivity assessment of viscoelasticity for polymer composite materials. *Sci Rep* 10(1):1–9

347. Saharuddin KD, Ariff MM, Bahiuddin I, Mazlan SA, Aziz SAA, Nazmi N, Fatah AYA, Mohmad K (2020) Constitutive models for predicting field-dependent viscoelastic behavior of magnetorheological elastomer using machine learning. *Smart Mater Struct* 29(8):087001

348. Wang YT, Zhang X, Liu XS (2021) Machine learning approaches to rock fracture mechanics problems: mode-I fracture toughness determination. *Eng Fract Mech* 253(March):107890. <https://doi.org/10.1016/j.engfracmech.2021.107890>

349. Kachanov L (1958) Predictive elastoplastic damage constitutive law: establishment of equivalence relation between intrinsic and extrinsic material parameters. *Izvestia Akademii Nauk SSSR, Otdelenie Tekhnicheskikh Nauk* 8:26–31

350. Abendroth M, Kuna M (2003) Determination of deformation and failure properties of ductile materials by means of the small punch test and neural networks. *Comput Mater Sci* 28:633–644. <https://doi.org/10.1016/j.commatsci.2003.08.031>

351. Abendroth M, Kuna M (2006) Identification of ductile damage and fracture parameters from the small punch test using neural networks. *Eng Fract Mech* 73(6):710–725. <https://doi.org/10.1016/j.engfracmech.2005.10.007>

352. Abbassi F, Belhadj T, Mistou S, Zghal A (2013) Parameter identification of a mechanical ductile damage using Artificial Neural Networks in sheet metal forming. *Mater Des* 45:605–615. <https://doi.org/10.1016/j.matdes.2012.09.032>

353. Unger JF, Könke C (2009) Neural networks as material models within a multiscale approach. *Comput Struct* 87(19–20):1177–1186. <https://doi.org/10.1016/j.compstruc.2008.12.003>

354. Yan S, Zou X, Ilkhani M, Jones A (2020) An efficient multiscale surrogate modelling framework for composite materials considering progressive damage based on artificial neural networks. *Composites B* 194(January):108014. <https://doi.org/10.1016/j.compositesb.2020.108014>

355. Fernández M, Rezaei S, Rezaei Mianroodi J, Fritzen F, Reese S (2020) Application of artificial neural networks for the prediction of interface mechanics: a study on grain boundary constitutive behavior. *Adv Model Simul Eng Sci* 7(1):1–27. <https://doi.org/10.1186/s40323-019-0138-7>

356. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 770–778

357. Haghigat E, Abouali S, Vaziri R (2023) Constitutive model characterization and discovery using physics-informed deep learning. *Eng Appl Artif Intell* 120:105828 <https://doi.org/10.1016/j.engappai.2023.105828>

358. Wang K, Sun W (2019) Meta-modeling game for deriving theory-consistent, microstructure-based traction-separation laws via deep reinforcement learning. *Comput Methods Appl Mech Eng* 346:216–241. <https://doi.org/10.1016/j.cma.2018.11.026>. arXiv: 1810.10535

359. Karapiperis K, Stainier L, Ortiz M, Andrade JE (2021) Data-Driven multiscale modeling in mechanics. *J Mech Phys Solids* 147(2020):104239. <https://doi.org/10.1016/j.jmps.2020.104239>

360. Theocaris PS, Panagiotopoulos PD (1993) Neural networks for computing in fracture mechanics. Methods and prospects of applications. *Comput Methods Appl Mech Eng* 106(1–2):213–228. [https://doi.org/10.1016/0045-7825\(93\)90191-Y](https://doi.org/10.1016/0045-7825(93)90191-Y)

361. Panagiotopoulos PD, Waszczyzyn Z (1999) The neural network approach in plasticity and fracture mechanics. In: Waszczyzyn Z (ed) Neural networks in the analysis and design of structures. Springer, Wien, pp 161–195

362. Liu X, Athanasiou CE, Padture NP, Sheldon BW, Gao H (2021) Knowledge extraction and transfer in data-driven fracture mechanics. *Proc Natl Acad Sci*. <https://doi.org/10.1073/pnas.2104765118>

363. Bourdin B, Francfort G, Marigo JJ (2000) Numerical experiments in revisited brittle fracture. *J Mech Phys Solids* 48(4):797–826. [https://doi.org/10.1016/S0022-5096\(99\)00028-9](https://doi.org/10.1016/S0022-5096(99)00028-9)

364. Aldakheel F, Satari R, Wriggers P (2021) Feed-forward neural networks for failure mechanics problems. *Appl Sci* (Switzerland). <https://doi.org/10.3390/app11146483>

365. Yu B et al (2018) The deep ritz method: a deep learning-based numerical algorithm for solving variational problems. *Commun Math Stat* 6(1):1–12

366. Motlagh YG, Jimack P, Borst R (2022) Deep learning phase-field model for brittle fractures. *Int J Numer Methods Eng* (June). <https://doi.org/10.1002/nme.7135>

367. Goswami S, Anitescu C, Chakraborty S, Rabczuk T (2020) Transfer learning enhanced physics informed neural network for phase-field modeling of fracture. *Theoret Appl Fract Mech* 106:102447

368. Goswami S, Anitescu C, Rabczuk T (2020) Adaptive fourth-order phase field analysis using deep energy minimization. *Theoret Appl Fract Mech* 107:102527

369. Manav M, Molinaro R, Mishra S, De Lorenzis L (2024) Phase-field modeling of fracture with physics-informed deep learning. *Comput Methods Appl Mech Eng* 429:117104. <https://doi.org/10.1016/j.cma.2024.117104>

370. Goswami S, Yin M, Yu Y, Karniadakis GE (2022) A physics-informed variational deepnet for predicting crack path in quasi-brittle materials. *Comput Methods Appl Mech Eng* 391:114587

371. Feng Y, Wang Q, Wu D, Luo Z, Chen X, Zhang T, Gao W (2021) Machine learning aided phase field method for fracture mechanics. *Int J Eng Sci* 169:1–25. <https://doi.org/10.1016/j.ijengsci.2021.103587>

372. Carrara P, De Lorenzis L, Stainier L, Ortiz M (2020) Data-driven fracture mechanics. *Comput Methods Appl Mech Eng* 372:113390. <https://doi.org/10.1016/j.cma.2020.113390>. arXiv: 2006.03133

373. Carrara P, Ortiz M, De Lorenzis L (2021) Data-driven rate-dependent fracture mechanics. *J Mech Phys Solids* 155(July):104559. <https://doi.org/10.1016/j.jmps.2021.104559>. arXiv:2103.12396

374. Paris PC, Erdogan F (1963) A critical analysis of crack propagation laws. *J Basic Eng* 85(4):528–533. <https://doi.org/10.1115/1.3656900>

375. Lee D-W, Hong S-H, Cho S-S, Joo W-S (2005) A study on fatigue damage modeling using neural networks. *J Mech Sci Technol* 19:1393–1404

376. Rovinelli A, Sangid MD, Proudhon H, Guilhem Y, Lebensohn RA, Ludwig W (2018) Predicting the 3d fatigue crack growth rate of small cracks using multimodal data via bayesian networks: in-situ experiments and crystal plasticity simulations. *J Mech Phys Solids* 115:208–229

377. Kalina KA, Gebhart P, Brummund J, Linden L, Sun W, Kästner M (2024) Neural network-based multiscale modeling of finite strain magneto-elasticity with relaxed convexity criteria. *Comput Methods Appl Mech Eng* 421:116739

378. Klein DK, Ortigosa R, Martínez-Frutos J, Weeger O (2024) Non-linear electro-elastic finite element analysis with neural network constitutive models. arXiv:2402.07007

379. Feyel F (2003) A multilevel finite element method (fe2) to describe the response of highly non-linear structures using generalized continua. *Comput Methods Appl Mech Eng* 192(28–30):3233–3244

380. Kanouté P, Boso D, Chaboche J-L, Schrefler B (2009) Multiscale methods for composites: a review. *Arch Comput Methods Eng* 16:31–75

381. Dehghani H, Zilian A (2020) Poroelastic model parameter identification using artificial neural networks: on the effects of heterogeneous porosity and solid matrix poisson ratio. *Comput Mech* 66(3):625–649

382. Dehghani H, Zilian A (2021) Ann-aided incremental multiscale-remodelling-based finite strain poroelasticity. *Comput Mech* 1–24

383. Ashworth M, Elsheikh AH, Doster F (2022) Machine learning-based multiscale constitutive modelling: development and application to dual-porosity mass transfer. *Adv Water Resour* 163:104166

384. Heider Y, Suh HS, Sun W (2021) An offline multi-scale unsaturated poromechanics model enabled by self-designed/self-improved neural networks. In: International journal for numerical and analytical methods in geomechanics

385. França FO (2018) A greedy search tree heuristic for symbolic regression. *Inf Sci* 442:18–32

386. Petersen BK, Landajuela M, Mundhenk TN, Santiago CP, Kim SK, Kim JT (2019) Deep symbolic regression: Recovering mathematical expressions from data via risk-seeking policy gradients. arXiv:1912.04871

387. Fuhr JN, Jadoon A, Weeger O, Seidl DT, Jones RE (2024) Poly-convex neural network models of thermoelasticity. arXiv:2404.15562

388. Bahmani B, Sun W (2021) A kd-tree-accelerated hybrid data-driven/model-based approach for poroelasticity problems with multi-fidelity multi-physics data. *Comput Methods Appl Mech Eng* 382:113868

389. Dafalias YF, Manzari MT (2004) Simple plasticity sand model accounting for fabric change effects. *J Eng Mech* 130(6):622–634

390. Sun X, Bahmani B, Vlassis NN, Sun W, Xu Y (2022) Data-driven discovery of interpretable causal relations for deep learning material laws with uncertainty propagation. *Granular Matter* 24(1):1–32

391. Schwer LE (2007) An overview of the ptc 60/v & v 10: guide for verification and validation in computational solid mechanics: transmitted by le schwer, chair ptc 60/v & v 10. *Eng Comput* 23(4):245–252

392. Sargent RG (2009) Verification and validation of simulation models. In: Proceedings of the 2009 winter simulation conference (WSC). IEEE, pp 162–176

393. Wallace DR, Fujii RU (1989) Software verification and validation: an overview. *IEEE Softw* 6(3):10–17

394. Thacker BH, Doebling SW, Hemez FM, Anderson MC, Pepin JE, Rodriguez EA (2004) Concepts of model verification and validation

395. Zienkiewicz OC, Taylor RL, Zhu JZ (2005) The finite element method: its basis and fundamentals. Elsevier, New York

396. Hughes TJR (2008) The finite element method: linear static and dynamic finite element analysis. Courier Corporation

397. Roache PJ (1998) Verification and Validation in Computational Science and Engineering, vol 895. Hermosa Albuquerque, NM

398. Jones AC, Wilcox RK (2008) Finite element analysis of the spine: towards a framework of verification, validation and sensitivity analysis. *Med Eng Phys* 30(10):1287–1304

399. Henninger HB, Reese SP, Anderson AE, Weiss JA (2010) Validation of computational models in biomechanics. *Proc Inst Mech Eng H* 224(7):801–812

400. Farrell K, Oden JT, Faghihi D (2015) A bayesian framework for adaptive selection, calibration, and validation of coarse-grained models of atomistic systems. *J Comput Phys* 295:189–208

401. Farrell-Maupin K, Oden J (2017) Adaptive selection and validation of models of complex systems in the presence of uncertainty. *Res Math Sci* 4(1):14

402. Ying X (2019) An overview of overfitting and its solutions. In: *Journal of physics: conference series*, vol 1168. IOP Publishing, Bristol, p 022022

403. Ghoshgoh B, Crowley M (2019) The theory behind overfitting, cross validation, regularization, bagging, and boosting: tutorial. [arXiv:1905.12787](https://arxiv.org/abs/1905.12787)

404. Bühlmann P, Van De Geer S (2011) Statistics for high-dimensional data: methods, theory and applications. Springer, Berlin

405. Ding J, Hu X-H, Gudivada V (2017) A machine learning based framework for verification and validation of massive scale image data. *IEEE Trans Big Data* 7(2):451–467

406. Pei K, Zhu L, Cao Y, Yang J, Vondrick C, Jana S (2017) Towards practical verification of machine learning: the case of computer vision systems. [arXiv:1712.01785](https://arxiv.org/abs/1712.01785)

407. Xiang W, Musau P, Wild AA, Lopez DM, Hamilton N, Yang X, Rosenfeld J, Johnson TT (2018) Verification for machine learning, autonomy, and neural networks survey. [arXiv:1810.01989](https://arxiv.org/abs/1810.01989)

408. Huang X, Kwiatkowska M, Wang S, Wu M (2017) Safety verification of deep neural networks. In: Computer aided verification: 29th international conference, CAV 2017, Heidelberg, Germany, July 24–28, 2017, Proceedings, Part I 30. Springer, Berlin, pp 3–29

409. Carlini N, Athalye A, Papernot N, Brendel W, Rauber J, Tsipras D, Goodfellow I, Madry A, Kurakin A (2019) On evaluating adversarial robustness. [arXiv:1902.06705](https://arxiv.org/abs/1902.06705)

410. Ruan W, Huang X, Kwiatkowska M (2018) Reachability analysis of deep neural networks with provable guarantees. [arXiv:1805.02242](https://arxiv.org/abs/1805.02242)

411. Huang X, Kroening D, Ruan W, Sharp J, Sun Y, Thamo E, Wu M, Yi X (2020) A survey of safety and trustworthiness of deep neural networks: verification, testing, adversarial attack and defence, and interpretability. *Comput Sc Rev* 37:100270

412. Conti S, Müller S, Ortiz M (2018) Data-driven problems in elasticity. *Arch Ration Mech Anal* 229(1):79–123

413. Vabalas A, Gowen E, Poliakoff E, Casson AJ (2019) Machine learning algorithm validation with a limited sample size. *PLoS ONE* 14(11):0224365

414. Bergmeir C, Benítez JM (2012) On the use of cross-validation for time series predictor evaluation. *Inf Sci* 191:192–213

415. Hoerl AE, Kennard RW (1970) Ridge regression: biased estimation for nonorthogonal problems. *Technometrics* 12(1):55–67

416. Van Laarhoven, T (2017) L2 regularization versus batch and weight normalization. [arXiv:1706.05350](https://arxiv.org/abs/1706.05350)

417. Loshchilov I, Hutter F (2017) Decoupled weight decay regularization. [arXiv:1711.05101](https://arxiv.org/abs/1711.05101)

418. Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. *J R Stat Soc Ser B* 67(2):301–320

419. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 15(1):1929–1958

420. Kalina KA, Linden L, Brummund J, Metsch P, Kästner M (2022) Automated constitutive modeling of isotropic hyperelasticity based on artificial neural networks. *Comput Mech* 69(1):213–232

421. González D, Chinesta F, Cueto E (2019) Thermodynamically consistent data-driven computational mechanics. *Continuum Mech Thermodyn* 31(1):239–253

422. Stutz D, Hein M, Schiele B (2019) Disentangling adversarial robustness and generalization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp 6976–6987

423. Wu D, Xia S-T, Wang Y (2020) Adversarial weight perturbation helps robust generalization. *Adv Neural Inf Process Syst* 33:2958–2969

424. Raghunathan A, Xie SM, Yang F, Duchi JC, Liang P (2019) Adversarial training can hurt generalization. [arXiv:1906.06032](https://arxiv.org/abs/1906.06032)

425. Blalock D, Gonzalez Ortiz JJ, Frankle J, Guttag J (2020) What is the state of neural network pruning? *Proc Mach Learn Syst* 2:129–146

426. Meyer KA, Ekre F (2023) Thermodynamically consistent neural network plasticity modeling and discovery of evolution laws. <https://doi.org/10.31224/2961>

427. Torens C, Juenger F, Schirmer S, Schopferer S, Maienschein TD, Dauer JC (2022) Machine learning verification and safety for unmanned aircraft-a literature study. In: AIAA Scitech 2022 Forum, p 1133

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.