



Delft University of Technology

HyperCAN

Hypernetwork-driven deep parameterized constitutive models for metamaterials

Zheng, Li; Kochmann, Dennis M.; Kumar, Siddhant

DOI

[10.1016/j.eml.2024.102243](https://doi.org/10.1016/j.eml.2024.102243)

Publication date

2024

Document Version

Final published version

Published in

Extreme Mechanics Letters

Citation (APA)

Zheng, L., Kochmann, D. M., & Kumar, S. (2024). HyperCAN: Hypernetwork-driven deep parameterized constitutive models for metamaterials. *Extreme Mechanics Letters*, 72, Article 102243. <https://doi.org/10.1016/j.eml.2024.102243>

Important note

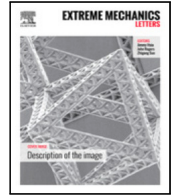
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

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

Takedown policy

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



HyperCAN: Hypernetwork-driven deep parameterized constitutive models for metamaterials

Li Zheng^a, Dennis M. Kochmann^a, Siddhant Kumar^{b,*}

^a Mechanics & Materials Laboratory, Department of Mechanical and Process Engineering, ETH Zürich, 8092, Switzerland

^b Department of Materials Science and Engineering, Delft University of Technology, 2628 CD Delft, Netherlands

ARTICLE INFO

Dataset link: <https://doi.org/10.3929/ethz-b-000699994>

Keywords:

Metamaterials
Constitutive modeling
Machine learning
Hypernetworks
Hyperelasticity

ABSTRACT

We introduce HyperCAN, a machine learning framework that utilizes hypernetworks to construct adaptable constitutive artificial neural networks for a wide range of beam-based metamaterials exhibiting diverse mechanical behavior under finite deformations. HyperCAN integrates an input convex neural network that models the nonlinear stress-strain map of a truss lattice, while ensuring adherence to fundamental mechanics principles, along with a hypernetwork that dynamically adjusts the parameters of the convex network as a function of the lattice topology and geometry. This unified framework demonstrates robust generalization in predicting the mechanical behavior of previously unseen metamaterial designs and loading scenarios well beyond the training domain. We show how HyperCAN can be integrated into multiscale simulations to accurately capture the highly nonlinear responses of large-scale truss metamaterials, closely matching fully resolved simulations while significantly reducing computational costs. This offers new efficient opportunities for the multiscale design and optimization of truss metamaterials.

1. Introduction

Architected materials have unlocked immense design possibilities by manipulating their architectures across scales. With recent advances in additive manufacturing, mechanical metamaterials, featuring beam- [1,2], plate- [3], or shell-based [4–6] architectures, have revolutionized material design by enabling tailored properties across diverse applications, including ultralight structures [7,8], auxetics [9–11], impact protective systems [12,13], and acoustic waveguides [14, 15]. Periodic beam-based metamaterials, commonly referred to as truss metamaterials or truss lattices and characterized by periodic arrangements of interconnected beam networks, offer a vast design landscape for previously unattainable properties and functionality by exploiting the topological and geometrical attributes of small-scale architectures [16–20]. To fully exploit the design potential of truss metamaterials, particularly in applications such as biomimetic implants [21,22], soft robotics [23,24], and energy absorption [25,26], it is crucial to understand their mechanical behavior under large deformation. A major challenge in modeling such truss lattices is the intensive computations required to capture the responses of individual structural members in the unit cell due to the complexity arising from geometric and material nonlinearity and associated instabilities [27, 28]. This calls for effective constitutive models which, depending on

the underlying unit cell design, facilitate efficient multiscale simulation and optimization.

Traditional modeling approaches have relied on formulating effective constitutive relations based on intuitive understanding and physical assumptions [29–33], with material-dependent parameters calibrated through phenomenological observations or experimental tests. Consequently, phenomenological constitutive models are often constrained by available experimental data and a-priori assumptions, lacking the flexibility to adapt to new materials or untested complex scenarios. Over the past decade, data-driven methods have emerged as a powerful and versatile alternative to circumvent the need for explicit formulations of constitutive relations [34]. By directly harnessing datasets of strain energies or stress responses, data-driven constitutive models approximate the constitutive behavior by, e.g., polynomial [35] or spline interpolation [36,37], Gaussian process regression [38,39], or reduced-order models [40,41]. Notably, artificial neural networks (NNs) have demonstrated remarkable success in learning constitutive laws across various materials for a range of material behaviors, from small strain elasticity [42] and finite strain hyperelasticity [43–45], to plasticity [46,47] and viscoelasticity [48], demonstrating exceptional capabilities as universal approximators. Despite the efficiency of NNs in capturing intricate nonlinear material responses, unrestricted NNs may easily violate underlying mechanical constraints and therefore lack

* Corresponding author.

E-mail addresses: dmk@ethz.ch (D.M. Kochmann), sid.kumar@tudelft.nl (S. Kumar).

<https://doi.org/10.1016/j.eml.2024.102243>

Received 12 August 2024; Received in revised form 9 October 2024; Accepted 9 October 2024

Available online 11 October 2024

2352-4316/© 2024 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

physical interpretability. To overcome this limitation, there is a growing interest in integrating ML tools with physical knowledge to build constitutive models that are both data-efficient and well-grounded in physical principles [49–54].

Strategies for developing physics-informed NN constitutive models for elastic finite strains typically involve learning the hyperelastic strain energy density function W with strain measures as input, such as the deformation gradient F or the Green–Lagrange strain tensor E (equivalently, the right Cauchy–Green deformation tensor C). Their respective conjugate stress measures, the first Piola–Kirchhoff stress P and the second Piola–Kirchhoff stress S , can be derived by differentiating the strain energy density. To ensure the NN model's adherence to fundamental physical considerations while maintaining flexibility and generality, physics-informed inductive biases in selecting the parameterization and model architecture can be advantageous. For instance, by expressing the energy function using E or C as inputs, the model inherently satisfies the objectivity constraints of W per construction [50, 55, 56]. Furthermore, the ellipticity of W in F is an important consideration, as it ensures material stability [57]. In general, directly enforcing a model to be elliptic a priori is cumbersome; hence, a common approach is to instead enforce polyconvexity, which implies ellipticity while not being too strong to exclude the non-uniqueness of the energy minimizer [58, 59]. Several data-driven constitutive models based on *input convex neural networks* (ICNNs) [60] have been proposed to construct polyconvex hyperelastic potentials [6, 45, 61, 62]. By incorporating polyconvex invariants (or the deformation gradient along with its cofactor and its determinant) as inputs to the neural network, ICNNs automatically satisfy the polyconvexity constraint through their architectural design and selection of activation functions. Polyconvex NNs have proven successful in accurately predicting anisotropic responses at finite deformations for various materials and, more importantly, extrapolating to untested deformation states, demonstrating that leveraging physical principles significantly enhances the model's interpretability and generalization capabilities.

Despite their efficiency and reliability, most existing NN constitutive models have been restricted to a single type of material. Once trained on responses for one specific material (or metamaterial unit cell), the model is fixed and cannot be directly employed for different materials without retraining. This is highly inefficient when considering the tremendous design space of metamaterials and their intricate structure-to-property maps. One potential solution is to incorporate constitutive models with an explicit parameterization [60], that is, the parameterized model takes both the parameters p characterizing the material and the deformation gradient F as inputs and predicts the strain energy density function as $W = W(F, p)$. The parameterized constitutive NN allows the model to account for variations in topological [6, 49] or material properties [63] when predicting material responses, thereby broadening its applicability across different material configurations.

Constitutive modeling for truss metamaterials, however, presents unique challenges and has remained largely under-explored. Truss lattices provide an extensive and diverse design space, enabled by variations in lattice topologies (i.e., connectivity of the beam network) and geometrical features (i.e., node placements, strut cross-sections, etc.). Despite this vast potential, existing works are often confined to a small catalog of lattices [61, 64, 65] due to the lack of generalized parameterization of trusses, not fully leveraging the available design freedom. Additionally, beam lattices exhibit complex nonlinear mechanical behavior at large strains, primarily due to their geometrical nonlinearities and mechanical instabilities such as beam buckling. The latter is highly sensitive to the configurations; e.g., small changes in the unit cell topology or strut orientation can drastically shift the deformation mechanisms (stretch- vs. bending-dominated) of truss lattices [17, 66]. Such sensitivity requires the NN model to effectively learn not only the individual effects of structural features and deformation states but also their combined effects on mechanical behaviors. This becomes particularly challenging for beam networks due to their

often high-dimensional and discrete parameterization. More recently, graph representations have become prominent as efficient tools for capturing the 3D structural features of truss lattices [19, 67, 68], used, e.g., in conjunction with graph neural networks to predict the mechanical response of those structures. Nevertheless, existing studies predominantly address only linear and small-strain responses [69, 70], or nonlinear responses subject to one specific loading scenario such as uniaxial compression/tension [71, 72], thus limiting their applicability to different loading conditions. To the best of our knowledge, there exists no constitutive modeling approach that is adaptable for various truss configurations across different topological and geometrical features as well as for different loading conditions, while fulfilling key constitutive conditions.

To address the aforementioned challenges, we propose a machine learning (ML) framework for constructing effective hyperelastic constitutive models of beam-based metamaterials at finite strains, covering a broad spectrum of truss architectures and anisotropic material behavior. Distinct from existing works that directly train a static NN constitutive model with stress responses for specific lattice configurations, the goal of our generalized constitutive modeling framework is to develop adaptable constitutive models capable of predicting the challenging behavior of truss metamaterials undergoing large deformation and instabilities, as well as generalizing and extrapolating across a vast range of metamaterial architectures and loading conditions beyond the training domain. To this end, we propose *HyperCAN*, which utilizes *hypernetworks* [73] to learn constitutive artificial neural networks for various metamaterial architectures by leveraging previously acquired knowledge. Hypernetworks have shown promising results in developing NNs with greater flexibility, adaptability, and generalization in various deep learning tasks, including multitasking [74, 75], transfer learning [76], and domain adaptation [77, 78]. By employing hypernetworks to customize weights of a diverse family of constitutive NNs, our approach provides a unified framework for learning constitutive models that can efficiently adapt to unseen scenarios, bypassing the need for fine-tuning multiple constitutive models.

Our proposed HyperCAN framework consists of two main components: a hypernetwork and a constitutive neural network. The constitutive NN uses an ICNN core, which encodes important mechanical considerations to construct the hyperelastic energy potential. The architectural design of ICNNs and proper parameterization automatically guarantee thermodynamic consistency, material objectivity, a stress-free reference configuration, and local material stability. The hypernetwork takes as input a unit cell design and aims to generate as output the weights of a constitutive NN. Both networks are trained end-to-end through backpropagation, which enables the hypernetwork to capture the intricate dependencies across various truss lattices and to adjust the constitutive relations accordingly. We demonstrate that our HyperCAN approach is not only capable of accurately describing the nonlinear hyperelastic effective behavior of various truss lattices but also more flexible and adaptive, showcasing great generalization and robustness to **unseen unit cell configurations and unseen loading scenarios**. Finally, the NN constitutive model is integrated into a multiscale finite element simulation for truss metamaterials, effectively capturing the highly nonlinear macroscopic responses of truss lattices with significantly reduced computational efforts compared to fully resolved simulations. Altogether, our proposed generalized ML framework opens new avenues for constructing efficient, highly adaptable, and physically sensible data-driven constitutive models for a broad range of architected materials. This, combined with the tremendous design space of the chosen graph-based truss representation, paves the way for fully exploring the potential of the multiscale design of metamaterials.

2. Methods

2.1. Constitutive requirements for hyperelasticity

To ensure that the NN constitutive model satisfies the necessary physical constraints, let us briefly discuss the theoretical basis of hyperelastic constitutive modeling relevant to this work. A hyperelastic constitutive model describes the material behavior up to large strains and can be characterized by a strain energy density $W = W(F)$ stored within a body $\Omega \subset \mathbb{R}^3$, where $F \in \text{GL}_+(3)$ denotes the deformation gradient. The first Piola–Kirchhoff stress $P(F)$ is derived as

$$P(F) = \frac{\partial W}{\partial F}(F). \quad (1)$$

To construct a physically valid hyperelastic strain energy density, the following constitutive requirements must be fulfilled:

- In the absence of deformation (i.e., in the reference configuration where $F = I$), the stress should vanish, i.e.,

$$P(I) = 0. \quad (2)$$

- The strain energy density must be objective, i.e., W must be invariant under any rotation tensor Q (satisfying $Q^T Q = I$ and $\det Q = 1$) of the deformation gradient F :

$$W(QF) = W(F) \quad \forall F \in \text{GL}_+(3), Q \in \text{SO}(3). \quad (3)$$

- The ellipticity (also known as rank-one convexity) condition of the strain energy density

$$(\xi \otimes \eta) : \frac{\partial^2 W(F)}{\partial F^2} : (\xi \otimes \eta) \geq 0 \quad \forall \xi, \eta \in \mathbb{R}^3 \quad (4)$$

ensures pointwise material stability. However, enforcing ellipticity a priori in model formulations is challenging. Consequently, a common approach in constitutive modeling is to instead enforce polyconvexity, which implies ellipticity. Specifically, $W(F)$ is polyconvex if and only if there exists a function $P : \text{GL}_+(3) \times \text{GL}_+(3) \times \mathbb{R} \rightarrow \mathbb{R}$ such that

$$W(F) = P(F, \text{Cof } F, \det F). \quad (5)$$

Based on Eq. (5), a straightforward approach to enforce polyconvexity is to formulate W as convex functions of input $\{F, \text{Cof } F, \det F\}$. This, however, violates the material objectivity in Eq. (3), since the deformation gradient and its cofactor are not invariant to the orientation of an observer. As a remedy, data augmentation approaches that introduce randomly sampled observers can be applied to approximate the objectivity condition [61,64]. An alternative is to employ polyconvex and objective invariants as inputs [45,49,61,79], which naturally fulfills the objectivity condition. However, the model capacity strongly depends on the predefined choice of invariants and specific material symmetry groups, which poses challenges in generalizing to a wide range of anisotropic materials or unseen loading scenarios beyond the training domain.

Therefore, considering that objectivity is an essential property that must be fulfilled non-trivially, we choose to express the strain energy density as a function of the Green–Lagrange strain tensor $E = (F^T F - I)/2$, i.e., in the form of $W = W(E)$, which inherently satisfies the material objectivity condition. Furthermore, the convexity of W in E implies the local convexity of W in F (see Refs. [62,80] for proofs). This is preferable over the global convexity of W in F , as the latter excludes the non-uniqueness of solutions due to structural buckling. These considerations motivate the use of ICNNs to construct a constitutive model that outputs a convex strain energy density $W_\theta^{\text{NN}}(E)$ as a function of the Green–Lagrange strain tensor E . As such a NN constitutive model may still violate the stress-free reference configuration condition in Eq. (2) without explicit constraints, we include linear

correction terms [45,62] in the strain energy density, ensuring that the model exhibits zero stress and zero strain energy density in the absence of deformation by construction. In summary, we propose to formulate the strain energy density as

$$W(E) = W_\theta^{\text{NN}}(E) + H : E + W^0, \quad (6)$$

where $W_\theta^{\text{NN}}(E)$ represents the ICNN model parameterized by θ . The energy correction term W^0 is a constant scalar that ensures zero strain energy density at zero deformation $F = I$, or equivalently, $E = 0$, i.e.,

$$W^0 = -W_\theta^{\text{NN}}(E) \Big|_{E=0} \rightarrow W(0) = 0. \quad (7)$$

The second Piola–Kirchhoff stress tensor and the components of the tangent modulus tensor can be derived by differentiating the strain energy density as

$$S(E) = \frac{\partial W}{\partial E}(E) = \frac{\partial W_\theta^{\text{NN}}}{\partial E}(E) + H, \quad (8)$$

$$\mathbb{C}(E) = \frac{\partial S}{\partial E}(E) = \frac{\partial^2 W_\theta^{\text{NN}}}{\partial E \partial E}(E) \quad \text{with} \quad \mathbb{C}_{ijkl} = \frac{\partial^2 W_\theta^{\text{NN}}}{\partial E_{ij} \partial E_{kl}},$$

The stress correction term involving second-order tensor H is chosen such that the model is stress-free for $F = I$ (equivalently, $E = 0$), i.e.,

$$H = - \frac{\partial W_\theta^{\text{NN}}}{\partial E} \Big|_{E=0} \rightarrow S(0) = 0. \quad (9)$$

Therefore, the neural network hyperelastic constitutive model defined in Eq. (6) ensures thermodynamic consistency, preserves the local convexity of the strain energy density, and fulfills both the stress-free reference configuration condition and the objectivity condition—without over-constraining the strain energy density to admit instability (as expected, e.g., from the effective response of truss metamaterials undergoing buckling).

2.2. ICNN-based constitutive modeling

We here briefly introduce the ICNN [60] architecture, which forms the core of our NN-based constitutive model. The architectural design and activation functions of the ICNN are specially chosen to ensure that it outputs a strain energy density $W_\theta^{\text{NN}}(E)$ that exhibits local convexity, thus encoding the required physical knowledge into the constitutive model (for proofs, see Ref. [60]). Here, we consider a fully convex NN based on a multilayer perceptron (MLP) with k hidden layers. Given the deformation gradient F , the model first computes the Green–Lagrange strain tensor $E = (F^T F - I)/2$, which is then vectorized as $y = [E_{11}, E_{12}, E_{13}, E_{21}, E_{22}, E_{23}, E_{31}, E_{32}, E_{33}]^T \in \mathbb{R}^{d_0}$ (with $d_0 = 9$) and passed to the ICNN core. The i th hidden layer ($i = 1, 2, \dots, k$) of size d_i takes the output from the previous layer, $z^{(i-1)}$, and computes the output as

$$z^{(i)} = \underbrace{\phi(\sigma(\mathcal{W}_z^{(i)} z^{(i-1)}))}_{\text{convex layers}} + \underbrace{\mathcal{W}_y^{(i)} y}_{\text{passthrough layers}} + \underbrace{b^{(i)}}_{\text{bias}} \quad (10)$$

where $\mathcal{W}_z^{(i)} \in \mathbb{R}^{d_i \times d_{i-1}}$ and $b^{(i)} \in \mathbb{R}^{d_i}$ are the weight matrix and the bias vector of fully connected layers, respectively, $\mathcal{W}_y^{(i)} \in \mathbb{R}^{d_i \times d_0}$ is the weight matrix of the *passthrough* layers that directly connect the inputs $y \in \mathbb{R}^{d_0}$ to the deeper hidden layers, and ϕ and σ are nonlinear element-wise functions.

The above network ensures convexity through several key restrictions. First, all weights in the fully connected layers are constrained to be *non-negative*. This is achieved by applying a nonlinear non-negative function σ to the weight matrix $\mathcal{W}_z^{(i)}$, while allowing $\mathcal{W}_y^{(i)}$ to be negative, thereby maintaining the model's predictive power without compromising convexity. Second, all activation functions ϕ must be *convex* and *non-decreasing*. In addition, W_θ^{NN} needs to be at least twice differentiable to enable the calculation of the tangent modulus $\mathbb{C}$ as shown in Eq. (8), which requires activation functions to have non-vanishing second derivatives. The aforementioned conditions motivate

the choice of squared-softplus and softplus activation functions for ϕ and σ , respectively:

$$\phi(x) = \alpha (\log(1 + e^x))^2 \quad \text{and} \quad \sigma(x) = \beta \log(1 + e^x), \quad (11)$$

where $\alpha, \beta \in \mathbb{R}^+$ are hyperparameters that control the curvature of functions $\phi(x)$ and $\sigma(x)$, respectively. The squaring of the softplus function ϕ provides smooth, non-negative, and convex activation, ensures stable training by preventing vanishing gradients [81–83], and enhances the expressiveness and accuracy of the constitutive model, e.g., by avoiding spurious local oscillations when predicting stress responses [45,62].

In summary, we construct the constitutive model $W_\theta^{\text{NN}} = F(E; \theta)$ based on ICNNs, where $\theta = \{\mathcal{W}_z^{(i)}, \mathcal{W}_y^{(i)}, b^{(i)} : i = 1, \dots, k\}$ represents the set of unknown NN weights and biases. The strain energy density W_θ^{NN} relies solely on the objective strain measure E , thus inherently satisfying the objectivity condition in Eq. (3). By applying a non-negative transformation function σ to weights and employing convex non-decreasing activation function ϕ , the NN constitutive model preserves the local convexity of the strain energy density, thus ensuring local material stability. Additionally, passthrough connections are integrated into each hidden layer to address the limited expressiveness due to the non-negativity constraint of layer weights. The resulting mechanics-informed NN model seeks to capture the complex, intractable mechanical relations and to accurately predict the highly nonlinear material behavior at finite strains. The predictive capabilities and physical insights provided by the NN constitutive model can further be exploited to enhance the stability and efficiency of nonlinear multiscale simulations of architected materials.

2.3. Generation of the metamaterial design space with diverse nonlinear responses

To develop efficient NN-based constitutive models for various truss metamaterials, we aim to construct a design space that encompasses a wide array of truss configurations, exhibiting diverse mechanical responses. We particularly focus on beam lattices based on the 3D periodic tessellation of a cubic representative volume element (RVE). We modify the graph-based representation of truss lattices, following Ref. [19], which allows for a unified and systematic representation of periodic trusses. By assuming symmetries across three mutually orthogonal planes, the RVE is partitioned into eight equal octants. Within each octant, we define a total of 14 possible node placements: 8 vertex nodes $\{v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ and 6 face nodes $\{f_0, f_1, f_2, f_3, f_4, f_5\}$, as shown in Fig. 1a. The vertex nodes are fixed to maintain connectivity on the outer boundaries of the unit cell, whereas the face nodes can freely move within the face. Each truss can be naturally translated into a graph representation $G = (\mathbf{A}, \mathbf{x})$, where $\mathbf{A} \in \{0, 1\}^{14 \times 14}$ denotes the adjacency matrix representing all beam connections and $\mathbf{x} \in \mathbb{R}^{14 \times 3}$ denotes the node positions. Beginning with 6 elementary unit cells as initial designs, we apply an iterative stochastic perturbation algorithm to generate new lattice designs by randomly inserting or removing nodes and connections. By perturbing both the truss connectivity and node positions, we construct a database of truss lattices containing 6,000 unique structures, achieving a reasonable balance between computational efficiency and the diversity of training space. Details regarding the parameterization and creation of the truss design space can be found in Ref. [19]. Fig. 1b shows representative examples of truss structures in our dataset.

To characterize the effective nonlinear responses of truss lattices, we perform finite element (FE) homogenization with periodic boundary conditions, leveraging the open-source FE code *ae108* [84]. Each individual strut is modeled as a linear elastic corotational (Euler–Bernoulli) beam element with a constant circular cross-section and Young’s modulus $E_s = 1$ (undergoing large rotations but small strains). The beam thickness of all struts is varied to maintain a constant relative density $\rho = 0.025$ across all structures. Effective deformation gradients F are applied on average onto a unit cell through displacement-based

periodic boundary conditions for $T > 0$ quasistatic load steps at uniform strain increments. For each load step, we compute the effective strain energy density W by averaging the strain energy stored in the beams over the unit cell. The first and second Piola–Kirchhoff stress tensors can be derived from the reaction forces on the outer nodes of the unit cells. Fig. 1b shows the homogenized stress–strain responses of representative unit cells subject to uniaxial compression, exhibiting distinct nonlinear behaviors, which underscores the challenges in learning the highly complex structure-dependent deformation mechanisms. Unlike existing approaches [72,85] whose NNs serve as surrogate models to predict stress responses for specific loading paths (e.g., uni-/biaxial compression), we aim to develop generalized constitutive models capable of handling various combinations of structural configurations and loading conditions. In this study, we focus on the compressive and shear responses of truss structures, showcasing critical phenomena such as buckling instabilities and localized deformation. These characteristics are essential for assessing the constitutive model’s robustness and accuracy in predicting highly nonlinear and anisotropic mechanical behaviors of truss structures. Additionally, the constitutive model adopts a generalized formulation, enabling straightforward extension to other deformation modes. To facilitate training the constitutive model, for each structure, we consider the following deformation paths: uniaxial compression (UC), biaxial compression (BC), and simple shear (SS). The corresponding deformation gradients are given by (without summation convention)

$$\begin{aligned} \mathbf{F}^{\text{UC}}(\lambda) &= \mathbf{I} + \lambda e_i \otimes e_i, & \text{with } i \in \{1, 2, 3\}, \quad \lambda \in [-0.25, 0]; \\ \mathbf{F}^{\text{BC}}(\lambda) &= \mathbf{I} + \lambda_1 e_i \otimes e_i + \lambda_2 e_j \otimes e_j, & \text{with } i, j \in \{1, 2, 3\}, j \neq i, \quad \lambda_1, \lambda_2 \in [-0.25, 0]; \\ \mathbf{F}^{\text{SS}}(\lambda) &= \mathbf{I} + \lambda e_1 \otimes e_2, & \text{with } \lambda \in [0, 0.5], \end{aligned} \quad (12)$$

where λ is a loading parameter uniformly sampled from the given ranges. Furthermore, to assess the generalization capabilities of the proposed constitutive modeling framework, we consider an unseen validation loading scenario, which involves a more complex deformation path: triaxial compression (TC), characterized by the following deformation gradient

$$\mathbf{F}^{\text{TC}}(\lambda) = \mathbf{I} + \lambda_1 e_1 \otimes e_1 + \lambda_2 e_2 \otimes e_2 + \lambda_3 e_3 \otimes e_3, \quad \text{with } \lambda_1, \lambda_2, \lambda_3 \in [-0.25, 0]. \quad (13)$$

The validation path TC remains unseen by the model and does not contribute to the training process. For each design, we consider a total of K different loading paths, each discretized into T load steps. For each path, we simulate the effective response of unit cells and collect the resulting T values of the hyperelastic strain energy density and second Piola–Kirchhoff stress $\{(G, E_t^{(k)}, S_t^{(k)}, W_t^{(k)}) : k = 1, \dots, K; t = 1, \dots, T\}$ to construct a comprehensive structure-response dataset.

2.4. Learning parameterized constitutive models for truss metamaterials via hypernetworks

To address the complexities involved in constructing constitutive models for a broad range of truss metamaterials, we propose the HyperCAN framework, which employs a hypernetwork [73] to learn the abstract parameter space characterizing the constitutive models. Specifically, hypernetworks utilize a meta-learning strategy, which decomposes the task of learning constitutive models for various material configurations into two separate networks: a hypernetwork and a target/main network. The target network F is designed to perform the actual learning tasks, such as classification and regression, whereas the hypernetwork H dynamically generates the weights for the target network F based on input conditions. Distinct from existing works [56, 61] on NN-based constitutive modeling that directly learn and update the constitutive NN parameters θ for a specific material type during training, we propose to use the hypernetwork to predict each individual

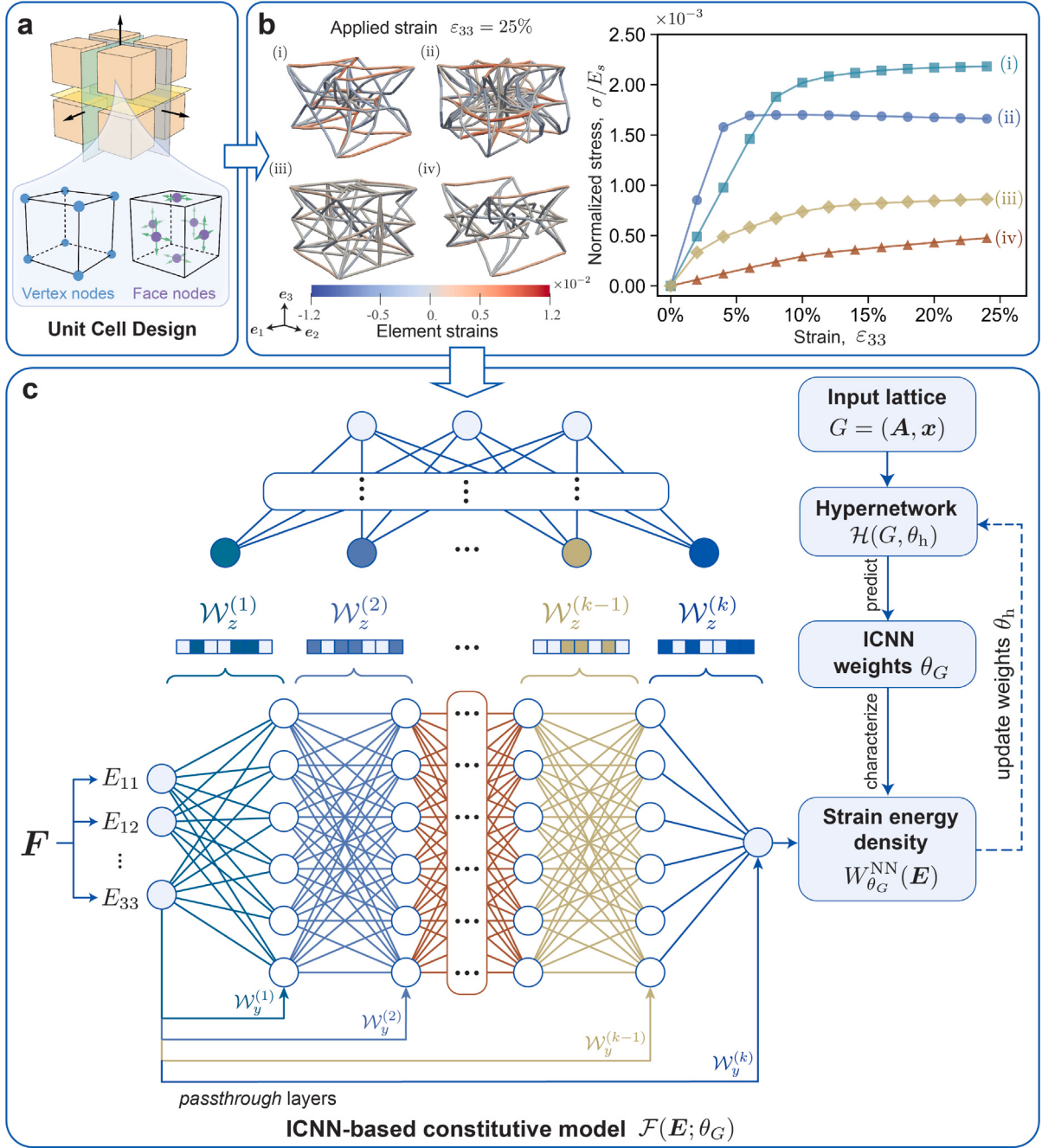


Fig. 1. Overview of the truss design space with nonlinear mechanical responses. **(a)** Cube decomposition generates the unit cell with possible node placements and the nodes' degrees of freedom defined on the octant. **(b)** Representative examples of deformed truss unit cells and their corresponding homogenized stress-strain responses under uniaxial compression along e_3 direction, showcasing the diverse design space of truss lattices with distinct deformation mechanisms. Reported stress values are normalized by the base material Young's modulus E_s . **(c)** Schematic of the HyperCAN framework for constitutive modeling of truss lattices. The hypernetwork $\mathcal{H}$ predicts the weight parameters θ_G that characterize an ICNN-based constitutive model for a given input graph representation of the truss lattice $G = (A, x)$. The constitutive model $\mathcal{F}(E; \theta_G)$ first calculates the Green-Lagrange strain measure E for given deformation gradient F , and then predicts a convex strain energy density $W_{\theta_G}^{NN}$ with respect to the vectorized input E through fully-connected layers and passthrough layers.

set of θ , including weights and biases for layers in W_{θ}^{NN} , taking the input material features that characterize, e.g., the material properties or the microstructure geometries. This reparameterization allows the hypernetwork to capture the complex parameter space for different metamaterials with a significantly reduced number of parameters [86, 87], leading to better adaptability and generalization of the target network to new, unseen data, as the hypernetwork can dynamically adjust its output weights to varying input conditions.

The schematic representation of the proposed HyperCAN framework is summarized in Fig. 1c. The target network, implemented as the ICNN-based constitutive model as described in Section 2.2, learns the strain energy density W_{θ}^{NN} while encoding necessary constitutive principles and preserving physical consistency. Instead of learning a static set of weights, the target network's parameters are generated on-the-fly by the hypernetwork conditioned on specific inputs, which allows the model to navigate the complex parameter spaces associated with

Table 1

Overview of datasets for training and comprehensive evaluation of the HyperCAN model. D^{train} is used for model training. D_L^{test} tests the model's capability to handle unseen loading scenarios on known truss structures, while D_G^{test} assesses generalization to entirely unseen truss structures under known loading conditions. $D_{G,L}^{\text{test}}$ features both unseen truss structures and loading scenarios, aiming to test the model's adaptability to completely unknown conditions. N is the number of unique truss structures, K represents the number of unique loading paths, and T is the number of loading steps per each. Deformation paths include UC (uniaxial compression), BC (biaxial compression), SS (simple shear), and TC (triaxial compression).

Dataset	Truss structures	Loading scenarios	Deformation paths	Sample size	Dataset size
D^{train}	seen	seen	UC, BC, SS	$N = 6000, K = 14, T = 100$	6,134,800
D_L^{test}	seen	unseen	UC, BC, SS, TC	$N = 6000, K = 7, T = 100$	3,163,800
D_G^{test}	unseen	seen	UC, BC, SS	$N = 3000, K = 14, T = 100$	3,035,700
$D_{G,L}^{\text{test}}$	unseen	unseen	UC, BC, TC	$N = 200, K = 500, T = 100$	4,344,800

various truss designs. That is, given an input truss lattice, represented by the graph $G = (\mathbf{A}, \mathbf{x})$, the hypernetwork $\mathcal{H}$ predicts a customized constitutive model, denoted as $W_{\theta_G}^{\text{NN}}$, for the given truss as

$$\theta_G = \mathcal{H}(G; \theta_h) \Rightarrow W_{\theta_G}^{\text{NN}} = F(E; \theta_G) = F(E; \mathcal{H}(G; \theta_h)). \quad (14)$$

The second Piola–Kirchhoff stress $\mathbf{S} = \partial W / \partial \mathbf{E}$ can be instantly obtained by differentiating the output of the predicted (target) constitutive NN, i.e., the strain energy density $W = W_{\theta_G}^{\text{NN}}$. By leveraging automatic differentiation [88] and backpropagation [89], the constitutive NN ensures smooth and analytically exact first and second derivatives, which is particularly beneficial for subsequent integration into FE simulations to improve numerical stability. The hypernetwork is designed as a feed-forward, multi-task learning architecture (see Appendix A for details). The MLP processes the graph representation of truss lattices as input, applies sequential transformations through several sharing hidden layers, and then predicts the weights for the fully connected layers and the passthrough layers of the (target) constitutive NN, respectively, via each task-specific layer. Here we consider a static target network architecture for our constitutive model, while the framework can be easily extended to other scenarios where the target network can have dynamic architectures to maintain more flexibility [73].

Given a representative dataset with N tuples of truss graph description G , Green–Lagrange strain tensors $\mathbf{E}$, second Piola–Kirchhoff stress tensors $\mathbf{S}$, and strain energy densities W (all measured along K loading paths, each discretized into T load steps) as

$$D = \left\{ \left(G^{(n)}, \mathbf{E}_t^{(n,k)}, \mathbf{S}_t^{(n,k)}, W_t^{(n,k)} \right) : n = 1, \dots, N; k = 1, \dots, K; t = 1, \dots, T \right\}, \quad (15)$$

the hypernetwork $\mathcal{H}$ is trained to minimize the loss function as

$$\begin{aligned} \theta_h \leftarrow \arg \min_{\theta_h} \lambda_S \underbrace{\frac{1}{|D|} \sum_{n=1}^N \sum_{k=1}^K \sum_{t=1}^T \|\mathbf{S}_t^{(n,k)} - \hat{\mathbf{S}}_t^{(n,k)}\|^2}_{\text{stress prediction loss}} \\ + \lambda_W \underbrace{\frac{1}{|D|} \sum_{n=1}^N \sum_{k=1}^K \sum_{t=1}^T \|W_t^{(n,k)} - \hat{W}_t^{(n,k)}\|^2}_{\text{energy prediction loss}}, \quad (16) \end{aligned}$$

with $\hat{W}_t^{(n,k)} = W_{\theta_G}^{\text{NN}}(\mathbf{E}_t^{(n,k)})$ and $\hat{\mathbf{S}}_t^{(n,k)} = \partial \hat{W}_t^{(n,k)} / \partial \mathbf{E}_t^{(n,k)}$,

where n, k , and t denote the indices for different truss structures, deformation paths, and loading steps, respectively. $\lambda_S > 0$ and $\lambda_W > 0$ are the weights for the stress prediction loss and the energy prediction loss, respectively. We choose to include the energy prediction loss with a smaller weight ($\lambda_S = 1, \lambda_W = 0.2$) as a regularization term, guiding the model towards predictions that are not only accurate in terms of stress predictions but also physically plausible in terms of energy landscape consistent with the stress responses, thereby enhancing the model's ability to extrapolate to unseen deformation states. Details regarding the selection of the loss weights are presented in Appendix B. By learning the mapping from truss configurations to

constitutive model weights, the hypernetwork encapsulates distinct material behaviors within a unified framework, which enables efficient training and inference for a wide range of truss metamaterials and, furthermore, enhances the model's generalization and extrapolation ability to unseen truss structures without the need for retraining from scratch for each new configuration.

We train our HyperCAN framework with the training dataset D^{train} , containing $K = 14$ different loading paths for 6,000 unique truss structures, with $T = 100$ loading steps for each load path. To systematically evaluate the adaptability and robustness of our HyperCAN model, we consider several test datasets, as outlined in Table 1. Specifically, D_L^{test} tests the model's ability to handle unseen loading scenarios for known truss structures, whereas D_G^{test} evaluates the model's generalization to unseen truss structures under familiar loading conditions. Note that, although D_L^{test} contains deformation paths {UC, BC, SS} that are also present in training, specific loading parameters within these paths are not seen during training, which ensures that the model is tested on truly unseen scenarios. Furthermore, we randomly sample 500 deformation paths including UC, BC, and TC with uniformly sampled loading parameters $\lambda_1, \lambda_2, \lambda_3 \in [-0.3, 0]$ (extending beyond those sampled in the training dataset). The constructed dataset $D_{G,L}^{\text{test}}$, composed exclusively of *unseen* loading paths applied to *unseen* truss designs, aims to provide a more stringent test of the constitutive model's extrapolation capabilities to completely unknown conditions. All details pertaining to the NN architectures, training protocols, and hyperparameters are presented in Appendix A.

3. Results

3.1. Performance of the constitutive model

We first examine our model's predictive performance on stress-strain responses for truss structures along unseen deformation paths from D_L^{test} . Reported stress values are normalized by the base material's Young's modulus E_s . Fig. 4a presents the accuracy of the model for the dataset D_L^{test} , demonstrated by the overall goodness-of-fit $R^2 \geq 99.4\%$ across all normal stress components (see Appendix B for detailed results). For this and subsequent R^2 -plots, stress values for each type of deformation path are normalized by their corresponding standard deviation for better visualization, given the significant variation in stress magnitudes across different deformations, such as simple shear and biaxial compression, as is commonly the case for anisotropic materials. Fig. 2 compares the predicted vs. true stress responses for four representative examples from D_L^{test} subject to biaxial compression and triaxial compression deformations, exhibiting distinct nonlinear behaviors (see also Supplementary Movies 1–3). The constitutive model W_{θ}^{NN} accurately predicts the mechanical responses for these unseen loading paths, achieving an averaged normalized root mean square error (NRMSE) of 1.35% (see Appendix B for detailed results). Altogether, this confirms that, by embedding fundamental principles of mechanics and material behavior directly into the model architecture, our constitutive model can not only capture the intricate mechanical behavior

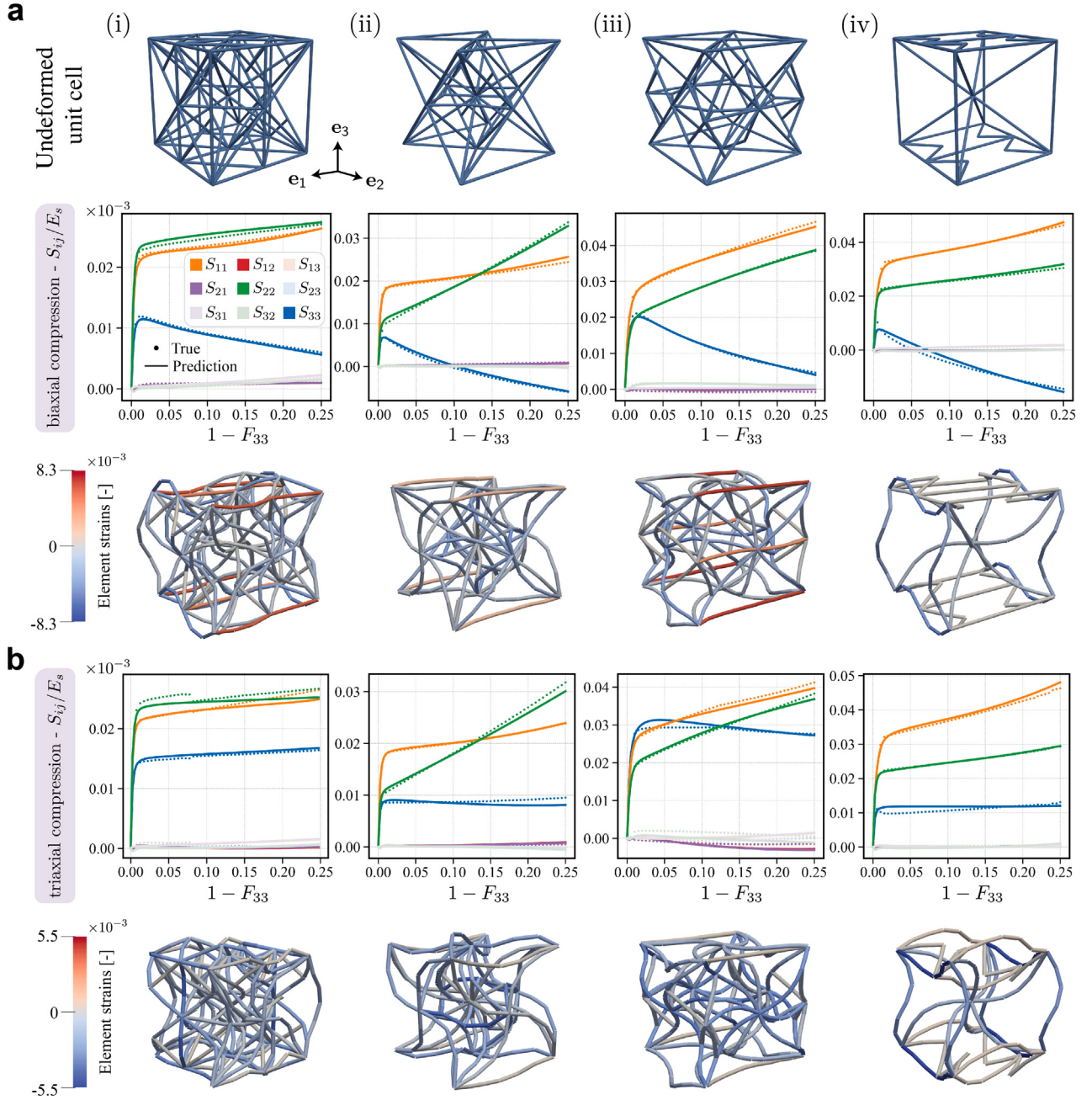


Fig. 2. Evaluation of constitutive model predictions on stress responses for unseen loading scenarios. Stress-strain plots showing the predicted (solid lines) vs. true (solid dots) responses for representative truss structures taken from D_L^{test} along (a) biaxial compression and (b) triaxial compression deformation paths sampled from outside of the training dataset. The resulting deformed unit cells are also shown alongside. Reported stress values are normalized by the base material Young's modulus E_s .

of truss metamaterials under various nonlinear loading conditions but also generalize effectively to unseen deformation paths.

Next, we assess the performance of the HyperCAN on D_G^{test} , which evaluates the model's generalization to unit cells not encountered during training. For each structure, the HyperCAN instantly predicts a constitutive model with its weights tailored to the specific truss topological and geometrical features, which allows the ML model to learn and share information from existing domains and adapt effectively to unknown domains. Fig. 4b shows the accuracy of the predicted stress responses for the dataset D_G^{test} , with $R^2 \geq 99.1\%$ across all normal stress components (see Appendix B for detailed results). Fig. 3 illustrates four representative unit cells sampled from the holdout set D_G^{test} , comparing

their true vs. predicted stress responses subjected to simple shear and biaxial compression (see also Supplementary Movies 4–5). The model achieves an averaged NRMSE of 2.18% on unseen structures, demonstrating robust generalization to novel designs. We observe that the unit cells shown in Fig. 2b(i) and Fig. 3b(ii) exhibit notable similarities by visual inspection, while their corresponding mechanical responses differ significantly. This highlights the challenges in developing unified constitutive models for a broad range of structural designs, which can accommodate the high sensitivity of the mechanical behavior to structural features. Overall, this underscores HyperCAN's efficacy in learning the underlying intricate dependencies between metamaterial

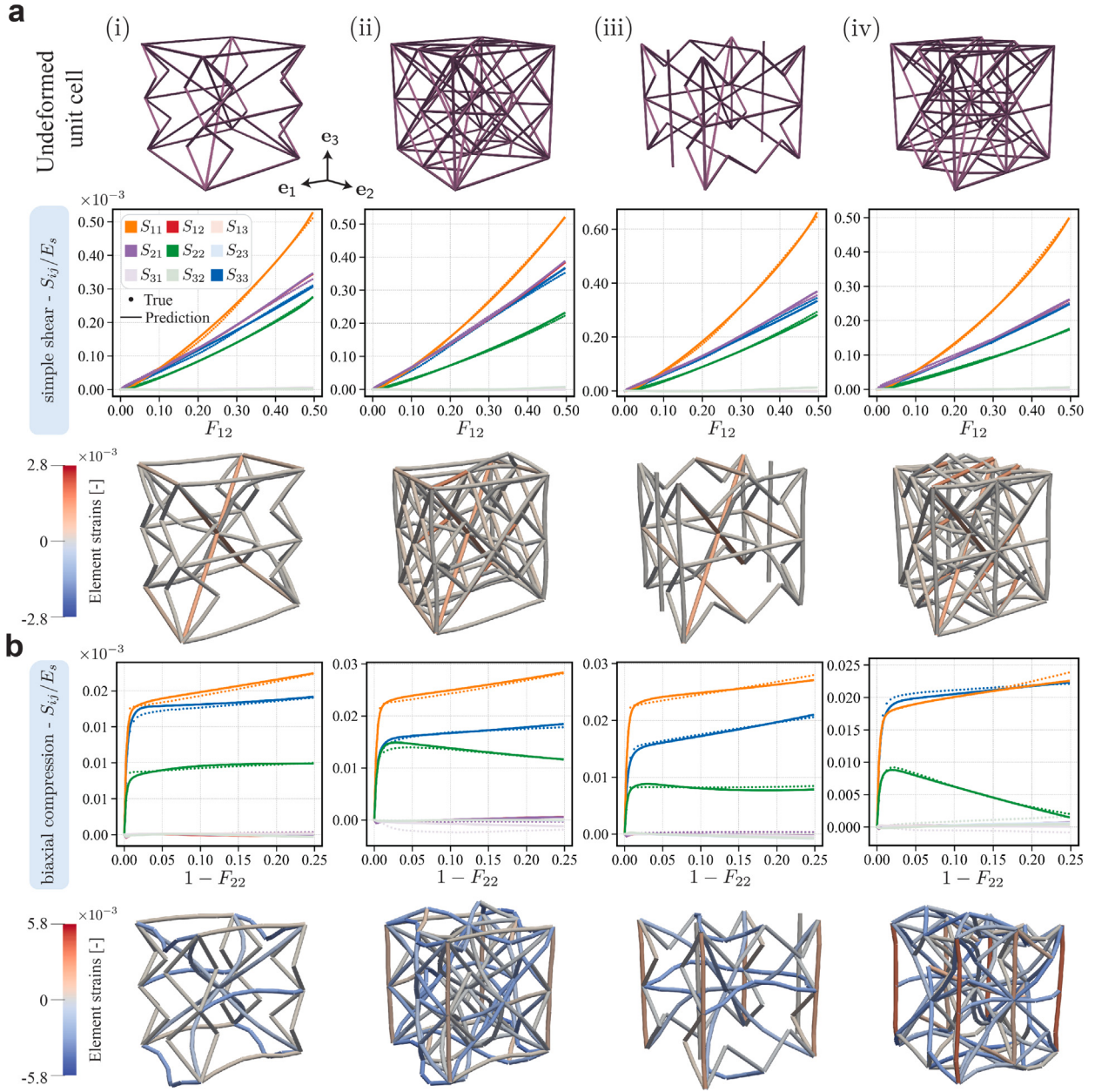


Fig. 3. Evaluation of constitutive model predictions on stress responses for unseen truss structures. Stress-strain plots showing the predicted (solid lines) vs. true (solid dots) responses for representative truss structures taken from D_G^{test} (i.e., not included during training) along (a) simple shear and (b) biaxial compression deformation paths. The resulting deformed unit cells are also shown alongside. Reported stress values are normalized by the base material Young's modulus E_s .

design and constitutive behavior for a broad truss design space, going beyond those designs previously encountered during training.

3.2. Generalization beyond the training domain

To further validate the model's versatility, we evaluate the model's predictive power for unseen designs with highly anisotropic properties, especially when going to large deformations beyond the training range. The dataset $D_{G,L}^{\text{test}}$, as outlined in Table 1, presents the most challenging test case, featuring both unseen truss structures and unseen loading scenarios with randomly sampled loading parameters, which serves as a rigorous assessment of the model's effectiveness in entirely unknown domains. To visualize the model's generalization capabilities, Fig. 5a compares different loading scenarios in the strain invariants space for both the training (seen) and evaluation (unseen) deformation paths

across all loading steps. The principal strain invariants are computed as

$$\bar{I}_1 = J^{-2/3} I_1, \quad \bar{I}_2 = J^{-4/3} I_2, \quad J = \det \mathbf{F} = I_3^{1/2}, \quad (17)$$

$$\text{with } I_1 = \text{tr } \mathbf{C}, \quad I_2 = \frac{1}{2} [(\text{tr } \mathbf{C})^2 - \text{tr}(\mathbf{C}^2)], \quad I_3 = \det \mathbf{C},$$

where $\mathbf{C} = \mathbf{F}^T \mathbf{F} = \mathbf{I} + 2\mathbf{E}$ is the right Cauchy–Green tensor. Additionally, we highlight two representative test deformation paths in Fig. 5 and evaluate the model's predictions on unseen truss structures (taken from D_G^{test} , consistent with Fig. 3) subject to these test paths. Fig. 5 highlights that the unseen deformation states used in D_L^{test} and $D_{G,L}^{\text{test}}$ considerably surpass the boundaries of those encountered during the model's training phase, showcasing the extensive range of the model's applicability. As shown in Fig. 5b, the model demonstrates remarkable accuracy in predicting responses at unseen deformation states that extend significantly beyond the training space (see also

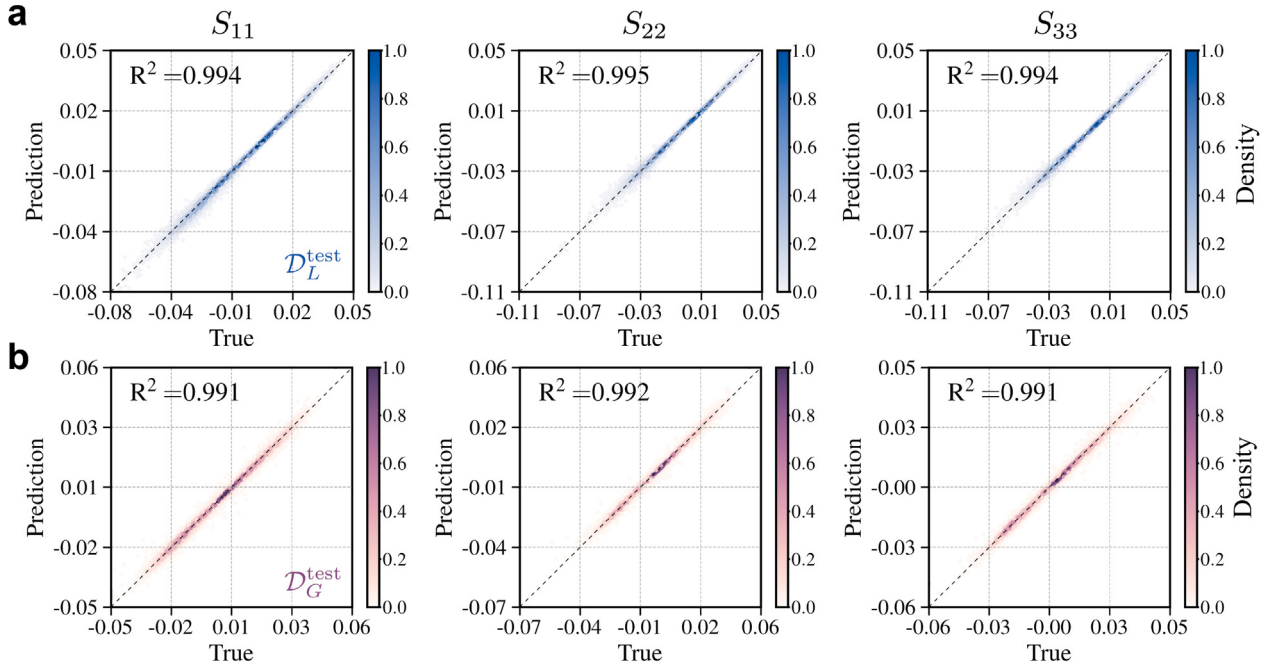


Fig. 4. Evaluation of constitutive model prediction accuracy. Distribution of predicted vs. ground truth stress responses (obtained by FE homogenization) for (a) unseen loading scenarios $\mathcal{D}_L^{\text{test}}$ and (b) unseen truss structures $\mathcal{D}_G^{\text{test}}$ for the normal components S_{11} , S_{22} , and S_{33} of the second Piola–Kirchhoff stress tensor (see Appendix B for detailed results). The color of each marker indicates the normalized density of the corresponding data point over the dataset distribution. Dashed lines denote ideal lines with zero intercept and unit slope; the goodness-of-fit R^2 is shown as a measure of accuracy.

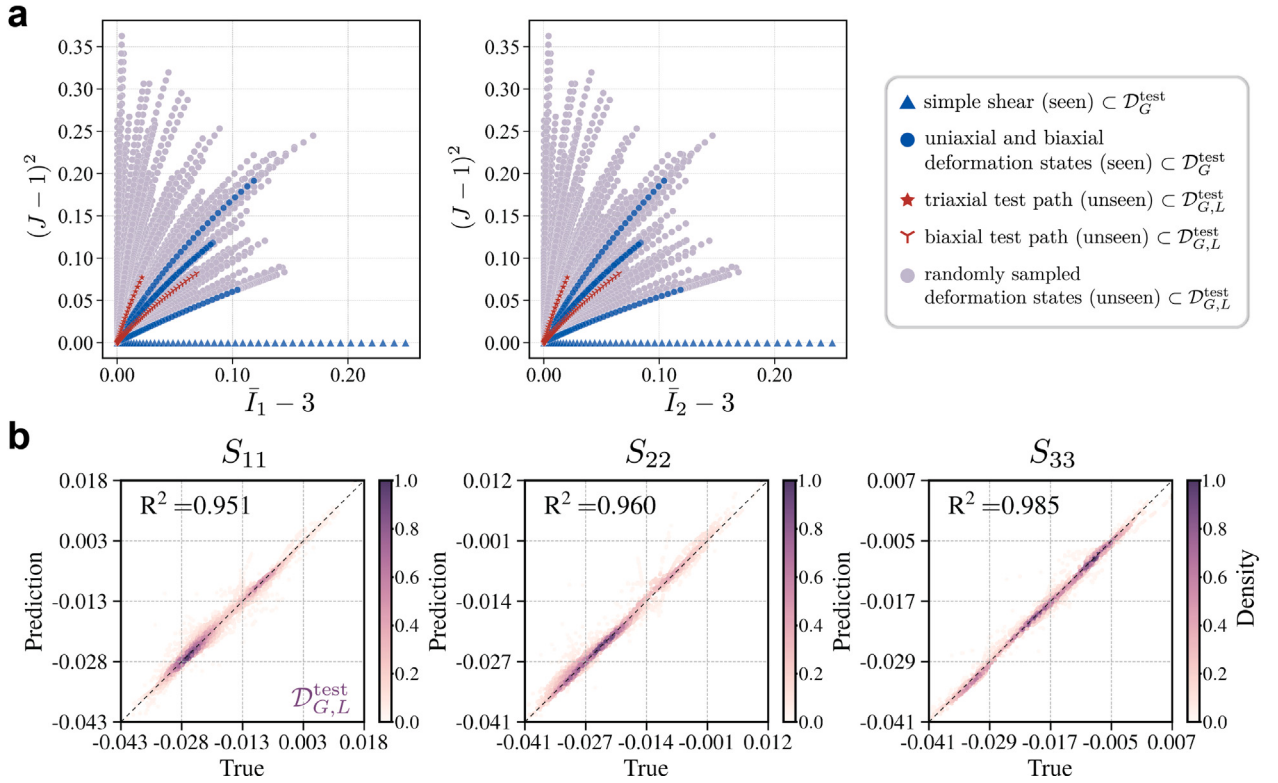


Fig. 5. Evaluation of constitutive model predictions beyond the training domain. (a) Principal strain invariant components $\bar{I}_1 - 3$, $\bar{I}_2 - 3$, and $(J - 1)^2$ at each deformation state used in training (seen) and evaluation (unseen) across all loading steps. Also shown are the principal strain invariants of the biaxial test and triaxial test deformation path used in Fig. 6b across all loading steps. (b) Distribution of predicted vs. ground truth stress responses for unseen truss structures along unseen deformation paths $\mathcal{D}_{G,L}^{\text{test}}$ for the normal components S_{11} , S_{22} , and S_{33} of the second Piola–Kirchhoff stress tensor (see Appendix B for detailed results). The color of each marker indicates the normalized density of the corresponding data point over the dataset distribution. Dashed lines denote ideal lines with zero intercept and unit slope; the goodness-of-fit R^2 is shown as a measure of accuracy.

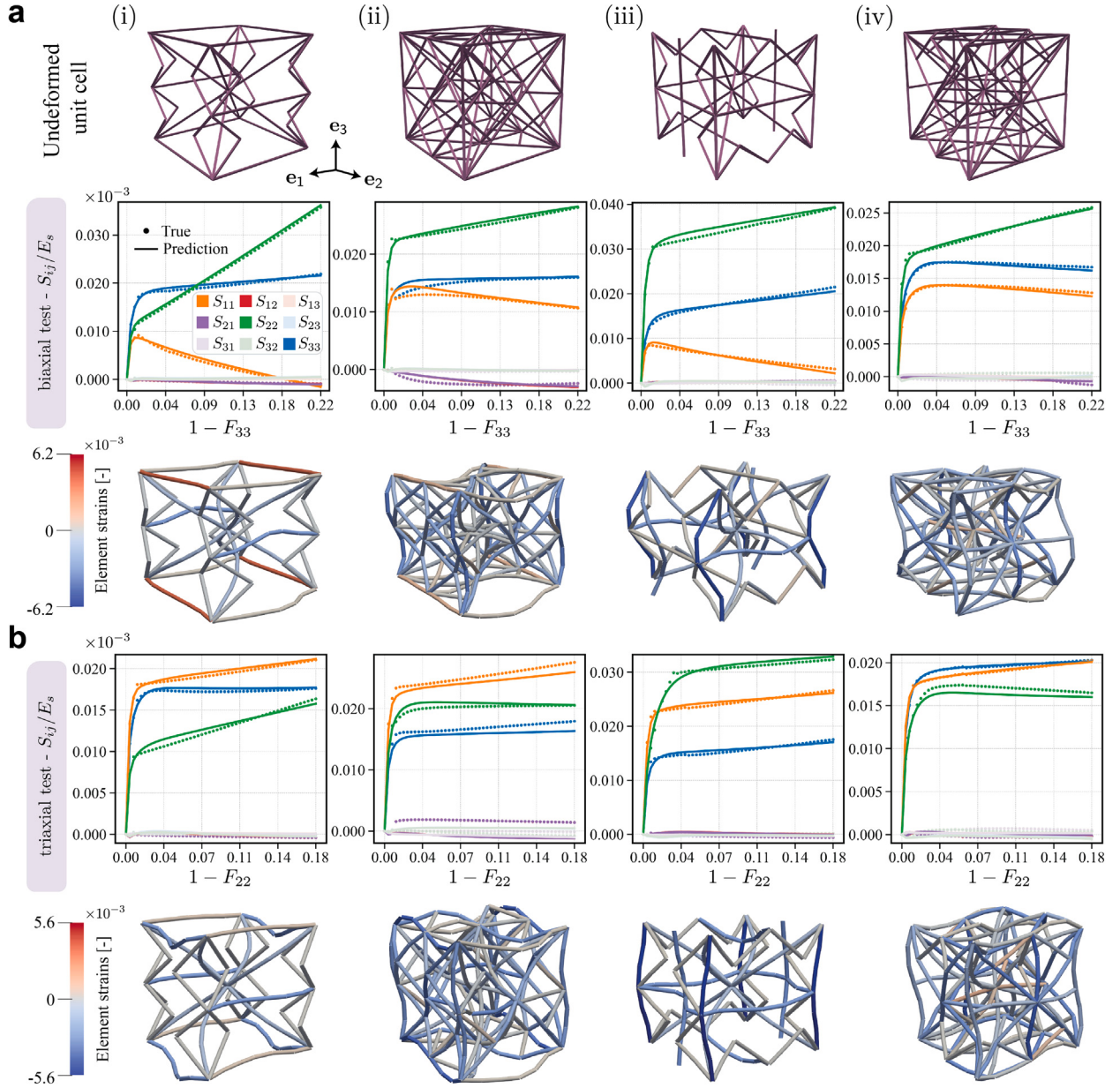


Fig. 6. Evaluation of constitutive model predictions on stress responses for unseen unit cells along unseen deformation paths. Stress–strain plots showing the predicted (solid dots) vs. true (solid lines) responses for representative truss structures taken from $D_{G,L}^{\text{test}}$ (i.e., not included during training) along (a) simple shear and (b) biaxial compression deformation paths. The resulting deformed unit cells are also shown alongside. Reported stress values are normalized by the base material Young’s modulus E_s .

Supplementary Movies 6–7), achieving an averaged NRMSE of 5.23% and $R^2 \geq 95.2\%$ across all unseen test cases (see Appendix B for detailed results). Furthermore, Fig. 6 shows that the model’s prediction agrees well with the ground truth for stress responses of unseen truss unit cells along the unseen loading paths, demonstrating the model’s robust generalization capabilities in handling completely unseen scenarios.

3.3. Multiscale simulations with hypercan

HyperCAN’s ability to generalize to unseen load conditions makes it a promising tool for efficient multiscale simulations of architected materials. One of the main challenges in modeling their highly nonlinear and anisotropic mechanical response is the identification of an effective constitutive model—especially when considering geometric nonlinearity and instability at finite strains. If such an effective material model is available, it can be used efficiently in FE simulations that replace the multitude of discrete structural elements by an effective

continuum. The challenge in this setting is the appearance of locally arbitrary deformation histories in the heterogeneously deforming continuum, for which the effective constitutive model must be suitable. To demonstrate the applicability of our framework, we employ HyperCAN within a 3D multiscale FE simulation framework to characterize the global mechanical responses, where the constitutive NN serves as a surrogate model to provide the local constitutive law of the macroscale body, offering a computational shortcut to expensive, fully resolved structural simulations.

We consider a macroscale continuous body $\Omega \subset \mathbb{R}^3$, whose underlying microstructure is the periodic tessellation of an RVE. The RVE, in turn, is composed of struts modeled as linear elastic corotational (Euler–Bernoulli) beam elements. In a classical FE^2 setting, for every macroscale point $\mathbf{x} \in \Omega$ the effective strain energy density $W(\mathbf{E})$ is computed on the fly from a homogenization simulation on the RVE-level [90]. As an efficient alternative, we here use $W_{\theta_G}^{\text{NN}}(\mathbf{E})$ as the effective strain energy density on the macroscale.

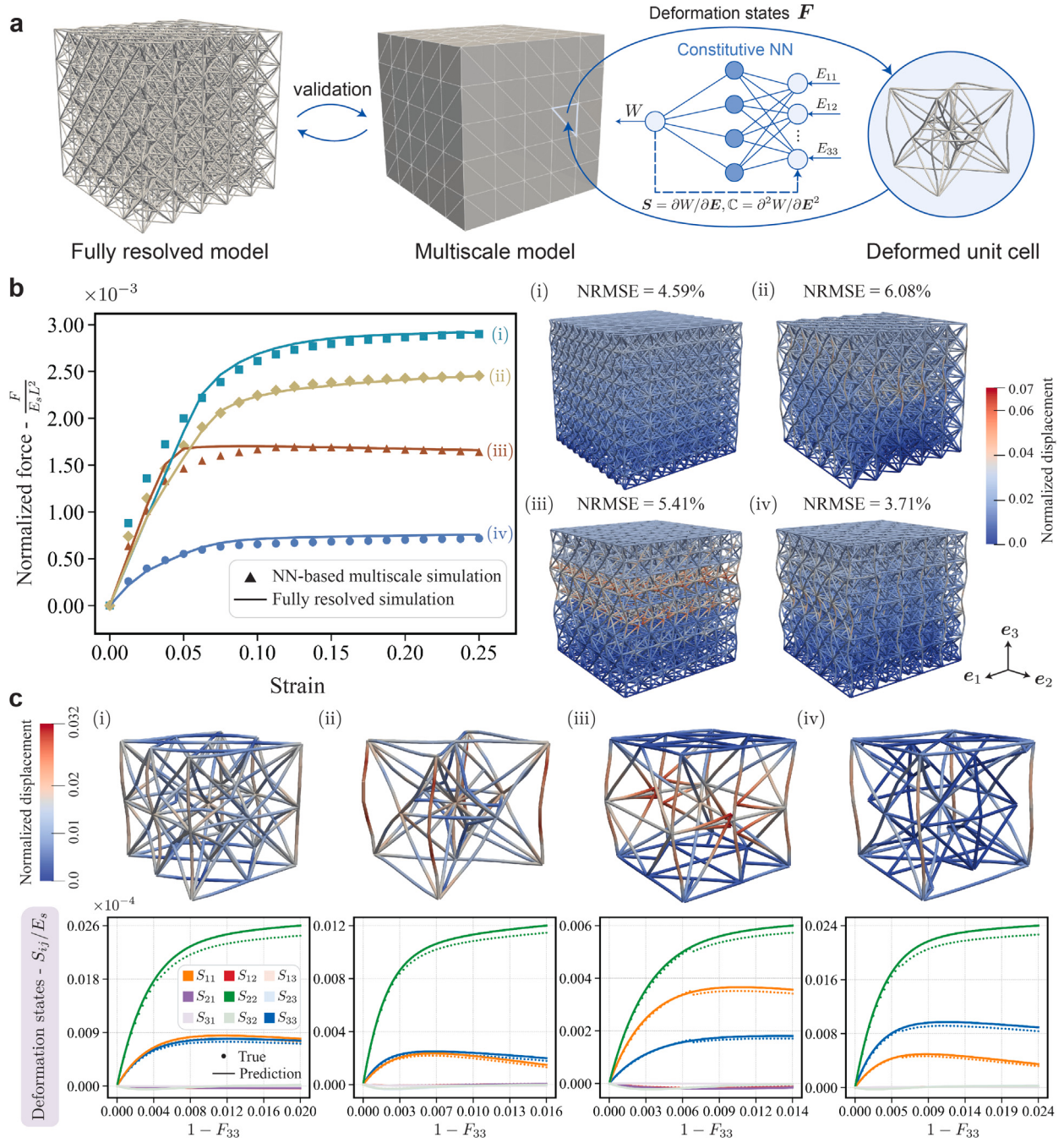


Fig. 7. Data-driven nonlinear multiscale simulation framework for truss metamaterials. (a) The discrete structure is simulated as a continuous body, which is discretized by tetrahedral finite elements. At every integration RVE point of the FE mesh, the effective constitutive behavior of a unit cell is predicted by the NN model, taking the (local) macroscopic deformation field $\mathbf{F}$ as input. The predicted RVE-averaged stresses $\mathbf{S}$ and tangent modulus $\mathbb{C}$ are calculated by differentiating the energy and returned to the macroscale. (b) Global mechanical responses (normalized by the base material Young's modulus E_s and the cube side length L) calculated by the NN-based continuum model are compared with those obtained from the fully resolved discrete beam model for four representative structures made of truss unit cells taken from $\mathcal{D}_{G,L}^{\text{test}}$ to validate our approach. Each of the four examples shows the corresponding deformed structure obtained by the fully resolved model and its displacement magnitude (normalized by the cube side length L) under an applied compressive displacement of 0.025 m. Also shown are the NRMSE of the compressive forces computed by the fully resolved model and the continuum model. (c) Deformed truss unit cell for each structure and its displacement magnitude (normalized by the unit cell dimension) are visualized at a compressive displacement of 0.025 m. Stress-strain plots compare the predicted (solid lines) vs. true (solid dots) homogenized responses of the four structures at the mesh integration point highlighted in (a) along their corresponding deformation paths.

Fig. 7a shows a schematic representation of the NN-based multiscale simulation framework for beam-based metamaterials. For a given architecture $G = (\mathbf{A}, \mathbf{x})$, the hypernetwork predicts a constitutive NN that serves as a surrogate model for the effective strain energy density. The effective second Piola-Kirchhoff stress tensor $\mathbf{S}$ and the tangent stiffness tensor $\mathbb{C}$ are computed efficiently from the NN model by automatic differentiation via Eq. (8), which are then returned to the

macroscale. The macroscale boundary value problem is solved by the FE method, which discretizes the macroscale body into tetrahedral elements. The resulting nonlinear system of equilibrium equations is solved by Newton-Raphson iteration.

As an example, we consider a cubic sample of size $0.1 \times 0.1 \times 0.1 \text{ m}^3$ composed of $5 \times 5 \times 5$ unit cells under uniaxial compression. Within each unit cell, beams are assumed to have constant circular cross-

sections characterized by an isotropic linear elastic base material with Young's modulus $E_s = 1$ and Poisson's ratio $\nu = 0.3$. Compressive displacements up to $u = -0.025\text{m}$ are applied incrementally across the top surface in the e_3 -direction, while the bottom surface is fixed and the other surfaces remain unconstrained. The deformation within the sample is not homogeneous in general and can show buckling instability. We employ the aforementioned NN-based multiscale simulation approach to study the global mechanical response of large truss structures made of different unit cell designs. Results are compared with a 3D fully resolved beam model to illustrate the validity of our NN-based continuum model.

Fig. 7b showcases four representative examples of deformed truss structures made of $5 \times 5 \times 5$ truss unit cells, and their corresponding overall force–displacement responses obtained from both the discrete fully resolved and NN-based multiscale simulations. The compression force is normalized by the base material's Young's modulus E_s and the side length of the cube L . Fig. 7c further illustrates the deformed truss unit cell for each structure, and additionally, the constitutive NN predictions for their homogenized responses along the corresponding loading path for the element integration point in the macroscale model shown in Fig. 7a (see also Supplementary Movie 8). The NN-based homogenized model reveals good agreement with the fully resolved model (with an overall NRMSE $\leq 6.08\%$) across the nonlinear deformation regime, which confirms that the NN model effectively captures the various truss responses at large deformation. In Fig. 7b(i) and (ii), we observe steady increases in force which eventually plateau at large displacements, while the samples in Fig. 7b(iv) and especially (iii) demonstrate distinct softening, which indicates buckling instabilities within the structure (of individual unit cells or rows of cells). Notably, the NN constitutive model accurately represents such phenomena in the global mechanical responses of large truss structures, as illustrated in Fig. 7b(iii) and (iv). This validation underscores the robustness and the generalization capability of the NN constitutive model in predicting the complex mechanical responses of beam lattices including buckling instabilities.

The close agreement of the NN-based homogenized models with the fully resolved simulation across various truss structures demonstrates the potential of HyperCAN to serve as an efficient surrogate in large-scale simulations, offering a cost-effective alternative to fully resolved structural calculations with orders of magnitude improvement in computational efficiency (see Appendix C for detailed comparisons of computational performance). Furthermore, the NN constitutive model inherently ensures smooth, continuous predictions and derivatives without spurious oscillations, e.g., as shown in Fig. 7c(iii). This smoothness, while potentially smoothing out abrupt changes in mechanical responses due to, e.g., localized buckling effects, offers significant advantages when employed in FE simulations under large deformations by enhancing computational efficiency and numerical stability. Evaluation of the once-trained NN-based model is considerably more efficient than on-the-fly homogenization [90] (see Table C.4), while also generalizing to metamaterial architectures, which offers opportunities for multiscale optimization [42,91].

4. Conclusion

The mechanical modeling of beam-based architected materials, featuring a vast topological and geometrical design space, presents challenges due to geometric nonlinearity and instability under large deformations. Here, we introduce HyperCAN, a unified framework for constructing parameter-efficient, mechanics-informed, and highly adaptable constitutive models for a broad spectrum of beam lattices exhibiting distinct nonlinear mechanical behaviors. The proposed NN constitutive models demonstrate remarkable robustness and generalization to unseen unit cell designs and unseen loading scenarios outside the training domain, further enabling efficient multiscale simulations that replace discrete structures by effective continua.

The HyperCAN framework tackles the challenges of learning constitutive NNs for different designs by employing hypernetworks, which dynamically adjust the constitutive relations (via the weights and biases of a constitutive NN) to a given design. This decomposition bypasses the need to store all structure–property relations within the constitutive model, thereby facilitating efficient learning and transferring to novel structures. We construct the constitutive models using ICNNs, which inherently fulfill thermodynamic consistency, local material stability, a stress-free reference configuration, and objectivity. Integrating fundamental physical principles and mechanics considerations into the constitutive NN significantly enhances the model's performance, especially the generalization to unseen strain states. Furthermore, the ICNN avoids non-smooth, oscillatory predictions of the stress response, ensuring smooth first and second derivatives of the effective strain energy density. This admits the incorporation of the constitutive NN into nonlinear multiscale simulations.

The presented constitutive modeling framework closes the gap in constructing efficient, physically sensible mappings between structural configurations to finite-strain responses for a wide class of beam-based architected materials. The graph-based representation with great versatility provides a unified parameterization for capturing complex interactions among topological features of truss unit cells [19], facilitating the exploration of a wide range of potential microstructure designs in multiscale design and optimization of architected materials. It also admits straightforward extension to more extensive design space of trusses, accommodating variations in, e.g., relative density or manufacturing imperfections [92], as well as to other classes of metamaterials, e.g., triply periodic minimal surface (TPMS) lattices [93,94] or spinodal morphologies [5,6,95,96] by modifying the design space parameterization. Moreover, the proposed framework can be adapted to address more complex material behavior, e.g., path- or history-dependent base material models [97], ranging from viscoelasticity and plasticity, by modifying the core architecture of constitutive NNs. For example, hypernetworks can be leveraged to generate parametric physics-informed neural networks (PINNs) that approximate the solutions to the underlying partial differential equation for a given parameterization [98]. This highlights the robustness and versatility of our HyperCAN framework, opening up new avenues for the multiscale design and analysis of a wide range of architected materials.

CRedit authorship contribution statement

Li Zheng: Writing – original draft, Visualization, Validation, Software, Methodology, Data curation. **Dennis M. Kochmann:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Siddhant Kumar:** Writing – review & editing, Supervision, Methodology, Conceptualization.

Code availability

The code used to train and evaluate the model is available at <https://github.com/li-zhengz/HyperCAN> and on Zenodo [99].

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

L.Z. was supported by ETH Zurich through the SynMatLab project.

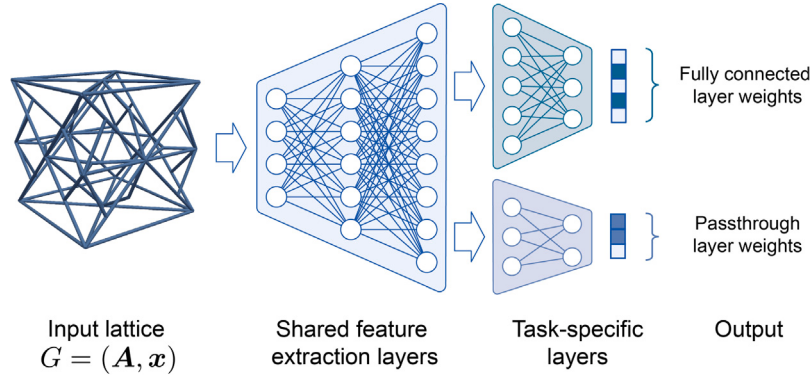


Fig. A.8. Schematic illustration of the multi-tasking hypernetwork architecture. The multi-tasking hypernetwork first processes the input truss graph representation through several shared feature extraction layers, and then feeds the intermediate outputs into separate task-specific branches corresponding to the prediction of fully connected layer weights and the passthrough layer weights of the ICNN constitutive model.

Appendix A. Model architecture and training protocols

In this work, we employ a multi-layer perceptron (MLP) to construct the hypernetwork $\mathcal{H}$, which predicts the weights of the ICNN constitutive model based on given input truss graph representations $G = (\mathbf{A}, \mathbf{x})$. We note that since the adjacency matrix $\mathbf{A}$ is symmetric, only the upper triangular part is considered by the model. Additionally, our focus is restricted to the node positions $\mathbf{x}$ of 6 face nodes $\{f_1, f_2, f_3, f_4, f_5, f_6\}$, as positions of other vertex nodes remain constant. Given the distinct functionalities and constraints on the weights of the fully connected and passthrough layers in the ICNN architecture, we adopt a multi-task architecture for the hypernetwork, where the intermediate outputs of the shared layers are fed into separate blocks of task-specific layers to predict the weights of different layers. A schematic representation of this multi-tasking hypernetwork architecture is depicted in Fig. A.8. Further details of the relevant hyperparameters of the HyperCAN framework, including the hypernetwork and the ICNN-based constitutive model, are provided in Table A.2. In Algorithm 1, we summarize a pseudocode for the training of the HyperCAN framework.

In our framework, the hypernetwork and ICNN are trained jointly in an end-to-end manner without any restrictions on their interaction. Specifically, the hypernetwork processes the graph-based representation of the truss configuration and generates the weights for the ICNN, and as such, there is no independent learning process for the ICNN itself. The ICNN then uses these weights to predict the strain energy density and compute the stress responses by differentiating the energy with respect to the input deformation measures. The ICNN's weights are entirely determined by the hypernetwork based on the input truss configurations, and consequently, the hypernetwork is the only component that is trained and optimized. During training, the gradients from the stress and energy prediction losses propagate through the ICNN back to the hypernetwork, which are then used to update the hypernetwork's parameters. This allows the hypernetwork to directly adjust the ICNN weights in response to the error between the predicted mechanical responses and the ground truth data.

The joint training process enables the model to effectively capture the complex mapping from truss configurations to mechanical responses in a unified optimization framework – eliminating the need for separate training stages for each component. Furthermore, this approach facilitates better generalization to unseen truss structures, as the hypernetwork learns to dynamically adjust the ICNN's behavior for a wide range of truss configurations and deformation scenarios without requiring retraining for new domains.

Appendix B. Model performance

Fig. B.9 shows the evolution of the loss and the normalized root mean square error (NRMSE, computed as in Eq. (B.1)) against the

Algorithm 1 HyperCAN training

Input: Graph representation of truss lattice $G = (\mathbf{A}, \mathbf{x})$

Input: Deformation gradient $\mathbf{F}$, strain energy density W , and Second Piola-Kirchhoff stress $\mathbf{S}$

- 1: Randomly initialize hypernetwork parameters: θ_h .
- 2: Initialize Adam optimizer with learning rate η .
- 3: **for** $e = 1, \dots, n_e$ **do** ▷ Iterate over training epochs
- 4: Loss: $\mathcal{L} \leftarrow 0$ ▷ Initialize loss
- 5: $\theta_G : \{(\mathcal{W}_z^{(i)}, \mathcal{W}_y^{(i)}) : i = 1, \dots, k\} \leftarrow \mathcal{H}(G; \theta_h)$ ▷ Compute layer weights of the ICNN
- 6: $\mathbf{E} \leftarrow \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{I})$ ▷ Compute Green-Lagrangian strain tensor
- 7: $\mathbf{y} \leftarrow [E_{11}, E_{12}, E_{13}, E_{21}, E_{22}, E_{23}, E_{31}, E_{32}, E_{33}]^T$ ▷ Vectorized input
- 8: $\mathbf{z}^{(1)} \leftarrow \phi(\mathcal{W}_y^{(1)} \mathbf{y} + \mathbf{b}^{(1)})$ ▷ Input layer of the ICNN
- 9: **for** $i = 2, \dots, k-1$ **do** ▷ Iterate over ICNN hidden layers
- 10: $\mathbf{z}^{(i)} \leftarrow \phi(\sigma(\mathcal{W}_z^{(i)} \mathbf{z}^{(i-1)} + \mathcal{W}_y^{(i)} \mathbf{y} + \mathbf{b}^{(i)}))$
- 11: **end for**
- 12: $\hat{W}_{\theta_G} \leftarrow \sigma(\mathcal{W}_z^{(k)} \mathbf{z}^{(k-1)} + \mathcal{W}_y^{(k)} \mathbf{y} + \mathbf{b}^{(k)})$ ▷ Compute strain energy density
- 13: $\hat{\mathbf{S}} \leftarrow \partial \hat{W}_{\theta_G} / \partial \mathbf{E}$ ▷ Compute Second Piola-Kirchhoff stress
- 14: $\mathcal{L} \leftarrow \mathcal{L} + \lambda_S (\hat{\mathbf{S}} - \mathbf{S})^2 + \lambda_W (\hat{W}_{\theta_G} - W)^2$ ▷ Update loss
- 15: Compute gradients $\nabla_{\theta_h} \mathcal{L}$ using automatic differentiation.
- 16: Update parameters θ_h using the Adam optimizer:
- 17: $\theta_h \leftarrow \text{Adam}(\theta_h, \nabla_{\theta_h} \mathcal{L}, \eta)$
- 18: **end for**
- 19: **Output:** Trained hypernetwork $\mathcal{H}(G; \theta_h)$; constitutive models $\hat{W}_{\theta_G} = F(\mathbf{E}; \mathcal{H}(G; \theta_h))$

number of training iterations. We further show the error maps of NRMSE measures for each stress component for datasets $\mathcal{D}_L^{\text{test}}$, $\mathcal{D}_G^{\text{test}}$ and $\mathcal{D}_{G,L}^{\text{test}}$ in Fig. B.10, corresponding to Fig. 4a, Fig. 4b, and Fig. 5b in the main article, respectively.

$$\text{RMSE}(\mathbf{S}, \hat{\mathbf{S}}) = \sqrt{\frac{1}{n} \sum_{i=1}^n \|\mathbf{S}^{(i)} - \hat{\mathbf{S}}^{(i)}\|^2}, \quad \text{NRMSE}(\mathbf{S}, \hat{\mathbf{S}}) = \frac{\text{RMSE}(\mathbf{S}, \hat{\mathbf{S}})}{\max(\mathbf{S}) - \min(\mathbf{S})} \quad (\text{B.1})$$

where $\mathbf{S} \in \mathbb{R}^9$ and $\hat{\mathbf{S}} \in \mathbb{R}^9$ denote the true and the predicted second Piola-Kirchhoff stresses, respectively, and n denote the size of the dataset.

Besides, we have investigated the effects of different values of energy prediction loss weight λ_W on the model's prediction performance. We choose to include the energy prediction loss in Eq. (16)

Table A.2
Dimensions and training hyperparameters of the HyperCAN ML framework.

	Hypernetwork $\mathcal{H}$			Constitutive model $\mathcal{F}_\omega$
	Shared layers	Fully-connected layer weights predictor	Passthrough layer weights predictor	
Input dimensions	117	256	256	9
Hidden dimensions	256,256,256,256	256,256,256,256	256,256,256,256	20,20,20
Output dimensions	256	1000	369	1
Activation functions	LeakyReLU [100] (with a negative slope of 0.01)			—
Optimization algorithm	Adam [101]			—
Learning rate	5×10^{-4}			—
Batch size	64			—
Number of epochs	1000			—

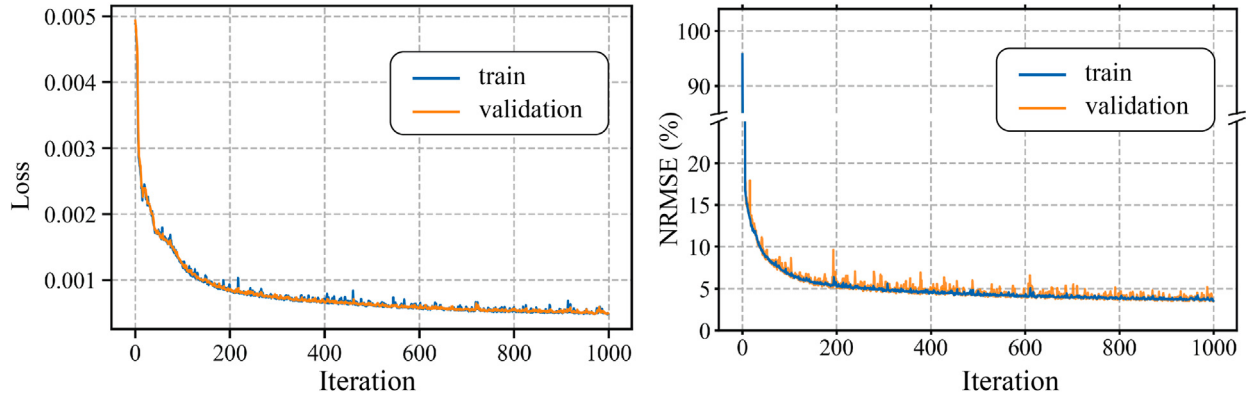


Fig. B.9. Error convergence evaluated across the entire training and validation dataset. Evolution of the (a) loss and (b) normalized root mean square error (NRMSE, expressed as a percentage) as a function of the number of training iterations.

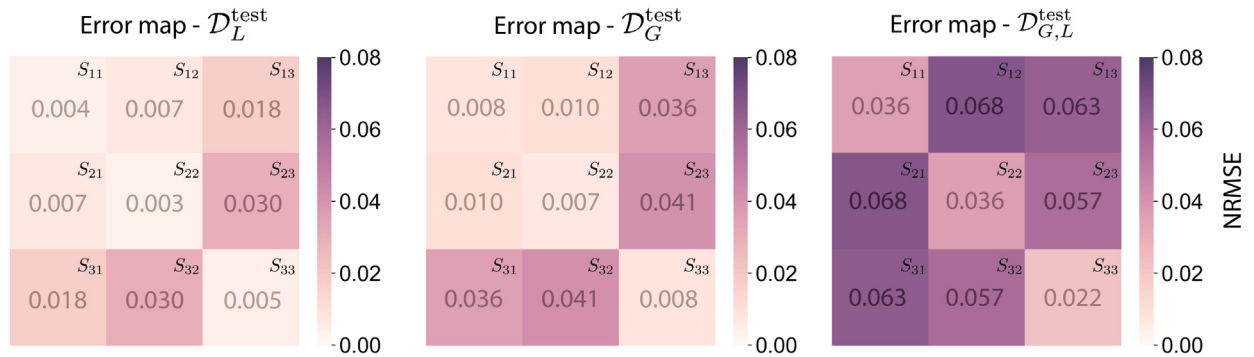


Fig. B.10. Error map of the normalized root mean square error (NRMSE) evaluated across different datasets. $\mathcal{D}_L^{\text{test}}$ evaluates the model's performance on unseen loading scenarios with known truss structures. $\mathcal{D}_G^{\text{test}}$ assesses generalization to unseen truss structures under known loading conditions. $\mathcal{D}_{G,L}^{\text{test}}$ tests the model's adaptability to completely unknown conditions, featuring both unseen truss structures and loading scenarios.

Table B.3
Comparison of the normalized root mean square error (NRMSE) evaluated across different datasets with varying energy prediction loss weight λ_W . $\mathcal{D}_L^{\text{test}}$ evaluates the model's performance on unseen loading scenarios with known truss structures. $\mathcal{D}_G^{\text{test}}$ assesses generalization to unseen truss structures under known loading conditions. $\mathcal{D}_{G,L}^{\text{test}}$ tests the model's adaptability to completely unknown conditions, featuring both unseen truss structures and loading scenarios.

NRMSE	$\mathcal{D}_L^{\text{test}}$			$\mathcal{D}_G^{\text{test}}$			$\mathcal{D}_{G,L}^{\text{test}}$		
	S_{11}	S_{22}	S_{33}	S_{11}	S_{22}	S_{33}	S_{11}	S_{22}	S_{33}
$\lambda_W = 0.0$	0.007	0.007	0.006	0.011	0.011	0.010	0.040	0.030	0.027
$\lambda_W = 0.1$	0.006	0.005	0.006	0.010	0.009	0.010	0.034	0.026	0.025
$\lambda_W = 0.2$	0.004	0.003	0.005	0.008	0.007	0.008	0.036	0.036	0.022
$\lambda_W = 0.5$	0.006	0.005	0.007	0.011	0.009	0.009	0.035	0.027	0.030
$\lambda_W = 1.0$	0.005	0.005	0.006	0.010	0.009	0.009	0.036	0.028	0.026

as a regularization term to prevent the model from predicting non-physical stress responses, particularly in unseen loading scenarios, where relying solely on stress data can potentially lead the model

to overfit specific patterns, such as abrupt changes or noise in the training data. The value of loss weight λ_W balances the emphasis between stress prediction accuracy and the generalization capacity of

Table C.4

Overview of the computational runtime, the software and hardware resources required for different tasks. The reported runtimes are rough average estimates.

Tasks	Software	Hardware	Runtime
Truss dataset generation	Python	CPU (20 cores) ^b	5 min
FE homogenization computations (of the full dataset)	In-house C++ FEM code	CPU (20 cores) ^b	6 days
Model training	PyTorch in Python	GPU ^c	4.5 h
Responses computation using NN model F^a	PyTorch in Python	GPU ^c	0.01 s
Responses computation using FE homogenization ^a	In-house C++ FEM code	CPU (10 cores) ^b	10 min
Multiscale simulation with discrete beam elements	In-house C++ FEM code	CPU (10 cores) ^b	40 min ^d
Multiscale simulation with NN-based continuum model	PyTorch in Python	CPU (single core) ^b	2 min ^d

^a Runtimes are reported for a single truss unit cell subject to a defined loading path with 100 load steps.

^b Computations were performed on the Euler V cluster of ETH Zurich with two 12-core Intel Xeon Gold 5118 processors and 96 GB of DDR4 memory at 2400 MHz.

^c Computations were performed on a single Nvidia RTX 4090 with CUDA 11.7.

^d Reported runtimes are averaged for representative examples in the main article only and may vary due to the convergence of the FE solver.

the constitutive model. To illustrate the impact of different λ_W values, we present the normalized root mean square error (NRMSE) of the normal components S_{11}, S_{22}, S_{33} of the second Piola–Kirchhoff stress tensor (for simplicity) evaluated on different datasets in Table B.3. It can be observed that choosing a value of $\lambda_W = 0.2$ achieves the best trade-off, as it sufficiently emphasizes the energy term to improve physical plausibility and generalization without significantly sacrificing the model's stress prediction accuracy.

Appendix C. Computational efficiency

To demonstrate the efficiency of our approach, we provide the computation runtimes, software, and hardware resources used for the data generation, training, and deployment of our framework in Table C.4. It is important to note that the multiscale simulations with the discrete beam-based structural elements were performed using a highly optimized in-house C++ FEM code [84], whereas the NN-based multiscale simulations were conducted in a pure Python FEM code with a NN as a constitutive model; the latter is inherently slower for large and complex computations [102]. Currently, no general FEM codebase accommodates both discrete beam-based structural simulations and NN-based continuum models, making a direct comparison of computational speeds challenging. Nevertheless, the conceptual advantage of NN-based continuum simulation over a fully resolved discrete simulation in terms of speed is evident from the 20x speedup (as outlined in Table C.4) with the potential for several more orders of magnitude given a competitive implementation that natively integrates FEM and NN-based continuum models.

Appendix D. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.eml.2024.102243>.

Data availability

The training and test dataset (consisting of truss structures and their effective stress-strain responses) and the pre-trained models are available in the ETHZ Research Collection [103] at <https://doi.org/10.3929/ethz-b-000699994>.

References

- [1] V.S. Deshpande, N.A. Fleck, M.F. Ashby, Effective properties of the octet-truss lattice material, *J. Mech. Phys. Solids* 49 (8) (2001) 1747–1769.
- [2] L.R. Meza, A.J. Zelhofer, N. Clarke, A.J. Mateos, D.M. Kochmann, J.R. Greer, Resilient 3D hierarchical architected metamaterials, *Proc. Natl. Acad. Sci.* 112 (37) (2015) 11502–11507.
- [3] K. Davami, L. Zhao, E. Lu, J. Cortes, C. Lin, D.E. Lilley, P.K. Purohit, I. Bargatin, Ultralight shape-recovering plate mechanical metamaterials, *Nature Commun.* 6 (1) (2015) 10019.
- [4] C. Bonatti, D. Mohr, Smooth-shell metamaterials of cubic symmetry: Anisotropic elasticity, yield strength and specific energy absorption, *Acta Mater.* 164 (2019) 301–321.
- [5] S. Kumar, S. Tan, L. Zheng, D.M. Kochmann, Inverse-designed spinodoid metamaterials, *npj Comput. Mater.* 6 (1) (2020) 73.
- [6] P. Thakolkaran, M.A. Espinal, S. Dhulipala, S. Kumar, C.M. Portela, Experiment-informed finite-strain inverse design of spinodal metamaterials, 2023, arXiv preprint [arXiv:2312.11648](https://arxiv.org/abs/2312.11648).
- [7] T.A. Schaedler, A.J. Jacobsen, A. Torrents, A.E. Sorensen, J. Lian, J.R. Greer, L. Valdevit, W.B. Carter, Ultralight metallic microlattices, *Science* 334 (6058) (2011) 962–965.
- [8] X. Zheng, H. Lee, T.H. Weisgraber, M. Shusteff, J. DeOtte, E.B. Duoss, J.D. Kuntz, M.M. Biener, Q. Ge, J.A. Jackson, et al., Ultralight, ultrastiff mechanical metamaterials, *Science* 344 (6190) (2014) 1373–1377.
- [9] S. Babae, J. Shim, J.C. Weaver, E.R. Chen, N. Patel, K. Bertoldi, 3D soft metamaterials with negative Poisson's ratio, *Adv. Mater.* 25 (36) (2013) 5044–5049.
- [10] A. Clausen, F. Wang, J.S.n. Jensen, O. Sigmund, J.A. Lewis, Topology optimized architectures with programmable Poisson's ratio over large deformations, *Adv. Mater.* 27 (37) (2015) 5523–5527.
- [11] A. Rafsanjani, D. Pasini, Bistable auxetic mechanical metamaterials inspired by ancient geometric motifs, *Extreme Mech. Lett.* 9 (2016) 291–296.
- [12] X. Chen, Q. Ji, J. Wei, H. Tan, J. Yu, P. Zhang, V. Laude, M. Kadic, Light-weight shell-lattice metamaterials for mechanical shock absorption, *Int. J. Mech. Sci.* 169 (2020) 105288.
- [13] T. Butruille, J.C. Crone, C.M. Portela, Decoupling particle-impact dissipation mechanisms in 3D architected materials, *Proc. Natl. Acad. Sci.* 121 (6) (2024) e2313962121.
- [14] M.R. Haberman, M.D. Guild, Acoustic metamaterials, *Phys. Today* 69 (6) (2016) 42–48.
- [15] G.-S. Liu, Y. Zhou, M.-H. Liu, Y. Yuan, X.-Y. Zou, J.-C. Cheng, Acoustic waveguide with virtual soft boundary based on metamaterials, *Sci. Rep.* 10 (1) (2020) 981.
- [16] J. Aizenberg, J.C. Weaver, M.S. Thanawala, V.C. Sundar, D.E. Morse, P. Fratzl, Skeleton of *Euplectella* sp.: structural hierarchy from the nanoscale to the macroscale, *Science* 309 (5732) (2005) 275–278.
- [17] T.S. Lumpe, T. Stankovic, Exploring the property space of periodic cellular structures based on crystal networks, *Proc. Natl. Acad. Sci.* 118 (7) (2021) e2003504118.
- [18] M. Maurizi, C. Gao, F. Berto, Inverse design of truss lattice materials with superior buckling resistance, *npj Comput. Mater.* 8 (1) (2022) 247.
- [19] L. Zheng, K. Karapiperis, S. Kumar, D.M. Kochmann, Unifying the design space and optimizing linear and nonlinear truss metamaterials by generative modeling, *Nature Commun.* 14 (1) (2023) 7563.
- [20] J.-H. Bastek, S. Kumar, B. Telgen, R.N. Glaesener, D.M. Kochmann, Inverting the structure–property map of truss metamaterials by deep learning, *Proc. Natl. Acad. Sci.* 119 (1) (2022) e2111505119.
- [21] M.-T. Hsieh, M.R. Begley, L. Valdevit, Architected implant designs for long bones: Advantages of minimal surface-based topologies, *Mater. Des.* 207 (2021) 109838.
- [22] C. Wang, Z. Vangelatos, C.P. Grigoropoulos, Z. Ma, Micro-engineered architected metamaterials for cell and tissue engineering, *Mater. Today Adv.* 13 (2022) 100206.
- [23] Y. Jiang, Q. Wang, Highly-stretchable 3D-architected mechanical metamaterials, *Sci. Rep.* 6 (1) (2016) 34147.
- [24] C.E. Gregg, D. Catanoso, O.I.B. Formoso, I. Kostitsyna, M.E. Ochalek, T.J. Olatunde, I.W. Park, F.M. Sebastianelli, E.M. Taylor, G.T. Trinh, et al., Ultralight, strong, and self-reprogrammable mechanical metamaterials, *Science Robotics* 9 (86) (2024) eadi2746.

- [25] P. Zhang, D. Qi, R. Xue, K. Liu, W. Wu, Y. Li, Mechanical design and energy absorption performances of rational gradient lattice metamaterials, *Compos. Struct.* 277 (2021) 114606.
- [26] X. Li, X. Yu, J.W. Chua, H.P. Lee, J. Ding, W. Zhai, Microlattice metamaterials with simultaneous superior acoustic and mechanical energy absorption, *Small* 17 (24) (2021) 2100336.
- [27] Y. He, Y. Zhou, Z. Liu, K. Liew, Buckling and pattern transformation of modified periodic lattice structures, *Extreme Mech. Lett.* 22 (2018) 112–121.
- [28] M. Jamshidian, N. Boddeti, D.W. Rosen, O. Weeger, Multiscale modelling of soft lattice metamaterials: Micromechanical nonlinear buckling analysis, experimental verification, and macroscale constitutive behaviour, *Int. J. Mech. Sci.* 188 (2020) 105956.
- [29] M. Mooney, A theory of large elastic deformation, *J. Appl. Phys.* 11 (9) (1940) 582–592.
- [30] R. Rivlin, Large elastic deformations of isotropic materials. I. Fundamental concepts, *Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* 240 (822) (1948) 459–490.
- [31] K. Valanis, R.F. Landel, The strain-energy function of a hyperelastic material in terms of the extension ratios, *J. Appl. Phys.* 38 (7) (1967) 2997–3002.
- [32] R.W. Ogden, Large deformation isotropic elasticity—on the correlation of theory and experiment for incompressible rubberlike solids, *Proc. R. Soc. A* 326 (1567) (1972) 565–584.
- [33] E.H. Twizell, R. Ogden, Non-linear optimization of the material constants in Ogden's stress-deformation function for incompressible isotropic elastic materials, *ANZIAM J.* 24 (4) (1983) 424–434.
- [34] T. Kirchdoerfer, M. Ortiz, Data-driven computational mechanics, *Comput. Methods Appl. Mech. Engrg.* 304 (2016) 81–101.
- [35] D. González, A. García-González, F. Chinesta, E. Cueto, A data-driven learning method for constitutive modeling: application to vascular hyperelastic soft tissues, *Materials* 13 (10) (2020) 2319.
- [36] J. Yvonnet, D. Gonzalez, Q.-C. He, Numerically explicit potentials for the homogenization of nonlinear elastic heterogeneous materials, *Comput. Methods Appl. Mech. Engrg.* 198 (33–36) (2009) 2723–2737.
- [37] T. Sussman, K.-J. Bathe, A model of incompressible isotropic hyperelastic material behavior using spline interpolations of tension–compression test data, *Commun. Numer. Methods Eng.* 25 (1) (2009) 53–63.
- [38] J.N. Fuhg, M. Marino, N. Bouklas, Local approximate Gaussian process regression for data-driven constitutive models: development and comparison with neural networks, *Comput. Methods Appl. Mech. Engrg.* 388 (2022) 114217.
- [39] K. Upadhyay, J.N. Fuhg, N. Bouklas, K.T. Ramesh, Physics-informed data-driven discovery of constitutive models with application to strain-rate-sensitive soft materials, *Comput. Mech.* (2024) <http://dx.doi.org/10.1007/s00466-024-02497-x>.
- [40] J. Oliver, M. Caicedo, A.E. Huespe, J. Hernández, E. Roubin, Reduced order modeling strategies for computational multiscale fracture, *Comput. Methods Appl. Mech. Engrg.* 313 (2017) 560–595.
- [41] S. Vijayaraghavan, L. Wu, L. Noels, S. Bordas, S. Natarajan, L.A. Beex, A data-driven reduced-order surrogate model for entire elastoplastic simulations applied to representative volume elements, *Sci. Rep.* 13 (1) (2023) 12781.
- [42] L. Zheng, S. Kumar, D.M. Kochmann, Data-driven topology optimization of spinodoid metamaterials with seamlessly tunable anisotropy, *Comput. Methods Appl. Mech. Engrg.* 383 (2021) 113894.
- [43] H. Yang, X. Guo, S. Tang, W.K. Liu, Derivation of heterogeneous material laws via data-driven principal component expansions, *Comput. Mech.* 64 (2019) 365–379.
- [44] S. Im, H. Kim, W. Kim, M. Cho, Neural network constitutive model for crystal structures, *Comput. Mech.* 67 (2021) 185–206.
- [45] P. Thakolkaran, A. Joshi, Y. Zheng, M. Flaschel, L. De Lorenzis, S. Kumar, NN-EUCLID: Deep-learning hyperelasticity without stress data, *J. Mech. Phys. Solids* 169 (2022) 105076.
- [46] J. Ghaboussi, J. Garrett Jr., X. Wu, Knowledge-based modeling of material behavior with neural networks, *J. Eng. Mech.* 117 (1) (1991) 132–153.
- [47] C. Bonatti, D. Mohr, One for all: Universal material model based on minimal state-space neural networks, *Sci. Adv.* 7 (26) (2021) eabf3658.
- [48] S. Jung, J. Ghaboussi, Neural network constitutive model for rate-dependent materials, *Comput. Struct.* 84 (15–16) (2006) 955–963.
- [49] K. Linka, M. Hillgärtner, K.P. Abdolazizi, R.C. Aydin, M. Itskov, C.J. Cyron, Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning, *J. Comput. Phys.* 429 (2021) 110010, <http://dx.doi.org/10.1016/j.jcp.2020.110010>.
- [50] N.N. Vlassis, R. Ma, W. Sun, Geometric deep learning for computational mechanics part I: Anisotropic hyperelasticity, *Comput. Methods Appl. Mech. Engrg.* 371 (2020) 113299.
- [51] K.A. Kalina, L. Linden, J. Brummund, P. Metsch, M. Kästner, Automated constitutive modeling of isotropic hyperelasticity based on artificial neural networks, *Comput. Mech.* 69 (1) (2022) 213–232, <http://dx.doi.org/10.1007/s00466-021-02090-6>.
- [52] D. Huang, K. Xu, C. Farhat, E. Darve, Learning constitutive relations from indirect observations using deep neural networks, *J. Comput. Phys.* 416 (2020) 109491, <http://dx.doi.org/10.1016/j.jcp.2020.109491>.
- [53] J. Dornheim, L. Morand, H.J. Nallani, D. Helm, Neural networks for constitutive modeling: From universal function approximators to advanced models and the integration of physics, *Arch. Comput. Methods Eng.* 31 (2) (2024) 1097–1127, <http://dx.doi.org/10.1007/s11831-023-10009-y>.
- [54] M. Flaschel, S. Kumar, L. De Lorenzis, Discovering plasticity models without stress data, *npj Comput. Mater.* 8 (1) (2022) 91, <http://dx.doi.org/10.1038/s41524-022-00752-4>.
- [55] K. Sagiya, K. Garikipati, Machine learning materials physics: deep neural networks trained on elastic free energy data from martensitic microstructures predict homogenized stress fields with high accuracy, 2019, arXiv preprint arXiv:1901.00524.
- [56] M. Fernández, M. Jamshidian, T. Böhlke, K. Kersting, O. Weeger, Anisotropic hyperelastic constitutive models for finite deformations combining material theory and data-driven approaches with application to cubic lattice metamaterials, *Comput. Mech.* 67 (2021) 653–677.
- [57] L. Zee, E. Sternberg, Ordinary and strong ellipticity in the equilibrium theory of incompressible hyperelastic solids, *Arch. Ration. Mech. Anal.* 83 (1) (1983) 53–90, <http://dx.doi.org/10.1007/BF00281087>.
- [58] J.M. Ball, Convexity conditions and existence theorems in nonlinear elasticity, *Arch. Ration. Mech. Anal.* 63 (4) (1976) 337–403, <http://dx.doi.org/10.1007/BF00279992>.
- [59] J. Schröder, P. Neff, Poly-, Quasi- and Rank-One Convexity in Applied Mechanics, vol. 516, Springer Science & Business Media, 2010.
- [60] B. Amos, L. Xu, J.Z. Kolter, Input convex neural networks, in: International Conference on Machine Learning, PMLR, 2017, pp. 146–155.
- [61] D.K. Klein, M. Fernández, R.J. Martin, P. Neff, O. Weeger, Polyconvex anisotropic hyperelasticity with neural networks, *J. Mech. Phys. Solids* 159 (2022) 104703.
- [62] F. As'ad, P. Avery, C. Farhat, A mechanics-informed artificial neural network approach in data-driven constitutive modeling, *Internat. J. Numer. Methods Engrg.* 123 (12) (2022) 2738–2759.
- [63] D.K. Klein, F.J. Roth, I. Valizadeh, O. Weeger, Parametrized polyconvex hyperelasticity with physics-augmented neural networks, *Data-Centric Eng.* 4 (2023) e25.
- [64] T. Gärtner, M. Fernández, O. Weeger, Nonlinear multiscale simulation of elastic beam lattices with anisotropic homogenized constitutive models based on artificial neural networks, *Comput. Mech.* 68 (5) (2021) 1111–1130, <http://dx.doi.org/10.1007/s00466-021-02061-x>.
- [65] M. Fernández, F. Fritzen, O. Weeger, Material modeling for parametric, anisotropic finite strain hyperelasticity based on machine learning with application in optimization of metamaterials, *Internat. J. Numer. Methods Engrg.* 123 (2) (2022) 577–609.
- [66] V. Deshpande, M. Ashby, N. Fleck, Foam topology: Bending versus stretching dominated architectures, *Acta Mater.* 49 (6) (2001) 1035–1040, [http://dx.doi.org/10.1016/S1359-6454\(00\)00379-7](http://dx.doi.org/10.1016/S1359-6454(00)00379-7).
- [67] P.P. Indurkar, S. Karlapati, A.J.D. Shaikhe, V.S. Deshpande, Predicting deformation mechanisms in architected metamaterials using GNN, 2022, arXiv preprint arXiv:2202.09427.
- [68] L. Makatura, B. Wang, Y.-L. Chen, B. Deng, C. Wojtan, B. Bickel, W. Matusik, Procedural metamaterials: A unified procedural graph for metamaterial design, *ACM Trans. Graph.* 42 (5) (2023) 1–19.
- [69] E. Ross, D. Hambleton, Using graph neural networks to approximate mechanical response on 3d lattice structures, in: Proceedings of AAG2020-Advances in Architectural Geometry, vol. 24, 2021, pp. 466–485.
- [70] P.P. Meyer, C. Bonatti, T. Tancogne-Dejean, D. Mohr, Graph-based metamaterials: Deep learning of structure-property relations, *Mater. Des.* 223 (2022) 111175.
- [71] M. Maurizi, C. Gao, F. Berto, Predicting stress, strain and deformation fields in materials and structures with graph neural networks, *Sci. Rep.* 12 (1) (2022) 21834.
- [72] L. Xiao, G. Shi, W. Song, Machine learning predictions on the compressive stress-strain response of lattice-based metamaterials, *Int. J. Solids Struct.* (2024) 112893.
- [73] D. Ha, A. Dai, Q.V. Le, Hypernetworks, 2016, arXiv preprint arXiv:1609.09106.
- [74] A. Navon, A. Shamsian, G. Chechik, E. Fetaya, Learning the pareto front with hypernetworks, 2020, arXiv preprint arXiv:2010.04104.
- [75] R.K. Mahabadi, S. Ruder, M. Dehghani, J. Henderson, Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks, 2021, arXiv preprint arXiv:2106.04489.
- [76] S.-A. Rebuffi, H. Bilen, A. Vedaldi, Efficient parametrization of multi-domain deep neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8119–8127.
- [77] Y. Li, R. Bhardwaj, A. Mehrish, B. Cheng, S. Poria, HyperTTS: Parameter efficient adaptation in text to speech using hypernetworks, 2024, arXiv:2404.04645.
- [78] A. Alanov, V. Titov, D.P. Vetrov, Hyperdomainnet: Universal domain adaptation for generative adversarial networks, *Adv. Neural Inf. Process. Syst.* 35 (2022) 29414–29426.
- [79] G. Liang, K. Chandrashekhara, Neural network based constitutive model for elastomeric foams, *Eng. Struct.* 30 (7) (2008) 2002–2011, <http://dx.doi.org/10.1016/j.engstruct.2007.12.021>.

- [80] D. Yang Gao, P. Neff, I. Roventa, C. Thiel, On the convexity of nonlinear elastic energies in the right Cauchy-green tensor, *J. Elasticity* 127 (2) (2017) 303–308, <http://dx.doi.org/10.1007/s10659-016-9601-6>.
- [81] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016.
- [82] D.-A. Clevert, T. Unterthiner, S. Hochreiter, Fast and accurate deep network learning by exponential linear units (elus), 2015, arXiv preprint [arXiv:1511.07289](https://arxiv.org/abs/1511.07289).
- [83] X. Glorot, A. Bordes, Y. Bengio, Deep sparse rectifier neural networks, in: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings*, 2011, pp. 315–323.
- [84] Mechanics and Materials Lab., Ae108, 2020, <http://dx.doi.org/10.5905/ethz-1007-257>.
- [85] G. Yu, L. Xiao, W. Song, Deep learning-based heterogeneous strategy for customizing responses of lattice structures, *Int. J. Mech. Sci.* 229 (2022) 107531, <http://dx.doi.org/10.1016/j.ijmecsci.2022.107531>.
- [86] T. Galanti, L. Wolf, On the modularity of hypernetworks, in: *Advances in Neural Information Processing Systems*, vol. 33, Curran Associates, Inc., 2020, pp. 10409–10419.
- [87] R.K. Mahabadi, S. Ruder, M. Dehghani, J. Henderson, Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks, 2021, <http://dx.doi.org/10.48550/arXiv.2106.04489>, [arXiv:2106.04489](https://arxiv.org/abs/2106.04489).
- [88] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch, in: *NIPS 2017 Workshop on Autodiff*, 2017, URL <https://openreview.net/forum?id=BJJsrmlfCZ>.
- [89] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, *Nature* 521 (7553) (2015) 436–444.
- [90] R.N. Glaesener, E.A. Träff, B. Telgen, R.M. Canonica, D.M. Kochmann, Continuum representation of nonlinear three-dimensional periodic truss networks by on-the-fly homogenization, *Int. J. Solids Struct.* 206 (2020) 101–113.
- [91] B. Telgen, O. Sigmund, D.M. Kochmann, Topology optimization of graded truss lattices based on on-the-fly homogenization, *J. Appl. Mech.* 89 (6) (2022) 061006.
- [92] R. Glaesener, S. Kumar, C. Lestringant, T. Butruille, C. Portela, D. Kochmann, Predicting the influence of geometric imperfections on the mechanical response of 2D and 3D periodic trusses, *Acta Mater.* 254 (2023) 118918, <http://dx.doi.org/10.1016/j.actamat.2023.118918>.
- [93] O. Al-Ketan, R.K. Abu Al-Rub, Multifunctional mechanical metamaterials based on triply periodic minimal surface lattices, *Adv. Eng. Mater.* 21 (10) (2019) 1900524, <http://dx.doi.org/10.1002/adem.201900524>.
- [94] O. Al-Ketan, D.-W. Lee, R. Rowshan, R.K. Abu Al-Rub, Functionally graded and multi-morphology sheet TPMS lattices: design, manufacturing, and mechanical properties, *J. Mech. Behav. Biomed. Mater.* 102 (2020) 103520, <http://dx.doi.org/10.1016/j.jmbbm.2019.103520>.
- [95] W. Deng, S. Kumar, A. Vallone, D.M. Kochmann, J.R. Greer, AI-enabled materials design of non-periodic 3D architectures with predictable direction-dependent elastic properties, *Adv. Mater.* (2024) <http://dx.doi.org/10.1002/adma.202308149>.
- [96] Y. Guo, S. Sharma, S. Kumar, Inverse designing surface curvatures by deep learning, *Adv. Intell. Syst.* 6 (6) (2024) <http://dx.doi.org/10.1002/aisy.202300789>.
- [97] J.N. Fuhg, G.A. Padmanabha, N. Bouklas, B. Bahmani, W. Sun, N.N. Vlassis, M. Flaschel, P. Carrara, L. De Lorenzis, A review on data-driven constitutive laws for solids, 2024, arXiv preprint [arXiv:2405.03658](https://arxiv.org/abs/2405.03658).
- [98] F. de Avila Belbute-Peres, Y.-f. Chen, F. Sha, HyperPINN: Learning parameterized differential equations with physics-informed hypernetworks, in: *The Symbiosis of Deep Learning and Differential Equations*, vol. 690, 2021.
- [99] L. Zheng, Hypercan, 2024, <http://dx.doi.org/10.5281/zenodo.13947082>, <https://github.com/li-zhengz/HyperCAN>.
- [100] A.L. Maas, A.Y. Hannun, A.Y. Ng, et al., Rectifier nonlinearities improve neural network acoustic models, in: *Proc. Icml*, vol. 30, Atlanta, GA, 2013, p. 3.
- [101] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2017, arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).
- [102] L. Prechelt, An empirical comparison of c, c++, java, perl, python, rexx and tcl, *IEEE Comput.* 33 (10) (2000) 23–29.
- [103] L. Zheng, Hypercan: Hypernetwork-driven deep parameterized constitutive models for metamaterials. Dataset and model checkpoints, 2024, <http://dx.doi.org/10.3929/ethz-b-000699994>.