J. Mech. Phys. Solids 196 (2025) 106015

Contents lists available at ScienceDirect

Journal of the Mechanics and Physics of Solids

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

Consistent machine learning for topology optimization with microstructure-dependent neural network material models

Harikrishnan Vijayakumaran [a], Jonathan B. Russ [b], Glaucio H. Paulino [b,c], Miguel A. Bessa [d,*]

[a] Department of Materials Science and Engineering, Delft University of Technology, Delft, 2628 CD, The Netherlands
[b] Department of Civil and Environmental Engineering, Princeton University, Princeton, NJ 08544, United States
[c] Princeton Materials Institute (PMI), Princeton University, Princeton, NJ 08544, United States
[d] School of Engineering, Brown University, Providence, RI 02912, United States

ARTICLE INFO

Keywords:
Topology optimization
Material-integrated design
Multi-scale structures
Functionally-graded materials
Neural networks
Deep learning

ABSTRACT

Additive manufacturing methods together with topology optimization have enabled the creation of multiscale structures with controlled spatially-varying material microstructure. However, topology optimization or inverse design of such structures in the presence of nonlinearities remains a challenge due to the expense of computational homogenization methods and the complexity of differentiably parameterizing the microstructural response. A solution to this challenge lies in machine learning techniques that offer efficient, differentiable mappings between the material response and its microstructural descriptors. This work presents a framework for designing multiscale heterogeneous structures with spatially varying microstructures by merging a homogenization-based topology optimization strategy with a consistent machine learning approach grounded in hyperelasticity theory. We leverage neural architectures that adhere to critical physical principles such as polyconvexity, objectivity, material symmetry, and thermodynamic consistency to supply the framework with a reliable constitutive model that is dependent on material microstructural descriptors. Our findings highlight the potential of integrating consistent machine learning models with density-based topology optimization for enhancing design optimization of heterogeneous hyperelastic structures under finite deformations.

1. Introduction

The growth of computational resources, together with advances in additive manufacturing (AM), have brought topology optimization (TO) to the forefront of engineering design. The foundational work by Bendsøe and Kikuchi (1988), which addressed the optimization of material distribution through a homogenized treatment of the microstructure, provided the basis for the approaches we have today. Despite this, the subsequently developed methods, such as the solid isotropic material with penalization (SIMP) method (Bendsøe, 1989; Zhou and Rozvany, 1991) and the level set method (Allaire et al., 2004) were favored over homogenization-based TO because of early challenges related to manufacturability, small lengthscale effects, and microstructural connectivity. Today, with advances in additive manufacturing technology, it is now possible to 3D print various graded microstructures (see e.g. Schumacher et al., 2015), which has prompted a resurgence of homogenization-based methods (Pantz and Trabelsi, 2008; Groen and Sigmund, 2017; Groen et al., 2020) in multiscale TO. These developments not only enabled the ability to design structures of unprecedented complexity at multiple scales but also their physical realization (Sanders et al., 2021).

* Corresponding author.
E-mail address: miguel_bessa@brown.edu (M.A. Bessa).

https://doi.org/10.1016/j.jmps.2024.106015
Received 25 August 2024; Received in revised form 11 December 2024; Accepted 13 December 2024
Available online 24 December 2024

Nomenclature

Abbreviations

R^2	Coefficient of determination
FE^2	Finite element squared
3D	Three-dimensional
AM	Additive manufacturing
FEM	Finite element method
FFN	Feed-forward neural network
ICNN	Input convex neural network
MAE	Mean absolute error
ML	Machine learning
NN	Neural network
PICNN	Partial input convex neural network
RMSE	Root mean squared error
RVE	Representative volume element
SIMP	Solid Isotropic Material with Penalization
TO	Topology optimization

List of Symbols

α	Particle volume fraction
$\bar{\rho}, \bar{\rho}_e$	Projected pseudo-density design variable and its element-wise value
$\boldsymbol{B}$	Left Cauchy–Green deformation tensor
$\boldsymbol{C}$	Right Cauchy–Green deformation
$\boldsymbol{E}$	Green–Lagrange strain
β_ψ	Projection strength parameter for the strain energy density interpolation factor
β_ρ	Projection strength parameter for the pseudo-density design variable
$\boldsymbol{F}$	Deformation gradient
$\boldsymbol{F}_{e\gamma_e}$	Effective deformation gradient in element e
$\boldsymbol{h}_{ei}$	Linear filter function in filtering operation
$\boldsymbol{\Lambda}, \tilde{\boldsymbol{\Lambda}}$	Principal stretch-based transformed input space for ICNN
$\boldsymbol{P}$	First Piola–Kirchhoff stress
$\boldsymbol{\phi}^\alpha, \phi^\alpha_{ij}$	Filter matrix for the particle volume fraction design variable
$\boldsymbol{\phi}^\rho, \phi^\rho_{ei}$	Filter matrix for the pseudo-density design variable
$\boldsymbol{Q}$	Orthogonal transformation tensor
$\boldsymbol{S}$	Second Piola–Kirchhoff stress
$\boldsymbol{\Theta}, \boldsymbol{\Theta}_e$	Set of design variables and element-wise set of design variables
$\boldsymbol{\varphi}, \boldsymbol{\varphi}_e$	Deformation field and its element-wise value
$\boldsymbol{x}_e, \boldsymbol{x}_i$	Centroid of element e and i respectively
$\mathcal{G}, \tilde{\mathcal{G}}$	Convex function
χ_e	SIMP interpolation factor for element e
$\mathcal{N}, \mathcal{N}_0$	Output of the ICNN and correction term
$\mathcal{T}$	Transformation layer in the ICNN-based model
ϵ	SIMP ersatz parameter
η_ψ	Projection threshold parameter for the strain energy density interpolation factor
η_ρ	Projection threshold parameter for the pseudo-density design variable
γ_e	Strain energy density interpolation factor for element e
$\hat{\alpha}, \hat{\alpha}_e$	Filtered particle volume fraction design variable and its element-wise value
$\hat{\rho}, \hat{\rho}_e$	Filtered pseudo-density design variable and its element-wise value
$\lambda_1, \lambda_2, \lambda_3$	Principal stretches of the deformation gradient
$\langle \boldsymbol{P}_\mu \rangle$	Volume averaged microscopic first Piola–Kirchhoff stress

The research in this field is far from complete, as the progression to multiscale TO within a geometrically and materially nonlinear setting has significant challenges. Parallel developments have been made in multimaterial additive manufacturing (Gaynor et al., 2014) which paved the way to the development of multimaterial TO (Sanders et al., 2018a,b) and the extension to hyperelastic

$\mathrm{cof}\,\boldsymbol{F}$	Cofactor of the deformation gradient
$\mathrm{d_m}$	Microstructural descriptor
$\nabla \boldsymbol{u}_e$	Displacement gradient in element e
∇_{sym}	Symmetric gradient operator
$\psi, \hat{\psi}, \tilde{\psi}$	Strain energy density
ψ_{e,γ_e}	Interpolated strain energy density in element e
ψ_e	Strain energy density in element e
ψ_L	Small deformation linear elastic strain energy density
ρ, ρ_e	Pseudo-density design variable and its element-wise value
e	Element index
$\boldsymbol{J}$	Determinant of the deformation gradient
p	SIMP penalization parameter
r_ρ	Filter radius for the pseudo-density design variable

multimaterial TO undergoing large deformations (Zhang et al., 2020). However, the existing TO literature has not yet demonstrated multiscale design with nonlinear material behavior at reasonable computational expense. The primary limitation that has to be addressed during nonlinear multiscale TO is a computational bottleneck that arises while linking the finer scales to the global, coarser scale. While homogenization techniques offer an excellent means to bridge scales in the forward problem when there exists a clear separation of scales, the standard analytical strategies fall short in the presence of material and geometrical nonlinearities. The classical alternative to this is to employ an incremental computational homogenization scheme in a nested setting, known as the FE^2 method (Feyel, 1999), to solve the forward problem. As pointed out in the extensive review by Geers et al. (2010), the computational expense associated with numerical homogenization calls for strategies that exploit them efficiently. The findings in Xia and Breitkopf (2016) also echo this concern that the principal difficulty in realizing nonlinear multiscale structures is indeed the excessive computational burden that arises due to multiple realizations of FE^2 computing within the TO problem. Their conclusions indicate that employing model-reduction techniques through database-type methods or data-driven techniques offers a viable compromise.

Amongst these methods, neural network (Hornik et al., 1989; Ghaboussi et al., 1991) and Gaussian process (Rasmussen and Williams, 2006) models are particularly attractive for explicitly mapping the microstructural descriptors to the homogenized response (Bessa et al., 2017). Notably, neural networks with recurrent units are even capable of modeling history-dependent material responses such as plasticity (Mozaffar et al., 2019). Furthermore, a natural advantage of employing neural networks to capture the homogenized constitutive behavior is the ease of obtaining derivatives for both the tangent stiffness of the material as well as the sensitivities in TO, leveraging automatic differentiation. However, the application of ML-based constitutive models in the context of nonlinear multiscale TO has been limited and demonstrated only to design structures undergoing infinitesimal elastic deformation (White et al., 2019; Chandrasekhar et al., 2023), where analytical homogenization theories would suffice. Simultaneously, there is a growing literature on the use of ML in TO, with the overarching goal of reducing computational cost. For example, the work of Chi et al. (2021) introduces a method to accelerate the optimization procedure by employing an ML technique to discover an underlying mapping between the design variables and the objective function sensitivity. Yet, as can be concluded from the reviews by Woldseth et al. (2022), Shin et al. (2023), substantial contributions on the use of neural network-based material models in multiscale TO for large deformation problems have yet to be made.

The fundamental limitation that has hindered the use of typical *black-box* neural network (NN) constitutive models is their failure to adhere to established physical principles, particularly in data-scarce situations, leading to concerns about accuracy and numerical stability. In the case of hyperelasticity, this includes thermodynamic consistency, objectivity, the existence of a *natural state* (Coleman and Noll, 1959), quasiconvexity, and volumetric growth conditions (Ball, 1976), in addition to material-dependent symmetry conditions. *Polyconvexity* is a convenient and tractable condition to ensure quasiconvexity (and thereby ellipticity), which is needed to prove the existence of minimizers for the variational problem being solved in the forward analysis. However, in the case of hyperelastic composites, like the ones we consider here, the homogenization step can lead to a loss of ellipticity (and therefore quasiconvexity) despite each phase individually being polyconvex (Braides, 1994). This implies that even under a data-rich scenario, a black-box model trained using homogenized response data can exhibit microstructural instabilities leading to numerical difficulties in the forward analyses performed in the optimization loop. Thus, it becomes all the more important to wrap the homogenized response with a polyconvex envelope (sometimes referred to as *polyconvexification*, see Avazmohammadi and Ponte Castañeda, 2016) to ensure that the response is stable during the optimization procedure. The development of the input convex neural networks (ICNNs) (Amos et al., 2017; Chen et al., 2019) has enabled significant progress in realizing NN-based models that attempt to incorporate polyconvexity among other relevant physical principles (As'ad et al., 2022; Klein et al., 2022). In particular, Linden et al. (2023) have provided a rigorous framework for using ICNN-based models in the context of hyperelasticity formulated in terms of strain invariants.

This article focuses on multiscale TO by introducing mechanically consistent machine learning material models into TO problems considering hyperelastic composites, therefore replacing the use of conventional or phenomenological hyperelastic constitutive

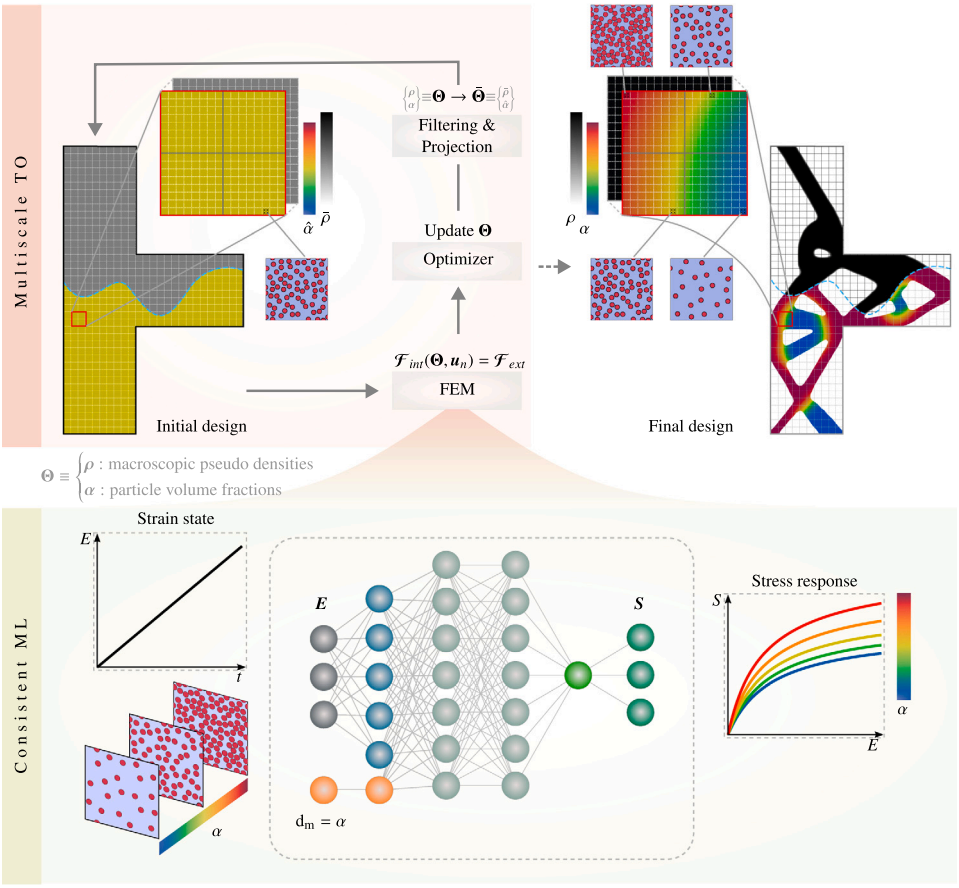


Fig. 1. An overview of the proposed consistent machine learning-driven topology optimization framework for multiscale hyperelastic structures. The consistent ML block learns the mapping between microstructural descriptors d_m (set as the particle volume fraction α) and the homogenized constitutive response in an offline phase adhering to the hyperelastic principles. The multiscale TO block treats α as an additional design variable using its filtered counterpart to approximate the spatially varying material response. By employing a differentiable ML material model in the forward analyses, the optimizer can efficiently update the design variables using analytically computed sensitivities of the objective/constraint functions obtained through automatic differentiation. The magnified regions in the multiscale TO block illustrate the filtered and projected pseudo-density field $\bar{\rho}$ and the filtered microstructural particle volume fraction field $\hat{\alpha}$ in the initial design state (left) and the final optimized state (right). Representative microstructures at select integration points are shown for the initial and final design states.

models in TO. The proposed methodology is valid when there exists scale separation between the material microstructure and the macroscale structural design, as illustrated in Fig. 1 and discussed in detail later. The goal of this contribution is to provide a design optimization framework for structures undergoing nonlinear material behavior while controlling macroscopic heterogeneity. When compared to the literature on nonlinear small-strain multiscale TO (Xia and Breitkopf, 2016, 2015), our consistent ML-based TO framework offers expressive yet flexible models that can learn microstructure-dependent homogenized responses in a finite strain setting. Our method deviates from the conventional FE^2-based concurrent TO strategies by circumventing the nested computational evaluations. Instead, we employ the microstructure-dependent ML model that converts the (average) macroscopic strain state to an (average) macroscopic stress state by replacing the microscale RVE with the ML model, enabling a two-field optimization problem as illustrated in Fig. 1. Furthermore, we demonstrate the importance of considering data-driven material models that include appropriate Physics constraints, which improves the robustness of the optimization process.

To that end, we propose an enhancement to the work of Linden et al. (2023) on ICNN-based models by means of a principal stretch-based hyperelastic model that adheres to the necessary conditions for isotropic hyperelasticity (Ball, 1976) and use it as a consistent machine learning block in the proposed TO framework illustrated in Fig. 1. A single microstructural descriptor, the microstructural volume fraction, is considered to describe the microstructural variations; however, the framework can be extended to include multiple microstructural descriptors, such as the orientation of the microstructural phases to capture anisotropy in the homogenized response, among others. The data-driven design and analysis framework developed by Bessa et al. (2017) and recently made open-source as the f3dasm package (van der Schelling et al., 2024) is employed to streamline the data generation using RVEs (Yi and Bessa, 2023) that are needed for training the consistent ML constitutive models. We first compare the performance of consistent ML models against classical single-scale phenomenological models through a series of TO examples for maximizing external work, providing confidence in the approach prior to demonstrating the multiscale topology optimization results.

Subsequently, we demonstrate the design optimization examples that result in structures with spatially varying microstructures with different particle volume fractions and contrast them with structures without microstructural variations (fixed microstructures with constant volume fraction).

2. Macroscopic constitutive modeling of hyperelastic materials

The multiscale topology optimization framework, shown in Fig. 1, is realized by using a consistent ML-based constitutive model that simply replaces the typical phenomenological constitutive model of the finite element analyses conducted during TO. The ML model is trained in an offline procedure by homogenizing the response of Representative Volume Elements (RVEs) under random deformation paths, as originally proposed in Bessa et al. (2017), Mozaffar et al. (2019). Therefore, prior to integrating the ML material model in the TO framework, a data-driven process is formulated to train the microstructure-dependent constitutive model. The process is detailed in Section 5, which can be broadly summarized in three steps: (1) design of experiments to generate samples of RVE microstructures based on the microstructural descriptors, (2) finite element analyses of the RVEs subject to applied average strain states, followed by a first-order homogenization scheme to determine the corresponding average stress states, and (3) training the ML model to establish the surrogate constitutive law whose inputs are the macroscopic strain states and the microstructural descriptors of interest (e.g., particle volume fraction of the RVE microstructure), and whose outputs are the homogenized, macroscopic stress states. As usual, we assume a separation of scales, with the characteristic dimension of the RVE being much smaller than that of the macroscopic problem while also being larger than the characteristic dimension of the microstructural features of each material phase. As mentioned, this first-order homogenization strategy employed for data generation is identical to past works (Yvonnet et al., 2013; Bessa et al., 2017), but Appendix F is also provided for readers unfamiliar with how the RVE boundary value problem is defined.

Throughout this manuscript, we refer to the kinematic and kinetic quantities at the macroscopic scale to avoid confusion between the single and multiple scale settings. We consider a total Lagrangian formulation and choose the Green–Lagrange strain tensor $\boldsymbol{E}$ as the primary kinematic variable, computed using the deformation gradient $\boldsymbol{F}$ and the right Cauchy–Green deformation tensor $\boldsymbol{C}$ via

$$\boldsymbol{C} := \boldsymbol{F}^{\mathsf{T}}\boldsymbol{F} \quad ; \quad \boldsymbol{E} := \frac{1}{2}(\boldsymbol{C} - \boldsymbol{1}) \,, \tag{1}$$

and the second Piola–Kirchhoff stress $\boldsymbol{S}$ as the corresponding energetically conjugate variable. To avoid interpenetration of matter, the condition $J := \det \boldsymbol{F} \overset{!}{>} 0$ must hold. In the multiscale setting, these quantities correspond to their homogenized counterparts in accordance with the Hill–Mandel lemma. From the solution to the (periodic) boundary value problem at the microscale subject to the macroscopic deformation gradient $\boldsymbol{F}$, the macroscopic Piola–Kirchhoff stress is obtained through a volume average of its microscopic counterpart, i.e. $\boldsymbol{P} \equiv \langle \boldsymbol{P}_{\mu} \rangle$. The second Piola–Kirchhoff stress is then obtained from the relation $\boldsymbol{S} := \boldsymbol{F}^{-1}\boldsymbol{P}$. With the definition of consistent macroscopic quantities, the conditions set for the constitutive relation between the second Piola–Kirchhoff stress $\boldsymbol{S}$ and the Green–Lagrange strain $\boldsymbol{E}$ are identical for the single and multiple scale settings, and hence the following discussion is valid for both.

2.1. Hyperelasticity conditions

In order to enforce consistency with the constitutive restrictions relevant to hyperelastic materials in the ML models, this section briefly reviews the fundamental concepts of hyperelasticity. The constitutive definition of hyperelastic materials is provided through a strain energy density function ψ, whose derivative with respect to a strain measure directly relates to the definition of the *thermodynamically consistent* conjugate stress measure (Coleman and Noll, 1959). If we define $\psi = \hat{\psi}(\boldsymbol{E})$, then the energetically conjugate stress measure — the second Piola–Kirchhoff stress, is obtained as $\boldsymbol{S} = \partial_{\boldsymbol{E}}\hat{\psi}$.

The reference configuration of the body must correspond to a *natural state* (Coleman and Noll, 1959), i.e. a zero-valued minimum stored energy density state that is stress-free:

$$\hat{\psi}(\boldsymbol{E}) \overset{!}{\geq} 0 \quad \text{s.t.} \quad \hat{\psi}(\boldsymbol{E})|_{\boldsymbol{E}=\boldsymbol{0}} \overset{!}{=} 0 \quad \text{and} \quad \boldsymbol{S}(\boldsymbol{E})|_{\boldsymbol{E}=\boldsymbol{0}} \overset{!}{=} \boldsymbol{0} \tag{2}$$

Following the *principle of objectivity* (Coleman and Noll, 1959), the material behavior should be independent of the observer, implying that the strain energy density remains invariant to orthogonal transformations $\boldsymbol{Q}$ to the displacement gradient $\boldsymbol{F}$. Since the right Cauchy–Green deformation $\boldsymbol{C}$ is objective by definition, i.e. $\boldsymbol{C} = \boldsymbol{F}^{\mathsf{T}}\boldsymbol{F} = \boldsymbol{F}^{\mathsf{T}}\boldsymbol{Q}^{\mathsf{T}}\boldsymbol{Q}\boldsymbol{F} = (\boldsymbol{Q}\boldsymbol{F})^{\mathsf{T}}(\boldsymbol{Q}\boldsymbol{F})$, the Green–Lagrange strain $\boldsymbol{E}$ is also objective, c.f. (1). Thus, formulating the strain energy density as $\hat{\psi}(\boldsymbol{E})$ below ensures objectivity:

$$\tilde{\psi}(\boldsymbol{Q}\boldsymbol{F}) = \hat{\psi}(\boldsymbol{E}) = \tilde{\psi}(\boldsymbol{F}) \tag{3}$$

Furthermore, the material body cannot be compressed to a volume of zero or expanded to an infinite volume. As a result, the strain energy density of hyperelastic materials has to satisfy the volumetric *growth condition*:

$$\psi \to \infty \quad \text{as} \quad (J \to 0^{+} \quad \vee \quad J \to \infty) \tag{4}$$

In order to satisfy the requirements for the existence of total potential energy minimizers associated with the boundary value problem, the strain energy density ψ should also be *quasiconvex*. [1] For a twice differentiable function, quasiconvexity implies the satisfaction of the *Legendre–Hadamard* condition or *ellipticity* and thereby – rank-1 convexity. However, directly imposing quasiconvexity on a strain energy density ψ is challenging, as the non-pointwise nature of the condition makes it intractable to verify whether a given function is quasiconvex. Furthermore, the stringent growth conditions assumed for quasiconvexity (Morrey, 1952) preclude any singular behavior, such as the strain energy density $\psi \to \infty$ as $J \to 0^+$.

Polyconvexity (Ball, 1976) is a more tractable condition that arises from generalizing the notions of duality for convex functions to the vectorial context, bridging the above limitations. A strain energy density $\tilde{\psi}(\boldsymbol{F})$ is said to be polyconvex if:

$$\tilde{\psi}(\boldsymbol{F}) = \mathcal{G}(\boldsymbol{F}, \operatorname{cof} \boldsymbol{F}, \det \boldsymbol{F}) \quad \text{where} \quad \mathcal{G}(\cdot) \text{ is a convex function} \tag{5}$$

By restricting the strain energy density to be polyconvex, one may arrive at far more physically useful functions within the context of hyperelasticity, i.e., not only does polyconvexity imply quasiconvexity, but it also establishes the existence of minimizers that are valid under weaker growth conditions necessary to accommodate singular behavior such as $\psi \to \infty$ as $J \to 0^+$.

For isotropic hyperelasticity, Theorem 5.2 from Ball (1976) provides the sufficient conditions for polyconvexity of the strain energy density function. Let $\lambda_1, \lambda_2, \lambda_3$ be the principal stretches associated with the deformation gradient $\boldsymbol{F}$ and $\tilde{\mathcal{G}}(\cdot)$ a convex function in its arguments. Then a strain energy density function ψ of the form:

$$\tilde{\psi}(\boldsymbol{F}) = \tilde{\mathcal{G}}(\lambda_1, \lambda_2, \lambda_3, \lambda_1\lambda_2, \lambda_2\lambda_3, \lambda_3\lambda_1, J) \quad J = \lambda_1\lambda_2\lambda_3 = \det \boldsymbol{F} \tag{6}$$

is isotropic and polyconvex as per (5) if $\tilde{\mathcal{G}}$ is symmetric and non-decreasing in the principal stretches $(\lambda_1, \lambda_2, \lambda_3)$ as well as their pair-wise products $(\lambda_1\lambda_2, \lambda_2\lambda_3, \lambda_3\lambda_1)$.

Note that such a strain energy density function ψ has to additionally satisfy the natural state conditions (2) and the volumetric growth conditions (4) to be a valid and useful hyperelastic model, respectively. We remark that many of the existing isotropic hyperelastic models such as Neo-Hookean, Mooney–Rivlin, Arruda–Boyce, Blatz–Ko, etc. (see Holzapfel, 2002) satisfy these conditions via constructing the strain energy density in terms of the invariants of some objective strain measure such as the right Cauchy–Green deformation tensor $\boldsymbol{C}$ or the left Cauchy–Green deformation tensor $\boldsymbol{B}$. However the sufficient conditions still correspond to (6).

2.2. Consistent machine learning for material modeling

Feed-forward neural networks (FFN) have universal approximation properties (Hornik et al., 1989; Cybenko, 1989) and can represent any continuous function to arbitrary accuracy provided enough data. As opposed to the classical approach of formulating a strain energy density function ψ based on physical intuition and then fitting the parameters to experimental data, neural networks, with their superior representation capabilities, can directly learn arbitrarily complex material models from experimental or numerically generated data for nonlinear elastic and plastic properties (Bessa et al., 2017). However, as discussed in Section 2.1, the constitutive model for hyperelasticity should correspond to a *thermodynamically consistent strain energy density function* ψ that satisfies conditions (2), (3), (4), and (5) along with the necessary material symmetry conditions (e.g., isotropy). Standard FFNs do not have inductive biases that guarantee that the strain energy density function they learn satisfies the conditions set forth for hyperelasticity. Thus, a feasible alternative consists of employing the so-called input convex neural networks (ICNNs) (Amos et al., 2017) and constructing a polyconvex strain energy density function that satisfies the conditions for hyperelasticity (Klein et al., 2022; As'ad et al., 2022; Linden et al., 2023; Kalina et al., 2023, 2024).

Amongst the literature on ICNN-based ML models for hyperelasticity, the work by Linden et al. (2023) was the first to enforce the physical principles pertaining to hyperelasticity (see Section 2.1) rigorously. First, as we explain in Appendix A, the ICNN architecture has to be modified so that the appropriate convexity and non-negativity constraints are applied to the input of the network. Secondly, the enforcement of the natural state condition (2) should not compromise the polyconvexity condition. This is a point we shall revisit later in this section. These two issues were resolved in Linden et al. (2023) and their successive works. However, their approach to modeling material symmetry through invariants introduces an undesirable limitation in the model expressivity, particularly in the case of isotropy. Appendix B demonstrates an example of the limitation of invariant-based ICNN models. Since our interest is in modeling isotropic hyperelasticity with as much expressivity as possible, we propose a different formulation.

In our work, we employ the alternative ICNN formulation from Chen et al. (2019) to define the strain energy density function ψ for hyperelastic materials (see Appendix A). Unlike the prior invariant-based model architectures, we propose a model architecture that enforces the sufficient condition for polyconvexity for isotropic materials (6) in terms of the principal stretches $\lambda_1, \lambda_2, \lambda_3$. Based on the arguments we present in Appendices A and B, we propose to employ an ICNN-based model that takes in input $\boldsymbol{\Lambda} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_1\lambda_2, \lambda_2\lambda_3, \lambda_3\lambda_1, J, -J\}$. In order to enforce the symmetry with respect to the principal stretches $(\lambda_1, \lambda_2, \lambda_3)$ and

[1] Solving a typical boundary value involving hyperelastic materials corresponds to finding the minimizer for the total elastic energy potential $\Pi^{\text{total}}(\boldsymbol{\varphi}, \boldsymbol{F}) = \int_{\mathcal{B}_0} \tilde{\psi}(\boldsymbol{F})\, \mathrm{d}V - \Pi^{\text{external}}(\boldsymbol{\varphi})$. If one ignores the body force potential $\Pi^{\text{external}}(\boldsymbol{\varphi})$, the total energy potential $\Pi^{\text{total}}(\boldsymbol{\varphi}, \boldsymbol{F})$ can be viewed as a *functional* of the stored energy density *function* $\tilde{\psi}$. The existence of a minimizer for the total elastic energy potential relies on the constitutive restriction of the hyperelastic strain energy density to be quasiconvex (Ball, 1976). This follows from (Morrey, 1952, 1966) that quasiconvexity of a *function* (together with certain continuity and growth hypotheses) is the necessary and sufficient condition for its *functional* to be weakly lower semicontinuous, which in turn establishes the existence of minimizers for the *functional*.

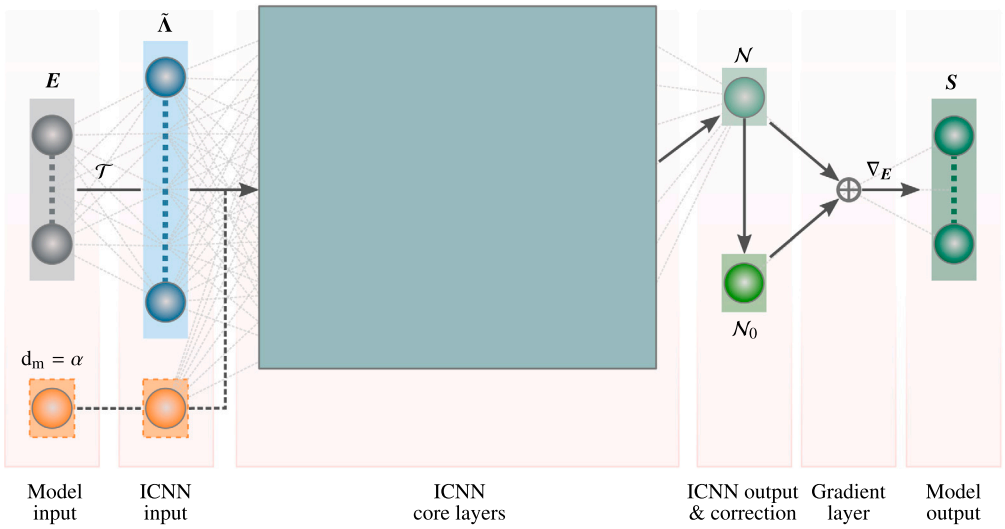


Fig. 2. Schematic illustration of the consistent machine learning model for isotropic hyperelasticity. The input to the neural network is $\boldsymbol{E}$, which passes through a fixed transformation layer $\mathcal{T}$ to arrive at $\tilde{\boldsymbol{\Lambda}}$ that subsequently goes through input convex neural layers to arrive at polyconvex output $\mathcal{N}$. This output is regularized for the natural state condition (2) and passed through a gradient layer, which obtains the second Piola–Kirchhoff stress $\boldsymbol{S}$ as in (10). Extension of the model to include microstructural descriptors d_m is achieved by appending the input space of the ICNN. The treatment of the microstructural descriptor d_m is such that the ICNN output (strain energy density) is convex with respect to the microstructural descriptor, which is a reasonable assumption for the volume fraction α of a stiff particle in a soft matrix.

their pair-wise products $(\lambda_1\lambda_2, \lambda_2\lambda_3, \lambda_3\lambda_1)$ while also remaining consistent with (6), we employ a simple and effective symmetric weight sharing strategy through which the symmetry condition for isotropy is inherently satisfied (see also Appendix B). The growth condition corresponding to $\psi \to \infty$ as $J \to \infty$ is handled naturally due to the convex nature of the ICNN. However, the singularity associated with $J = 0$ requires special treatment due to limited data availability in the region $J \to 0^+$. Yet we argue that the behavior of the strain energy density as $J \to 0^+$ has to be learned from the experimental data. For this reason, rather than adding a fixed coercive function to the strain energy density definition, as suggested in Linden et al. (2023) and other conventional hyperelastic models, we add a convex non-decreasing term to the input to our isotropic ICNN. We, therefore, choose the term $f(J) = -\ln J$ that is a convex, decreasing function in J ($J \overset{!}{>} 0$). By constructing the strain energy density as an ICNN with inputs $\tilde{\boldsymbol{\Lambda}} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_1\lambda_2, \lambda_2\lambda_3, \lambda_3\lambda_1, J, -J, -\ln J\}$, the singularity point becomes embedded in the model, and the volumetric growth condition (4) becomes satisfied and learnable from the experimental data.

Next, we address the natural state condition (2). This is achieved by adding a correction function $\mathcal{N}_0$ to the ICNN output that ensures that the strain energy density function ψ and the second Piola–Kirchhoff stress $\boldsymbol{S}$ vanishes at the natural state $\boldsymbol{E} = \boldsymbol{0}$. The overall strain energy density function ψ can be formulated as:

$$\hat{\psi}(\boldsymbol{E}) = \tilde{\mathcal{G}}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E})) = \mathcal{N}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E})) + \underbrace{\mathcal{N}_0^{\text{stress}}(\boldsymbol{E}) + \mathcal{N}_0^{\text{energy}}}_{\mathcal{N}_0(\boldsymbol{E})} \quad \text{where} \quad \mathcal{N}_0^{\text{energy}} = -\mathcal{N}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E}))\Big|_{\boldsymbol{E}=\boldsymbol{0}} \tag{7}$$

A stress correction term $\mathcal{N}_0^{\text{stress}}(\boldsymbol{E})$ formulated in the form:

$$\mathcal{N}_0^{\text{stress}}(\boldsymbol{E}) = -\partial_{\boldsymbol{E}}\mathcal{N}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E}))\Big|_{\boldsymbol{E}=\boldsymbol{0}} : \boldsymbol{E} \tag{8}$$

satisfies the natural state condition similarly to the correction terms proposed in As'ad et al. (2022) but violates polyconvexity in addition to the material symmetry condition, as remarked in Linden et al. (2023). To address this, we propose a stress correction term following the projection approach inspired by the weighted sum of derivatives method proposed in Linden et al. (2023) to obtain a polyconvex stress correction term:

$$\mathcal{N}_0^{\text{stress}}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E})) = \mathcal{N}_0^{\text{stress}}(J) = -c_0(J-1) \quad \text{with} \quad c_0 = -\sum_{i=1}^{9} \xi_i \partial_{\tilde{\Lambda}_i}\mathcal{N}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E}))\Big|_{\boldsymbol{E}=\boldsymbol{0}} \tag{9}$$

where $\boldsymbol{\xi} = \{1/3, 1/3, 1/3, 2/3, 2/3, 2/3, 1, -1, 1\}$ correspond to weights for the derivatives with respect to the inputs $\tilde{\boldsymbol{\Lambda}} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_1\lambda_2, \lambda_2\lambda_3, \lambda_3\lambda_1, J, -J, -\ln J\}$. A detailed derivation of the polyconvex stress correction term is provided in Appendix C. The second

Piola–Kirchhoff stress $\boldsymbol{S}$ is then obtained via differentiation of the strain energy density function $\hat{\psi}$ with respect to the Green Lagrange strain measure:

$$\begin{aligned}\boldsymbol{S} &= \partial_{\boldsymbol{E}}\hat{\psi}(\boldsymbol{E}) = \partial_{\boldsymbol{E}}\mathcal{N}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E})) + \partial_{\boldsymbol{E}}\mathcal{N}_0(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E})) \\ &= \partial_{\boldsymbol{E}}\mathcal{N}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E})) - \sum_{i=1}^{9}\xi\partial_{\tilde{\Lambda}_i}\mathcal{N}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E}))\Big|_{\boldsymbol{E}=\boldsymbol{0}}\boldsymbol{J}\boldsymbol{C}^{-1}\end{aligned} \tag{10}$$

The overall architecture of the resulting ICNN-based consistent ML model for hyperelasticity is illustrated in Fig. 2. The input to the neural network is the Green–Lagrange strain tensor $\boldsymbol{E}$, which is transformed to the input $\tilde{\boldsymbol{\Lambda}}$ for the ICNN through a fixed transformation layer $\mathcal{T}$. *The ICNN then computes the strain energy density function $\hat{\psi}(\boldsymbol{E})$ that satisfies objectivity, material symmetry, polyconvexity as well as the natural state condition (2) and the thermodynamically consistent second Piola–Kirchhoff stress $\boldsymbol{S}$ is obtained as in (10).* We remark that it is possible to use the ICNN architecture to define an isotropic hyperelastic model that satisfies the conditions for hyperelasticity (c.f. Linden et al., 2023). However, such a model can have limitations due to the hard enforcement of convexity on the pre-set invariants. A detailed discussion of this point is presented in Appendix B.

Extension of the consistent ML hyperelastic model to include microstructural descriptors is achieved by extending the input space of the ICNN so that the constitutive mapping now corresponds to $\boldsymbol{S}(\boldsymbol{E}, \mathrm{d_m}) = \partial_{\boldsymbol{E}}\hat{\psi}(\boldsymbol{E}, \mathrm{d_m})$. In the most generalized setting, the microstructural descriptor $\mathrm{d_m}$ could be included in the ML model through the partial input convex neural network (PICNN) (Amos et al., 2017), wherein no convexity-related constraints are imposed on output strain energy density function ψ with respect to the microstructural descriptor. In the present study involving two-phase hyperelastic composites with a soft matrix and stiff particle, we consider the microstructural descriptor $\mathrm{d_m}$ to be the volume fraction α of the stiff particle phase, which could be considered to hold a convex relationship with the strain energy density function ψ. As a result, the microstructural descriptor $\mathrm{d_m} = \alpha$ is introduced in the model through a simple extension of the input space of the ICNN, as illustrated in Fig. 2.

3. Topology optimization formulation

We employ a standard density-based topology optimization scheme in which each finite element, e, is assigned a scalar pseudo density parameter, $\rho_e \in [0, 1]$, reflecting the presence $\rho_e = 1$ or absence $\rho_e = 0$ of the material within the element. The design problem is regularized by means of a standard linear filtering technique (Bourdin, 2001) with a user-specified length scale, r_ρ, in order to mitigate the checkerboard effect and mesh dependency issues, resulting in the filtered design variables $\hat{\rho}_e$. This may be mathematically expressed as:

$$\hat{\rho}_e = \phi^{\rho}_{ei}\rho_i, \quad \phi^{\rho}_{ei} = \frac{h_{ei}V_i \text{ (no sum)}}{\sum_{j=1}^{N_{\text{elem}}} h_{ej}V_j}, \quad h_{ei} = \max\left(0,\ r_\rho - \left|\left|\boldsymbol{x}_e - \boldsymbol{x}_i\right|\right|\right) \tag{11}$$

where $\boldsymbol{x}_e$ represents the centroid of element e and $\|\cdot\|$ denotes the Euclidean norm.

Subsequently, the filtered design is projected by means of a smooth Heaviside function (Wang et al., 2010), in order to significantly reduce intermediate densities that may develop as a result of the filtering technique.

$$\bar{\rho}_e = \frac{\tanh(\beta_\rho\eta_\rho) + \tanh(\beta_\rho(\hat{\rho}_e - \eta_\rho))}{\tanh(\beta_\rho\eta_\rho) + \tanh(\beta_\rho(1 - \eta_\rho))} \tag{12}$$

The filtered and projected pseudo-density field $\bar{\rho}_e$ is then used in a SIMP interpolation scheme (Zhou and Rozvany, 1991; Bendsøe, 1989) during the forward analysis. To ensure numerical stability in the large deformation finite element simulations, we adopt the energy interpolation scheme proposed in Wang et al. (2014) which takes the form,

$$\psi_e = \chi_e(\bar{\rho}_e)\psi_{e,\gamma_e} \quad \text{with} \quad \chi_e(\bar{\rho}_e) = \epsilon + (1 - \epsilon)\bar{\rho}_e^p \tag{13}$$

where p is SIMP penalization parameter to penalize intermediate densities. The function ψ_{e,γ_e} is the interpolated strain energy density and is defined as

$$\begin{aligned}\psi_{e,\gamma_e} &= \psi(\boldsymbol{E}(\boldsymbol{F}_{e\gamma_e})) - \psi_L(\gamma_e\nabla_{\text{sym}}\boldsymbol{u}_e) + \psi_L(\nabla_{\text{sym}}\boldsymbol{u}_e) \\ &= \psi(\boldsymbol{E}(\boldsymbol{F}_{e\gamma_e})) - (1 - \gamma_e^2)\psi_L(\nabla_{\text{sym}}\boldsymbol{u}_e)\end{aligned} \tag{14}$$

which consists of the solid phase strain energy density $\psi(\cdot)$ and small deformation linear elastic stored energy density $\psi_L(\cdot)$. The effective deformation gradient and symmetric gradient operator used above are defined as

$$\boldsymbol{F}_{e\gamma_e} = \nabla\gamma_e\boldsymbol{\varphi}_e = \boldsymbol{1} + \gamma_e\nabla\boldsymbol{u}_e \quad \text{and} \quad \nabla_{\text{sym}}(\cdot) = 1/2(\nabla(\cdot) + \nabla(\cdot)^{\mathsf{T}}) \tag{15}$$

where the interpolation factor γ_e takes the value 1 for solid elements and 0 for void elements. This results in a geometrically nonlinear treatment for solid elements and small strain kinematics for the void elements. A smooth Heaviside projection function is used to compute the strain energy density interpolation parameter γ_e, providing a differentiable transition between the two kinematic formulations:

$$\gamma_e = \frac{\tanh(\beta_\psi\eta_\psi) + \tanh(\beta_\psi(\chi_e - \eta_\psi))}{\tanh(\beta_\psi\eta_\psi) + \tanh(\beta_\psi(1 - \eta_\psi))} \tag{16}$$

Algorithm 1 Incremental update of the nodal displacements & load factor

Input: Nodal displacements $\boldsymbol{u}_{n-1}$ and load factor ζ_{n-1} at pseudo-time t_{n-1}; Generalized applied displacement c_n
Output: Nodal displacements $\boldsymbol{u}_n$ and load factor ζ_n at pseudo-time t_n

Initialize Newton iteration counter $k = 1$
Initialize $\boldsymbol{u}_n^{(k-1)} = \boldsymbol{u}_{n-1}$, $\zeta_n^{(k-1)} = \zeta_{n-1}$
Initialize $\Delta\boldsymbol{u}_n^{(k-1)} = \boldsymbol{0}$, $\Delta\zeta_n^{(k-1)} = 0$, $\Delta c_n = c_n - c_{n-1}$
while $\left|\left|\mathbf{R}\left(\boldsymbol{\Theta}, \boldsymbol{u}_n^{(k-1)}, \zeta_n^{(k-1)}\right)\right|\right|_2 > \mathrm{tol}_{\mathrm{NR}} \cdot \left|\left|\mathbf{R}\left(\boldsymbol{\Theta}, \boldsymbol{u}_n^{(0)}, \zeta_n^{(0)}\right)\right|\right|_2$
  Compute $\mathbf{K}_T = \frac{\partial \mathbf{R}}{\partial \boldsymbol{u}}(\boldsymbol{\Theta}, \boldsymbol{u}_n^{(k-1)}, \zeta_n^{(k-1)})$
  Compute $\delta\boldsymbol{u}_R = -\mathbf{K}_T^{-1}\mathbf{R}(\boldsymbol{\Theta}, \boldsymbol{u}_n^{(k-1)}, \zeta_n^{(k-1)})$
  Compute $\delta\boldsymbol{u}_{f_0} = \mathbf{K}_T^{-1}\boldsymbol{f}_0$
  Compute $\delta\zeta = \frac{\Delta c_n - \boldsymbol{f}_0 \cdot \Delta\boldsymbol{u}_n^{(k-1)} + \boldsymbol{f}_0 \cdot \delta\boldsymbol{u}_R}{\boldsymbol{f}_0 \cdot \delta\boldsymbol{u}_{f_0}}$
  Compute $\delta\boldsymbol{u} = \delta\boldsymbol{u}_R + \delta\zeta\ \delta\boldsymbol{u}_{f_0}$
  Update $\Delta\boldsymbol{u}_n^{(k)} = \Delta\boldsymbol{u}_n^{(k-1)} + \delta\boldsymbol{u}$
  Update $\boldsymbol{u}_n^{(k)} = \boldsymbol{u}_{n-1} + \Delta\boldsymbol{u}_n^{(k)}$
return $\boldsymbol{u}_n = \boldsymbol{u}_n^{(k)}$, $\zeta_n = \zeta_n^{(k)}$

The set of design variables controlled by the optimizer, denoted by $\boldsymbol{\Theta} := \{\Theta_e | \forall e = 1 \cdots N_{elem}\}$ represents the macroscopic density design variable ρ_e, i.e. $\Theta_e \equiv \{\rho_e\}$ in the single-scale setting or additionally include the microstructural design variable, i.e. $\Theta_e \equiv \{\rho_e, \alpha_e\}$ at each element e in the multiscale setting. In a similar manner to the macroscopic density field, a design length scale on the particle volume fraction field is prescribed via a linear filtering operation to produce a filtered particle volume fraction, $\hat{\alpha}_e$, in each element to obtain a smooth variation of the field across the topology. Consistent with Eq. (11), we represent this operation, $\hat{\alpha}_i = \phi_{ij}^{\alpha}\alpha_j$, via the separate filter matrix $\boldsymbol{\phi}^{\alpha}$ constructed with user-specified radius, r_{α}. To simplify notation, the set of all *physical design variables*, i.e., the filtered (and projected) design variables, is labeled as $\bar{\boldsymbol{\Theta}}$. Depending upon the problem setting, $\bar{\boldsymbol{\Theta}}$ may be composed of the set of all filtered and projected macroscopic density design variables $\bar{\boldsymbol{\rho}}$ (single-scale) or additionally include the microstructural design variables, $\bar{\boldsymbol{\Theta}} \equiv \{\bar{\boldsymbol{\rho}}, \hat{\boldsymbol{\alpha}}\}$ (multiscale).

A total Lagrangian formulation is used to compute the internal force in each element (e) of the form:

$$\boldsymbol{\mathcal{F}}_{int_e}(\bar{\Theta}_e, \boldsymbol{u}_{e,n}) = \frac{\partial}{\partial \boldsymbol{u}_{e,n}} \int_{\Omega_{0_e}} \psi_e \, \mathrm{d}V \tag{17}$$

at pseudo-time increment, n, with integration over the element in the reference configuration, $\Omega_{0_e} \subset \Omega_0$. Subsequently, the element contributions are then assembled into their global counterpart, $\boldsymbol{\mathcal{F}}_{int}$. Global equilibrium is then achieved via the solution of the residual equations at the nth pseudo-time increment. As $\bar{\boldsymbol{\Theta}}$ may be represented as a function of $\boldsymbol{\Theta}$, the global residual equations may be written as:

$$\begin{aligned} \mathbf{R}(\boldsymbol{\Theta}, \boldsymbol{u}_n, \zeta_n) &= \boldsymbol{\mathcal{F}}_{int}(\boldsymbol{\Theta}, \boldsymbol{u}_n) - \zeta_n \boldsymbol{f}_0 = \boldsymbol{0} \\ \boldsymbol{f}_0 \cdot \boldsymbol{u}_n &= c_n \end{aligned} \tag{18}$$

The modified generalized displacement control method (Leon et al., 2014) is an effort to mitigate instabilities associated with load control in large strain topology optimization problems, while also providing more flexibility over the load distribution than standard displacement control. Note that $\boldsymbol{f}_0$ is a constant vector containing the load distribution, whereas the load factor, ζ_n, controls the magnitude of the applied load in each pseudo-time increment. The load factor at each step is determined via the additional equation in (18), representing a user-specified weighted average of the displacements with the constant nonzero external applied force distribution vector, $\boldsymbol{f}_0$. This weighted average displacement is constrained to be equal to the parameter c_n at the nth pseudo-time step. The forward analyses begin in an undeformed state corresponding to zero nodal displacements and zero loads (i.e., $\boldsymbol{u}_0 = \boldsymbol{0}$, $\zeta_0 = 0$). With $\boldsymbol{u}_{n-1}, \zeta_{n-1}$ known, the nodal displacements and load factor at the subsequent pseudo-time increment, n, are obtained via the procedure provided in Algorithm 1 with evenly spaced generalized applied displacements, c_n, between $c_0 = 0$ and the user-specified value, c_N, at the final step, corresponding to $n = N$.

The optimization problem is formulated as a work maximization problem subject to constraints imposed on the material volume fractions. This is defined mathematically via the following:

$$\begin{aligned} \max_{\boldsymbol{\Theta}} \quad & W_{ext} = \sum_{n=1}^{N} \frac{1}{2}\left(\zeta_n + \zeta_{n-1}\right)\left(c_n - c_{n-1}\right) \\ \text{s.t.} \quad & g^{(m)}(\boldsymbol{\Theta}) \le g^{(m)}_{\max}, \quad m = 1, \ldots, M \\ & \boldsymbol{\Theta}_{\min} \preceq \boldsymbol{\Theta} \preceq \boldsymbol{\Theta}_{\max} \\ \text{with:} \quad & \mathbf{R}(\boldsymbol{\Theta}, \boldsymbol{u}_n, \zeta_n) = \boldsymbol{\mathcal{F}}_{int}(\boldsymbol{\Theta}, \boldsymbol{u}_n) - \zeta_n \boldsymbol{f}_0 = \boldsymbol{0} \quad \forall n = 1, \ldots, N \\ & \boldsymbol{f}_0 \cdot \boldsymbol{u}_n = c_n \end{aligned} \tag{19}$$

where $\boldsymbol{\Theta}_{\min}$ and $\boldsymbol{\Theta}_{\max}$ represent the side constraint bounds on the design variables. For benchmark problems, there is a single constraint (i.e. $M = 1$) corresponding to the material volume fraction:

$$g^{(1)}(\boldsymbol{\Theta}) = g(\boldsymbol{\Theta}) = \frac{1}{\sum_{e=1}^{N_{elem}} V_e} \sum_{e=1}^{N_{elem}} \bar{\rho}_e V_e \tag{20}$$

where V_e is the volume of element e. The macroscopic design density variables are bounded by $\rho_e \in [0, 1]$ at the element level.

Subsequently, for the two-scale optimization examples, two separate constraints are included on each material phase (i.e. $M = 2$):

$$\begin{aligned} g^{(1)}(\boldsymbol{\Theta}) &= g^{\text{inc}}(\boldsymbol{\Theta}) = \frac{1}{\sum_{e=1}^{N_{elem}} V_e} \sum_{e=1}^{N_{elem}} \hat{\alpha}_e \bar{\rho}_e V_e \\ g^{(2)}(\boldsymbol{\Theta}) &= g^{\text{mat}}(\boldsymbol{\Theta}) = \frac{1}{\sum_{e=1}^{N_{elem}} V_e} \sum_{e=1}^{N_{elem}} (1 - \hat{\alpha}_e) \bar{\rho}_e V_e \end{aligned} \tag{21}$$

where the first constraint equation corresponds to the particle volume fraction and the second restricts the quantity of matrix material. Here, in addition to the macroscopic density design variables side constraint $\rho_e \in [0, 1]$, the particle volume fraction design variables are bounded by $\alpha_e \in [\alpha_\ell, \alpha_u]$ at the element level, where α_ℓ and α_u are the lower and upper bounds on the particle volume fraction design variables.

4. Sensitivity analysis

Here we provide the derivation for the sensitivity of the external work objective function with respect to the design variables, $\boldsymbol{\Theta}$. First, the sequence of algebraic operations reducing to our expression for the external work integrated using the trapezoidal rule is shown below.

$$\begin{aligned} W_{ext} &= \sum_{n=1}^{N} \frac{1}{2} \left(\boldsymbol{f}_{ext}^{(n)} + \boldsymbol{f}_{ext}^{(n-1)} \right) \cdot \left(\boldsymbol{u}_n - \boldsymbol{u}_{n-1} \right) = \sum_{n=1}^{N} \frac{1}{2} \left(\zeta_n \boldsymbol{f}_0 + \zeta_{n-1} \boldsymbol{f}_0 \right) \cdot \left(\boldsymbol{u}_n - \boldsymbol{u}_{n-1} \right) \\ &= \sum_{n=1}^{N} \frac{1}{2} \left(\zeta_n + \zeta_{n-1} \right) \left(\boldsymbol{f}_0 \cdot \boldsymbol{u}_n - \boldsymbol{f}_0 \cdot \boldsymbol{u}_{n-1} \right) = \sum_{n=1}^{N} \frac{1}{2} \left(\zeta_n + \zeta_{n-1} \right) \left(c_n - c_{n-1} \right) \end{aligned} \tag{22}$$

To obtain the sensitivity of this function, we directly differentiate the residual equation at time step, n.

$$\begin{aligned} \frac{d}{d\boldsymbol{\Theta}} \mathbf{R}(\boldsymbol{\Theta}, \boldsymbol{u}_n, \zeta_n) &= \frac{d}{d\boldsymbol{\Theta}} \boldsymbol{\mathcal{F}}_{int}(\boldsymbol{\Theta}, \boldsymbol{u}_n) - \boldsymbol{f}_0 \frac{d\zeta_n}{d\boldsymbol{\Theta}} = \mathbf{0} \\ &= \frac{\partial \boldsymbol{\mathcal{F}}_{int}}{\partial \boldsymbol{\Theta}} + \frac{\partial \boldsymbol{\mathcal{F}}_{int}}{\partial \boldsymbol{u}_n} \frac{d\boldsymbol{u}_n}{d\boldsymbol{\Theta}} - \boldsymbol{f}_0 \frac{d\zeta_n}{d\boldsymbol{\Theta}} = \mathbf{0} \\ &= \frac{\partial \boldsymbol{\mathcal{F}}_{int}}{\partial \boldsymbol{\Theta}} + \mathbf{K}_T \frac{d\boldsymbol{u}_n}{d\boldsymbol{\Theta}} - \boldsymbol{f}_0 \frac{d\zeta_n}{d\boldsymbol{\Theta}} = \mathbf{0} \\ \implies \frac{d\boldsymbol{u}_n}{d\boldsymbol{\Theta}} &= \mathbf{K}_T^{-1} \left(\boldsymbol{f}_0 \frac{d\zeta_n}{d\boldsymbol{\Theta}} - \frac{\partial \boldsymbol{\mathcal{F}}_{int}}{\partial \boldsymbol{\Theta}} \right) \end{aligned} \tag{23}$$

Recalling the constraint, $\boldsymbol{f}_0 \cdot \boldsymbol{u}_n = c_n$, we also have the following useful relationship:

$$\begin{aligned} &\frac{d}{d\boldsymbol{\Theta}} \left(\boldsymbol{f}_0 \cdot \boldsymbol{u}_n \right) = \boldsymbol{f}_0 \cdot \frac{d\boldsymbol{u}_n}{d\boldsymbol{\Theta}} = 0 \\ \implies &\boldsymbol{f}_0 \cdot \frac{d\boldsymbol{u}_n}{d\boldsymbol{\Theta}} = \boldsymbol{f}_0 \cdot \mathbf{K}_T^{-1} \left(\boldsymbol{f}_0 \frac{d\zeta_n}{d\boldsymbol{\Theta}} - \frac{\partial \boldsymbol{\mathcal{F}}_{int}}{\partial \boldsymbol{\Theta}} \right) = 0 \end{aligned} \tag{24}$$

Rearranging this final equation we have the sensitivity of the load factor at time step, n,

$$\frac{d\zeta_n}{d\boldsymbol{\Theta}} = \frac{\boldsymbol{f}_0 \cdot \mathbf{K}_T^{-1} \cdot \frac{\partial \boldsymbol{\mathcal{F}}_{int}}{\partial \boldsymbol{\Theta}}}{\boldsymbol{f}_0 \cdot \mathbf{K}_T^{-1} \cdot \boldsymbol{f}_0} = \frac{\delta \boldsymbol{u}_{f_0} \cdot \frac{\partial \boldsymbol{\mathcal{F}}_{int}}{\partial \boldsymbol{\Theta}}}{\boldsymbol{f}_0 \cdot \delta \boldsymbol{u}_{f_0}} \tag{25}$$

which does not require the solution of any additional system of equations (assuming that the $\delta \boldsymbol{u}_{f_0}$ vectors were saved during the forward analysis). All required partial derivatives can be easily obtained using automatic differentiation. Eq. (25) can then be used to obtain the final sensitivity of the external work as provided in Eq. (22) since each c_n is a constant.

$$\frac{dW_{ext}}{d\boldsymbol{\Theta}} = \sum_{n=1}^{N} \frac{1}{2} \left(\frac{d\zeta_n}{d\boldsymbol{\Theta}} + \frac{d\zeta_{n-1}}{d\boldsymbol{\Theta}} \right) \left(c_n - c_{n-1} \right) \tag{26}$$

Finally, the chain rule is required to obtain the derivative with respect to the design variables, $\{\boldsymbol{\rho}, \boldsymbol{\alpha}\}$, through the projection function and the corresponding filters in the standard manner.

5. Numerical experiments and results

In the present study, we consider two-phase composite material microstructures with stiff particles embedded in a soft matrix. An Arruda–Boyce (AB) hyperelastic model (Arruda and Boyce, 1993) is used to represent the soft rubbery matrix, while the stiff

particle phase is represented by a Neo-Hookean model (see Bergstrom, 2015). The Arruda–Boyce model is parameterized by the initial bulk modulus κ_0, the initial shear modulus μ_0, and λ_m, which is associated with the chain locking stretch. The strain energy density function is given by:

$$\boxed{\psi_{AB} = \mu \sum_{i=1}^{5} a_i \, \beta^{\,i-1} (\bar{I}_{C1}^{\,i} - 3^i) + \frac{\kappa_0}{2}\left(\frac{J^2-1}{2} - \ln J\right)} \quad \text{where}$$

$$\mu = \mu_0 \left(1 + \frac{3}{5\lambda_m^2} + \frac{99}{175\lambda_m^4} + \frac{513}{875\lambda_m^6} + \frac{42039}{67375\lambda_m^8}\right)^{-1} \quad \text{and} \tag{27}$$

$$\beta = \frac{1}{\lambda_m^2}; \quad a_1 = \frac{1}{2}; \quad a_2 = \frac{1}{20}; \quad a_3 = \frac{11}{1050}; \quad a_4 = \frac{19}{7000}; \quad a_5 = \frac{519}{673750}$$

Here, $\bar{I}_{C1} = J^{-2/3} I_{C1}$ is the modified first invariant of the right Cauchy–Green deformation tensor $\boldsymbol{C}$, and J is the determinant of the deformation gradient tensor $\boldsymbol{F}$. The Neo-Hookean model is parameterized by the initial shear modulus μ_0 and the initial bulk modulus κ_0:

$$\psi_{NH} = \frac{\mu_0}{2}\left(\bar{I}_{C1} - 3\right) + \frac{\kappa_0}{2}(J-1)^2 \tag{28}$$

Note that the initial bulk modulus is related to the initial Poisson's ratio and shear modulus μ_0 as $\kappa_0 = \frac{2\mu_0(1+\nu_0)}{3(1-2\nu_0)}$. The material properties considered for the matrix phase are $\mu_0^{\text{mat}} = 3.098\times10^{-1}$ MPa, $\lambda_m^{\text{mat}} = 5.083$ to closely resemble natural rubber (Treloar, 1944; Bergstrom, 2015) with $\nu_0^{\text{mat}} = 0.45$. The material properties for the particle phase chosen such that its Young's modulus is 100 times stiffer than that of the matrix, with a Poisson's ratio $\nu_0^{\text{inc}} = 0.3$, resulting in $\mu_0^{\text{inc}} = 3.455\times10^{1}$ MPa. Prior to demonstrating the multiscale optimization results, we first report the performance of the consistent ML models against classical single-scale phenomenological models. In order to make a meaningful comparison, we choose the soft rubbery material for the single-scale assessment, using the soft matrix material parameters presented above. For the sake of brevity, we present the results under 2D plane strain considerations in this section, while a 3D proof of concept example is presented in Appendix D.3

5.1. Design of experiments for generating dataset for model calibration

Two datasets are generated to calibrate two distinct models: dataset $\mathcal{D}^s$ to train the model $\mathcal{M}^s$ for single-scale TO evaluations, and dataset $\mathcal{D}^m$ to train the microstructure-dependent model $\mathcal{M}^m$ for two-scale TO evaluations. The dataset $\mathcal{D}^s$ consists of input–output tuples of the Green–Lagrange strain $\boldsymbol{E}$ and the corresponding second Piola–Kirchhoff stress $\boldsymbol{S}$, with the ith data sample corresponding to $(\boldsymbol{E}, \boldsymbol{S})^i$. The dataset $\mathcal{D}^m$ for the model $\mathcal{M}^m$ for the two-scale TO has an additional microstructural descriptor $\mathrm{d_m}$, which here is the volume fraction of the particle phase in the RVE, α — making the ith data sample $((\alpha, \boldsymbol{E}), \boldsymbol{S})^i$. The strain states for data generation are obtained through Sobol sampling of the principal stretch space spanned by $\lambda_d \in [0.75, 1.75]$ and random sampling of orthonormal principal directions $\mathbf{N}_d$ for $d \in (1, \ldots, N_D)$ where N_D is the spatial dimensionality of the problem. The right stretch tensor is then obtained as $\boldsymbol{U} = \sum \lambda_d \mathbf{N}_d \otimes \mathbf{N}_d$, from which the Green–Lagrange strain tensor is computed as $\boldsymbol{E} = \frac{1}{2}(\boldsymbol{U}^T\boldsymbol{U} - \boldsymbol{I})$. This ensures that the strain sampling is space filling in the principal stretch space and non-informative sample points are limited. Under plane strain conditions, $N_D = 2$, and λ_3 is set to unity. The target second Piola–Kirchhoff stresses for the datasets $\mathcal{D}^s$ comprised of $2^{12} = 4096$ sample strain states are obtained by a direct evaluation of the second Piola–Kirchhoff stress tensor $\boldsymbol{S} = \partial_{\boldsymbol{E}}\psi_{AB}$ at the sampled Green–Lagrange strain states. In order to generate the second Piola–Kirchhoff stresses in the dataset $\mathcal{D}^m$ for the microstructure-dependent model, the data-driven design and analysis framework developed by Bessa et al. (2017), Yi and Bessa (2023), van der Schelling et al. (2024) is employed to perform RVE simulations with periodic boundary conditions in the commercial FE software ABAQUS (Systèmes, 2021), including the subsequent volumetric averaging of the stress measure. The microstructural geometry is fully described by the volume fraction α of the particle phase ($0.1 \leq \alpha \leq 0.5$), which is generated by randomly distributing circular particles of radius r_μ in a square RVE domain of side length L_{RVE}. Following the approach discussed by Bessa et al. (2017), the macroscopic deformation gradient tensor $\boldsymbol{F}$ necessary for applying the RVE boundary conditions is obtained in its rotational invariant form:

$$\boldsymbol{F} = \boldsymbol{U} = \boldsymbol{C}^{1/2} = (2\boldsymbol{E} + \mathbf{1})^{1/2} \tag{29}$$

For the considered materials and particle radius ($r_\mu = 0.15\,\text{mm}$), a 4 mm RVE side length ($L_{\text{RVE}} = 4.0\,\text{mm}$) was found to be sufficiently large to achieve a statistically representative response for volume fraction, α. The macroscopic second Piola Kirchhoff stresses are obtained corresponding to 2^{10} macroscopic Green–Lagrange strain states for representative realizations of the microstructure for each volume fraction $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$, totaling to 5×2^{10} data samples. Upon eliminating data samples corresponding to the non-converged RVE simulations, the final dataset $\mathcal{D}^m$ consists of 4787 data samples.

Table 1
Model and training hyperparameters.

Hyperparameter	$\mathcal{M}^s$	$\mathcal{M}^m$
No. hidden layers	2	2
No. neurons per layer	8	16
Batch size	128	128
Initial learning rate	1×10^{-3}	5×10^{-3}
Exponential decay rate	0.5	0.5
Decay transition epoch interval	3000	3000
Max epochs	15 000	15 000
Early stopping patience	5000	5000

Table 2
Performance metrics of the trained models.

Model	Metric	$\mathcal{D}^s_{\text{train}}$	$\mathcal{D}^s_{\text{val}}$	$\mathcal{D}^s_{\text{test}}$
$\mathcal{M}^s$	$1 - R^2$	3.41×10^{-8}	3.66×10^{-8}	3.22×10^{-8}
	RMSE	1.14×10^{-4}	1.14×10^{-4}	1.14×10^{-4}
	MAE	8.47×10^{-5}	8.51×10^{-5}	8.52×10^{-5}

Model	Metric	$\mathcal{D}^m_{\text{train}}$	$\mathcal{D}^m_{\text{val}}$	$\mathcal{D}^m_{\text{test}}$
$\mathcal{M}^m$	$1 - R^2$	1.13×10^{-4}	1.20×10^{-4}	1.53×10^{-4}
	RMSE	1.04×10^{-2}	1.11×10^{-2}	1.19×10^{-2}
	MAE	4.23×10^{-3}	4.32×10^{-3}	4.41×10^{-3}

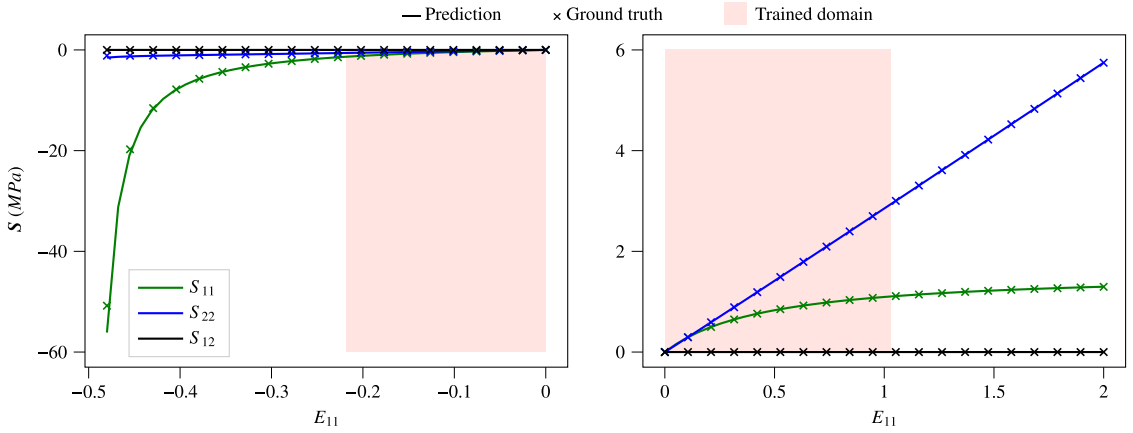


Fig. 3. Stress response of the single-scale consistent ML model compared to the ground truth phenomenological model for a uniaxial strain loading case. The plot is split into two subplots to better visualize the compressive and tensile loading ranges.

5.2. Model training

The strategy for training the ML models is standardized and described in this section. The dataset $\mathcal{D}$ associated with a model $\mathcal{M}$ is split into training, validation, and test datasets in the ratio 60 : 20 : 20. The mean squared error loss function between the predicted second Piola–Kirchhoff stress $\boldsymbol{S}$ and the target second Piola–Kirchhoff stress $\boldsymbol{S}^*$, given by:

$$\mathcal{L}_{\square} = \frac{1}{|\mathcal{D}_{\square}|} \sum_{|\mathcal{D}_{\square}|} \left\| \boldsymbol{S} - \boldsymbol{S}^* \right\|^2, \tag{30}$$

is used as the objective function in the training process, where $\square$ is a placeholder for the dataset split under consideration, and $|\mathcal{D}_{\square}|$ is the number of samples in the dataset split and $\|\cdot\|$ denotes the Euclidean norm. The model training is performed using the Adam optimizer (Kingma and Ba, 2014) with decoupled weight decay regularization (Loshchilov and Hutter, 2019), together with an exponential decaying learning rate scheduler by minimizing the loss function $\mathcal{L}_{\text{train}}$ on the training dataset $\mathcal{D}_{\text{train}}$. The validation loss $\mathcal{L}_{\text{val}}$ on the validation dataset $\mathcal{D}_{\text{val}}$ is monitored during the training process to prevent overfitting through early stopping, as well as to tune the hyperparameters of the model. The hyperparameters of the models and the training process, tuned through a simple grid search method, are summarized in Table 1. The predictive capability of the tuned model is then evaluated on the test dataset $\mathcal{D}_{\text{test}}$. Table 2 summarizes the performance of the trained models on the training, validation, and test datasets in terms of root mean squared error (RMSE), mean absolute error (MAE), and coefficient of determination (R^2) metrics.[2]

[2] The machine learning workflow is implemented in JAX (Bradbury et al., 2018) and realized through the Equinox library (Kidger and Garcia, 2021), and leverages the gradient processing and optimization library Optax (DeepMind et al., 2020). All the ML trainings are performed on a standard desktop with NVIDIA RTX 3050 Ti GPU.

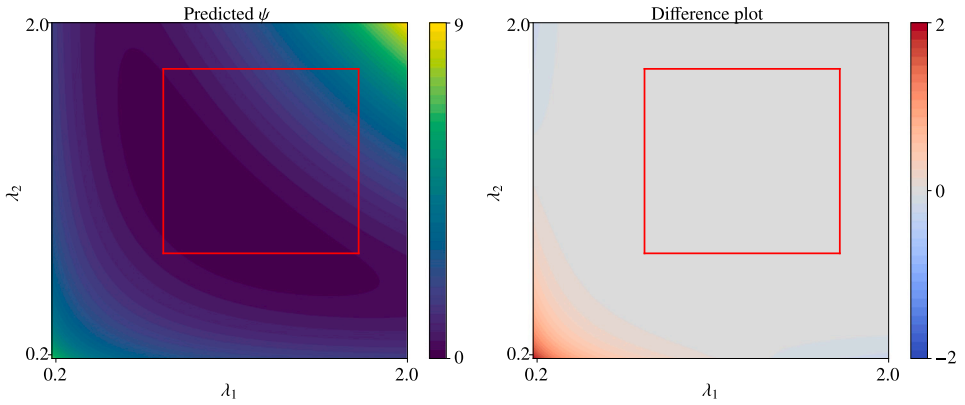


Fig. 4. Strain energy density contour plot of the single-scale consistent ML model and the corresponding difference plot to the ground truth phenomenological model. The region within the red box corresponds to the trained domain.

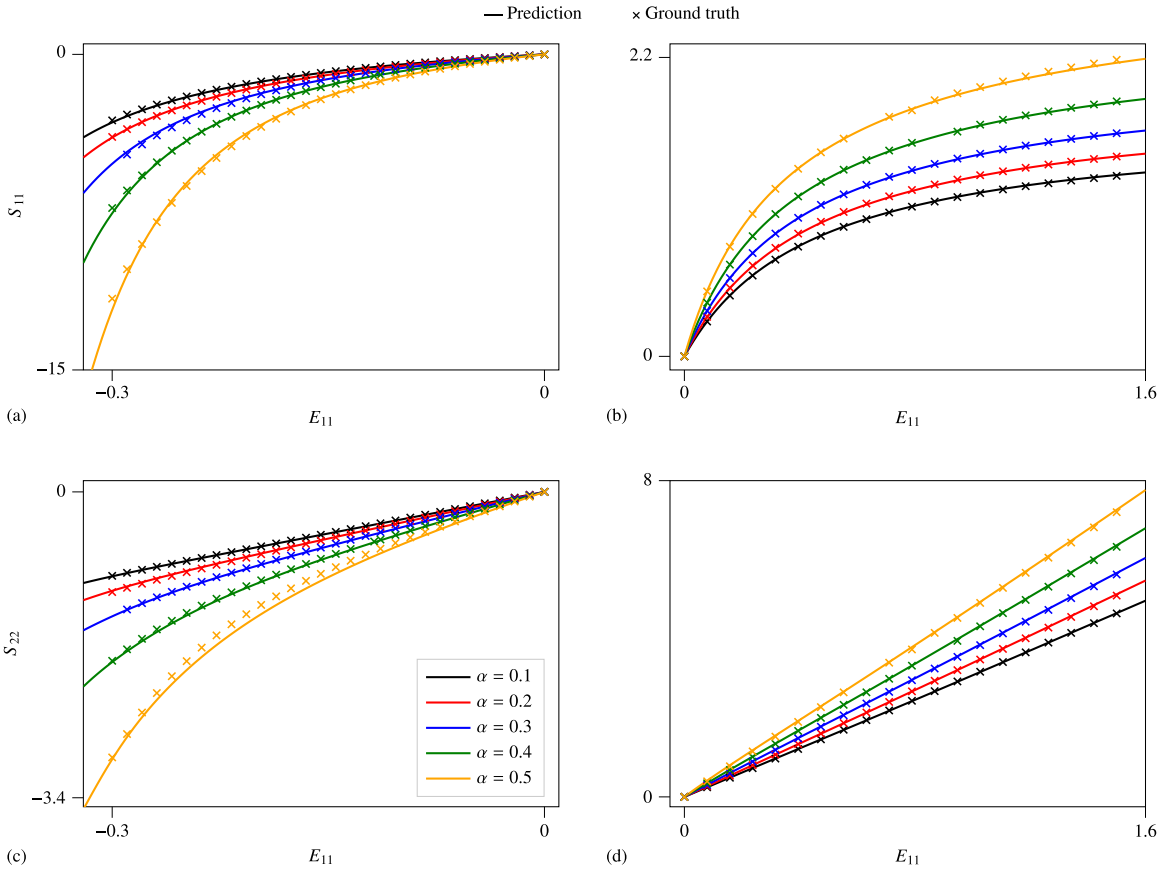


Fig. 5. Stress response of the microstructure-dependent consistent ML model compared to the ground truth obtained through RVE simulations for a uniaxial strain loading scenario for (left) compressive loading, and (right) tension loading range. The top and bottom plots show the stress components S_{11} and S_{22} separately.

5.3. Evaluation of model effectiveness

We begin by evaluating the effectiveness of the consistent ML model in representing classical single-scale phenomenological models. As mentioned earlier, an Arruda Boyce hyperelastic model (27) corresponding to the soft rubbery matrix phase is considered the ground truth for evaluating the approximation capability of the consistent ML material models. Fig. 3 illustrates the constitutive response of the trained model $\mathcal{M}^s$ compared to the ground truth phenomenological model for uniaxial loading (i.e., variation of

Table 3
Topology optimization problem hyperparameters.

Hyperparameter	Symbol	Value
Filter radius*	r_ρ	4.0
SIMP penalty parameter (initial value)	p	1.0
SIMP ersatz parameter	ϵ	10^{-5}
Projection strength (initial value)	β_ρ	1.0
Projection threshold	η_ρ	0.5
Energy interpolation transition strength	γ_ρ	32.0
Energy interpolation transition threshold	η_γ	0.01
Linearized elastic energy Young's modulus	E_{lin}	8.984×10^{-1}
Linearized elastic energy Poisson's ratio	ν_{lin}	0.45
Forward problem number of time steps	N	8
Relative tolerance for Newton convergence	tol_{NR}	10^{-6}
MMA move limit		0.15

∗ Filter radius for *cantilever beam* problem is 6.0.

E_{11}, with all other strain components set to zero). The figure highlights how closely the trained model $\mathcal{M}^s$ approximates the ground truth phenomenological model, which is in agreement with the performance metrics provided in Table 2. Importantly, the model shows impressive predictive performance beyond the training range. To visualize the predictive performance of the model more comprehensively, the strain energy density is compared to the ground truth phenomenological model in Fig. 4. From the plot, visualized in the principal stretch space, we see that the trained model $\mathcal{M}^s$ exhibits high interpolation accuracy, with acceptable extrapolation performance attributed largely to the relevant restrictions in the consistent ML model architecture. The extrapolation quality is observed to deteriorate as the principal strain state $\{\lambda_1, \lambda_2\} \to \mathbf{0}$, which corresponds to the asymptote as $J \to 0$, yet the model is able to predict the strain energy density response in the vicinity of the singularity with reasonable accuracy.

Next, we assess the ability of the consistent ML model to capture the microstructure-dependent response variations encapsulated within the two-phase composite material dataset $\mathcal{D}^m$. We are now specifically interested in evaluating how well the model $\mathcal{M}^m$ learns the mapping between the particle volume fraction α of the microstructure and the homogenized material response. The stress response corresponding to uniaxial loading for particle volume fractions $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$ in both compression and tension is shown in Fig. 5. Note that this data is not present in the training dataset. Naturally, similar to the single-scale case, the microstructure-dependent model $\mathcal{M}^m$ preserves the hyperelasticity conditions. More importantly, the model captures the dependence of α on the stress response very well, suggesting that the assumption of convexity with respect to α in the model architecture was appropriate.

5.4. Single-scale topology optimization

We now verify the performance of the consistent ML model in the context of topology optimization via direct comparison against results obtained using the "classical" or phenomenological model. The intent of this exercise is to provide confidence in the trained ML model for use in topology optimization by demonstrating nearly identical designs in a setting with directly comparable conventional results. Note that the update of the design variables is performed using the method of moving asymptotes (MMA) (Svanberg, 1987).

In the main text, we present the results for two examples — the T-bracket and a cantilever beam. In Appendix D.1, we also provide the results for a portal frame example, and a three-dimensional fixed–fixed beam example is presented in Appendix D.3. The results in these appendices are consistent with the examples discussed in the main text.

We define each design domain with reference to a base dimension, $L = 100\,\text{mm}$, and the parameters provided in Table 3 are used unless otherwise specified. A continuation scheme is employed on both the SIMP penalty exponent and the projection strength parameter. The SIMP penalty exponent is initialized to $p = 1$ and increased in increments of $\Delta p = 1$ each continuation step up to a maximum value of $p = 4$. In subsequent continuation steps, the projection strength parameter is doubled up to a maximum value computed according to da Silva et al. (2019), from an initial value of $\beta_p = 1$. The next continuation step begins when either the percent difference in the objective function over the previous five iterations has fallen below a tolerance of 0.1 or a maximum number of 50 iterations is reached. A minimum of 20 optimization iterations is also enforced in each step.

5.4.1. Single-scale topology optimization: T-bracket example

The T-bracket design domain and boundary conditions are illustrated in Fig. 6a, corresponding to an average downward displacement of $c_N = 0.3L$ applied at the center of the right-most edge. The domain is discretized into 35,400 four-node quadrilateral elements within which the displacement field is interpolated using standard bilinear shape functions. In this example, we employ a volume fraction upper bound of $g_{\max} = 0.5$. The convergence history is provided in Fig. 7, where the continuation updates on the SIMP penalty exponent p and projection strength parameter β_ρ are apparent through the sharp changes in the objective function values. Apart from the expected jumps in the objective function values at optimization iterations corresponding to the continuation steps, the convergence history is smooth with few oscillations. The visualized designs in Fig. 7 show the progression of the topology as the optimization process proceeds using the ML material model. The final design closely resembles the design obtained using the ground truth phenomenological model, as evident from the difference plot provided in Fig. 6b and the similar convergence history shown in Fig. 7. A value of $\delta_\rho = 1$ indicates material addition, and $\delta_\rho = -1$ indicates material removal from the design obtained using the ground truth phenomenological model. The deformed configuration of the final design is shown in Fig. 6c.

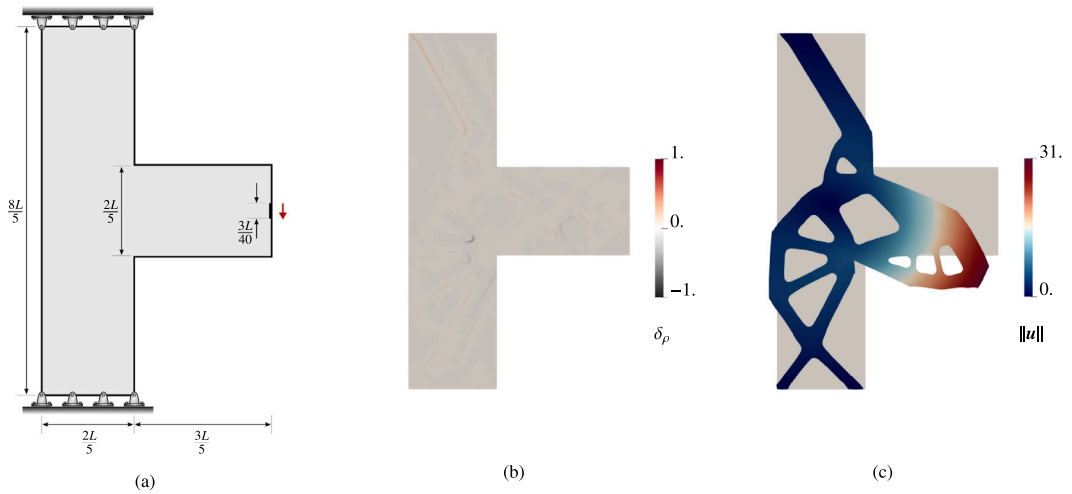


Fig. 6. (a) T-bracket boundary value problem. (b) Difference plot between the design obtained using the ML model and the design obtained using the ground truth phenomenological model, where $\delta_\rho = 1$ indicates material addition, and $\delta_\rho = -1$ indicates material removal. (c) Deformed configuration of the design obtained using the ML model.

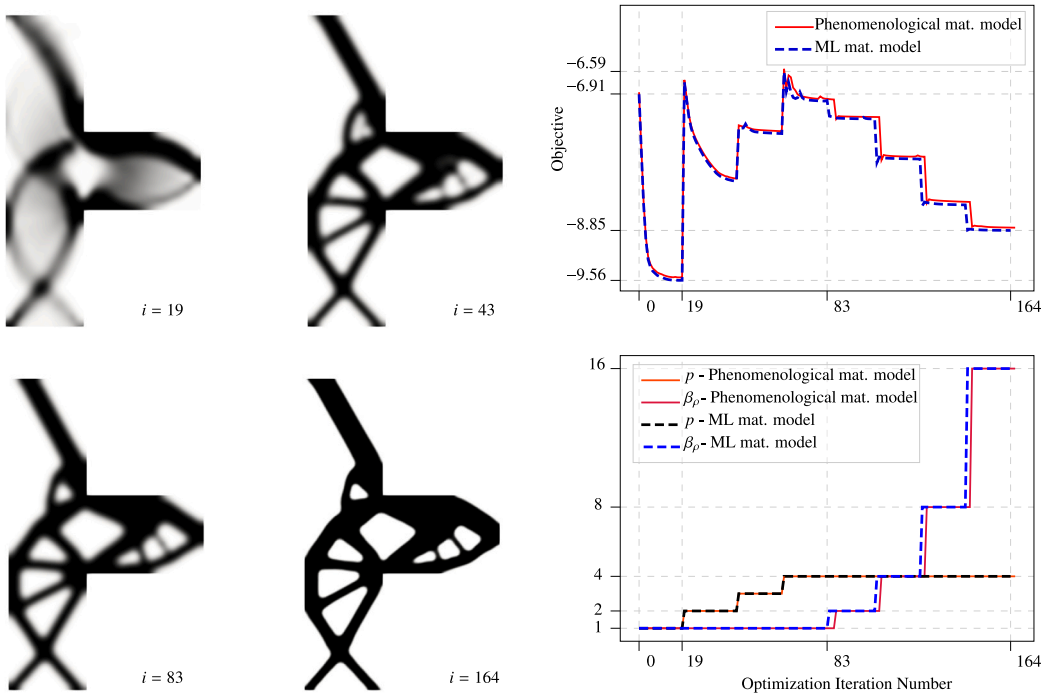


Fig. 7. Convergence history for the T-bracket example. Illustrated designs correspond to the optimization formulation using the ML model at iterations with selected continuation steps.

5.4.2. *Single-scale topology optimization: Cantilever beam example*

We now consider a cantilever beam with the design domain, as shown in Fig. 8(a). The cantilever beam is fixed along the left edge, and a vertically downward average displacement $c_N = 0.6L$ is applied near the center of the right edge. The domain is discretized using 4-node quadrilateral elements with a uniform mesh of edge length 0.5 mm, adding up to a total of 40000 elements. This example is particularly interesting as the cantilever beam problem subject to large deformation is a challenging case for hyperelastic materials due to the presence of buckling instabilities. We remark that for this problem, we considered hyperparameters, viz. material volume fraction constraint $g_{\max} = 0.6$ and the filter radius $r_\rho = 6.0$, that lead to final designs not susceptible to large buckling instabilities.

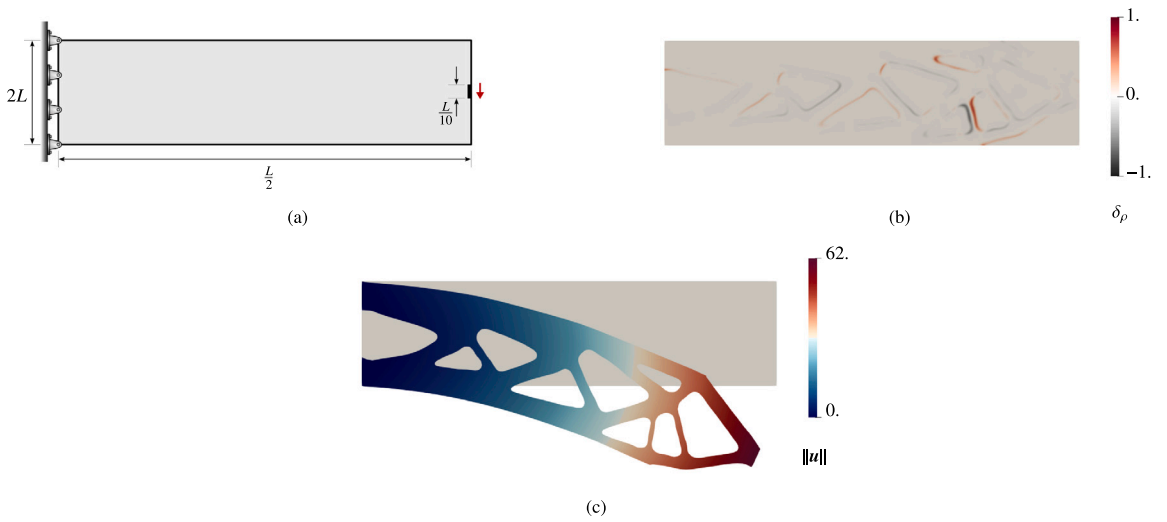


Fig. 8. (a) Cantilever boundary value problem. (b) Difference plot of the design obtained using the ML model to the design obtained using the ground truth phenomenological model, where $\delta_\rho = 1$ indicates material addition, and $\delta_\rho = -1$ indicates material removal. (c) Deformation of the design obtained using the ML model.

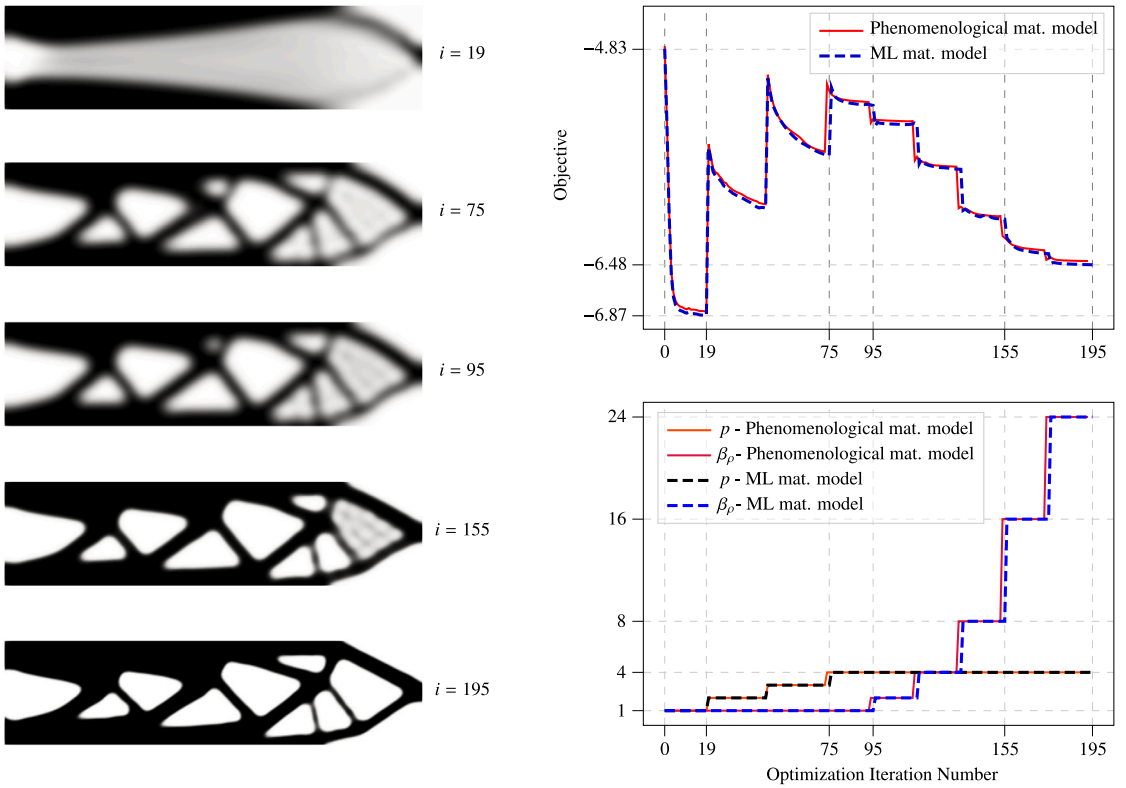


Fig. 9. Convergence history for cantilever beam example. Illustrated designs correspond to the optimization formulation using the ML model at iterations with selected continuation steps.

Although not the focus of this article, future work could include buckling criteria explicitly as part of the optimization formulation to systematically prevent such behavior.

Nonetheless, minor instabilities occur during the optimization process, which has a visible effect on the convergence history as well as the intermediate designs, highlighting the challenging nature of the cantilever beam problem, as shown in Fig. 9. We observe

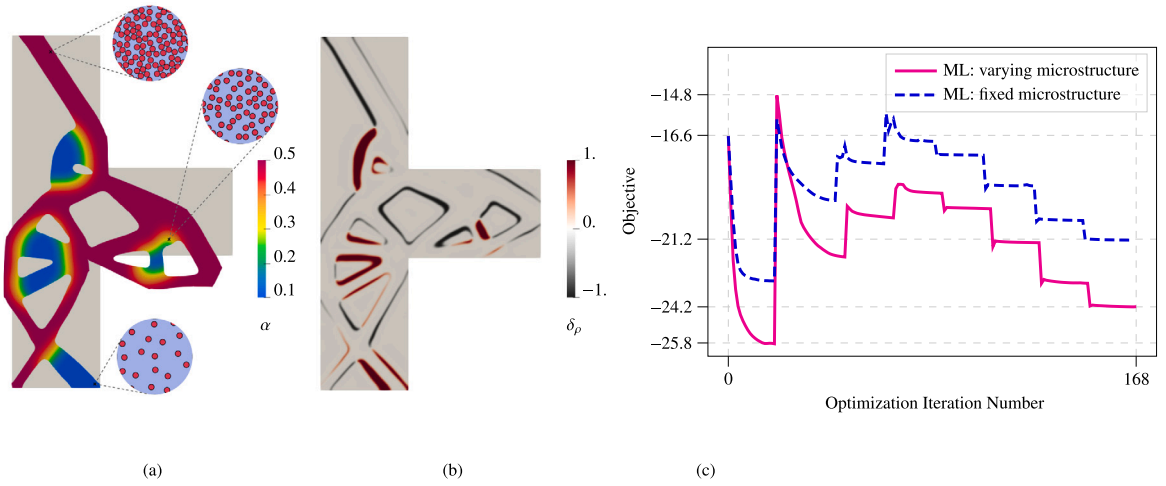


Fig. 10. (a) Deformed configuration of the final design obtained using the ML model with varying microstructure (b) Difference plot of the design obtained using the ML model with varying microstructure to the design obtained using the ML model with fixed microstructure. (c) Convergence history for the T-bracket TO problem with varying microstructure versus fixed microstructure.

that while the continuation updates on the SIMP penalty exponent p produces sharp changes in the objective function values, the projection strength parameter β_ρ updates, especially in the later stages of the optimization process, do not lead to significantly abrupt changes in the objective. This is due to the interference of minor instabilities altering the deformation state of the design, resulting in the prolonged presence of gray regions in the design, which is only slowly resolved with the β_ρ updates. See the progression of the topology as the optimization process proceeds using the ML material model in Fig. 9 for reference. Interestingly, this leads the convergence histories to differ very slightly, although the final objective values remain close to one another. As a result, we see that the final design obtained using the ML model resembles the design obtained using the ground truth phenomenological model. Nonetheless, there are subtle differences around the boundaries of the two designs, as shown in the difference plot in Fig. 8(b). The deformed configuration of the final design obtained using the ML model is depicted in Fig. 8(c).

5.5. Multi-scale topology optimization

We now demonstrate the ability of the framework to perform simultaneous design optimization at two spatial scales. The optimization problem considered here involves maximizing the external work (19) subject to material volume fraction constraints applied separately to the matrix and particle phases, as specified in (21). Note that in this section, we no longer have the benefit of a ground truth phenomenological model with which we can compare our results. For a fair evaluation of any potential benefits of the two-scale optimization, i.e., optimizing both the pseudo density field ρ and the particle phase volume fraction α, we compare the results with a baseline case in which the microstructure is fixed (i.e., $\alpha = \alpha_0$ is fixed) while also satisfying the same material volume fraction constraints. In both cases, we employ the microstructure-dependent ML model $\mathcal{M}^m$ to represent the hyperelastic composite. The value of α_0 for the baseline case in each example is set based on the side constraint bounds on the total volume fraction of the particle phase ($g^{\text{inc}}_{\max}$) and matrix phase ($g^{\text{mat}}_{\max}$), i.e., $\alpha_0 = g^{\text{inc}}_{\max}/(g^{\text{inc}}_{\max} + g^{\text{mat}}_{\max})$. Similarly, the total material volume fraction for the baseline case is set to $g_{\max,0} = g^{\text{inc}}_{\max} + g^{\text{mat}}_{\max}$.

Two examples are considered in this section (the T-bracket and the cantilever beam), which can be compared with the single-scale topology optimization results. The geometry, finite element discretization, and boundary conditions for these examples are discussed in Section 5.4. We remark that Appendix D.1 contains an additional multiscale TO example for the portal frame boundary value problem.

5.5.1. Multi-scale topology optimization: T-bracket example

The T-bracket example is subject to total volume fraction side constraint bounds of $g^{\text{mat}}_{\max} = 0.3$ and $g^{\text{inc}}_{\max} = 0.2$. Therefore, the baseline case with fixed microstructure has a fixed particle volume fraction $\alpha_0 = 0.4$ and a total volume fraction constraint of $g_{\max,0} = 0.5$. The convergence history of the objective function for this example is shown in Fig. 10c. Consistent with intuition, the objective value for the case in which microstructural variations are allowed is indeed lower. From the final design with varying microstructure (Fig. 10a) and the difference plot with the baseline case with fixed microstructure (Fig. 10b), we observe that the number of macroscopic voids remains unchanged, but their shapes change. We also notice that the material addition is occurs close to regions chosen to have lower particle volume fraction α by the optimizer, and the material removal is made in regions dominated by higher particle volume fractions.

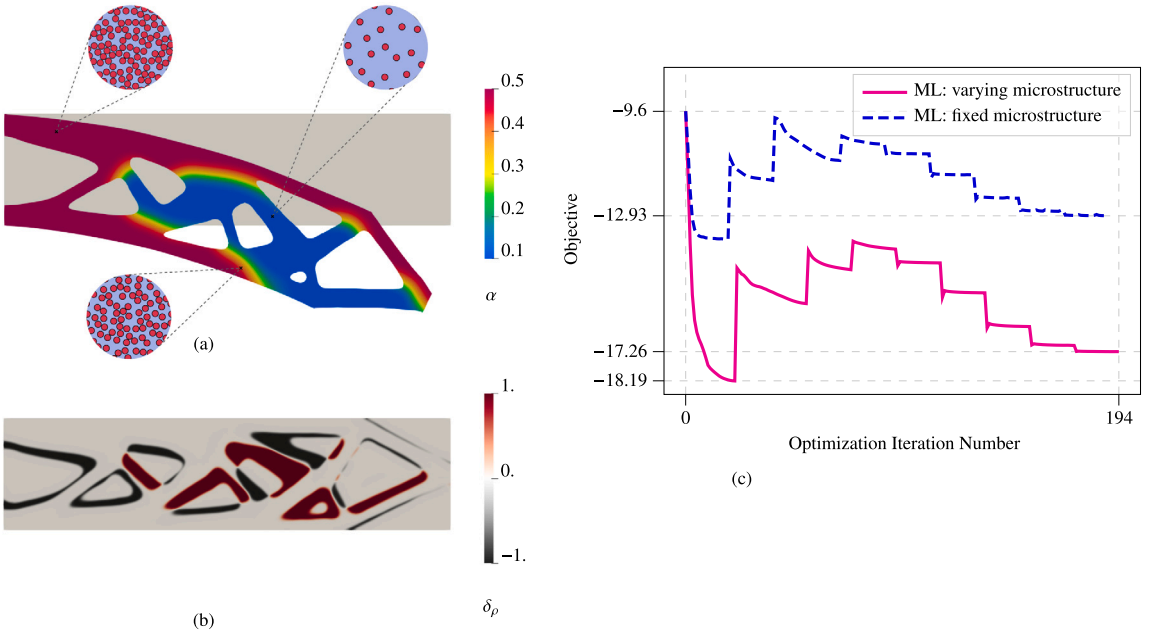


Fig. 11. (a) Deformed configuration of the final design obtained using the ML model with varying microstructure (b) Difference plot of the design obtained with varying microstructure to the design obtained with fixed microstructure. (c) Convergence history for the cantilever TO problem with varying microstructure vs. fixed microstructure.

Table 4
Performance comparison of designs obtained using varying and fixed microstructure.

Problem	Design	External work		External load factor	
		(mJ/mm)	Improvement	(N/mm)	Improvement
T-bracket	Fixed microstructure	21.244	–	1.4808	–
	Varying microstructure	24.179	13.8%	1.6892	14.1%
Cantilever	Fixed microstructure	12.895	–	0.43766	–
	Varying microstructure	17.263	33.9%	0.60115	37.4%
Portal frame	Fixed microstructure	9.1433	–	0.84417	–
	Varying microstructure	10.522	15.1%	0.97354	15.3%

5.5.2. *Multi-scale topology optimization: Cantilever beam example*

The final example considered here corresponds to the cantilever beam. Consistent with the example in Section 5.4.2, the volume fraction upper bounds for the matrix and particle phases are set to $g^{\text{mat}}_{\text{max}} = 0.4$ and $g^{\text{inc}}_{\text{max}} = 0.2$, corresponding to a total volume fraction upper bound of $g_{\text{max},0} = 0.6$ for the baseline case with fixed microstructure. The fixed microstructure particle volume fraction that satisfies the material volume fraction constraint is then set to $\alpha_0 = \frac{1}{3}$. It can be seen from the final design with varying microstructure (Fig. 11a) and the difference plot with the baseline case (Fig. 11b) that the former achieves an improved objective value by adjusting the internal material-void distribution in addition to the variation of the particle volume fraction α. Again, the material addition predominantly occurs close to regions of lower particle volume fraction α, whereas material removal happens close to regions with higher particle volume fraction. Similar to the previous examples, we observe a lower objective value when allowing the microstructure to vary, demonstrating the added benefit of the proposed approach.

5.5.3. *Multi-scale topology optimization: Discussion*

The results from the above examples demonstrate the potential of the proposed framework to obtain improved designs by exploring the design space at multiple spatial scales. In Table 4, we provide a comparison of the external work and the external load factor for the cases with varying and fixed microstructure throughout the macroscopic design. The reader is referred to Appendix D.2 for additional information regarding the portal frame example.

We observe that the designs obtained using the varying microstructure have a significantly better objective value (external work) and, thereby, a higher external load factor compared to the designs obtained using the fixed microstructure. More importantly, these results demonstrate for the first time the large potential improvements that may be obtained by exploring multiscale designs for structures exhibiting large nonlinear responses. If improvements on the order of 15% to 35% are possible by considering simple microstructures controlled by a single descriptor (particle volume fraction), future improvements will surely be more significant.

6. Conclusions

A framework is presented for designing multiscale heterogeneous structures with spatially varying hyperelastic microstructures. Central to this work is the ML material model, which captures the homogenized response of the microstructure as a differentiable function of microstructural descriptors while adhering to important physical principles.

Our results show that ensuring polyconvexity, among other relevant restrictions, not only results in consistent ML models of high accuracy but also produces reliable final designs in the topology optimization process. An equally important benefit of adhering to these physical principles is the guarantee of the existence of solutions in the forward problem. The differentiable mapping of the microstructural descriptors to the homogenized response enables the computation of the tangent stiffness as well as the sensitivities of the relevant objective/constraint functions with respect to the design variables, facilitating efficient, simultaneous design optimization at two spatial scales. The presented approach provides an effective method for designing future functionally graded structures and materials with impactful microstructural details. Our findings indicate that two-scale optimization can significantly improve performance compared to the baseline case with spatially uniform microstructure. The examples presented produce optimized designs that align with our intuition and demonstrate that expanding the design space to multiple scales allows the optimizer to achieve higher design performance. Although the focus of this work has been to build a proof of concept computational framework, we hope this may be used by designers and manufacturers as a tool to explore new design spaces for structures with spatially varying microstructures.

The proposed framework is general and can be extended to include additional microstructural descriptors, such as the orientation of microstructural phases to capture anisotropy in the homogenized response. In this future work, we intend to explore this through partial input convex neural networks (Amos et al., 2017). An extension of the consistent ML to handle path and history-dependent material for topology optimization is also envisioned, wherein we also anticipate a potential acceleration of the optimization process due to the absence of Newton–Raphson iterations (Mozaffar et al., 2019). While manufacturability and variability in loading conditions have not been in the scope of this work, the literature on additive manufacturing shows immense potential for the realization of such designs (e.g., Truby and Lewis, 2016; Xu et al., 2021; Zhu et al., 2024). We hope to leverage our past experience incorporating manufacturability constraints in TO (Vatanabe et al., 2016) to explore this exciting field in the future.

CRediT authorship contribution statement

Harikrishnan Vijayakumaran: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Jonathan B. Russ:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Investigation, Formal analysis, Conceptualization. **Glaucio H. Paulino:** Writing – review & editing, Supervision, Formal analysis, Conceptualization. **Miguel A. Bessa:** Writing – review & editing, Supervision, Funding acquisition, Formal analysis, Conceptualization.

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

H.V. and M.A.B. gratefully acknowledge the financial support from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement 'LIGHTEN – Ultralight membrane structures toward a sustainable environment' H2020-MSCA-ITN-2020-LIGHTEN-956547. G.H.P. acknowledges support from the National Science Foundation grants 2105811 and 2323415.

Appendix A. Modification to ICNN for modeling hyperelasticity

Here, we recall three fundamental facts about convexity from Boyd and Vandenberghe (2004) that are useful within the context of input convex neural networks.

1. If $f_i : \boldsymbol{R}^n \to \boldsymbol{R}$ are convex functions and $c_i \geq 0$, then the function $g : \boldsymbol{R}^n \to \boldsymbol{R}$ defined as:
$$g(\boldsymbol{y}) = \sum_{i=1}^{m} c_i f_i(\boldsymbol{y}), \quad \mathbf{dom}\, g = \bigcap_{i=1}^{m} \mathbf{dom}\, f_i \tag{A.1}$$
is convex.
2. The composition of a convex function with an affine mapping preserves convexity. If the function $f : \boldsymbol{R}^n \to \boldsymbol{R}$ is a convex function, then a function $g : \boldsymbol{R}^m \to \boldsymbol{R}$ defined as
$$g(\boldsymbol{y}) = f(\boldsymbol{W}\boldsymbol{y} + \boldsymbol{b}), \quad \mathbf{dom}\, g = \{\boldsymbol{y} \mid \boldsymbol{W}\boldsymbol{y} + \boldsymbol{b} \in \mathbf{dom}\, f\} \tag{A.2}$$
is also convex, with $\boldsymbol{W} \in \boldsymbol{R}^{n\times m}$, $\boldsymbol{b} \in \boldsymbol{R}^n$.

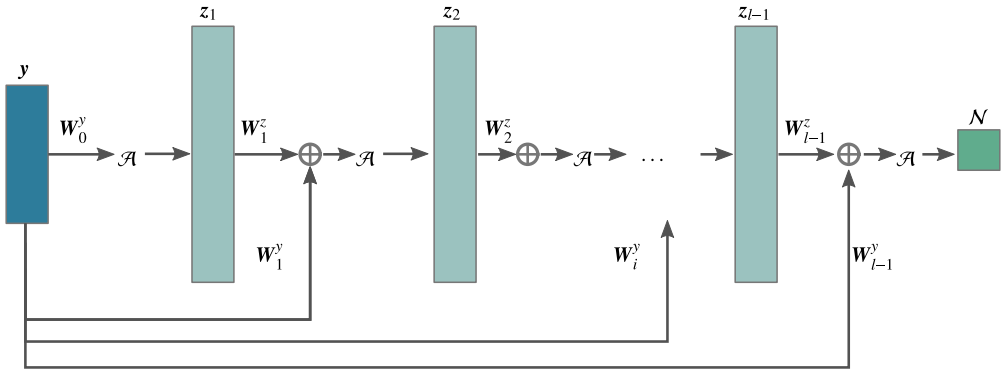


Fig. A.12. Input neural network architecture.

3. The composition of a convex function with a convex non-decreasing function is convex. If $h : \boldsymbol{R}^m \to \boldsymbol{R}^n$ is a convex function and $f : \boldsymbol{R}^n \to \boldsymbol{R}$ is a convex non-decreasing function, then the composition $g : \boldsymbol{R}^m \to \boldsymbol{R}$ defined as:

$$g(\boldsymbol{y}) = f(h(\boldsymbol{y})), \quad \mathbf{dom}\, g = \{\boldsymbol{y} \in \mathbf{dom}\, h \mid h(\boldsymbol{y}) \in \mathbf{dom}\, f\} \tag{A.3}$$

is convex.

The ICNN proposed by Amos et al. (2017) as shown in Fig. A.12 uses (A.2) together with (A.3) in the first layer to achieve convexity. In each of the subsequent layers (A.3) is used together with (A.1) for each core sub-layer, and (A.2) for each passthrough sub-layer. Its mathematical definition for general vectorial input is defined as $f(\boldsymbol{y};\boldsymbol{\theta})$:

$$\begin{aligned} &\boldsymbol{z}_{i+1} = \mathcal{A}_i(\boldsymbol{W}_i^{(z)}\boldsymbol{z}_i + \boldsymbol{W}_i^{(y)}\boldsymbol{y} + \boldsymbol{b}_i) \quad \text{for} \quad i = 0, \dots, l-1 \\ &f(\boldsymbol{y},\boldsymbol{\theta}) = \boldsymbol{z}_l \end{aligned} \tag{A.4}$$

where $\boldsymbol{z}_i$ denotes the layers (with $\boldsymbol{z}_0, \boldsymbol{W}_0^z \equiv \boldsymbol{0}$) to which convex and non-decreasing element-wise activation functions $\mathcal{A}_i(\cdot)$ are applied. The learnable parameters are $\boldsymbol{\theta} = \{\boldsymbol{W}_{0:l-1}^{(z)}, \boldsymbol{W}_{1:l-1}^{(y)}, \boldsymbol{b}_{0:L-1}\}$ where $\boldsymbol{W}_{0:l-1}^{(z)}$ are non-negative. For ensuring C2 continuity, the activation function $\mathcal{A}_i(\cdot)$ is chosen to be the softplus function and the positive weights are enforced by wrapping the weights $\boldsymbol{W}_i^{(z)}$ in with a squared softplus function, following As'ad et al. (2022).

As noted in Section 2.1, one basic requirement in hyperelastic material modeling is that the strain energy density function ψ is polyconvex. Namely,

$$\tilde{\psi}(\boldsymbol{F}) = \mathcal{G}(\boldsymbol{F}, \operatorname{cof}\boldsymbol{F}, \det\boldsymbol{F}) \tag{A.5}$$

where $\mathcal{G}(\cdot)$ is a convex function in $(\boldsymbol{F}, \operatorname{cof}\boldsymbol{F}, \det\boldsymbol{F})$. In conventional constitutive modeling, the strain energy density function is often expressed as a (convex) function of the invariants of an objective strain measure. The classic example of this corresponds to the invariants $\boldsymbol{\iota} = \{I_{C1}, I_{C2}, I_{C3}\}$ of the right Cauchy–Green tensor, $\boldsymbol{C} = \boldsymbol{F}^{\mathsf{T}}\boldsymbol{F}$, since they satisfy polyconvexity and have desirable symmetry features. However, such invariants cannot be used as input in the original ICNN implementation. The first layer and the passthrough layers involve an *unrestricted* transformation (A.2) with weights that can be positive or negative, which, when applied to the invariants, does not preserve convexity in the original parameters. Therefore, if it is desired to use the invariants, the treatment should be in accordance with the convex non-decreasing composition (A.3), which implies that the transformation (A.2) should be restricted to non-negative weights. However, if the invariant $J = \sqrt{I_{C3}} = \det\boldsymbol{F}$ is used instead of I_{C3}, there need not be any restrictions on the specific weights mapping J. In effect, when using invariants as input, the ICNN should be modified so that it results in a function that is convex and non-decreasing in the invariants.

A similar argument holds when using the principal stretches or any of its convex combinations as input, as shown in Appendix B. The only exception to this is the invariant J, since the strain energy density function need not be non-decreasing in J. The modified version of the ICNN per Chen et al. (2019), provides a straightforward implementation for handling convexity and ensuring that the output is non-decreasing in particular inputs. In this definition, an ICNN is expressed as a function, $f(\boldsymbol{y};\boldsymbol{\theta})$ defined by:

$$\begin{aligned} &\boldsymbol{z}_{i+1} = \mathcal{A}_i(\boldsymbol{W}_i^{(z)}\boldsymbol{z}_i + \boldsymbol{W}_i^{(\hat{y})}\hat{\boldsymbol{y}} + \boldsymbol{b}_i) \quad \text{for} \quad i = 0, \dots, l-1 \\ &f(\hat{\boldsymbol{y}},\boldsymbol{\theta}) = \boldsymbol{z}_l \end{aligned} \tag{A.6}$$

where $\hat{\boldsymbol{y}}$ represents the expanded input $\hat{\boldsymbol{y}} = [\boldsymbol{y}, -\boldsymbol{y}]^{\mathsf{T}}$, $\boldsymbol{z}_i$ denotes the layers (with $\boldsymbol{z}_0, \boldsymbol{W}_0^z \equiv \boldsymbol{0}$) to which convex and non-decreasing element-wise activation functions $\mathcal{A}_i(\cdot)$ are applied, and $\boldsymbol{\theta} = \{\boldsymbol{W}_{0:l-1}^{(z)}, \boldsymbol{W}_{1:l-1}^{(\hat{y})}, \boldsymbol{b}_{0:L-1}\}$ denotes the learnable parameters where $\boldsymbol{W}_{0:l-1}^{(z)}$ and $\boldsymbol{W}_{0:l-1}^{(\hat{y})}$ are non-negative. Since all layers, including the passthrough layers, have non-negative weights, if the expanded input is not used (i.e., $\hat{\boldsymbol{y}} \equiv \boldsymbol{y}$), the output of the ICNN, as defined by Chen et al. (2019), becomes convex and non-decreasing in the inputs. By *not* using the expanded inputs, except for J, the ICNN (A.6) can be used to model hyperelastic materials with strain energy density functions that are convex in J and convex non-decreasing in the remaining inputs.

Amongst the ICNN-based neural network models for hyperelasticity present in literature, Linden et al. (2023) and their subsequent works have addressed this aspect by having non-negative weights overall. However, the authors omit the skip connections presented in Chen et al. (2019).

Appendix B. Limitations of invariant-based ICNN models

When choosing a set of invariants as inputs, an ICNN should be convex, non-decreasing in those invariants (except for J, as ψ need not be non-decreasing in J) c.f. Appendix A. We theorize that such an ICNN would be limited in its ability to model an arbitrary polyconvex strain energy density function ψ.

Here, we limit the current discussion to isotropic cases where the conditions discussed in (6) hold. We aim to provide a simple counter-example to show that a strain energy density in accordance with (6) cannot always be represented by a composition of convex and non-decreasing functions of pre-chosen invariants (convex in J). Consider the Ogden model (Holzapfel, 2002), which has a strain energy density function given by:

$$\begin{aligned}\psi_O(\boldsymbol{F}) &= \psi_{\text{iso}}(\bar{\lambda}_1, \bar{\lambda}_2, \bar{\lambda}_3) + \psi_{\text{vol}}(J) \\ &= \left\{ \sum_{i=1}^{N} \frac{\mu_i}{a_i} (\bar{\lambda}_1^{a_i} + \bar{\lambda}_2^{a_i} + \bar{\lambda}_3^{a_i} - 3) \right\} + \frac{\kappa}{\beta^2} (\beta \ln J + J^{-\beta} - 1)\end{aligned} \tag{B.1}$$

We adopt $\beta = -2$ as in Simo and Miehe (1992). For the sake of simplicity, we consider the case of $N = 1$, $a_1 = a = 6$, and replace $\mu_1 = \mu$. Note that $\bar{\lambda}_i = J^{-1/3}\lambda_i$ and $J = \lambda_1\lambda_2\lambda_3$. Then we have:

$$\begin{aligned}\psi_O(\boldsymbol{F}) &= \frac{\mu}{6}(\bar{\lambda}_1^6 + \bar{\lambda}_2^6 + \bar{\lambda}_3^6 - 3) + \frac{\kappa}{4}(J^2 - 1 - 2\ln J) \\ &= \frac{\mu}{6}(J^{-2}(\lambda_1^6 + \lambda_2^6 + \lambda_3^6) - 3) + \frac{\kappa}{4}(J^2 - 1 - 2\ln J)\end{aligned} \tag{B.2}$$

This model is polyconvex by definition in (6), which can also be verified with a simple test. The strain energy density (B.2) should be convex, non-decreasing in $\{\lambda_1, \lambda_2, \lambda_3\}$ and convex in J. Let us obtain the derivatives of the strain energy density function (B.2) with respect to $\boldsymbol{\Lambda} = \{\lambda_1, \lambda_2, \lambda_3, J\}$:

$$\nabla_{\boldsymbol{\Lambda}}\psi_O = \begin{bmatrix} \mu J^{-2}\lambda_1^5 \\ \mu J^{-2}\lambda_2^5 \\ \mu J^{-2}\lambda_3^5 \\ -\frac{1\mu}{3}J^{-3}(\lambda_1^6 + \lambda_2^6 + \lambda_3^6) + \frac{\kappa}{2}(J - J^{-1}) \end{bmatrix} \tag{B.3}$$

For all $\mu > 0$, the derivatives of $\{\lambda_1, \lambda_2, \lambda_3\}$ are non-negative. Hence, the strain energy density function (B.2) satisfies the non-decreasing condition in $\{\lambda_1, \lambda_2, \lambda_3\}$.

The Hessian of the strain energy density function (B.2) with respect to $\boldsymbol{\Lambda}$ is:

$$\nabla^2_{\boldsymbol{\Lambda}}\psi_O = \begin{bmatrix} 5\mu J^{-2}\lambda_1^4 & 0 & 0 & -2\mu J^{-3}\lambda_1^5 \\ 0 & 5\mu J^{-2}\lambda_2^4 & 0 & -2\mu J^{-3}\lambda_2^5 \\ 0 & 0 & 5\mu J^{-2}\lambda_3^4 & -2\mu J^{-3}\lambda_3^5 \\ -2\mu J^{-3}\lambda_1^5 & -2\mu J^{-3}\lambda_2^5 & -2\mu J^{-3}\lambda_3^5 & \mu J^{-4}(\lambda_1^6 + \lambda_2^6 + \lambda_3^6) + \frac{\kappa}{2}(1 + J^{-2}) \end{bmatrix} \tag{B.4}$$

The principal minors M_{11}, M_{22}, and M_{33} are clearly positive for $\mu > 0$. It is, therefore, sufficient that $M_{44} > 0$ for the Hessian (B.4) to be positive semi-definite. Checking this, we have:

$$M_{44} = \begin{vmatrix} 5\mu J^{-2}\lambda_1^4 & 0 & 0 & -2\mu J^{-3}\lambda_1^5 \\ 0 & 5\mu J^{-2}\lambda_2^4 & 0 & -2\mu J^{-3}\lambda_2^5 \\ 0 & 0 & 5\mu J^{-2}\lambda_3^4 & -2\mu J^{-3}\lambda_3^5 \\ -2\mu J^{-3}\lambda_1^5 & -2\mu J^{-3}\lambda_2^5 & -2\mu J^{-3}\lambda_3^5 & \mu J^{-4}(\lambda_1^6 + \lambda_2^6 + \lambda_3^6) + \frac{\kappa}{2}(1 + J^{-2}) \end{vmatrix} \tag{B.5}$$

which, after simplification, produces:

$$M_{44} = \frac{25\mu^3}{2} J^{-6}(2\mu(\lambda_1^6 + \lambda_2^6 + \lambda_3^6) + 5\kappa(J^4 + J^2)) \tag{B.6}$$

For $\mu, \kappa > 0$, the minor M_{44} is positive. Hence the strain energy density function (B.2) is convex in $\boldsymbol{\Lambda}$.

Now, let the chosen invariants be $\check{\imath} = \{I_{C1}, I_{C2}, J\}$:

$$\begin{aligned} I_{C1} &= \lambda_1^2 + \lambda_2^2 + \lambda_3^2 \\ I_{C2} &= \lambda_1^2\lambda_2^2 + \lambda_2^2\lambda_3^2 + \lambda_3^2\lambda_1^2 \\ J &= \lambda_1\lambda_2\lambda_3 \end{aligned} \tag{B.7}$$

The strain energy density function (B.2) can be expressed as a function of these invariants as:

$$\begin{aligned}\breve{\psi}_O(\tilde{\boldsymbol{\imath}}) &= \frac{\mu}{6}(J^{-2}(I_{C1}^3 - 3I_{C1}I_{C2} + 3J^2) - 3) + \frac{\kappa}{4}(J^2 - 1 - 2\ln J) \\ &= \frac{\mu}{6}J^{-2}(I_{C1}^3 - 3I_{C1}I_{C2}) + \frac{\kappa}{4}(J^2 - 1 - 2\ln J)\end{aligned} \tag{B.8}$$

Here, we provide the extra steps for clarity:

$$\begin{aligned}I_{C1}^3 &= (\lambda_1^2 + \lambda_2^2 + \lambda_3^2)^3 \\ &= \lambda_1^6 + \lambda_2^6 + \lambda_3^6 \\ &\quad + 3(\lambda_1^4\lambda_2^2 + \lambda_1^2\lambda_2^4 + \lambda_2^4\lambda_3^2 + \lambda_2^2\lambda_3^4 + \lambda_3^4\lambda_1^2 + \lambda_3^2\lambda_1^4) + 6\lambda_1^2\lambda_2^2\lambda_3^2 \\ I_{C1}I_{C2} &= (\lambda_1^2 + \lambda_2^2 + \lambda_3^2)(\lambda_1^2\lambda_2^2 + \lambda_2^2\lambda_3^2 + \lambda_3^2\lambda_1^2) \\ &= \lambda_1^4\lambda_2^2 + \lambda_1^2\lambda_2^2\lambda_3^2 + \lambda_1^4\lambda_3^2 \\ &\quad + \lambda_2^4\lambda_1^2 + \lambda_1^2\lambda_2^2\lambda_3^2 + \lambda_2^4\lambda_3^2 \\ &\quad + \lambda_1^2\lambda_2^2\lambda_3^2 + \lambda_3^4\lambda^2 + \lambda_3^4\lambda_1^2 \\ &= (\lambda_1^4\lambda_2^2 + \lambda_1^2\lambda_2^4 + \lambda_2^4\lambda_3^2 + \lambda_2^2\lambda_3^4 + \lambda_3^4\lambda_1^2 + \lambda_3^2\lambda_1^4) + 3\lambda_1^2\lambda_2^2\lambda_3^2 \\ 3I_{C1}I_{C2} &= 3(\lambda_1^4\lambda_2^2 + \lambda_1^2\lambda_2^4 + \lambda_2^4\lambda_3^2 + \lambda_2^2\lambda_3^4 + \lambda_3^4\lambda_1^2 + \lambda_3^2\lambda_1^4) + 9\lambda_1^2\lambda_2^2\lambda_3^2 \\ I_{C1}^3 - 3I_{C1}I_{C2} &= \lambda_1^6 + \lambda_2^6 + \lambda_3^6 - 3\lambda_1^2\lambda_2^2\lambda_3^2 \\ 3J^2 &= 3\lambda_1^2\lambda_2^2\lambda_3^2 \\ I_{C1}^3 - 3I_{C1}I_{C2} + 3J^2 &= \lambda_1^6 + \lambda_2^6 + \lambda_3^6\end{aligned} \tag{B.9}$$

The derivative of the strain energy density function (B.8) with respect to these invariants is:

$$\nabla_{\tilde{\boldsymbol{\imath}}}\breve{\psi}_O = \begin{bmatrix} \frac{\mu}{2}J^{-2}(I_{C1}^2 - I_{C2}) \\ -\frac{\mu}{2}J^{-2}I_{C1} \\ -\frac{\mu}{3}J^{-3}(I_{C1}^3 - 3I_{C1}I_{C2}) + \frac{\kappa}{2}(J - J^{-1}) \end{bmatrix} \tag{B.10}$$

The derivative w.r.t I_{C2} is negative for all I_{C1}, for any $\mu > 0$. This implies that the strain energy density function (B.8) is *not* non-decreasing in I_{C2}. Differentiating again, we obtain the Hessian of the strain energy density function (B.8) with respect to these invariants:

$$\nabla^2_{\tilde{\boldsymbol{\imath}}}\breve{\psi}_O = \begin{bmatrix} \mu J^{-2}I_{C1} & -\frac{\mu}{2}J^{-2} & -\mu J^{-3}(I_{C1}^2 - I_{C2}) \\ -\frac{\mu}{2}J^{-2} & 0 & \mu J^{-3}I_{C1} \\ -\mu J^{-3}(I_{C1}^2 - I_{C2}) & \mu J^{-3}I_{C1} & \mu J^{-4}(I_{C1}^3 - 3I_{C1}I_{C2}) + \frac{\kappa}{2}(1 + J^{-2}) \end{bmatrix} \tag{B.11}$$

The principal minors of the Hessian (B.8) are:

$$\begin{aligned}M_{11} &= \mu J^{-2}I_{C1}\ , \\ M_{22} &= -\frac{\mu}{4}J^{-4} < 0\ , \\ M_{33} &= -\frac{\mu^2}{8}J^{-8}(2\mu(I_{C1}^3 + I_{C1}I_{C2}) + \kappa J^2(1 + J^2)) \\ &= -\left[\frac{\mu^3}{4}J^{-8}(I_{C1}^3 + I_{C1}I_{C2}) + \frac{\mu^2\kappa}{8}J^{-6}(1 + J^2)\right] < 0\ ,\end{aligned} \tag{B.12}$$

two of which are negative for any $\mu, \kappa > 0$. This implies that the Hessian is *not* positive semi-definite and, consequently, that the strain energy density function (B.8) is *not* convex in I_{C1}, I_{C2}.

Hence we argue that an ICNN that is convex and non-decreasing in the invariants I_{C1}, I_{C2}, J will not be able to represent the strain energy density function (B.2). On the contrary, for isotropic hyperelasticity, if we built an ICNN that satisfied (6) directly, then it would be able to represent any arbitrary isotropic hyperelastic strain energy density function, given a sufficient number of layers and neurons. This can be achieved using an ICNN (A.6) with the input $\hat{\boldsymbol{y}} = \boldsymbol{\Lambda} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_1\lambda_2, \lambda_2\lambda_3, \lambda_3\lambda_1, J, -J\}$ with symmetric enforcing $\boldsymbol{W}_i^{(\hat{y})}$ through weight sharing for all the direct connections from the input (i.e., the first layer and all the passthrough layers). Thus, $\boldsymbol{W}_i^{(\lambda_1)} = \boldsymbol{W}_i^{(\lambda_2)} = \boldsymbol{W}_i^{(\lambda_3)}$ and $\boldsymbol{W}_i^{(\lambda_1\lambda_2)} = \boldsymbol{W}_i^{(\lambda_2\lambda_3)} = \boldsymbol{W}_i^{(\lambda_3\lambda_1)}$ for $i = 0, \dots, l-1$ in (A.6).

Appendix C. Derivation of the stress correction term for the isotropic hyperelastic neural network model

The starting point for the derivation is due to the observation made in Linden et al. (2023), that a stress correction term $\mathcal{N}_0^{\text{stress}}$ of the form:

$$\mathcal{N}_0^{\text{stress}}(\boldsymbol{E}) = -\partial_{\boldsymbol{E}}\mathcal{N}(\tilde{\boldsymbol{\Lambda}}(\boldsymbol{E}))\Big|_{\boldsymbol{E}=\boldsymbol{0}} : \boldsymbol{E} \tag{C.1}$$

violates the polyconvexity and material symmetry conditions. As an alternative, they propose a stress correction term based on a weighted sum of derivatives method, where the wights are obtained via exploiting the chain rule and the behavior of derivative

of strain invariants at undeformed state. Here we adapt this approach to the principal stretch based isotropic hyperelastic neural network model we have developed.

Recall the input to the model $\tilde{\mathbf{\Lambda}} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_1\lambda_2, \lambda_2\lambda_3, \lambda_3\lambda_1, J, -J, -\ln J\}$ as proposed in Section 2.2. Due to the symmetry enforcement we have made, the derivative of the neural network output $\mathcal{N}$ with respect to the input $\tilde{\mathbf{\Lambda}}$ respects the enforced symmetry. As, a result, we can consider a stress correction term for the strain energy function of the form:

$$\mathcal{N}_0^{\text{stress}} = -\sum_{i=1}^{9} \partial_{\tilde{\Lambda}_i} \mathcal{N}(\mathbf{\Lambda}(\tilde{\boldsymbol{E}}))\Big|_{\boldsymbol{E}=\mathbf{0}} (\tilde{\Lambda}_i - \tilde{\Lambda}_i^0) \tag{C.2}$$

The corresponding stress contribution could be obtained by the chain rule as:

$$\boldsymbol{S}_0^{\text{stress}} = -\sum_{i=1}^{9} \partial_{\tilde{\Lambda}_i} \mathcal{N}(\tilde{\mathbf{\Lambda}}(\boldsymbol{E}))\Big|_{\boldsymbol{E}=\mathbf{0}} \partial_{\boldsymbol{E}} \tilde{\Lambda}_i \tag{C.3}$$

Due to symmetry enforcement, the above expression simplifies to:

$$\boldsymbol{S}_0^{\text{stress}} = -(\sum_{i=1}^{3} h_1 \partial_{\boldsymbol{E}} \tilde{\Lambda}_i + \sum_{j=4}^{6} h_2 \partial_{\boldsymbol{E}} \tilde{\Lambda}_j + h_3 \partial_{\boldsymbol{E}} \tilde{\Lambda}_7 + h_4 \partial_{\boldsymbol{E}} \tilde{\Lambda}_8 + h_5 \partial_{\boldsymbol{E}} \tilde{\Lambda}_9) \tag{C.4}$$

which when evaluated at $\boldsymbol{E} = \mathbf{0}$ gives:

$$\begin{aligned}
\boldsymbol{S}_0^{\text{stress}}\Big|_{\boldsymbol{E}=\mathbf{0}} &= -(h_1\mathbf{1} + 2h_2\mathbf{1} + h_3\mathbf{1} - h_4\mathbf{1} - h_5\mathbf{1}) \\
&= -(h_1 + 2h_2 + h_3 - h_4 - h_5)\mathbf{1} \quad \text{where} \\
h_1 &= \partial_{\tilde{\Lambda}_1}\mathcal{N}\Big|_{\boldsymbol{E}=\mathbf{0}} = \partial_{\tilde{\Lambda}_2}\mathcal{N}\Big|_{\boldsymbol{E}=\mathbf{0}} = \partial_{\tilde{\Lambda}_3}\mathcal{N}\Big|_{\boldsymbol{E}=\mathbf{0}}; \\
h_2 &= \partial_{\tilde{\Lambda}_4}\mathcal{N}\Big|_{\boldsymbol{E}=\mathbf{0}} = \partial_{\tilde{\Lambda}_5}\mathcal{N}\Big|_{\boldsymbol{E}=\mathbf{0}} = \partial_{\tilde{\Lambda}_6}\mathcal{N}\Big|_{\boldsymbol{E}=\mathbf{0}}; \\
h_3 &= \partial_{\tilde{\Lambda}_7}\mathcal{N}\Big|_{\boldsymbol{E}=\mathbf{0}}; \quad h_4 = \partial_{\tilde{\Lambda}_8}\mathcal{N}\Big|_{\boldsymbol{E}=\mathbf{0}}; \quad h_5 = \partial_{\tilde{\Lambda}_9}\mathcal{N}\Big|_{\boldsymbol{E}=\mathbf{0}}
\end{aligned} \tag{C.5}$$

Note that the stress contribution $\boldsymbol{S}_0^{\text{stress}}$ coming from the correction term $\mathcal{N}_0^{\text{stress}}$ is not polyconvex. However, we can exploit the fact that an affine transformation of J preserves polyconvexity, and the fact that the derivative of $\partial_{\boldsymbol{E}} J = J\boldsymbol{C}^{-1}$, which when evaluated at $\boldsymbol{E} = \mathbf{0}$ becomes identity $\mathbf{1}$, to arrive at a polyconvex stress correction term. The polyconvex alternative that result in the requisite stress contribution at $\boldsymbol{E} = \mathbf{0}$ as given in (C.5) may be obtained by defining a stress correction term:

$$\mathcal{N}_0^{\text{stress}} = -\sum_{i=1}^{9} \xi_i \partial_{\tilde{\Lambda}_i} \mathcal{N}(\tilde{\mathbf{\Lambda}}(\boldsymbol{E}))\Big|_{\boldsymbol{E}=\mathbf{0}} (J - 1) \tag{C.6}$$

where $\boldsymbol{\xi} = \{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}, \frac{2}{3}, \frac{2}{3}, 1, -1, 1\}$ are the weights that result in the requisite stress contribution as in (C.5). The corresponding stress contribution can be obtained as:

$$\boldsymbol{S}_0^{\text{stress}} = -\sum_{i=1}^{9} \xi_i \partial_{\tilde{\Lambda}_i} \mathcal{N}(\tilde{\mathbf{\Lambda}}(\boldsymbol{E}))\Big|_{\boldsymbol{E}=\mathbf{0}} J\boldsymbol{C}^{-1} \tag{C.7}$$

It can be clearly seen that evaluation of (C.7) at $\boldsymbol{E} = \mathbf{0}$ gives the desired stress correction term as in (C.5):

$$\begin{aligned}
\boldsymbol{S}_0^{\text{stress}}\Big|_{\boldsymbol{E}=\mathbf{0}} &= -\sum_{i=1}^{9} \xi_i \partial_{\tilde{\Lambda}_i} \mathcal{N}(\tilde{\mathbf{\Lambda}}(\boldsymbol{E}))\Big|_{\boldsymbol{E}=\mathbf{0}} \mathbf{1} \\
&= -\sum_{i=1}^{9} \xi_i h_i \mathbf{1} \\
&= -(h_1 + 2h_2 + h_3 - h_4 - h_5)\mathbf{1}
\end{aligned} \tag{C.8}$$

Appendix D. Additional topology optimization examples assessing ML material model

D.1. Single-scale topology optimization: Portal frame example

Next, we consider a 2D portal frame with the design domain shown in Fig. D.13(a). Due to the symmetry in the geometry and loading conditions, only half of the domain is considered for the TO problem. The support leg of the portal frame is fixed in the vertical direction, and horizontal displacement is constrained along the edge subject to the symmetry condition. A vertically downward *average applied displacement* $c_N = 0.2L$ is applied at the center of the portal frame. The domain is discretized using 4-noded quadrilateral elements with an unstructured mesh of nominal edge length 0.5 mm, resulting in a total of 28902 elements. The TO problem is formulated with the objective of maximizing the external work done by the applied displacement, subject to a material volume fraction constraint set by $g_{\max} = 0.5$.

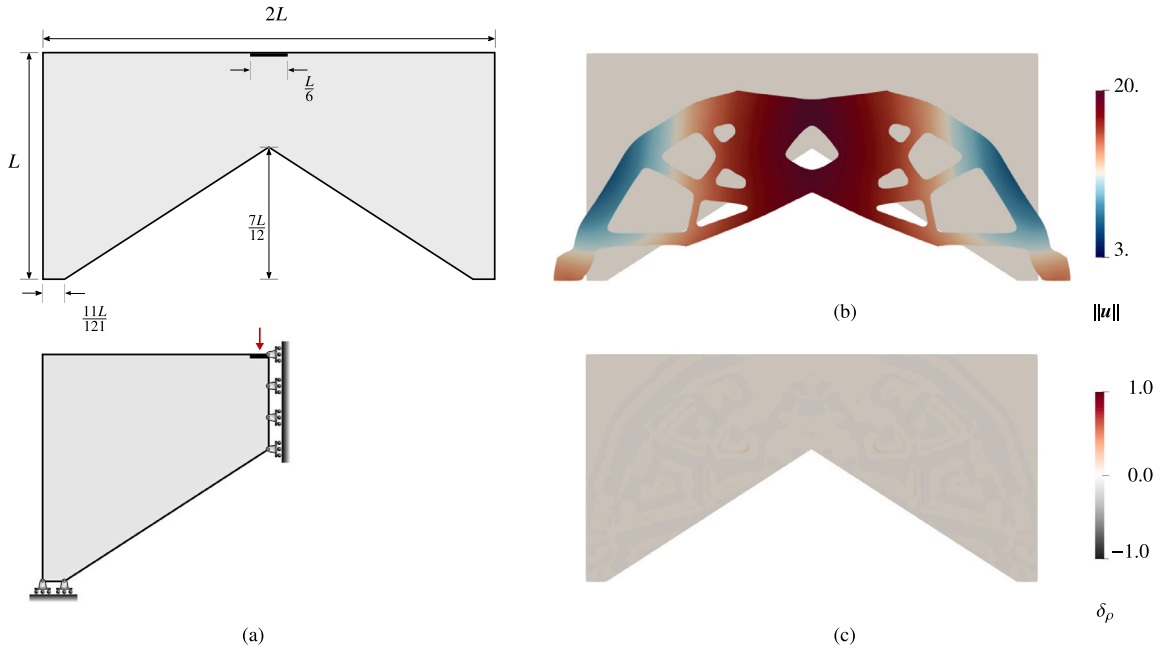


Fig. D.13. (a) Portal frame BVP (b) Deformation of the design obtained using the ML model (c) Difference plot of the design obtained using the ML model to the design obtained using the ground truth phenomenological model.

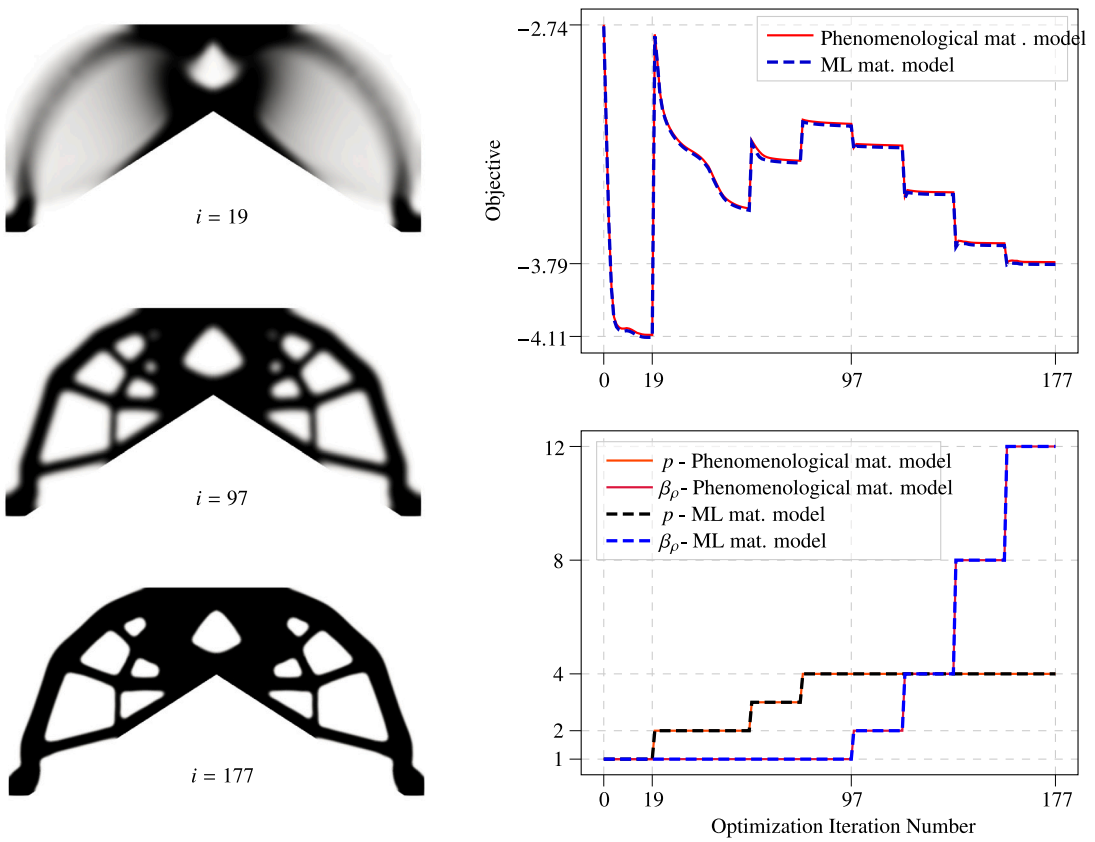


Fig. D.14. Convergence history for portal frame TO problem. Visualized designs correspond to those corresponding to TO using ML model at iterations where critical continuation parameter updates occur.

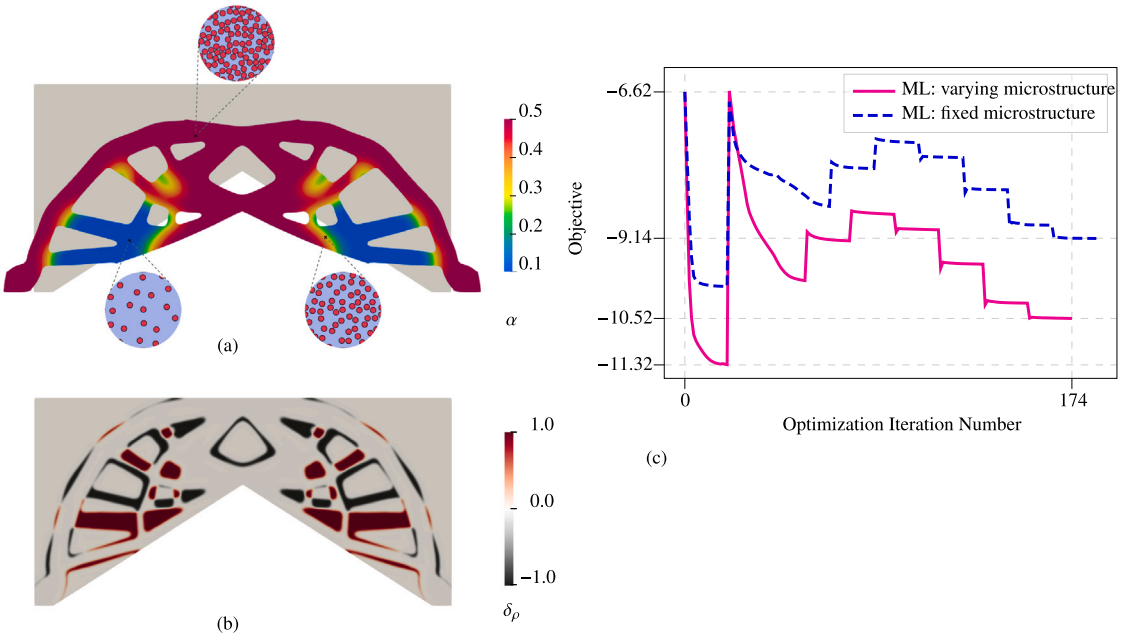


Fig. D.15. (a) Deformed configuration of the final design obtained using the ML model with varying microstructure (b) Difference plot between the design obtained with varying microstructure and the design obtained with fixed microstructure. (c) Convergence history for the portal frame TO problem with varying microstructure vs. fixed microstructure.

The convergence history of the objective function for the portal frame, shown in Fig. D.14, indicates that the optimization process using the ML model and the ground truth phenomenological model progresses in a similar manner. The objective curves and the continuation scheme curves overlap for both the cases, which results in the intermediate as well as the final designs (see Fig. D.14) obtained using the two material models to be almost identical. The difference plot in Fig. D.13(c) reinforces this observation. Fig. D.13(b) illustrates the magnitude of deformation of the final design obtained using the ML model in its deformed state.

D.2. Multi-scale topology optimization: Portal frame example

The portal frame example with varying microstructure and the baseline fixed microstructure are set up with the same side constraint bounds as the T-bracket example in Section 5.5.1. Visible differences are observed in the final topologies between the varying microstructure and fixed microstructure cases, with the former containing 13 void regions and the latter containing 15. The difference plot in Fig. D.15b highlights that although the external outlines of the designs are similar, there is considerable rearrangement of the material in the internal regions of the designs. Similar to the T-bracket example, allowing the microstructure to vary results in a lower objective value. The convergence history for the portal frame example in Fig. D.15c shows that the baseline case with fixed microstructure undergoes a delayed update in the continuation scheme corresponding to the update in the penalty parameter $p = 3$. This induces a rightward shift in the convergence history when compared with the case with varying microstructure. Consistent with the observations in the T-bracket example, the final design with varying microstructure (Fig. D.15a) has material addition predominantly occurring close to regions with lower particle volume fraction α and material removal happening close to regions with higher particle volume fractions.

D.3. Single-scale topology optimization: 3D benchmark example with fixed–fixed beam

Finally, we consider a 3D fixed–fixed beam with the design domain, as shown in Fig. D.16(a). The primary intention of considering this example is to assess the "sensitivity" of the optimization process to the differences in the phenomenological and ML models, as well as to show that the framework is easily extensible to 3D problems. In order to train a 3D ML model, we employed the design of experiment approach described in Section 5.1 to generate the training dataset $\mathcal{D}^s_{3D}$ consisting of $2^{13} = 8192$ sample strain states by considering the spatial dimensionality $N_D = 3$. The model and training hyperparameters and the performance metrics are given in Tables D.5 and D.6 respectively.

The beam is fixed along the left and right faces and a downward *average applied displacement* $c_N = 0.3L$ is applied at the center of the top surface of the beam. Due to symmetry in the geometry and loading conditions, only a quarter of the domain is considered for this example. The domain is discretized using 8-node hexahedral elements with a uniform mesh of edge length 2 mm, adding up

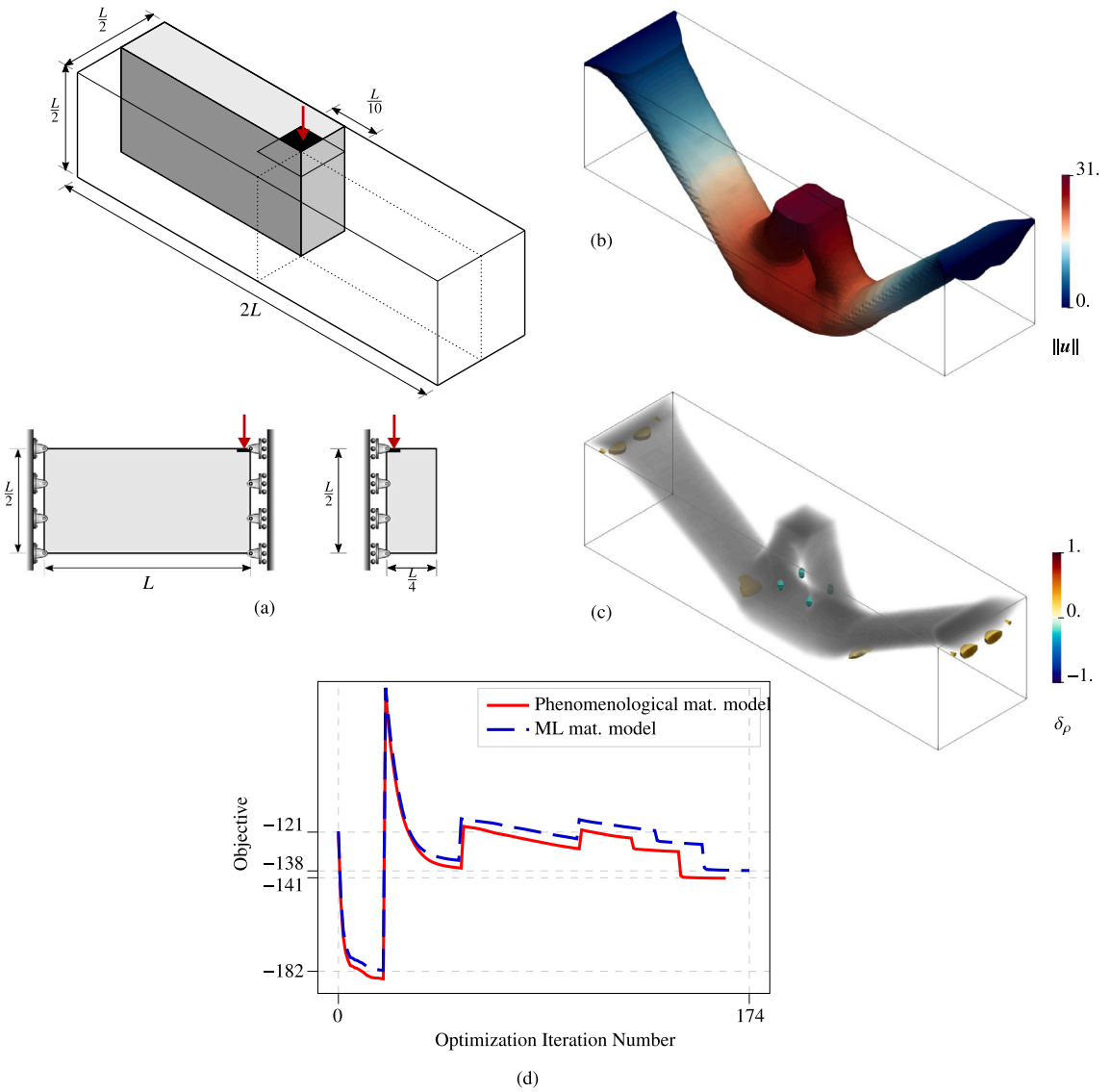


Fig. D.16. (a) Fixed–fixed beam BVP (b) Deformation of the design obtained using the ML model (c) Difference plot of the design obtained using the ML model to the design obtained using the ground truth phenomenological model. A transparent shadow of the design obtained using the phenomenological model is overlaid on the difference plot for reference. (d) Convergence history for the fixed–fixed beam 3D TO benchmark.

Table D.5
Model and training hyperparameters.

Hyperparameter	Value
No. hidden layers	2
No. neurons per layer	8
Initial learning rate	1×10^{-2}

Hyperparameter	Value
Exponential decay rate	0.5
Decay transition epoch interval	1000
Decay end value	5×10^{-4}

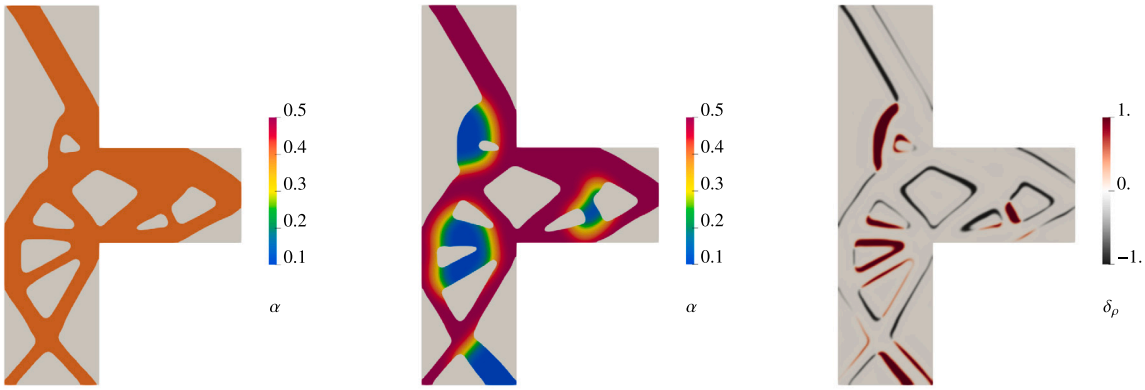
Hyperparameter	Value
Max epochs	15 000
Batch size	128
Early stopping patience	5000

to 16250 elements. A volume fraction constraint upper bound of $g_{\max} = 0.3$ is imposed in this example. The optimization parameters are as shown in Table 3, with the exception of the number of time steps, which is set to 15 for this example.

We observe that the final objective values using the ML model and the ground truth phenomenological model are almost identical. The difference plot in Fig. D.16(c) shows no significant material addition or deletion in the final design obtained using the ML model compared to the design obtained using the ground truth phenomenological model. The convergence history for the ML model shows minor differences; however, the lateral shift is caused by slight delays in convergence within each continuation segment.

Table D.6
Performance metrics of the trained model.

Model	Metric	Training dataset	Validation dataset	Test dataset
$\mathcal{M}^{s}_{3D}$	$1 - R^2$	2.06×10^{-5}	2.03×10^{-5}	2.15×10^{-5}
	RMSE	4.63×10^{-3}	4.60×10^{-3}	4.61×10^{-3}
	MAE	3.05×10^{-3}	3.05×10^{-3}	3.05×10^{-3}



(a) (b) (c)

Fig. D.17. Design obtained using the ML model with (a) fixed microstructure ($\alpha = \alpha_0 = 0.4$) and (b) Varying microstructure ($\alpha \in [0.1, 0.5]$) for the T-bracket TO problem. (c) Difference plot between the designs obtained using the varying microstructure and the fixed microstructure, obtained by subtracting the design field ρ obtained for the fixed microstructure from that obtained for the varying microstructure.

D.4. Supplementary results for the multi-scale topology optimization problems

In addition to the results presented in Section 5.5 for the T-bracket example and the cantilever example, as well as the results for portal frame in Appendix D.2, we present supplementary results in Figs. D.17–D.19 for clarity.

Appendix E. Additional results concerning homogenized response of the representative volume elements

The homogenized response of the RVEs is not readily computable through simple analytical homogenization approaches such as the rule of mixtures. Fig. E.20 shows the prediction of a rule of mixtures based model for the stress response of RVEs as a function of the particle volume fraction α, which is obtained analytically as:

$$\boldsymbol{S}_{\text{mix}} = \partial_{\boldsymbol{E}} \psi_{\text{mix}} \quad \text{where} \quad \psi_{\text{mix}} = \alpha \psi_{\text{inc}} + (1 - \alpha) \psi_{\text{mat}} \tag{E.1}$$

where, $\psi_{\text{inc}} = \psi_{NH}$ and $\psi_{\text{mat}} = \psi_{AB}$ with the material parameters as given in Section 5. It can be clearly seen that the dependence of the particle volume fraction is non-linear as the linear superposition, as in (E.1), does not capture the ground truth response obtained through computational homogenization. This is due to the fact that the effective homogenized response has complex dependencies due to the non-linear nature of the constituent phases, the geometrically non-linear setting, and the random nature of the particles. In contrast, the microstructure-dependent ML model is able to capture the non-linear dependence of the stress response on the particle volume fraction α, as shown in Fig. 5.

Another aspect of the RVEs considered in this work is the isotropic nature of their homogenized response. Although the microstructure by itself is heterogeneous at the microscale, the random yet space-filling nature of the considered particles results in a homogeneous response that is isotropic. Fig. E.21 shows the stress response of the RVEs for uniaxial loading in both E_{11} and E_{22}. The figure has the same configuration as that shown in Fig. 5, but additionally illustrates the stress response for the symmetric loading scenario (i.e., variation of both E_{11} and E_{22} separately, with all other strain components set to zero). Please note that we plot both the (E_{11}, S_{11}) and (E_{22}, S_{22}) pairs on the same axes for ease of comparison.

Appendix F. Boundary value problem and homogenization of RVE

The visibility of a material's configurational features depends on the scale at which it is inspected. At a larger *macro* scale, the material may seem completely uniform or *homogeneous*, while at another finer *micro* scale, the material can reveal a complex *heterogeneous* microstructure. Generally, these *scales* are considered distinct or *separate* when the contrast between them is sufficiently high. In such cases, classical, first-order theories may be applied to describe the homogenized behavior observable from the

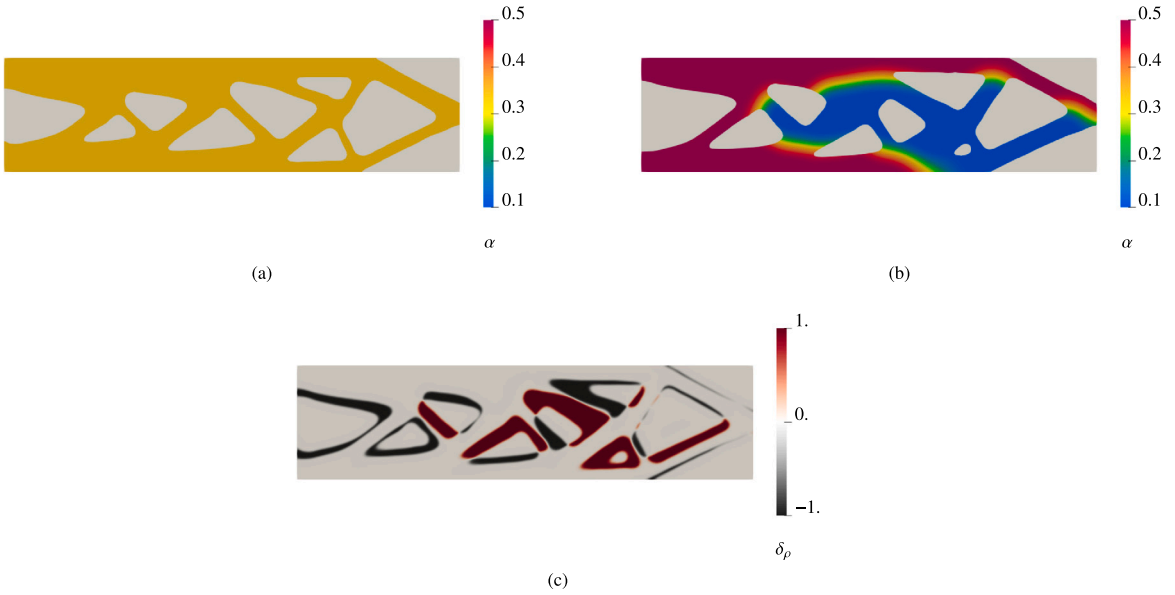


Fig. D.18. Design obtained using the ML model with (a) fixed microstructure ($\alpha = \alpha_0 = 1/3$) and (b) Varying microstructure ($\alpha \in [0.1, 0.5]$) for the cantilever beam TO problem. (c) Difference plot between the designs obtained using the varying microstructure and the fixed microstructure, obtained by subtracting the design field ρ obtained for the fixed microstructure from that obtained for the varying microstructure.

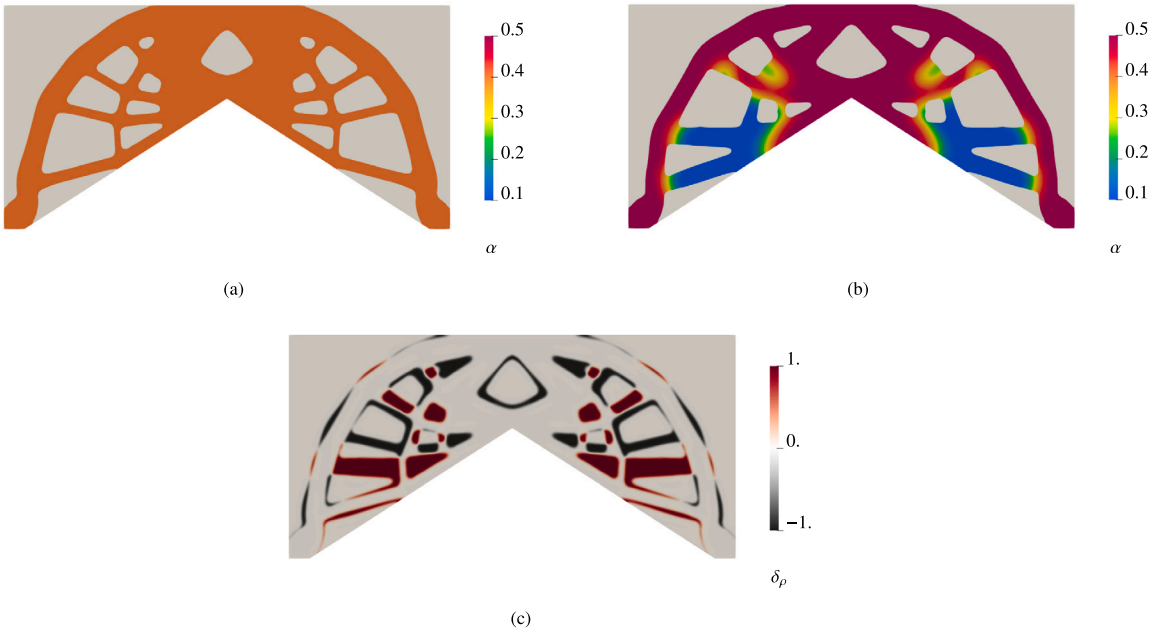


Fig. D.19. Design obtained using the ML model with (a) fixed microstructure ($\alpha = \alpha_0 = 0.4$) and (b) Varying microstructure ($\alpha \in [0.1, 0.5]$) for the portal frame TO problem. (c) Difference plot between the designs obtained using the varying microstructure and the fixed microstructure, obtained by subtracting the design field ρ obtained for the fixed microstructure from that obtained for the varying microstructure.

macroscale. When creating a Representative Volume Element of a composite material, there are two important aspects to consider: (1) the material is usually assumed to have periodic microstructure, and (2) the RVE needs to be sufficiently large such that when the microstructure is randomized, its response does not change. Simultaneously, if first-order homogenization is assumed, then the

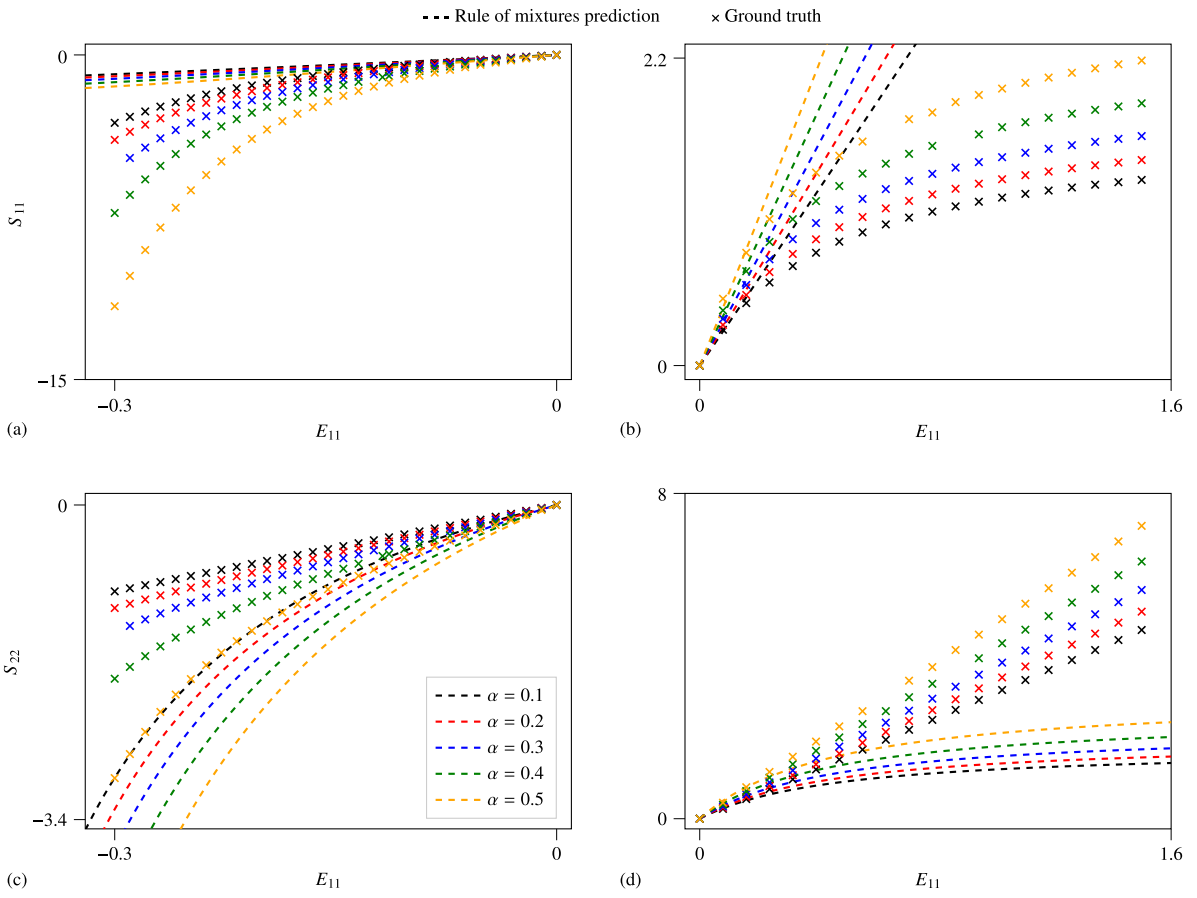


Fig. E.20. Stress response of the rule of mixtures model (E.1) compared to the ground truth obtained through RVE simulations for a uniaxial strain loading scenario for (left) compressive loading, and (right) tension loading range. The top and bottom plots show the stress components S_{11} and S_{22} separately.

boundary value problem of the RVE is directly translated from the macroscopic strain at a point in the domain of the structure. If one defines the characteristic length of a representative volume element (RVE) at the microscale as l_{micro} and that of the macroscale continuum as l_{macro}, the first-order homogenization theory assumes that $l_{\text{micro}} \ll l_{\text{macro}}$. This means that the size of the RVE is sufficiently small, and that the macroscale (structure) is separated from the microscale (RVE).

In this section, we briefly review the essential details of the homogenization step employed in the design of experiments for data generation. Let Ω represent the domain of the RVE and $\partial\Omega$ the boundary of the RVE composed of the Dirichlet boundary ($\partial\Omega_{\boldsymbol{u}}$) and Neumann boundary ($\partial\Omega_t = \partial\Omega \setminus \partial\Omega_{\boldsymbol{u}}$). Then the macroscopic kinematics at any time instant may be imposed on the microscale by defining a periodic RVE boundary value problem in terms of the macroscopic deformation gradient $\boldsymbol{F}$:

$$\boldsymbol{u}_\mu(\boldsymbol{Y}) = (\boldsymbol{F}(\boldsymbol{X}) - \boldsymbol{1})\boldsymbol{Y} + \tilde{\boldsymbol{u}}_\mu(\boldsymbol{Y}) = \boldsymbol{u}_\mu^{\text{lin}} + \tilde{\boldsymbol{u}}_\mu(\boldsymbol{Y}) \quad \text{on} \quad \partial\Omega_u \tag{F.1}$$

where $\boldsymbol{X}$ and $\boldsymbol{Y}$ correspond to typical points in the reference configuration at the macroscale and microscale, respectively. Here, $\tilde{\boldsymbol{u}}_\mu(\boldsymbol{Y})$ corresponds to the microscopic displacement fluctuation term that is bound to a kinematically admissible space constrained by the periodic boundary condition. Note that kinematic homogenization or volume averaging of the microscopic deformation gradient $\boldsymbol{F}_\mu$ enables us to transition from the microscale to the macroscale:

$$\langle \boldsymbol{F}_\mu \rangle = \frac{1}{V_\mu} \int_\Omega \boldsymbol{F}_\mu(\boldsymbol{Y}) \mathrm{d}V = \boldsymbol{F} \tag{F.2}$$

where V_μ denotes the RVE volume in the reference configuration. With these definitions in place, the solution to the boundary value problem associated with the RVE can be obtained by solving the microscale equilibrium equation:

$$\begin{aligned} \operatorname{div}[\boldsymbol{P}_\mu(\boldsymbol{Y})] &= \boldsymbol{0} \quad \text{on} \quad \Omega \\ \boldsymbol{P}_\mu(\boldsymbol{Y})\boldsymbol{n}(\boldsymbol{Y}) - \boldsymbol{t}_\mu(\boldsymbol{Y}) &= \boldsymbol{0} \quad \text{on} \quad \partial\Omega_t \end{aligned} \tag{F.3}$$

expressed in terms of the microscale first Piola–Kirchhoff stress $\boldsymbol{P}_\mu$, subject to the boundary conditions in (F.1). From the Hill–Mandel lemma it follows that:

$$\langle \boldsymbol{P}_\mu : \boldsymbol{F}_\mu \rangle = \langle \boldsymbol{P}_\mu \rangle : \langle \boldsymbol{F}_\mu \rangle = \boldsymbol{P} : \boldsymbol{F} \tag{F.4}$$

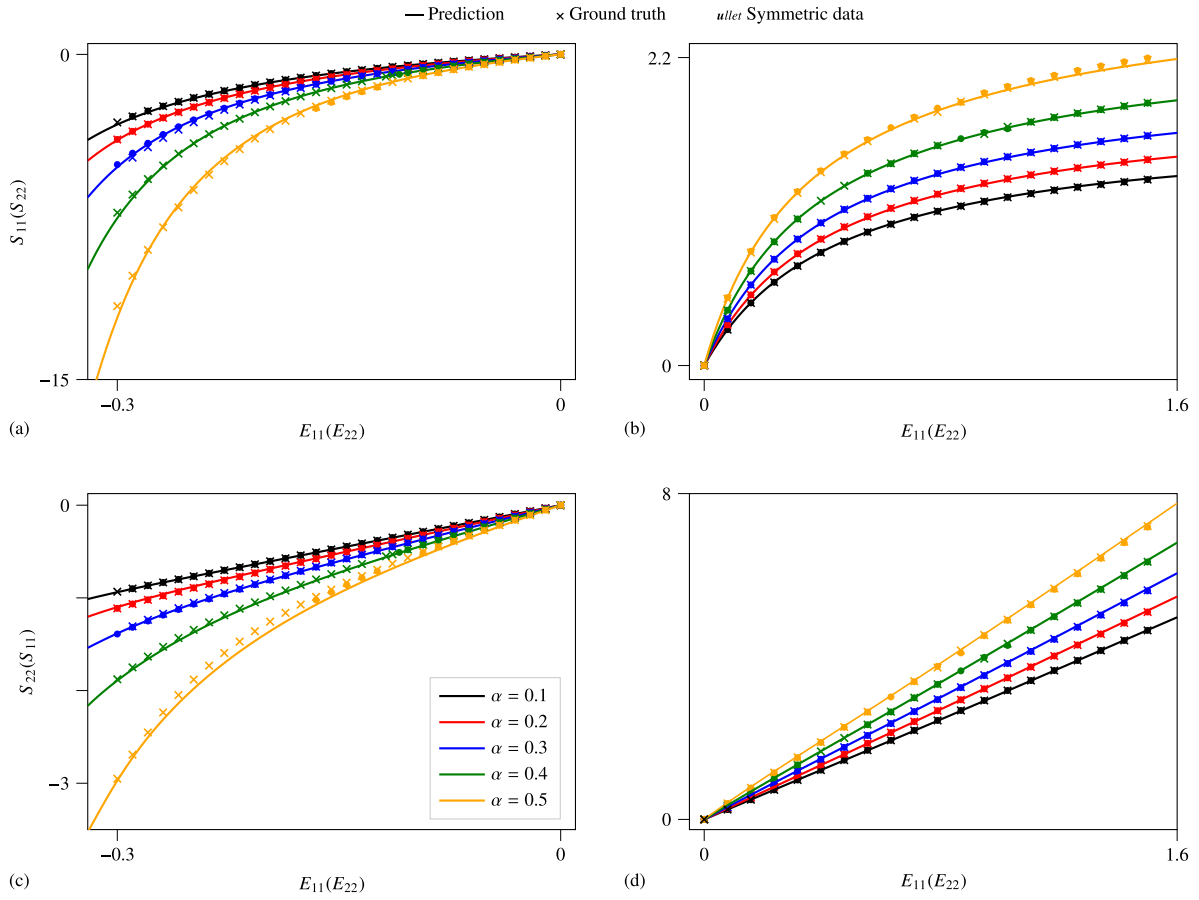


Fig. E.21. Stress response of RVEs of varying volume fractions ($\alpha \in [0.1, 0.5]$) for the uniaxial strain loading (variation of E_{11}), plotted together with the stress response corresponding to the symmetric loading (variation of E_{22}). The labels in $(\cdot)$ correspond to the symmetric data. The prediction from the microstructure-dependent consistent ML model closely matches the ground truth obtained through RVE simulations and the symmetric loading scenario.

which when read together with (F.2) gives us the relationship to obtain the macroscopic counterpart of the first Piola–Kirchhoff stress

$$\boldsymbol{P} = \langle \boldsymbol{P}_\mu \rangle = \frac{1}{V_\mu} \int_\Omega \boldsymbol{P}_\mu(\boldsymbol{Y}) \, \mathrm{d}V = \frac{1}{V_\mu} \int_{\partial\Omega_t} \boldsymbol{t}_\mu \otimes \boldsymbol{Y} \, \mathrm{d}A \tag{F.5}$$

Subsequently, the macroscopic second Piola–Kirchhoff stress $\boldsymbol{S}$ may be obtained as

$$\boldsymbol{S} = \boldsymbol{F}^{-1} \boldsymbol{P} \tag{F.6}$$

Data availability

Data will be made available on request.

References

Allaire, G., Jouve, F., Toader, A.M., 2004. Structural optimization using sensitivity analysis and a level-set method. J. Comput. Phys. 194 (1), 363–393. http://dx.doi.org/10.1016/j.jcp.2003.09.032.

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

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

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

Avazmohammadi, R., Ponte Castañeda, P., 2016. Macroscopic constitutive relations for elastomers reinforced with short aligned fibers: Instabilities and post-bifurcation response. J. Mech. Phys. Solids 97, 37–67. http://dx.doi.org/10.1016/j.jmps.2015.07.007.

Ball, J.M., 1976. Convexity conditions and existence theorems in nonlinear elasticity. Arch. Ration. Mech. Anal. 63 (4), 337–403. http://dx.doi.org/10.1007/bf00279992.

Bendsøe, M.P., 1989. Optimal shape design as a material distribution problem. Struct. Optim. 1 (4), 193–202. http://dx.doi.org/10.1007/bf01650949.

Bendsøe, M.P., Kikuchi, N., 1988. Generating optimal topologies in structural design using a homogenization method. Comput. Methods Appl. Mech. Engrg. 71 (2), 197–224. http://dx.doi.org/10.1016/0045-7825(88)90086-2.

Bergstrom, J.S., 2015. Mechanics of Solid Polymers: Theory and Computational Modeling. Elsevier, http://dx.doi.org/10.1016/c2013-0-15493-1.

Bessa, M.A., Bostanabad, R., Liu, Z., Hu, A., Apley, D.W., Brinson, C., Chen, W., Liu, W.K., 2017. A framework for data-driven analysis of materials under uncertainty: Countering the curse of dimensionality. Comput. Methods Appl. Mech. Engrg. 320, 633–667.

Bourdin, B., 2001. Filters in topology optimization. Internat. J. Numer. Methods Engrg. 50 (9), 2143–2158. http://dx.doi.org/10.1002/nme.116.

Boyd, S.P., Vandenberghe, L., 2004. Convex Optimization. Cambridge University Press.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M.J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., Zhang, Q., 2018. JAX: composable transformations of python+numpy programs. URL http://github.com/google/jax.

Braides, A., 1994. Loss of polyconvexity by homogenization. Arch. Ration. Mech. Anal. 127 (2), 183–190. http://dx.doi.org/10.1007/bf00377660.

Chandrasekhar, A., Sridhara, S., Suresh, K., 2023. Graded multiscale topology optimization using neural networks. Adv. Eng. Softw. 175, 103359. http://dx.doi.org/10.1016/j.advengsoft.2022.103359.

Chen, Y., Shi, Y., Zhang, B., 2019. Optimal control via neural networks: A convex approach. In: International Conference on Learning Representations. URL https://openreview.net/forum?id=H1MW72AcK7.

Chi, H., Zhang, Y., Tang, T.L.E., Mirabella, L., Dalloro, L., Song, L., Paulino, G.H., 2021. Universal machine learning for topology optimization. Comput. Methods Appl. Mech. Engrg. 375, 112739. http://dx.doi.org/10.1016/j.cma.2019.112739.

Coleman, B.D., Noll, W., 1959. On the thermostatics of continuous media. Arch. Ration. Mech. Anal. 4 (1), 97–128. http://dx.doi.org/10.1007/bf00281381.

Cybenko, G., 1989. Approximation by superpositions of a sigmoidal function. Math. Control Signals Systems 2 (4), 303–314. http://dx.doi.org/10.1007/bf02551274.

da Silva, G.A., Beck, A.T., Sigmund, O., 2019. Stress-constrained topology optimization considering uniform manufacturing uncertainties. Comput. Methods Appl. Mech. Engrg. 344, 512–537. http://dx.doi.org/10.1016/j.cma.2018.10.020.

DeepMind, Babuschkin, I., Baumli, K., Bell, A., Bhupatiraju, S., Bruce, J., Buchlovsky, P., Budden, D., Cai, T., Clark, A., Danihelka, I., Dedieu, A., Fantacci, C., Godwin, J., Jones, C., Hemsley, R., Hennigan, T., Hessel, M., Hou, S., Kapturowski, S., Keck, T., Kemaev, I., King, M., Kunesch, M., Martens, L., Merzic, H., Mikulik, V., Norman, T., Papamakarios, G., Quan, J., Ring, R., Ruiz, F., Sanchez, A., Sartran, L., Schneider, R., Sezener, E., Spencer, S., Srinivasan, S., Stanojević, M., Stokowiec, W., Wang, L., Zhou, G., Viola, F., 2020. The DeepMind JAX ecosystem. URL http://github.com/google-deepmind.

Feyel, F., 1999. Multiscale FE2 elastoviscoplastic analysis of composite structures. Comput. Mater. Sci. 16 (1–4), 344–354.

Gaynor, A.T., Meisel, N.A., Williams, C.B., Guest, J.K., 2014. Multiple-material topology optimization of compliant mechanisms created via polyjet three-dimensional printing. J. Manuf. Sci. Eng. 136 (6), http://dx.doi.org/10.1115/1.4028439.

Geers, M.G., Kouznetsova, V.G., Brekelmans, W., 2010. Multi-scale computational homogenization: Trends and challenges. J. Comput. Appl. Math. 234 (7), 2175–2182.

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

Groen, J.P., Sigmund, O., 2017. Homogenization-based topology optimization for high-resolution manufacturable microstructures. Internat. J. Numer. Methods Engrg. 113 (8), 1148–1163. http://dx.doi.org/10.1002/nme.5575.

Groen, J.P., Stutz, F.C., Aage, N., Bærentzen, J.A., Sigmund, O., 2020. De-homogenization of optimal multi-scale 3D topologies. Comput. Methods Appl. Mech. Engrg. 364, 112979. http://dx.doi.org/10.1016/j.cma.2020.112979.

Holzapfel, G.A., 2002. Nonlinear Solid Mechanics: a Continuum Approach for Engineering Science. Kluwer Academic Publishers Dordrecht.

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

Kalina, K.A., Gebhart, P., Brummund, J., Linden, L., Sun, W., Kästner, M., 2024. Neural network-based multiscale modeling of finite strain magneto-elasticity with relaxed convexity criteria. Comput. Methods Appl. Mech. Engrg. 421, 116739. http://dx.doi.org/10.1016/j.cma.2023.116739.

Kalina, K.A., Linden, L., Brummund, J., Kästner, M., 2023. FE ANN: an efficient data-driven multiscale approach based on physics-constrained neural networks and automated data mining. Comput. Mech. 71 (5), 827–851. http://dx.doi.org/10.1007/s00466-022-02260-0.

Kidger, P., Garcia, C., 2021. Equinox: neural networks in JAX via callable PyTrees and filtered transformations. In: Differentiable Programming workshop at Neural Information Processing Systems 2021.

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

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

Leon, S.E., Lages, E.N., de Araújo, C.N., Paulino, G.H., 2014. On the effect of constraint parameters on the generalized displacement control method. Mech. Res. Commun. 56, 123–129. http://dx.doi.org/10.1016/j.mechrescom.2013.12.009.

Linden, L., Klein, D.K., Kalina, K.A., Brummund, J., Weeger, O., Kästner, M., 2023. Neural networks meet hyperelasticity: A guide to enforcing physics. J. Mech. Phys. Solids 179, 105363. http://dx.doi.org/10.1016/j.jmps.2023.105363.

Loshchilov, I., Hutter, F., 2019. Decoupled weight decay regularization. In: International Conference on Learning Representations. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Morrey, C.B., 1952. Quasi-convexity and the lower semicontinuity of multiple integrals.. Pacific J. Math. 2 (4), 25–53.

Morrey, C.B., 1966. Multiple Integrals in the Calculus of Variations. Springer, Berlin Heidelberg, http://dx.doi.org/10.1007/978-3-540-69952-1.

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

Pantz, O., Trabelsi, K., 2008. A post-treatment of the homogenization method for shape optimization. SIAM J. Control Optim. 47 (3), 1380–1398. http://dx.doi.org/10.1137/070688900.

Rasmussen, C.E., Williams, C.K.I., 2006. Gaussian Processes for Machine Learning, vol. 2, (3), MIT press Cambridge, MA,

Sanders, E.D., Aguiló, M.A., Paulino, G.H., 2018a. Multi-material continuum topology optimization with arbitrary volume and mass constraints. Comput. Methods Appl. Mech. Engrg. 340, 798–823. http://dx.doi.org/10.1016/j.cma.2018.01.032.

Sanders, E.D., Pereira, A., Aguiló, M.A., Paulino, G.H., 2018b. PolyMat: an efficient matlab code for multi-material topology optimization. Struct. Multidiscip. Optim. 58 (6), 2727–2759. http://dx.doi.org/10.1007/s00158-018-2094-0.

Sanders, E.D., Pereira, A., Paulino, G.H., 2021. Optimal and continuous multilattice embedding. Sci. Adv. 7 (16), http://dx.doi.org/10.1126/sciadv.abf4838.

Schumacher, C., Bickel, B., Rys, J., Marschner, S., Daraio, C., Gross, M., 2015. Microstructures to control elasticity in 3D printing. ACM Trans. Graph. 34 (4), 1–13. http://dx.doi.org/10.1145/2766926.

Shin, S., Shin, D., Kang, N., 2023. Topology optimization via machine learning and deep learning: a review. J. Comput. Des. Eng. 10 (4), 1736–1766. http://dx.doi.org/10.1093/jcde/qwad072.

Simo, J.C., Miehe, C., 1992. Associative coupled thermoplasticity at finite strains: Formulation, numerical analysis and implementation. Comput. Methods Appl. Mech. Engrg. 98 (1), 41–104. http://dx.doi.org/10.1016/0045-7825(92)90170-o.

Svanberg, K., 1987. The method of moving asymptotes—a new method for structural optimization. Internat. J. Numer. Methods Engrg. 24 (2), 359–373. http://dx.doi.org/10.1002/nme.1620240207.

Systèmes, D., 2021. ABAQUS/Standard User's Manual, Version 2021. Dassault Systèmes Simulia Corp, United States.

Treloar, L.R.G., 1944. Stress-strain data for vulcanised rubber under various types of deformation. Trans. Faraday Soc. 40, 59. http://dx.doi.org/10.1039/tf9444000059.

Truby, R.L., Lewis, J.A., 2016. Printing soft matter in three dimensions. Nature 540 (7633), 371–378. http://dx.doi.org/10.1038/nature21003.

van der Schelling, M., Ferreira, B., Bessa, M., 2024. F3dasm: Framework for data-driven design and analysis of structures and materials. J. Open Source Soft. 9 (100), 6912.

Vatanabe, S.L., Lippi, T.N., Lima, C.R.d., Paulino, G.H., Silva, E.C., 2016. Topology optimization with manufacturing constraints: A unified projection-based approach. Adv. Eng. Softw. 100, 97–112. http://dx.doi.org/10.1016/j.advengsoft.2016.07.002.

Wang, F., Lazarov, B.S., Sigmund, O., 2010. On projection methods, convergence and robust formulations in topology optimization. Struct. Multidiscip. Optim. 43 (6), 767–784. http://dx.doi.org/10.1007/s00158-010-0602-y.

Wang, F., Lazarov, B.S., Sigmund, O., Jensen, J.S., 2014. Interpolation scheme for fictitious domain techniques and topology optimization of finite strain elastic problems. Comput. Methods Appl. Mech. Engrg. 276, 453–472. http://dx.doi.org/10.1016/j.cma.2014.03.021.

White, D.A., Arrighi, W.J., Kudo, J., Watts, S.E., 2019. Multiscale topology optimization using neural network surrogate models. Comput. Methods Appl. Mech. Engrg. 346, 1118–1135. http://dx.doi.org/10.1016/j.cma.2018.09.007.

Woldseth, R.V., Aage, N., Bærentzen, J.A., Sigmund, O., 2022. On the use of artificial neural networks in topology optimisation. Struct. Multidiscip. Optim. 65 (10), http://dx.doi.org/10.1007/s00158-022-03347-1.

Xia, L., Breitkopf, P., 2015. Multiscale structural topology optimization with an approximate constitutive model for local material microstructure. Comput. Methods Appl. Mech. Engrg. 286, 147–167. http://dx.doi.org/10.1016/j.cma.2014.12.018.

Xia, L., Breitkopf, P., 2016. Recent advances on topology optimization of multiscale nonlinear structures. Arch. Comput. Methods Eng. 24 (2), 227–249. http://dx.doi.org/10.1007/s11831-016-9170-7.

Xu, W., Jambhulkar, S., Zhu, Y., Ravichandran, D., Kakarla, M., Vernon, B., Lott, D.G., Cornella, J.L., Shefi, O., Miquelard-Garnier, G., Yang, Y., Song, K., 2021. 3D printing for polymer/particle-based processing: A review. Composites B 223, 109102. http://dx.doi.org/10.1016/j.compositesb.2021.109102.

Yi, J., Bessa, M.A., 2023. Rvesimulator: An automated representative volume element simulator for data-driven material discovery. In: AI for Accelerated Materials Design - NeurIPS 2023 Workshop. URL https://openreview.net/forum?id=511z1DGjPi.

Yvonnet, J., Monteiro, E., He, Q.C., 2013. Computational homogenization method and reduced database model for hyperelastic heterogeneous structures. Int. J. Multiscale Comput. Eng. 11 (3), 201–225. http://dx.doi.org/10.1615/intjmultcompeng.2013005374.

Zhang, X.S., Chi, H., Paulino, G.H., 2020. Adaptive multi-material topology optimization with hyperelastic materials under large deformations: A virtual element approach. Comput. Methods Appl. Mech. Engrg. 370, 112976. http://dx.doi.org/10.1016/j.cma.2020.112976.

Zhou, M., Rozvany, G.I.N., 1991. The COC algorithm, Part II: Topological, geometrical and generalized shape optimization. Comput. Methods Appl. Mech. Engrg. 89 (1–3), 309–336. http://dx.doi.org/10.1016/0045-7825(91)90046-9.

Zhu, C., Gemeda, H.B., Duoss, E.B., Spadaccini, C.M., 2024. Toward multiscale, multimaterial 3D printing. Adv. Mater. 36 (34), http://dx.doi.org/10.1002/adma.202314204.