



HAL
open science

Micromechanics-Informed Parametric Deep Material Network for Physics Behavior Prediction of Heterogeneous Materials with a Varying Morphology

Tianyi Li

► **To cite this version:**

Tianyi Li. Micromechanics-Informed Parametric Deep Material Network for Physics Behavior Prediction of Heterogeneous Materials with a Varying Morphology. 2023. hal-04212612v3

HAL Id: hal-04212612

<https://hal.science/hal-04212612v3>

Preprint submitted on 14 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Micromechanics-Informed Parametric Deep Material Network for Physics Behavior Prediction of Heterogeneous Materials with a Varying Morphology

Tianyi Li^a

^a*Dassault Systèmes, 10 rue Marcel Dassault, 78140 Vélizy-Villacoublay, France*

Abstract

Deep Material Network (DMN) has recently emerged as a data-driven surrogate model for heterogeneous materials. Given a particular microstructural morphology, the effective linear and nonlinear behaviors can be successfully approximated by such physics-based neural-network like architecture. In this work, a novel micromechanics-informed parametric DMN (MIpDMN) architecture is proposed for multiscale materials with a varying microstructure characterized by several parameters. A single-layer feedforward neural network is used to account for the dependence of DMN fitting parameters on the microstructural ones. Micromechanical constraints are prescribed both on the architecture and the outputs of this new neural network. The proposed MIpDMN is also recast in a multiple physics setting, where physical properties other than the mechanical ones can also be predicted. In the numerical simulations conducted on three parameterized microstructures, MIpDMN demonstrates satisfying generalization capabilities when morphology varies. The effective behaviors of such parametric multiscale materials can thus be predicted and encoded by MIpDMN with high accuracy and efficiency.

Keywords: Deep material network, Parameterized microstructures, Structure-property relationships, Neural network, Machine learning

1. Introduction

The effective macroscopic behaviors of heterogeneous materials can be predicted by the physical properties of the constituents and the underlying microstructures using a multiscale modeling approach [1, 2, 3]. Among such models, mean-field homogenization schemes rely on the micromechanical assumptions (inclusion shapes, interaction between the constituents) of the microstructure being considered and are frequently used to predict linear and nonlinear behaviors of specific composites (such as the fiber-reinforced plastics) [4, 5]. Thanks to their analytical nature, they are computationally efficient and can be used as a local constitutive model on each integration point for concurrent structural-scale simulation of industrial components. However, their inherent simplifying hypotheses on the idealized microstructure may not be appropriate for other complex morphologies. In this case, full-field simulation of the micromechanical problem would be necessary, following a computational homogenization approach [6]. For instance, finite element (FE) simulation of a Representative Volume Element (RVE) of the underlying microstructure can be carried out to compute the local solution fields. The accurate linear and nonlinear effective responses can then be obtained through field averaging. However, such full-field method involves much higher computational cost and is impractical to be run concurrently in multiscale simulations on industrial parts.

Recently, the Deep Material Network (DMN) method [7] has emerged as a novel data-driven surrogate model for such multiscale materials. Based on linear elastic training data generated by high-fidelity computational homogenization (FE-RVE simulation, for instance), DMN is capable of predicting their elastic and inelastic behaviors with high accuracy and efficiency even in the finite-strain range. Compared to other data-driven approaches for material modeling [8] that rely on the (effective) strain-stress pair data, DMN learns instead the morphological characterizations of the microstructure via its homogenization function $(\mathbb{C}_1, \mathbb{C}_2) \mapsto \bar{\mathbb{C}}$. Here, $(\mathbb{C}_1, \mathbb{C}_2)$ are the stiffness tensors of the constituents and $\bar{\mathbb{C}}$ is the effective stiffness tensor. Due to its physics-based neural-network like architecture, DMN demonstrates great expressive power with much fewer fitting parameters compared to traditional machine learning methods. Thanks to these interesting properties, it is now gaining popularity in the computational mechanics community:

Email address: tianyi.li@3ds.com (Tianyi Li)

- In [9], DMN has been successfully applied for 3-d particle-reinforced composites, polycrystalline materials and woven composites involving three modeling scales. Nonlinear complex material behaviors such as hyperelasticity with stress softening and rate-dependent crystal plasticity are investigated under a finite-strain setting.
- A new rotation-free formulation along with a flattened architecture is proposed in [10]. It has been tested for short-fiber plastics and metal matrix composites with an elasto-plastic behavior in the matrix phase.
- Applications to short-fiber reinforced materials are also considered further in [11, 12, 13]. The thermomechanical behaviors of such composites can be successfully captured by DMN. Concurrent multiscale simulations combining a macroscopic finite element model and DMN on each integration point are performed for static and dynamic problems on structural components.
- More complex material behaviors such as viscoplasticity or failure are investigated in [14, 15, 16, 17]. New training strategies are also proposed to obtain more accurate nonlinear responses.
- In [18], other DMN architectures are explored with mean-field homogenization based building blocks. Applications to woven microstructures are considered. A generalization of the original DMN framework based on interactions between discrete material nodes is also proposed in [19, 20]. It has been successfully applied to porous microstructures with an elasto-plastic behavior.

Despite these successes indicated by numerical evidences, the expressive power of DMN remains to be fully understood from a theoretical perspective. In [10], its micromechanical justifications are provided in the context of generalized standard materials [21]. Thanks to the hierarchical architecture based on an appropriate laminate microstructure as the building block, DMN is thermodynamically consistent and verifies the micromechanical bounds on the effective linear and nonlinear behaviors. Actually, DMN can be identified as a multiple-rank laminate microstructure in homogenization theory [22, 23], in which each constituent of a laminate is also a laminate. Sequential laminates (also called coated laminates) are a subset among such hierarchical microstructures, where one of the constituents in the lamination process is always the same. It is well known that the Hashin-Shtrikman bounds [24] can be attained by such finite-rank sequential laminates. In the case of an effectively isotropic composite of two isotropic phases, such bounds can be realized by a rank-3 laminate in 2-d or a rank-6 laminate for 3-d elasticity [25]. As noted in [23, pg. 294], sequential laminates can approximate the homogenization function $(\mathbb{C}_1, \mathbb{C}_2) \mapsto \bar{\mathbb{C}}$ associated with a two-phase microstructure to the second order in $\mathbb{C}_1 - \mathbb{C}_2$. Since sequential laminates are a proper subset of multi-rank laminates, these results can be naturally extended to DMN and hence provide some preliminary theoretical justifications of its expressive power.

In general, we are often dealing with a class of microstructures that present similar morphologies. These microstructures are characterized by several parameters defining the geometrical shapes of each constituent inside the RVE. For instance, the volume fraction parameter vf describes the overall relative volume of each constituent. Other parameters also exist, depending on the exact morphological characterization of the microstructure. These micromechanical parameters may have an important influence on the effective properties. Hence, it would be beneficial that DMN also captures such structure-property relationships with high accuracy and efficiency. In [26, 27], a transfer-learning based approach is proposed to construct a DMN database for such parameterized microstructures. Training of several DMN instances is performed sequentially following a pre-determined path in the parametric space. Piecewise linear interpolation is then used to include the dependence of DMN fitting parameters on the microstructural parameters. In their work, they only focus on microstructures that can be characterized by vf and a second-order orientation tensor [28].

Meanwhile, a regression-based approach is proposed in [11] for short-fiber reinforced materials with a constant fiber volume fraction. The fitting parameters of one single DMN instance are assumed dependent on the varying (principal) fiber orientation tensor (a_1, a_2, a_3) . The offline training is performed using a total loss function that aggregates multiple microstructures with different (a_1, a_2, a_3) values. The main objective of this paper is to extend this formulation to generic parameterized microstructures, with varying volume fractions and possibly other geometrical parameters. In particular, we investigate of the use of a single-layer feedforward neural network to account for the dependence of DMN parameters on the microstructural ones. Additional micromechanics-based constraints are included to improve generalization capability of our parametric DMN architecture.

Another objective of this paper is to recast DMN in a multiple physics setting. In the literature, DMN has been used extensively as a surrogate of the possibly nonlinear *mechanical* behaviors of heterogeneous materials. Other physical behaviors have not been explored except in [12], where a two-way coupled thermomechanical problem is considered. The temperature-dependent stress-strain behaviors as well as the mechanically-induced self-heating

are predicted. However, effective physical properties, such as thermal conductivity or the coefficient of thermal expansion, are not explicitly computed by DMN. In this paper, we demonstrate that physical properties other than the mechanical ones can also be accurately predicted by DMN, even though it is trained using isothermal linear elastic mechanical data. It should be noted that multiphysics coupling is not considered in this paper.

The paper is organized as follows. The original DMN formulation [7, 9] is first reviewed and then recast in a multiple physics setting in Sect. 2. Extensions to parameterized microstructures are then considered in Sect. 3, in which we introduce a new micromechanics-informed parametric DMN architecture. The proposed method is further evaluated on three parameterized microstructures in Sect. 4. The obtained numerical results highlight the effects of the introduced physical constraints. Finally, the conclusions from the current work are summarized in Sect. 5.

2. Deep Material Network

Instead of learning the macroscopic (nonlinear) strain-stress behaviors of a particular two-phase microstructure Ω , as shown in Fig. 1, DMN learns its linear elastic homogenization function

$$(\mathbb{C}_1, \mathbb{C}_2) \mapsto \bar{\mathbb{C}}. \quad (1)$$

The effective computation of Eq. (1) on complex microstructures Ω is in general realized by computational homogenization like FE-RVE methods. The homogenization function encodes in particular the microstructure morphology, which can be defined by the characteristic function

$$\chi(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} \in \text{Phase 1,} \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

In this regard, DMN is not a surrogate of the *effective behaviors* of the microstructure but the microstructure *per se*.

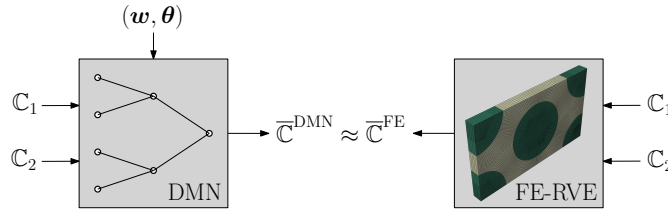


Figure 1: DMN learns the morphology of a particular microstructure Ω through its homogenization function.

2.1. Network architecture and fitting parameters

In this work, we follow the original formulation of [7, 9] based on a perfect binary tree architecture and a rotation-based formulation for laminates, with minor modifications that will be explained in the sequel.

DMN is a multiple-rank laminate microstructure as defined in [23, Chap. 9], comprised of hierarchically nested laminates of laminates on different length scales, see Fig. 2. The rank L of such multiple-rank laminates, also called the number of DMN layers, characterizes the number of nesting levels. Its architecture corresponds topologically to a perfect binary tree. Each “node” is a rank-1 laminate microstructure, serving as the “mechanistic building blocks” [7, 9, 10] or “neurons” in this neural-network like architecture.

On the input layer, each leaf laminate of the binary tree receives the linear elastic properties of the two phases $(\mathbb{C}_1, \mathbb{C}_2)$ expressed in their respective material frames. All these $N = 2^L$ entries are called DMN material nodes, since each of them will be associated with a constitutive model of the phase 1 or 2 for online multiple physics behavior prediction. The index set of the material nodes will be denoted \mathbb{I} . It can be partitioned evenly into two subsets for each of the phase.

$$\mathbb{I} = \{1, \dots, N\} = \mathbb{I}_1 \cup \mathbb{I}_2, \quad \mathbb{I}_1 = \{2j - 1 \mid j = 1, \dots, 2^{L-1}\}, \quad \mathbb{I}_2 = \{2j \mid j = 1, \dots, 2^{L-1}\}. \quad (3)$$

On the output layer, the effective properties of the microstructure $\bar{\mathbb{C}}$ are predicted through the root laminate and are given in the global frame of the microstructure Ω . The homogenization function $\text{DMN}_L : (\mathbb{C}_1, \mathbb{C}_2) \mapsto \bar{\mathbb{C}}$ of such L -layer DMN can be defined recursively as follows

$$\bar{\mathbb{C}} = \text{DMN}_L(\mathbb{C}_1, \mathbb{C}_2) = \begin{cases} \text{Lam}(\mathbb{C}_1, \mathbb{C}_2) & L = 1, \\ \text{Lam}(\text{DMN}_{L-1}(\mathbb{C}_1, \mathbb{C}_2), \text{DMN}_{L-1}(\mathbb{C}_1, \mathbb{C}_2)) & L > 1. \end{cases} \quad (4)$$

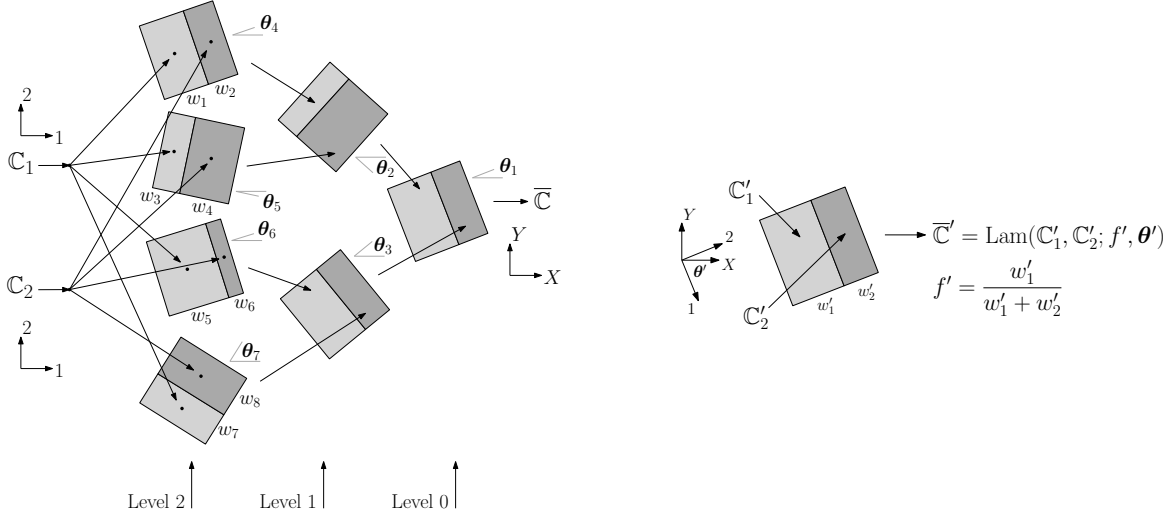


Figure 2: An L -layer DMN architecture can be represented by a perfect binary tree with each “node” being a rank-1 laminate microstructure. The DMN material nodes $1 \leq i \leq N = 2^L$ are represented by dots \cdot on the input layer and receive the linear elastic properties $(\mathbb{C}_1, \mathbb{C}_2)$ of the two phases. In this example with 3 DMN layers, three nesting levels is involved in order to compute the homogenization function $\text{DMN}_3 : (\mathbb{C}_1, \mathbb{C}_2) \mapsto \bar{\mathbb{C}}$. The fitting parameters are the weights \mathbf{w} defined on the leaf laminates and the rotations $\boldsymbol{\theta}$ on each laminate.

In Eq. (4), each Lam denotes the homogenization function of the rank-1 laminate associated with different weights (w'_1, w'_2) and rotations $\boldsymbol{\theta}'$, see Fig. 2. Compared to the original formulation [7], we omit the material rotation matrices for phase constitutive properties on the input layer. Note that by omitting these input rotations, the material frames of the input properties will actually coincide with the local laminate frames.

A rank-1 laminate is a microstructure in which its characteristic function Eq. (2) only varies in the direction of lamination $\chi(\mathbf{x}) = \chi(\mathbf{x} \cdot \mathbf{n})$. In 3-d, its homogenization function $\text{Lam} : (\mathbb{C}'_1, \mathbb{C}'_2) \mapsto \bar{\mathbb{C}}'$ can thus be characterized by only two parameters: the volume fraction f' of one of the phases, and a 3×3 rotation matrix $\mathbf{R}(\boldsymbol{\theta}') \in \text{SO}(3)$, where $\text{SO}(3)$ is the special orthogonal group acting on \mathbb{R}^3 . The primes on the parameters and on the input and output tensors indicate that they don't represent necessarily the properties of the constituents (except for the input layer), nor the effective behaviors of the microstructure (except for the output layer). The constituent properties $(\mathbb{C}'_1, \mathbb{C}'_2)$ are expressed in the laminate frame $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$, see Fig. 2. The lamination direction \mathbf{n} aligns with this local frame: \mathbf{e}_2 for 2-d microstructures [7], and \mathbf{e}_3 for 3-d ones [9]. The homogenized behaviors of the laminate are first computed locally, then rotated to the frame of another laminate on the next nesting level or the global frame for the laminate on level 0. Thanks to the simplifying geometry, the homogenization function Lam is analytical.

In summary, the fitting parameters of an L -layer DMN are

- Weights $\mathbf{w} = (w_1, \dots, w_N)$, defined on the leaf laminates. Its length $N = 2^L$ is the number of DMN material nodes. They are required to be non-negative, hence an adequate nonlinear activation function σ is applied

$$\mathbf{w} = \sigma(\mathbf{z}) \geq 0, \quad \mathbf{z} \in \mathbb{R}^N. \quad (5)$$

ReLU $\sigma(z) = \max(0, z)$ is used in [7, 9, 10, 11, 18], while Softplus $\sigma(z) = \log(1 + \exp(\beta z))/\beta$ is used in [19, 20] as a smooth approximation to ReLU.

These weights are propagated from the leaf laminates to the root so that each laminate receives different (w'_1, w'_2) , see Fig. 2. Denoting the weights on level $0 \leq i < L$ by $\mathbf{w}'^{(i)}$, with $\mathbf{w}'^{(L-1)} = \mathbf{w}$ the weights on the leaf laminates, we have

$$\mathbf{w}'_j^{(i)} = \mathbf{w}'_{2j-1}^{(i+1)} + \mathbf{w}'_{2j}^{(i+1)}, \quad j = 1, \dots, 2^{i+1}. \quad (6)$$

For each laminate, the volume fraction f' can then be computed

$$f' = \frac{w'_1}{w'_1 + w'_2}. \quad (7)$$

Due to Eq. (6) and Eq. (7), the DMN homogenization function Eq. (4) is invariant with respect to a scaling on DMN weights $\mathbf{w} \mapsto k\mathbf{w}$ for $k > 0$.

- Rotations $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_{2^L-1})$, defined for each laminate. Its length $2^L - 1$ equals to the number of “nodes” in a perfect binary tree. Its elements can be organized using breadth-first ordering like in Fig. 2. Each element $\boldsymbol{\theta}_i$ represents a 3×3 rotation matrix $\mathbf{R}(\boldsymbol{\theta}_i)$. In [9, 18], the three Euler angles are used. In this work however, we use the quaternion representation which avoids singularity issues with Euler angles [29]. Hence, $\boldsymbol{\theta}$ is a $(2^L - 1) \times 4$ matrix.

The total number of fitting parameters can be found in Tab. 1.

L -layer DMN	Weights	Rotations	Total
Number of fitting parameters	$N = 2^L$	$4 \times (2^L - 1)$	$5 \times 2^L - 4$

Table 1: Number of fitting parameters for an L -layer DMN, using the network architecture explained in Sect. 2.1 with in particular the quaternion representation for the rotation matrices.

The training of these fitting parameters follows a data-driven approach, by comparing the DMN prediction of the homogenized stiffness tensor and that predicted by computational homogenization, see Fig. 1. The machine learning procedure is described in [9, 10] and will be presented in Sect. 3.5 for our parametric DMN architecture.

After training, DMN not only provides an accurate and efficient surrogate of the linear homogenization function Eq. (4), it serves also as a surrogate microstructure to predict its nonlinear behaviors. In Appendix A, the nonlinear online prediction architecture of DMN [7] is recalled. In addition, we introduce and numerically investigate the use of acceleration methods [30] to further improve computational efficiency.

2.2. Multiple physics property prediction

As noted at the beginning of Sect. 2 and in Fig. 1, DMN should not be regarded as a surrogate of the *effective mechanical behaviors* but the microstructure *per se*. It is hence desirable that other physical properties can be predicted along with the mechanical ones at the online prediction stage. In this paper, we propose to consider the following two additional physical properties

1. Thermal conductivity \mathbf{k} , which is a symmetric second-order tensor for anisotropic behaviors.
2. Coefficient of thermal expansion $\boldsymbol{\alpha}$, also a symmetric second-order tensor in the anisotropic case.

It turns out that the online prediction of \mathbf{k} and $\boldsymbol{\alpha}$ requires only an adequate redefinition of the building block (neuron), denoted by Lam in Eq. (4). In particular, the DMN network architecture as well as the fitted parameters after offline training remain exactly the same. This property naturally recasts DMN in a multiple physics setting. Although each DMN neuron corresponds to a laminate microstructure, the exact formula of Lam will now depend on the physics being considered. Below, we will provide the definition of Lam for these additional physics properties. For completeness, its original definition [7, 9] for (linear elastic) mechanical behaviors is also included.

To ease the notation, we will now ignore the primes on the input and output tensors in the laminate homogenization function, see Fig. 1. For instance, for linear elasticity we now have

$$\bar{\mathbf{C}} = \text{Lam}_{\mathbf{C}}(\mathbf{C}_1, \mathbf{C}_2).$$

It should be remembered that the input tensors don’t necessarily refer to those of the two constituents (except for the input layer), and the output effective tensor is not necessarily the final effective property tensor of the microstructure (except for the output layer).

Linear elastic behaviors

The linear elastic behaviors of the laminate microstructure is governed by static equilibrium. Based on the lamination direction \mathbf{n} , the stress components can be partitioned into two parts: a tangential part $\boldsymbol{\sigma}^t$ and a normal part $\boldsymbol{\sigma}^n$. For example, if the lamination direction aligns with the local \mathbf{e}_3 vector, then in Mandel notation we have

$$\boldsymbol{\sigma}^t = (\sigma_{11}, \sigma_{22}, \sqrt{2}\sigma_{12}), \quad \boldsymbol{\sigma}^n = (\sigma_{33}, \sqrt{2}\sigma_{13}, \sqrt{2}\sigma_{23}). \quad (8)$$

The strain tensor as well as the stiffness tensors can be partitioned similarly. The interface condition prescribes that the normal part of the stress tensor $\boldsymbol{\sigma}^n$ is continuous, while the tangential part of the strain tensor $\boldsymbol{\varepsilon}^t$ is continuous. It can be encoded by the following matrix equation

$$\begin{bmatrix} \boldsymbol{\varepsilon}_1^t \\ \boldsymbol{\sigma}_1^n \end{bmatrix} = \begin{bmatrix} \boldsymbol{\varepsilon}_2^t \\ \boldsymbol{\sigma}_2^n \end{bmatrix} \implies \underbrace{\begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{C}_1^{nt} & \mathbf{C}_1^{nn} \end{bmatrix}}_{\bar{\mathbf{C}}_1} \begin{bmatrix} \boldsymbol{\varepsilon}_1^t \\ \boldsymbol{\sigma}_1^n \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{C}_2^{nt} & \mathbf{C}_2^{nn} \end{bmatrix}}_{\bar{\mathbf{C}}_2} \begin{bmatrix} \boldsymbol{\varepsilon}_2^t \\ \boldsymbol{\sigma}_2^n \end{bmatrix}, \quad (9)$$

where \mathbb{C}_1 and \mathbb{C}_2 are the 6×6 stiffness tensors of the two laminate phases in Mandel notation.

Using the definition of the effective strain tensor $\bar{\boldsymbol{\varepsilon}} = f\boldsymbol{\varepsilon}_1 + (1-f)\boldsymbol{\varepsilon}_2$, where f is the volume fraction of the phase 1, we obtain the following formula that computes $\boldsymbol{\varepsilon}_1$ from $\bar{\boldsymbol{\varepsilon}}$

$$((1-f)\widehat{\mathbb{C}}_1 + f\widehat{\mathbb{C}}_2)\boldsymbol{\varepsilon}_1 = \widehat{\mathbb{C}}_2\bar{\boldsymbol{\varepsilon}} \implies \boldsymbol{\varepsilon}_1 = \mathbb{A}\bar{\boldsymbol{\varepsilon}} = \widehat{\mathbb{C}}^{-1}\widehat{\mathbb{C}}_2\bar{\boldsymbol{\varepsilon}}, \quad \widehat{\mathbb{C}} = (1-f)\widehat{\mathbb{C}}_1 + f\widehat{\mathbb{C}}_2, \quad (10)$$

where $\mathbb{A} = \widehat{\mathbb{C}}^{-1}\widehat{\mathbb{C}}_2$ is the strain localization tensor.

Finally, with the effective stress tensor $\bar{\boldsymbol{\sigma}} = f\boldsymbol{\sigma}_1 + (1-f)\boldsymbol{\sigma}_2 = f\mathbb{C}_1\boldsymbol{\varepsilon}_1 + (1-f)\mathbb{C}_2\boldsymbol{\varepsilon}_2$, the effective stiffness tensor can be obtained as follows

$$\bar{\boldsymbol{\sigma}} = \bar{\mathbb{C}}\bar{\boldsymbol{\varepsilon}}, \quad \bar{\mathbb{C}} = f(\mathbb{C}_1 - \mathbb{C}_2)\mathbb{A} + \mathbb{C}_2. \quad (11)$$

The rotation can then be applied adequately for the fourth-order stiffness tensor $\bar{\mathbb{C}}$.

Thermal conductivity

Now, the neuron homogenization function computes the second-order effective (anisotropic) conductivity tensor $\bar{\mathbf{k}}$ from those of its constituents \mathbf{k}_1 and \mathbf{k}_2

$$\bar{\mathbf{k}} = \text{Lam}_{\mathbf{k}}(\mathbf{k}_1, \mathbf{k}_2).$$

The laminate microstructure is governed by steady-state heat equation and the Fourier's law $\mathbf{q} = -\mathbf{k}\nabla T$ that relates the temperature gradient ∇T to the heat flux \mathbf{q} through the conductivity tensor. As in Eq. (8), the heat flux and the temperature gradient can be decomposed into a tangential and a normal part

$$\mathbf{q}^t = (q_1, q_2), \quad \mathbf{q}^n = (q_3).$$

Similar to Eq. (9), the interface condition can be described by the following linear system

$$\begin{bmatrix} \nabla T_1^t \\ \mathbf{q}_1^n \end{bmatrix} = \begin{bmatrix} \nabla T_2^t \\ \mathbf{q}_2^n \end{bmatrix} \implies \underbrace{\begin{bmatrix} \mathbf{I}_{2 \times 2} & \mathbf{0}_{2 \times 1} \\ \mathbf{k}_1^{nt} & \mathbf{k}_1^{nn} \end{bmatrix}}_{\widehat{\mathbf{k}}_1} \begin{bmatrix} \nabla T_1^t \\ \nabla T_1^n \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{I}_{2 \times 2} & \mathbf{0}_{2 \times 1} \\ \mathbf{k}_2^{nt} & \mathbf{k}_2^{nn} \end{bmatrix}}_{\widehat{\mathbf{k}}_2} \begin{bmatrix} \nabla T_2^t \\ \nabla T_2^n \end{bmatrix}.$$

Following the same procedure as before, the effective conductivity tensor is given by

$$\bar{\mathbf{q}} = -\bar{\mathbf{k}}\bar{\nabla T}, \quad \bar{\mathbf{k}} = f(\mathbf{k}_1 - \mathbf{k}_2)\mathbb{A} + \mathbf{k}_2, \quad \mathbb{A} = \widehat{\mathbf{k}}^{-1}\widehat{\mathbf{k}}_2, \quad \widehat{\mathbf{k}} = (1-f)\widehat{\mathbf{k}}_1 + f\widehat{\mathbf{k}}_2. \quad (12)$$

The rotation can be applied to the second-order tensor $\bar{\mathbf{k}}$.

Coefficient of thermal expansion

The effective coefficient of thermal expansion (CTE), as a second-order tensor $\bar{\boldsymbol{\alpha}}$ in the anisotropic case, can be computed along with the effective stiffness tensor. Now, the laminate homogenization function becomes

$$(\bar{\mathbb{C}}, \bar{\boldsymbol{\alpha}}) = \text{Lam}_{\mathbb{C}, \boldsymbol{\alpha}}(\mathbb{C}_1, \mathbb{C}_2, \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2).$$

While $\bar{\mathbb{C}}$ is still given by Eq. (11), the effective CTE can be obtained by using the relationship [31, 32] between $\bar{\boldsymbol{\alpha}}$ and the effective compliance tensor $\bar{\mathbb{S}} = \bar{\mathbb{C}}^{-1}$. For completeness, its derivation is now recalled below.

Consider an arbitrary two-phase microstructure with fixed material orientation. It is subjected to a uniform temperature difference ΔT and a traction vector $\bar{\boldsymbol{\sigma}}\mathbf{n}$ on its boundary, such that the strain, stress and temperature fields are *uniform* inside the microstructure. Using the thermoelastic constitutive equation, we obtain

$$\boldsymbol{\varepsilon}_1 = \boldsymbol{\varepsilon}_2 = \mathbb{S}_1\bar{\boldsymbol{\sigma}} + \boldsymbol{\alpha}_1\Delta T = \mathbb{S}_2\bar{\boldsymbol{\sigma}} + \boldsymbol{\alpha}_2\Delta T \implies \bar{\boldsymbol{\sigma}} = (\mathbb{S}_1 - \mathbb{S}_2)^{-1}(\boldsymbol{\alpha}_2 - \boldsymbol{\alpha}_1)\Delta T. \quad (13)$$

In Eq. (13), similar to stiffness and compliance tensors, the second-order CTE tensors are also expressed in Mandel notation by a 6×1 vector. The expression $\mathbb{S}^{-1}\boldsymbol{\alpha} = \mathbb{C}\boldsymbol{\alpha}$ is hence understood as matrix multiplication similar to $\mathbb{C}\boldsymbol{\varepsilon}$.

The volume-averaged stress $\bar{\boldsymbol{\sigma}}_1$ of the phase 1 can be related to $\bar{\boldsymbol{\sigma}}$ and ΔT by using the mechanical stress concentration tensor \mathbb{B} and the thermal stress concentration tensor \mathbf{b}

$$\bar{\boldsymbol{\sigma}}_1 = \mathbb{B}\bar{\boldsymbol{\sigma}} + \mathbf{b}\Delta T. \quad (14)$$

Under the previous thermoelastic loading, using Eq. (13), we obtain hence

$$\mathbf{b} = (\mathbf{I}_{6 \times 6} - \mathbb{B})(\mathbb{S}_1 - \mathbb{S}_2)^{-1}(\boldsymbol{\alpha}_2 - \boldsymbol{\alpha}_1). \quad (15)$$

This universal relationship Eq. (15) is valid for arbitrary thermoelastic loading of the microstructure. It implies that the thermal stress concentration tensor \mathbf{b} can be uniquely determined as long as the mechanical one is given.

The effective thermoelastic constitutive equation for the microstructure is given by

$$\bar{\boldsymbol{\varepsilon}} = \bar{\mathbb{S}}\bar{\boldsymbol{\sigma}} + \bar{\boldsymbol{\alpha}}\Delta T. \quad (16)$$

With Eq. (14) under isothermal loading $\Delta T = 0$, the effective compliance tensor can be computed by

$$\bar{\mathbb{S}} = f(\mathbb{S}_1 - \mathbb{S}_2)\mathbb{B} + \mathbb{S}_2, \quad (17)$$

where f is the volume fraction of the phase 1 as before. Combining Eq. (14) to Eq. (17) with $\bar{\boldsymbol{\sigma}} = f\boldsymbol{\sigma}_1 + (1-f)\boldsymbol{\sigma}_2 = \mathbf{0}$, we obtain

$$\begin{aligned} \bar{\boldsymbol{\varepsilon}} &= \bar{\boldsymbol{\alpha}}\Delta T = f\boldsymbol{\varepsilon}_1 + (1-f)\boldsymbol{\varepsilon}_2 = (f\boldsymbol{\alpha}_1 + (1-f)\boldsymbol{\alpha}_2)\Delta T + f(\mathbb{S}_1 - \mathbb{S}_2)\boldsymbol{\sigma}_1 \\ &= (f\boldsymbol{\alpha}_1 + (1-f)\boldsymbol{\alpha}_2)\Delta T + (\bar{\mathbb{S}} - f\mathbb{S}_1 - (1-f)\mathbb{S}_2)(\mathbb{S}_1 - \mathbb{S}_2)^{-1}(\boldsymbol{\alpha}_1 - \boldsymbol{\alpha}_2)\Delta T. \end{aligned} \quad (18)$$

After some manipulations, we deduce that the effective CTE tensor reads

$$\bar{\boldsymbol{\alpha}} = \boldsymbol{\alpha}_1 + (\bar{\mathbb{S}} - \mathbb{S}_1)(\mathbb{S}_1 - \mathbb{S}_2)^{-1}(\boldsymbol{\alpha}_1 - \boldsymbol{\alpha}_2). \quad (19)$$

Note that Eq. (19) is not only valid for each laminate microstructure (DMN neuron) but also for the microstructure being approximated by DMN, as long as material orientation is fixed for both phases. In the latter case, first $\bar{\mathbb{C}}$ is predicted by using Eq. (4), then Eq. (19) can be directly applied using $\bar{\mathbb{S}} = \bar{\mathbb{C}}^{-1}$ computed by DMN. The only requirement is that the (effective) thermoelastic constituent properties are expressed in the same global frame.

When at least one of the phases contains varying local material orientation, Eq. (19) needs to be applied as a *neuron* operation similar to Eq. (11) and Eq. (12), from the input layer to the output layer. For each laminate microstructure, the following operations are expressed and performed in the same laminate frame:

1. Compute $\bar{\mathbb{C}}$ using Eq. (11).
2. Compute $\bar{\boldsymbol{\alpha}}$ using Eq. (19) and $\bar{\mathbb{S}} = \bar{\mathbb{C}}^{-1}$ just obtained.
3. Apply appropriate rotation operations for $\bar{\mathbb{C}}$ and $\bar{\boldsymbol{\alpha}}$.

Rotations must be applied at the end, since Eq. (19) is only valid when expressed in the same frame. The rotated $\bar{\mathbb{C}}$ and $\bar{\boldsymbol{\alpha}}$ then become the inputs for a laminate of the next nesting level (or directly the final DMN outputs).

3. Extension to parameterized microstructures

In general, we are often dealing with a class of morphologically similar microstructures, described by one or several parameters $\mathbf{p} = (p_1, p_2, \dots)$. In the case of the unidirectional fiber composite, illustrated in Fig. 3, they are parameterized by the single *vf* parameter which indicates the volume fraction of the fibers.

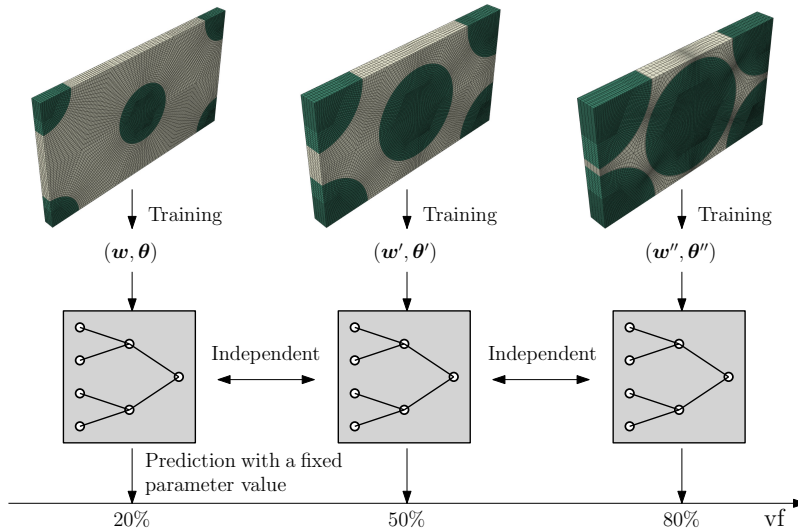


Figure 3: With the original formulation, each DMN is trained independently to learn a particular microstructure Ω_{p_i} .

The original DMN formulation [7] is not adapted for such parameterized microstructures $\mathbf{p} \mapsto \Omega_{\mathbf{p}}$, since each DMN is trained to learn a particular microstructure $\Omega_{\mathbf{p}_i}$ with a fixed parameter value \mathbf{p}_i . After independent training on such n microstructures, we only obtain a *discrete* set of DMN objects

$$\{\text{DMN}_{(\mathbf{w}_i, \boldsymbol{\theta}_i)}, \quad i = 1, \dots, n\}.$$

Each $\text{DMN}_{(\mathbf{w}_i, \boldsymbol{\theta}_i)}$ is tailored to approximate the homogenized behavior of a *particular* microstructure $\Omega_{\mathbf{p}_i}$. Without another training (which requires the costly computational homogenization data), it is not possible to predict the behavior of a new microstructure $\Omega_{\mathbf{p}'}$ that is absent in the previous set.

3.1. Transfer-learning based interpolative DMN

In [26], a transfer-learning strategy is proposed to interpolate different DMN models trained at different parameter values \mathbf{p}_i . The functional dependence $\mathbf{p} \mapsto (\mathbf{w}(\mathbf{p}), \boldsymbol{\theta}(\mathbf{p}))$ is defined by interpolating the DMN parameters $(\mathbf{w}_i, \boldsymbol{\theta}_i)$ between different microstructural parameters \mathbf{p}_i . This assumes implicitly that the same DMN architecture with the same number of layers is used. If each DMN is trained independently using random initialization of its fitting parameters, $\mathbf{p} \mapsto (\mathbf{w}(\mathbf{p}), \boldsymbol{\theta}(\mathbf{p}))$ would not be *smooth*, which leads to less accurate predictions when interpolating between known \mathbf{p}_i . This motivates hence a transfer-learning based training strategy, illustrated in Fig. 4.

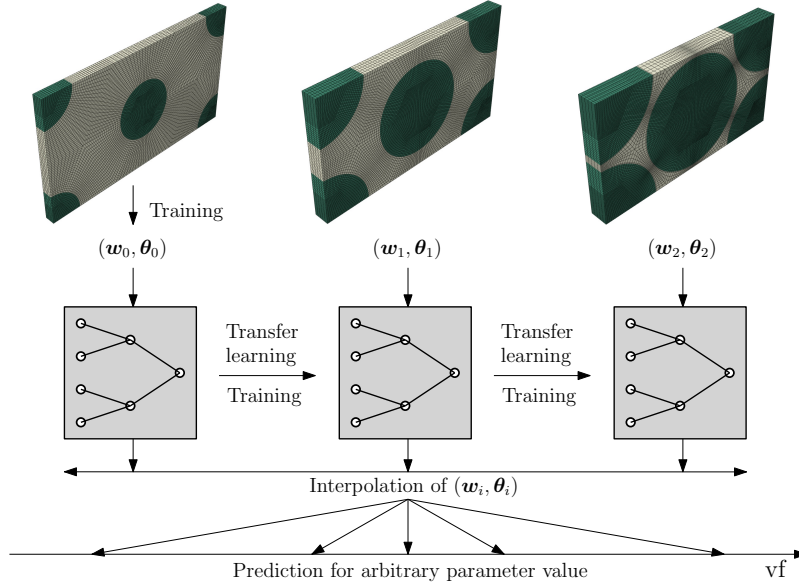


Figure 4: Transfer-learning based interpolative deep material network.

A *pre-determined sequential* path $(\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n)$ in the parametric space is required to carry out transfer-learning training on DMN parameters

$$\text{DMN}_{(\mathbf{w}_0, \boldsymbol{\theta}_0)} \rightarrow \text{DMN}_{(\mathbf{w}_1, \boldsymbol{\theta}_1)} \rightarrow \dots \rightarrow \text{DMN}_{(\mathbf{w}_n, \boldsymbol{\theta}_n)}.$$

The training of $(\mathbf{w}_i, \boldsymbol{\theta}_i)$ is initialized using previously trained $(\mathbf{w}_{i-1}, \boldsymbol{\theta}_{i-1})$, for $i > 0$. Only the first training at \mathbf{p}_0 is performed using standard random initialization. Transfer learning not only ensures a smooth interpolation between DMN parameters, it also accelerates training of subsequent DMN models for new microstructural parameters \mathbf{p}_i .

While this transfer-learning based approach has been tested with success for microstructures depending solely on the volume fraction parameter [26], and those depending on the volume fraction and the orientation tensor [27]. In our opinion it has also some limitations:

- Definition of a *pre-determined training sequence* may become difficult especially for higher parametric dimensions. For microstructures with multiple geometrical parameters of different natures, it is not trivial to choose the starting point \mathbf{p}_0 as well as the sequential interpolation points \mathbf{p}_i . Furthermore, the expressive power of DMN in the parametric space may also depend on the chosen training path $(\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n)$. The number of DMN active material nodes (those with a positive weight) may decrease in the process of transfer learning with the ReLU activation function. An example is given in [26].

- Extrapolation especially for higher parametric dimensions. Based on interpolation points, it would be difficult to extrapolate “outside” the training domain. In [26, 27] for instance, extrapolation in the parametric space is not considered.
- Computational efficiency concerns. Since transfer learning is sequential in nature, DMN at different microstructural parameters can only be trained one after another. Also, each DMN instance associated with its parameters needs to be stored for interpolation.

3.2. Micromechanics-informed parametric DMN

In this work, we propose a novel micromechanics-informed parametric DMN (MIpDMN) architecture dedicated for parameterized microstructures. The functional dependence of DMN parameters ($\mathbf{w}, \boldsymbol{\theta}$) on the microstructural parameters \mathbf{p} is now directly accounted for by a single-layer feedforward neural network. We obtain thus

$$\mathbf{w}(\mathbf{p}) = \sigma(\mathbf{W}_1 \mathbf{p} + \mathbf{w}_0), \quad (20a)$$

$$\boldsymbol{\theta}(\mathbf{p}) = \boldsymbol{\Theta}_1 \mathbf{p} + \boldsymbol{\theta}_0, \quad (20b)$$

where \mathbf{W}_1 and $\boldsymbol{\Theta}_1$ are “weight” tensors of appropriate dimensions, \mathbf{w}_0 and $\boldsymbol{\theta}_0$ are “bias” tensors and σ denotes a nonlinear activation function as in Eq. (5). Several remarks are in order:

- The tensors \mathbf{W}_1 and $\boldsymbol{\Theta}_1$ characterize the dependence of DMN parameters \mathbf{w} and $\boldsymbol{\theta}$ on microstructural parameters, while \mathbf{w}_0 and $\boldsymbol{\theta}_0$ remain constant while microstructural parameters change.
- When the microstructural parameters have little effect on the effective behaviors of the microstructures, we would have $\mathbf{W}_1 = \boldsymbol{\Theta}_1 = \mathbf{0}$ and thus recover the original DMN formulation [7]. The “bias” parameters \mathbf{w}_0 and $\boldsymbol{\theta}_0$ solely are needed to approximate the homogenized behavior of this parameterized microstructure.
- Due to the presence of a nonlinear activation function applied after affine transformation, the functional dependence of the DMN weights Eq. (20a) is different from piecewise linear interpolation used in the transfer-learning based interpolative DMN approach [26, 27]. Even when the ReLU activation function is used, Eq. (20a) implies that each w_i is *individually* piecewise linear with different slope-changing points. In [26, 27] however, linear interpolation is performed *globally* on the DMN weights vector \mathbf{w} .
- Similar functional dependence for the DMN rotations Eq. (20b) has been proposed in [11]. They have shown that an affine dependence works best in terms of accuracy and generalization capabilities, in comparison with other nonlinear functions. Hence, hidden layers as well as nonlinear activation functions are not considered for $\boldsymbol{\theta}(\mathbf{p})$ in this work.

The neural network proposed in Eq. (20) follows a *fully-connected* architecture in the sense that the DMN weights \mathbf{w} and rotations $\boldsymbol{\theta}$ depend on *all* the microstructural parameters. Meanwhile, several findings in the literature motivate a *micromechanics-informed* (MI) architecture by separating the dependence of DMN weights and rotations:

- For parameterized microstructures with a single *volume fraction* parameter, the naïve approach, as proposed by [26] in which only the DMN weights vary with *vf* and the DMN rotations remain constant, actually predicts physically plausible *vf*-dependence of the homogenized behaviors.
- For microstructures with parameters that do not change the volume fraction, such as for short-fiber reinforced composites with a fixed volume fraction but different fiber orientations, [11] indicates that only the DMN rotations need to vary with such parameters, with a constant DMN weights vector.

It should be noted that these results originate directly from the micromechanics-based design of DMN [10]. In order that the parametric DMN reflects these micromechanical properties, the microstructural parameters are partitioned

$$\mathbf{p} = (\text{vf}, \mathbf{q}) \in \mathbb{R} \times \mathbb{R}^q, \quad (21)$$

where *vf* is the volume fraction of the phase 2, while \mathbf{q} denotes all other q *independent* purely morphological parameters that are *orthogonal to* *vf*. Given the partition Eq. (21), the MI architecture is now given by

$$\mathbf{w}(\mathbf{p}) = \mathbf{w}(\text{vf}) = \sigma(\text{vf} \cdot \mathbf{w}_1 + \mathbf{w}_0), \quad (22a)$$

$$\boldsymbol{\theta}(\mathbf{p}) = \boldsymbol{\theta}(\mathbf{q}) = \boldsymbol{\Theta}_1 \mathbf{q} + \boldsymbol{\theta}_0, \quad (22b)$$

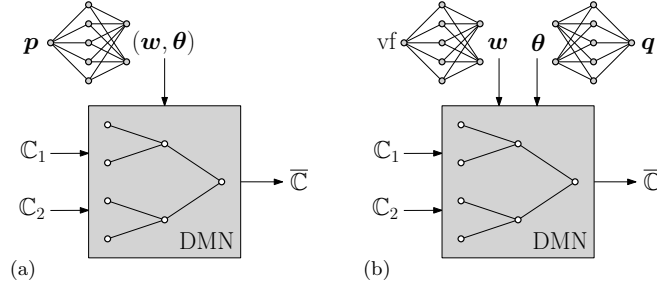


Figure 5: Parametric deep material network: (a) fully-connected architecture; (b) micromechanics-informed one.

where \mathbf{w}_0 and \mathbf{w}_1 are vectors of length N , $\boldsymbol{\theta}_0$ is a $(2^L - 1) \times 4$ matrix and $\boldsymbol{\Theta}_1$ is a $(2^L - 1) \times 4 \times q$ tensor. The fully-connected architecture Eq. (20) and the MI one Eq. (22) are compared in Fig. 5.

The partition Eq. (21) can be made more mathematically precise through the n -point correlation functions [33] which provide a hierarchy of statistical descriptions of the microstructure. The vf parameter only quantifies the relative volume proportion of each phase and corresponds to the 1-point correlation function. The other q purely morphological parameters do not change the current volume fraction (which is specified by vf) and contain information from higher-order correlation functions (2-point, 3-point, etc.). The *orthogonal* partition can be achieved through a principal component analysis [34, 35, 36] on such correlation functions. The vf parameter can be extracted as a major principal feature orthogonal to other principal parameters \mathbf{q} that are purely morphological.

The advantages of the MI architecture Eq. (22) compared to the fully-connected one is two-fold. Firstly, fewer fitting parameters are required due to the partition Eq. (21), which increases computational efficiency. Secondly, as we shall see through numerical examples, it provides comparable expressive power and may also enhance generalization ability in the parametric space.

Compared to the transfer learning-based interpolative DMN described in Sect. 3.1, a *unique* offline training is now required to optimize the fitting parameters of MIpDMN, see Fig. 6. The expressive power of DMN is evaluated *jointly* using the linear elastic behavior data at each \mathbf{p}_i in the parametric space. Furthermore, a neural-network functional dependence naturally defines interpolation and extrapolation inside or outside the training domain and extends easily to higher parametric dimensions.

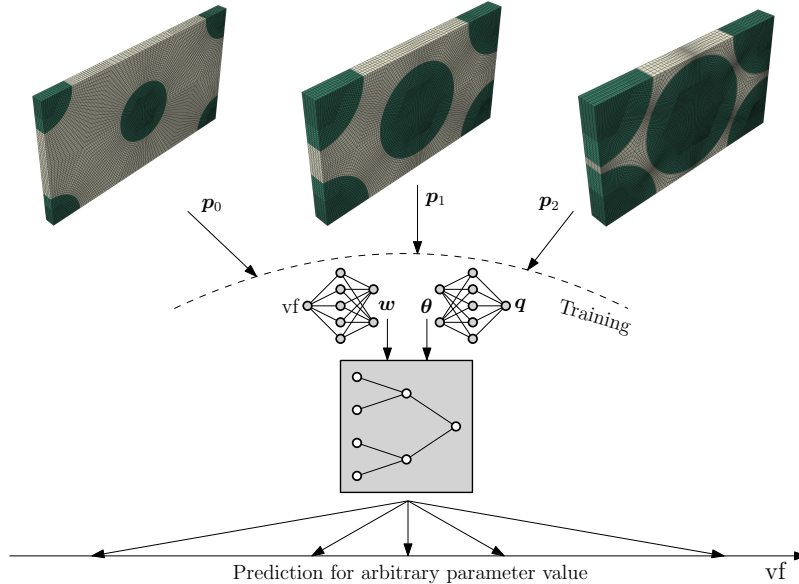


Figure 6: Offline training for MIpDMN based on the linear elastic behavior data at each \mathbf{p}_i in the parametric space.

This micromechanics-informed parametric DMN will be denoted in the sequel by MIpDMN. To summarize, it is defined by:

- DMN homogenization function Eq. (4) which learns the morphology of a particular microstructure $\Omega_{\mathbf{p}}$.

- Micromechanics-informed single-layer feedforward neural network Eq. (22) which learns the functional dependence of DMN parameters on microstructural parameters \mathbf{p} . It contains the fitting parameters of MIpDMN: $(\mathbf{w}_0, \mathbf{w}_1, \theta_0, \Theta_1)$.

With an L -layer architecture, the number of fitting parameters can be found in Tab. 2. The offline training of these fitting parameters follows a data-driven machine learning approach which will be described in Sect. 3.5.

L -layer MIpDMN	Weights	Rotations
Number of fitting parameters	$2N = 2^{L+1}$	$4(q+1) \times (2^L - 1)$

Table 2: Fitting parameters and their numbers for an L -layer MIpDMN.

3.3. Volume fraction constraint

Apart from the MI architecture on the functional dependence $\mathbf{p} \mapsto (\mathbf{w}, \theta)$, some physical constraints can also be prescribed on the outputs of this neural network to further improve its generalization ability, similar to the physics-informed machine learning approach for nonlinear partial differential equations [37].

It is now well known that the DMN weights vector \mathbf{w} reflects the actual volume fraction information of the microstructure Ω being considered [7, 10]. Specifically, the volume fraction of the phase 2 learned by DMN is given by

$$\mathbf{vf}(\mathbf{w}) = \frac{\sum_{i \in \mathbb{I}_2} w_i}{\sum_{i \in \mathbb{I}} w_i} \approx \mathbf{vf}_\Omega. \quad (23)$$

In [16], this physical property is explicitly included in the loss function as an additional constraint in learning a particular microstructure. Using our MI functional dependence Eq. (22a), Eq. (23) can be further enforced at *all volume fraction* values

$$\mathbf{vf}(\mathbf{w}) = \mathbf{vf}(\sigma(\mathbf{vf} \cdot \mathbf{w}_1 + \mathbf{w}_0)) = \mathbf{vf}, \quad \forall \mathbf{vf} \in [0, 1]. \quad (24)$$

This volume fraction constraint is prescribed on DMN weights \mathbf{w} .

Due to the presence of a nonlinear activation function σ , Eq. (24) is nonlinear and the unknowns $(\mathbf{w}_0, \mathbf{w}_1)$ can not be solved explicitly. Furthermore, and luckily enough since it contributes to the expressive power of MIpDMN, there may exist multiple solutions that satisfy Eq. (24). Anticipating the fact that this constraint will also be included during the offline training of our MIpDMN, we adopt a machine-learning approach to weakly enforce this volume fraction constraint through a loss function

$$\mathcal{L}_{\mathbf{vf}} = \frac{1}{n_{\mathbf{vf}}} \sum_{i=1}^{n_{\mathbf{vf}}} (\mathbf{vf}(\sigma(\mathbf{vf}_i \cdot \mathbf{w}_1 + \mathbf{w}_0)) - \mathbf{vf}_i)^2, \quad (25)$$

where \mathbf{vf}_i denotes the collocation points at which Eq. (24) is weakly prescribed and $n_{\mathbf{vf}}$ is the number of such collocation points. The sampling of these collocation points, the initialization of $(\mathbf{w}_0, \mathbf{w}_1)$ and the optimization algorithm will be described together with the offline training of MIpDMN in Sect. 3.5.

Using 5 DMN layers and the ReLU activation function, an example of the minimization of Eq. (25) is shown in Fig. 7. The loss function is decreasing, which demonstrates that our MI functional dependence Eq. (22a) of \mathbf{w} is capable of satisfying the volume fraction constraint Eq. (24). In this example, after about 200 epochs, the maximum absolute error of $|\mathbf{vf}(\mathbf{w}) - \mathbf{vf}|$ in $[0, 1]$ becomes approximately 0.5%.

Another physical information that can be extracted from Eq. (22a) is the number of active DMN material nodes. For arbitrary $\mathbf{vf} \in [0, 1]$, the number of active nodes for the phase $p \in \{1, 2\}$ is given by

$$\text{card}\{i \in \mathbb{I}_p \mid w_i(\mathbf{vf}) > 0\},$$

where $\text{card}(\cdot) = |\cdot|$ is the cardinality of a set. In practice, machine epsilon is used instead of 0. The ratio of active DMN nodes for each phase can be obtained by dividing their respective number of active nodes by $\frac{1}{2}N$, which is the number of total material nodes per phase. In the whole parametric space, the number of *globally* active nodes is defined by

$$\text{card}\{i \in \mathbb{I} \mid w_i(\mathbf{vf}) > 0, \quad \forall \mathbf{vf} \in [0, 1]\}.$$

In Fig. 8, these quantities are plotted as a function of \mathbf{vf} , the volume fraction of the phase 2. In order to satisfy Eq. (24), the ratio of active nodes of the phase 1 decreases from 100% to less than 70%, while for the phase 2 this ratio increases from less than 60% up to more than 90%. Our MI functional dependence Eq. (22a) of \mathbf{w} is hence capable of adapting the weights of each phase as a function of the microstructure volume fraction.

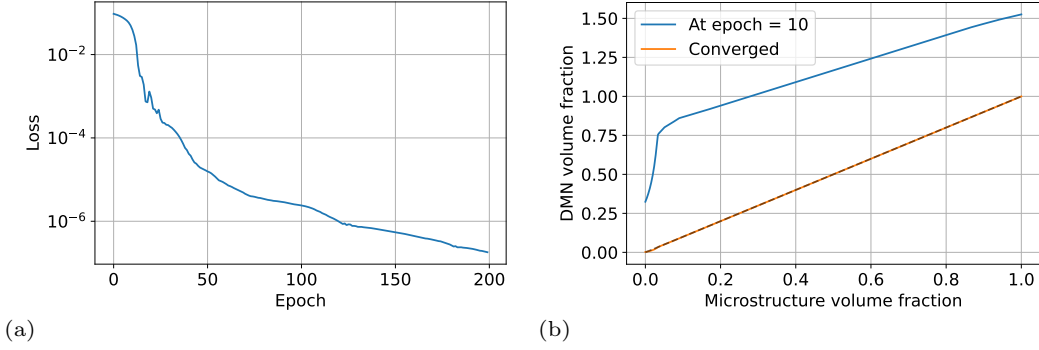


Figure 7: Volume fraction constraint prescribed on $\mathbf{w}(\mathbf{p})$: (a) minimization of the loss function Eq. (25); (b) predicted DMN volume fraction at the early stage and at the end of training.

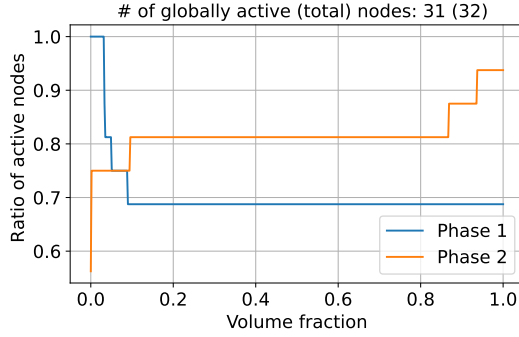


Figure 8: Variation of the ratio of active DMN nodes with the volume fraction of the phase 2.

3.4. Orientation constraint

DMN not only captures the microstructure morphology, it also learns material orientation distribution function $\Omega \rightarrow \text{SO}(3)$ in the microstructure [9]. Using orientation tensors, we propose an orientation constraint to be prescribed on DMN weights \mathbf{w} and rotations $\boldsymbol{\theta}$, in order that the parametric DMN generalizes such material orientation knowledge in the whole parametric space.

Orientation tensors [28] describe concisely the statistical information of the orientation distribution of *unit vectors*. Given an orientation distribution function $f : \mathbb{S}^2 \rightarrow \mathbb{R}$, where \mathbb{S}^2 denotes the two-dimensional surface of a unit sphere, the 3×3 second-order orientation tensor is defined by

$$\mathbf{a} = \int_{\mathbb{S}^2} f(\mathbf{e}) \mathbf{e} \otimes \mathbf{e} \, dS, \quad \|\mathbf{e}\| = 1, \quad (\mathbf{e} \otimes \mathbf{e})_{ij} = e_i e_j. \quad (26)$$

It can be easily shown that \mathbf{a} is symmetric and $\text{tr}(\mathbf{a}) = 1$, due to the normalization constraints of \mathbf{e} and of the probability density f . Higher order orientation tensors do exist [38], however the second-order one Eq. (26) is the most frequently used to characterize local fiber orientations due to manufacturing processes and their influence on material properties [39, 40].

In this work, we propose a generalization of such (second-order) orientation tensors for orientation distributions of *rotations*. Contrary to transversely isotropic fibers for which a single unit vector suffices to characterize its material frame, general anisotropic materials (like orthotropic ones) require a rotation matrix \mathbf{R} to describe the transformation from its material frame $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ to the global one. In such cases, the orientation distribution function is now defined for rotation matrices $f : \text{SO}(3) \rightarrow \mathbb{R}$. Such probability density function can also be found for the texture analysis of polycrystalline materials [41] or rigid body dynamics [42]. Given that each column $\mathbf{R}^{(i)}$ of \mathbf{R} essentially expresses $\mathbf{e}_i \in \mathbb{S}^2$ in the global frame, it defines an orientation distribution of the material frame, cf. Fig. 9 for the tows in this woven microstructure.

Given this interpretation, three (second-order) orientation tensors can be defined for each axis of the material frame

$$\mathbf{a}^{(i)} = \int_{\text{SO}(3)} f(\mathbf{R}) \mathbf{R}^{(i)} \otimes \mathbf{R}^{(i)} \, dR, \quad (27)$$

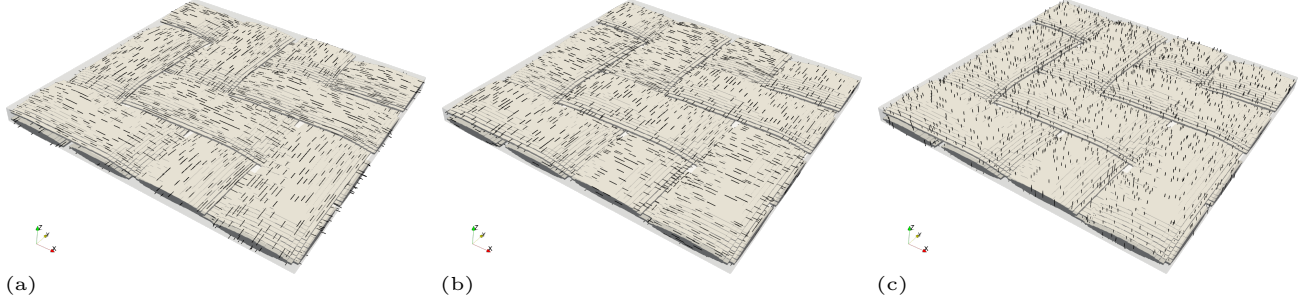


Figure 9: Orientation distribution of the material frame of the tows in the woven microstructure: (a) \mathbf{e}_1 ; (b) \mathbf{e}_2 ; (c) \mathbf{e}_3 .

where dR is the volume element for $SO(3)$ and Einstein summation is not implied. For discrete probability functions defined on a mesh, the integral can be understood as weighted-averaging using the element volumes as weights. Note that in the case of a two-phase microstructure, Eq. (27) can be computed for each of the phase. Each of these orientation tensors satisfies $\text{tr}(\mathbf{a}^{(i)}) = 1$. Due to the orthonormality of $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$, we also have

$$\mathbf{a}^{(1)} + \mathbf{a}^{(2)} + \mathbf{a}^{(3)} = \begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix}.$$

Using Eq. (27) in the case of Fig. 9, the material frame orientation tensors for the tows are given by

$$\mathbf{a}^{(1)} = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{a}^{(2)} = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{a}^{(3)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (28)$$

The tows are thus isotropically oriented in the X - Y plane for \mathbf{e}_1 and \mathbf{e}_2 , and unidirectionally oriented in the Z axis for \mathbf{e}_3 . For the matrix phase in Fig. 9, its material frame coincides with the global frame. In this case, \mathbf{a} is similar to the unidirectional orientation tensor and satisfies

$$a_{ii}^{(i)} = 1, \quad a_{jk}^{(i)} = 0 \quad \text{for other components.} \quad (29)$$

Suggested by [27], such orientation tensors can also be defined for DMN. As explained in Sect. 2.1, DMN rotations $\boldsymbol{\theta}$ define the rotation matrices between the local frame of the current laminate to that of another laminate on the next nesting level. They can thus be composed to obtain the effective rotation matrix from the material frame $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ (leaf laminates) to the global frame of the microstructure Ω (root laminate). Let $\mathbf{p}(i)$ denote the parent of a laminate i in the DMN binary tree architecture. For instance, in the 3-layer DMN example shown in Fig. 2, we have $\mathbf{p}(4) = 2$ for the leaf laminate 4 and $\mathbf{p}^2(4) = \mathbf{p}(\mathbf{p}(4)) = \mathbf{p}(2) = 1$ which is the root laminate. For each leaf laminate i which carries the DMN material nodes, the effective rotation matrix from the material frame to the global one is given by

$$\tilde{\mathbf{R}}_i = \mathbf{R}(\boldsymbol{\theta}_{\mathbf{p}^{L-1}(i)}) \cdots \mathbf{R}(\boldsymbol{\theta}_{\mathbf{p}^2(i)}) \mathbf{R}(\boldsymbol{\theta}_{\mathbf{p}(i)}) \mathbf{R}(\boldsymbol{\theta}_i), \quad \tilde{\mathbf{R}}_i \in SO(3). \quad (30)$$

Note that for an L -layer DMN, we have necessarily $\mathbf{p}^{L-1}(i) = 1$ for arbitrary leaf laminate i . Due to the absence of input rotation matrices for $(\mathbb{C}_1, \mathbb{C}_2)$, the material nodes $1 \leq i \leq N$ that share the same leaf laminate also obtain the same effective rotation matrix. For instance, for the material nodes 3 and 4 contained in the leaf laminate 5 in Fig. 2, their effective rotation is

$$\tilde{\mathbf{R}}_5 = \mathbf{R}(\boldsymbol{\theta}_1) \mathbf{R}(\boldsymbol{\theta}_2) \mathbf{R}(\boldsymbol{\theta}_5).$$

Using these effective rotations on leaf laminates, similar to Eq. (27), the DMN material frame orientation tensors can be computed for each phase p

$$\mathbf{a}_p^{(i)}(\mathbf{w}, \boldsymbol{\theta}) = \frac{\sum_{j \in \mathbb{I}_p} w_j \tilde{\mathbf{R}}_j^{(i)} \otimes \tilde{\mathbf{R}}_j^{(i)}}{\sum_{j \in \mathbb{I}_p} w_j}, \quad p \in \{1, 2\}. \quad (31)$$

Compared to the DMN volume fraction Eq. (23), the computation of DMN orientation tensors requires both DMN weights \mathbf{w} and rotations $\boldsymbol{\theta}$. Using the MI architecture Eq. (22), we propose the following orientation constraint

through the definition of a loss function

$$\mathcal{L}_a = \frac{1}{n_a} \sum_{p=1}^2 \sum_{i=1}^{n_a} \|\mathbf{a}_p(\mathbf{w}(\mathbf{p}_i), \boldsymbol{\theta}(\mathbf{p}_i)) - \mathbf{a}_{\Omega,p}(\mathbf{p}_i)\|^2, \quad \|\mathbf{a}\|^2 = \sum_{i=1}^3 \sum_{j=1}^3 \sum_{k=1}^3 (a_{jk}^{(i)})^2. \quad (32)$$

In Eq. (32), \mathbf{p}_i are the n_a collocation points in the microstructural parametric space and $\mathbf{a}_{\Omega,p}(\mathbf{p}_i)$ are the corresponding orientation tensors of the parameterized microstructure, for the phase p . The training strategy of Eq. (32) will be described further in Sect. 3.5.

With 5 DMN layers and the ReLU activation function, an example of the minimization of Eq. (32) is shown in Fig. 10. The microstructure is parameterized by two geometrical parameters $\mathbf{p} = \mathbf{q} = (q_1, q_2) \in [0, 1]^2$ that do not change w.r.t. Due to Eq. (22a), the DMN weights do not vary with \mathbf{p} . We suppose that the material frame of both phases coincides with the global one, hence in Eq. (32) the unidirectional orientation tensor Eq. (29) is used as the target values. The decreasing loss history indicates that our MI architecture Eq. (22) is capable of satisfying the orientation constraint.

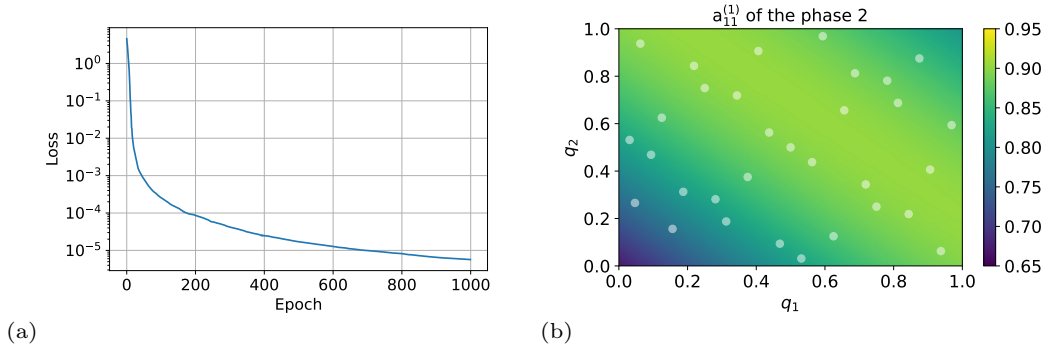


Figure 10: Orientation constraint prescribed on $\mathbf{w}(\mathbf{p})$ and $\boldsymbol{\theta}(\mathbf{p})$: (a) minimization of the loss function Eq. (32); (b) material frame orientation tensor component $a_{11}^{(1)}$ for the 1st material axis \mathbf{e}_1 of the phase 2, at epoch 10 of training. The points in (b) represent the $N = 2^5 = 32$ collocation points.

At the early stage of the training, $a_{11}^{(1)}$ of the phase 2 still presents variations in the parametric space (q_1, q_2) . Its mean and standard deviation values at epoch 10^1 and at epoch 10^3 are indicated in Tab. 3. After training, $a_{11}^{(1)} \approx 1$ becomes quasi-uniform as expected.

$a_{11}^{(1)}$	Mean	Standard deviation
At epoch 10^1	0.866	0.0440
At epoch 10^3	0.999	0.000686

Table 3: Mean and standard deviation of $a_{11}^{(1)}$ of the phase 2 in the parametric space at epoch 10^1 and at epoch 10^3 .

3.5. Offline training of MIpDMN

As shown in Fig. 6, the MIpDMN fitting parameters in Eq. (22) are optimized *jointly* using the linear elastic behavior data at different microstructural parameters. Each sample s in the training data contains the homogenized linear elastic stiffness tensor $\overline{\mathbb{C}}_s^{\text{FE}}$ of a particular microstructure with parameters \mathbf{p}_s , given input linear elastic behaviors of both phases $(\mathbb{C}_1, \mathbb{C}_2)_s$, see Fig. 1. Recall that $\overline{\mathbb{C}}_s^{\text{FE}}$ is in generally performed by computational homogenization like FE-RVE. The generation of such synthetic dataset requires sampling both in the input material space and in the parametric space of the microstructure

$$(\mathbb{C}_1, \mathbb{C}_2)_s \in \mathbb{M}, \quad \mathbf{p}_s = (\text{vf}, \mathbf{q})_s \in \mathbb{P} \subset \mathbb{R} \times \mathbb{R}^q \quad \implies \overline{\mathbb{C}}_s^{\text{FE}}. \quad (33)$$

In Eq. (33), \mathbb{M} represents the discrete material sample set containing various input material properties of $(\mathbb{C}_1, \mathbb{C}_2)$, while \mathbb{P} is the parametric sample set. In this work, these two samplings are performed *independently* then combined using a *Cartesian product*. For different \mathbf{p}_s , the *same* \mathbb{M} is used to evaluate the linear elastic homogenization

behavior. With this Cartesian product approach, generalization error can be quantified unambiguously in the material sampling space and in the parametric space. To maintain the training cost manageable, in this work we limit the number of total samples (FE-RVE simulations) to be $|\mathbb{P}| \times |\mathbb{M}| = 1500$. Hence, if more \mathbf{p}_s are present, less $(\mathbb{C}_1, \mathbb{C}_2)_s$ will be sampled. For comparison, in the original DMN [9] and the transfer-learning approach [26], they have fixed $|\mathbb{M}|$ to be 500.

In [11], another strategy is proposed for short-fiber reinforced plastics with fixed volume fraction but arbitrary fiber orientations. The material samples set \mathbb{M} is assigned to the parametric sample set \mathbb{P} in a *cyclic* fashion, resulting thus in a reduced number of $|\mathbb{M}|$ samples. Since different material samples are used at different \mathbf{p}_s , it is important that no bias is introduced when distributing \mathbb{M} into \mathbb{P} .

Material sampling. We follow the original material sampling method proposed by [9], assuming that $(\mathbb{C}_1, \mathbb{C}_2)$ are both orthotropic in their respective material frames. In total, $9 + 9 = 18$ material parameters are required to characterize their orthotropic elastic behaviors and 1 additional scaling parameter is used to introduce contrasts in the elastic moduli between the two phases. In this work, Latin hypercube sampling [43] is used to sample this 19-dimensional space.

After sampling, the generated $(\mathbb{C}_1, \mathbb{C}_2)_i$ are randomly partitioned into a training set and a validation set. An example of material sampling is given in Fig. 11, for the unidirectional fiber composite example in Sect. 4.1. Anisotropy in the phase 2 is similar to Fig. 11(a) and is not shown. In this case, since the actual fibers are much stiffer than the matrix, the scaling parameter is adapted to generate appropriate contrasts in the synthetic elastic moduli between the two phases. From Fig. 11(b), it can be seen that ratios between the Young’s moduli range from 10^{-1} to 10^4 .

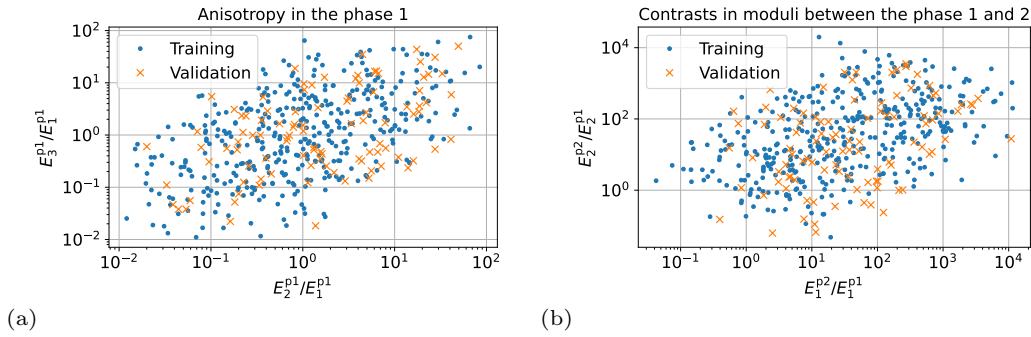


Figure 11: Material sampling for input orthotropic stiffness tensors $(\mathbb{C}_1, \mathbb{C}_2)$.

Sampling in the microstructural parametric space. Two different sampling procedures need to be performed in the microstructural parametric space. The first one defines the parameters values $\mathbf{p}_s \in \mathbb{P}$ in Eq. (33) on which $\overline{\mathbb{C}}_s^{\text{FE}}$ is computed together with the previous material sampling \mathbb{M} . For different microstructures, different sampling strategy may be used to well cover the region of interest in the parametric space. For instance, in [11], the fiber orientation space is sampled by subdividing recursively the orientation triangle into sub-triangles. For microstructures depending solely on the volume fraction parameter [26], equidistant sampling can be used in the vf-interval of interest. In general, all the microstructural parameters $\mathbf{p} = (\text{vf}, \mathbf{q})$ are sampled jointly in order that less sampling points are needed to optimally fill the parametric space.

The other sampling concerns the physical constraints prescribed on Eq. (22). The volume fraction constraint Eq. (25) and the orientation constraint Eq. (32) are weakly enforced using collocation points in the parametric space. Since the costly computational homogenization is not carried out for these points, more collocation points can be used.

The volume fraction constraint Eq. (25) requires a sampling of the vf parameter. A uniform sampling in $[0, 1]$ is simply used to generate them: $\text{vf}_i = i/(N - 1)$, with $i = 0, 1, \dots, N - 1$. The number of collocation points is taken to be the number of DMN material nodes N . A numerical study not reported here shows that N is sufficient to prescribe Eq. (24) with an acceptable accuracy, which is demonstrated by the example presented in Sect. 3.3.

The orientation constraint Eq. (32) requires collocation points \mathbf{p}_i in the parametric space, since it is prescribed both on DMN weights and rotations. Assume that each microstructural parameter is rescaled such that $\mathbf{p} \in [0, 1]^{q+1}$. Sobol low-discrepancy sequence [44] is used to generate these collocation points which presents good space filling properties. If internal bounds within the parameters are present, such as for the fiber orientation space [11],

collocation points outside the feasible region are removed. The number of collocation points is also taken to be the number of DMN material nodes N . Sobol sequence is deterministic so that the same collocation points are used given the number of layers L . Numerical example shown in Sect. 3.5 shows that the orientation constraint can be well prescribed with this sampling strategy.

Loss function. Based on the previous material and parameter sampling, a total loss function \mathcal{L} is defined as follows

$$\mathcal{L} = \frac{1}{|\mathbb{P}|} \sum_{\mathbb{P}} \mathcal{L}_{\mathbf{p}} + \lambda_{\text{vf}} \mathcal{L}_{\text{vf}} + \lambda_{\mathbf{a}} \mathcal{L}_{\mathbf{a}}, \quad \mathcal{L}_{\mathbf{p}} = \frac{1}{|\mathbb{M}|} \sum_{\mathbb{M}} e_i^2, \quad e_i = \frac{\|\bar{\mathbb{C}}_i^{\text{DMN}} - \bar{\mathbb{C}}_i^{\text{FE}}\|}{\|\bar{\mathbb{C}}_i^{\text{FE}}\|} \quad (34)$$

In Eq. (34), \mathcal{L}_{vf} is the volume fraction constraint Eq. (25) and $\mathcal{L}_{\mathbf{a}}$ is the orientation constraint Eq. (32). Similar to [9], the data loss at a fixed microstructural parameter value $\mathcal{L}_{\mathbf{p}}$ is the mean squared error in the material sample set \mathbb{M} comparing the DMN predictions and the computational homogenization ones, with the relative Frobenius norm on the stiffness tensors. In the total loss, $\mathcal{L}_{\mathbf{p}}$ is then averaged among different samples of the microstructural parameters.

The scaling factors λ_{vf} and $\lambda_{\mathbf{a}}$ are *symbolically* present in Eq. (34). In the physics-informed machine learning community, they are introduced to balance the interplay between various data-based and PDE residual terms [45]. In this work, we simply set $\lambda_{\text{vf}} = \lambda_{\mathbf{a}} = 1$. Our numerical simulations indicate that both the data loss and the constraint terms are already of the same order $\mathcal{O}(1)$ initially. Furthermore, no improvement in terms of training, validation and test accuracy has been found by adjusting these scaling factors (for instance, by setting $\lambda_{\text{vf}} = \lambda_{\mathbf{a}} = 10$). This may be related to the physics-based nature of DMN, the mathematically concise constraints Eq. (25) and Eq. (32) which are derivative-free, and the optimization algorithm described below.

In [9, 10], a penalty term is also introduced to control the magnitude of \mathbf{z} in Eq. (5). It is motivated by the fact that the ReLU activation function is unbounded and the DMN homogenization function Eq. (4) is invariant with respect to the scaling $\mathbf{z} \mapsto k\mathbf{z}$ for $k > 0$. In this work, we don't find it necessary to prescribe such constraints on our fitting parameters $(\mathbf{w}_0, \mathbf{w}_1)$.

Optimization algorithm. The total loss function Eq. (34) is minimized using gradient-based methods, similar to other DMN formulations [7, 10, 18, 19]. Our MIpDMN architecture is implemented using PyTorch [46]. The derivatives of the loss function Eq. (34) with respect to the fitting parameters Eq. (22) can thus be easily computed with automatic differentiation. Single precision is used for offline training.

Our MIpDMN forward function $(\mathbf{p}, \mathbb{C}_1, \mathbb{C}_2) \mapsto \bar{\mathbb{C}}$ is vectorized over both the material sampling dimensions $(\mathbb{C}_1, \mathbb{C}_2) \in \mathbb{M}$ and the parameter sampling dimension $\mathbf{p} \in \mathbb{P}$, in order to achieve optimal parallel computational efficiency. In [7, 11, 18, 19], a mini-batch (or even only one sample) is randomly drawn from the whole dataset at each iteration of an epoch to introduce randomness in gradient descent and promote more frequent parameter updates. In the machine learning community, this is known to improve generalization capability [47]. In this work however, a batch-learning approach is adopted, where the entire dataset is trained in a single batch at every epoch. Our simulation results indicate that the trained MIpDMN still generalizes well without significant overfitting, thanks to its physics-based nature. Furthermore, the training time can be reduced due to vectorization.

We have compared different first-order gradient-based optimizers provided in PyTorch, and have found that the resilient backpropagation (Rprop) algorithm [48] works best in terms of convergence (loss values), stability (noise) and efficiency (time to train each epoch) in this batch-learning setting. Compared to other algorithms like Adam [49] used in the DMN community [11, 20], Rprop only considers the signs of gradients and each weight is updated independently using a dynamically adapted step size. For batch learning, Rprop with default hyperparameters outperforms Adam even if the hyperparameters of the latter are tuned [50]. Hence, our MIpDMN is trained over 10000 epochs using Rprop with an initial learning rate (step-size) of 10^{-2} .

Initialization. Depending on the activation function used in Eq. (22a), different initialization can be used for \mathbf{w}_0 . For ReLU, uniform distribution $\mathbf{w}_0 \sim \mathcal{U}(0.2, 0.8)$ as proposed by [7] is applied. For \mathbf{w}_1 which characterizes the dependence of the DMN weights on vf, it is zero-initialized. Compared to a random initialization, numerical simulations demonstrate that it would result in better convergence of the loss function.

Regarding the DMN rotations in Eq. (22b), similarly Θ_1 is zero-initialized. The dependence of $\mathbf{p} \mapsto \theta(\mathbf{p})$ is hence also learned from zero. For the constant part θ_0 , a random initialization on quaternions is used. Each component is initialized using the standard normal distribution, and then normalized to obtain a unit norm. We don't find it necessary to control the norm of each DMN rotation θ in Eq. (22b) through the use of an additional loss term in Eq. (34).

Since initialization can have an impact on trained DMN parameters, training is in general repeated 20 times and the model with the least final loss value is chosen for further numerical investigations.

4. Numerical examples

In this section we will numerically evaluate our MIPDMN architecture on three parameterized microstructures. For the effective stiffness and thermal conductivity tensors, the FE-RVE simulation data are generated by imposing periodic boundary conditions [6] on the displacement or the temperature field. The effective CTE data are obtained by post-processing the thermal strains due to a uniform unit temperature increase, using Eq. (16). The finite element model is set up via the Abaqus FE-RVE plugin [51], then solved by Abaqus 2023 using 24 threads of Intel(R) Xeon(R) Gold 5220R CPU 2.20GHz. Training of MIPDMN is performed on an NVIDIA GeForce RTX 2080 Ti GPU card with single precision.

4.1. Unidirectional fiber composite

We first consider a unidirectional fiber composite with varying fiber volume fractions. The 3-d finite element model is built with hexagonal fiber packing. In total, 5 FE-RVE models are constructed, see Fig. 12. Three of them ($\text{vf} = 0.2$, $\text{vf} = 0.5$ and $\text{vf} = 0.8$) constitute the parametric sample set \mathbb{P} and are used to generate training dataset, while the other two are used to test interpolation accuracy.

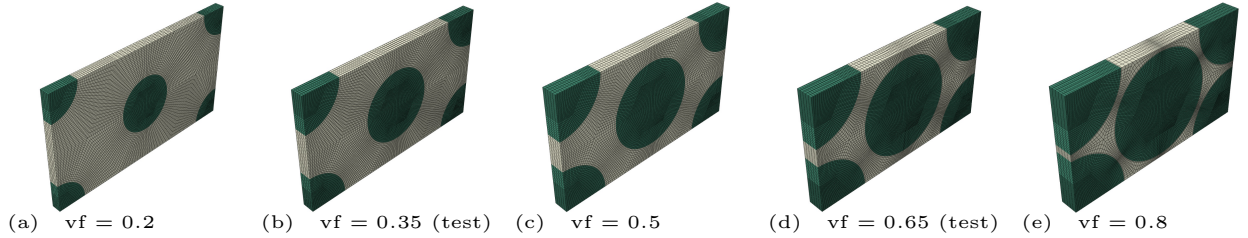


Figure 12: FE-RVE models for the unidirectional fiber composite with varying fiber volume fractions. The models with $\text{vf} = 0.2$, $\text{vf} = 0.5$ and $\text{vf} = 0.8$ are used to generate training dataset, while those with $\text{vf} = 0.35$ and $\text{vf} = 0.65$ are used to test interpolation accuracy.

Material sampling \mathbb{M} is performed using the method described in Sect. 3.5. In total, the same 500 input orthotropic material properties are sampled for all the five microstructures, see Fig. 11. Since the actual composite is made of a polypropylene matrix reinforced with glass fibers, the synthetic fiber stiffnesses \mathbb{C}_2 are also generated to be statistically higher than the matrix ones \mathbb{C}_1 . The real linear elastic properties of the two isotropic phases are indicated in Tab. 4.

	E (MPa)	ν	k (W/(m·K))	α (K ⁻¹)
Matrix	3300	0.41	0.27	7×10^{-5}
Fiber	72000	0.22	0.93	5×10^{-6}

Table 4: Real linear elastic properties, thermal conductivity and CTE of the two phases for the unidirectional fiber composite and the ellipsoidal inclusion composite.

For training microstructures $\text{vf} = 0.2$, $\text{vf} = 0.5$ and $\text{vf} = 0.8$, 400 of the 500 samples are used as training dataset, while the other 100 are reserved for validation. For the others $\text{vf} = 0.35$ and $\text{vf} = 0.65$, all the 500 samples are used for testing the interpolation accuracy. The FE-RVE models contain approximately 100×10^3 to 200×10^3 degrees of freedom and require up to 6 seconds for each run.

Since the volume fraction (of the fibers) is the only parameter of this microstructure, the MIPDMN architecture Eq. (22) implies that the DMN weights \mathbf{w} vary with vf according to Eq. (22a), while the DMN rotations $\boldsymbol{\theta}$ remain constant. The fitting parameters are hence $(\mathbf{w}_0, \mathbf{w}_1, \boldsymbol{\theta}_0)$. This is somehow similar to the “naïve” approach described in [26]. Given a pre-trained base DMN with $\mathbf{w}^{(b)}$ and corresponding volume fraction $\text{vf}^{(b)}$, the DMN weights at a new vf are scaled appropriately based on the volume fraction

$$w_i = \begin{cases} \frac{1 - \text{vf}}{1 - \text{vf}^{(b)}} w_i^{(b)} & i \in \mathbb{I}_1, \\ \frac{\text{vf}}{\text{vf}^{(b)}} w_i^{(b)} & i \in \mathbb{I}_2. \end{cases} \quad (35)$$

The DMN rotations remain unchanged. Compared to Eq. (35), the vf -dependence of \mathbf{w} in our MIPDMN is not directly prescribed but learned from data.

Training of our MIpDMN is conducted following Sect. 3.5, with the two physical constraints Eq. (25) and Eq. (32). The ReLU activation function is used. Since the material frames of both phases coincide with the global one, the unidirectional material frame orientation tensor Eq. (29) is used as the target value for the orientation constraint. With 5 DMN layers, we first compare Rprop with the frequently used Adam optimizer in terms of loss history, in Fig. 13. A total of 20 trainings are realized using random initialization for each optimizer. For Adam, two (initial) learning rates are used, which are common values for DMN training [11, 20]. With a larger $lr = 10^{-2}$, Adam results in a much more noisy loss history which could be solved by learning rate decay [11, 20], at the cost of introducing more hyperparameters. For both learning rates, Adam converges slower than Rprop with no improvement in the training accuracy. Similar to the finding of [50], our numerical simulations indicate that Rprop with no hyperparameter tuning outperforms Adam for the batch-training of MIpDMN.

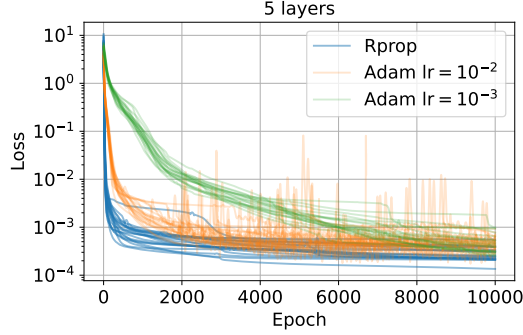


Figure 13: Loss history with 5 DMN layers with the Rprop and the Adam optimizers using two learning rates (lr).

Similar to the original DMN formulations [7], adding more layers also increases the expressive power of MIpDMN. From Fig. 14(a), the final loss values with 7 layers are lower than those obtained with 5 layers. The total loss can be partitioned into two parts according to Eq. (34): the FE-RVE data part and the physical constraints part. Their respective history is shown in Fig. 14(b) for the training realization with the least final loss value, for 7 layers. Each of the three “FE-RVE” curves represents one particular training microstructure $vf = 0.2$, $vf = 0.5$ or $vf = 0.8$, and is monotonically decreasing simultaneously with similar values. The physical constraints part also converges well and is lower than the FE-RVE part by an order of magnitude. Among the 20 training realizations, the median training time for 5 or 7 layers is only respectively 570 s and 786 s. Compared to the training times (several hours) reported in [9, 11], we believe that our batch-learning approach along with the vectorization of the forward function help to speed up training significantly. In the sequel, we will report the results obtained with 5 layers which provides satisfactory accuracy.

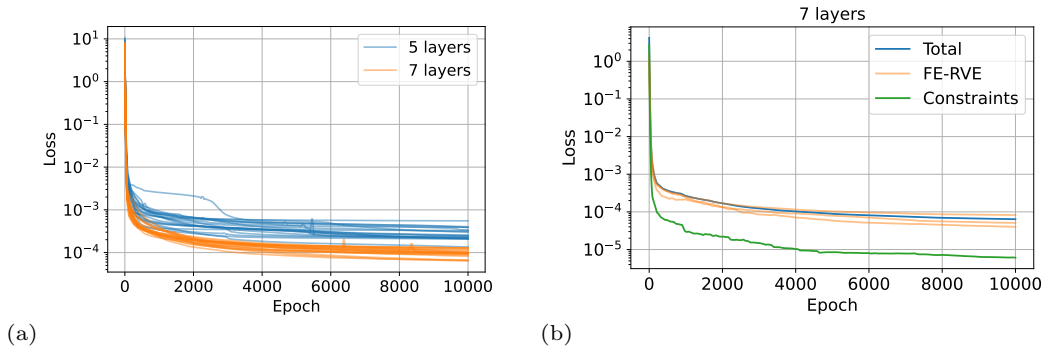


Figure 14: MIpDMN training for the unidirectional fiber composite: (a) loss histories with 5 and 7 DMN layers; (b) partition of the total loss into the FE-RVE data part and the physical constraints part.

In Fig. 15, the statistical distribution of the relative Frobenius-norm error e_i in Eq. (34) between DMN predictions and FE-RVE ones on each material sample is shown for the microstructure with $vf = 0.5$. The 0.1, 0.5 (median) and 0.9-quantiles are also indicated. Even though the maximum relative error can reach 6% for some localized samples, the error is less than 2.4% for 90% of them. We believe that these quantiles are less sensitive to outliers and hence more appropriate for comparing different models.

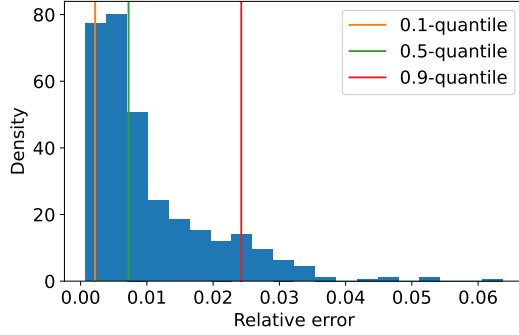


Figure 15: Statistical distribution of the relative error e_i for the microstructure with $vf = 0.5$.

We first analyze the influence of physical constraints on the MI architecture Eq. (22). The training, validation and test errors are compared in Fig. 16 using their respective 0.1, 0.5 (median) and 0.9-quantiles at different volume fractions. The MIpDMN model without physical constraints is trained similarly with 20 realizations and the one with the least final loss value is chosen. Validation error refers to the relative error computed on the validation dataset composed of 100 material samples. Overfitting is not observed for both cases, since the validation errors are comparable to the training errors. When the physical constraints are not included, the training errors at $vf = 0.5$ and $vf = 0.8$ are similar or only slightly lower at $vf = 0.2$ than the case when they are considered. However, the inclusion of the physical constraints reduces the interpolative test errors at $vf = 0.35$ and $vf = 0.65$ for previously unseen microstructures. The physical constraints may hence improve generalization ability of MIpDMN.

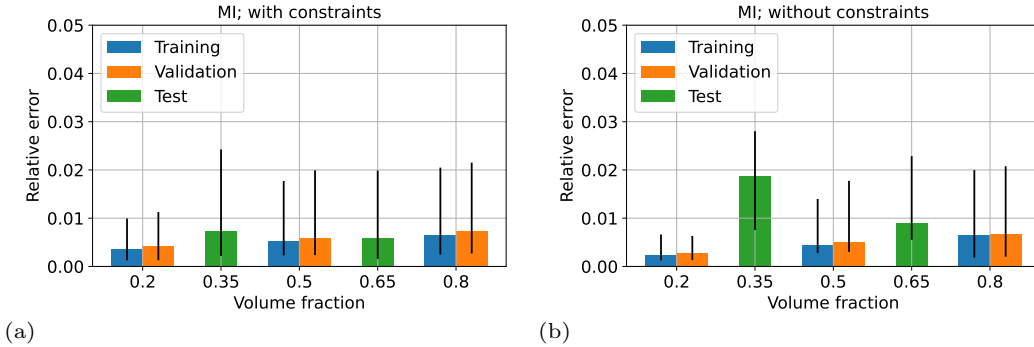


Figure 16: Training, validation and test errors of MIpDMN at different volume fractions: (a) with physical constraints; (b) without physical constraints.

In Fig. 17, we analyze the influence of the physical constraints on the fully-connected architecture Eq. (20). Now, the DMN rotations also become a function of the volume fraction, leading to an additional fitting parameter Θ_1 during training. The physical constraints reduce overfitting at the data points. By comparing Fig. 16(a) and Fig. 17(a), we observe similar training, validation and test errors between the MI and the fully-connected architectures, when the physical constraints are included. The MI architecture Eq. (22) hence demonstrates similar expressive power compared to the fully-connected one, even with a constant DMN rotations vector θ_0 for different volume fractions. When the physical constraints are not considered, the interpolative test errors become significantly larger (even the scale of the y -axis needs to be changed) in Fig. 17(b) for the fully-connected architecture. Comparing Fig. 16(b) and Fig. 17(b), we conclude that the MI architecture itself may help to improve generalization capability.

In Fig. 18(a), the DMN volume fraction prediction Eq. (23) is presented as a function of the microstructure volume fraction. For all cases, DMN recovers well the volume fractions 0.2, 0.5 and 0.8 at the FE-RVE data points. The microstructure morphologies are hence well learned by DMN using linear elastic data [7, 10]. However, away from these three data points, the DMN volume fraction prediction may differ from the actual microstructure one. At $vf = 0.35$, when Eq. (25) is explicitly included during training, the relative errors $|vf_{DMN} - vf_{\Omega}|/vf_{\Omega}$ are smaller than 0.4% for the two parametric DMN architectures. However, when it is not the case, we obtain 2% with the MI architecture and even 9% with the fully-connected one. This may explain the higher interpolative testing errors at $vf = 0.35$ for these two models in Fig. 16(b) and Fig. 17(b). The straight line Eq. (24) can only be well recovered when the volume fraction constraint Eq. (25) is explicitly included.

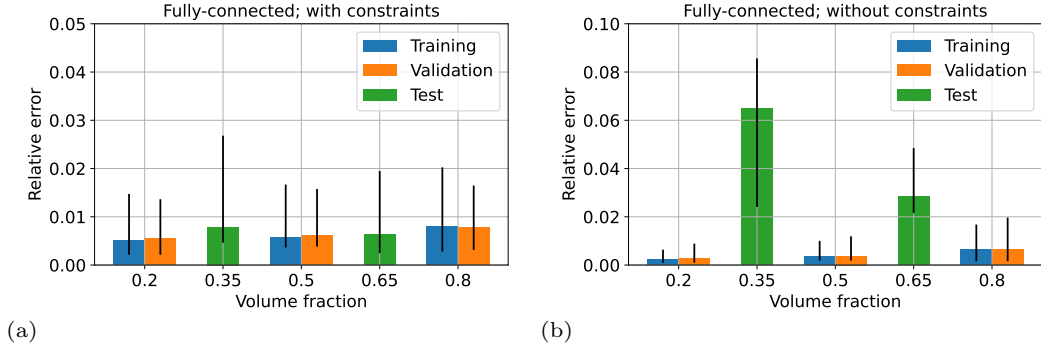


Figure 17: Training, validation and test errors of a fully-connected parametric DMN at different volume fractions: (a) with physical constraints; (b) without physical constraints.

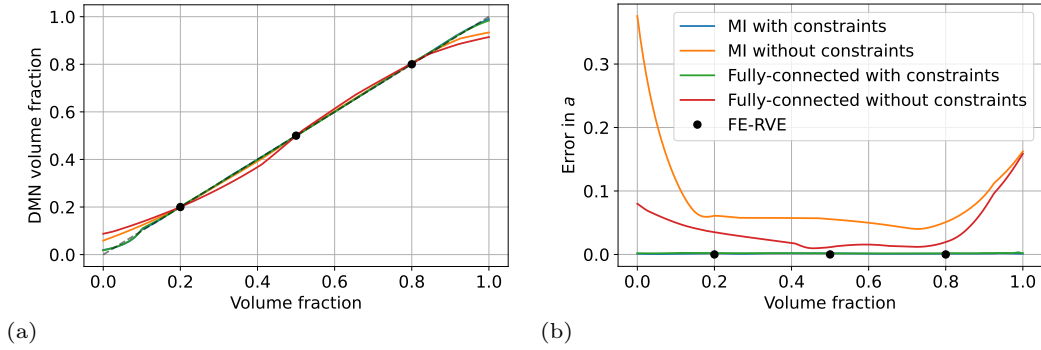


Figure 18: Physical constraints verification for the micromechanics-informed (MI) and the fully-connected architectures: (a) DMN volume fraction prediction; (b) error on the material frame orientation tensors.

Similarly, the error on the material frame orientation tensors $\|\mathbf{a}_p(\mathbf{v}f_i) - \mathbf{a}_{\Omega,p}(\mathbf{v}f_i)\|$ using the norm defined in Eq. (32) is presented in Fig. 18(b). The errors on the fibers and on the matrix are summed together. When Eq. (32) is not included during training, deviation from the theoretical unidirectional material frame orientation tensor Eq. (29) can be observed, even though the error is well bounded between $[0.2, 0.8]$. Similar to Fig. 18(a), physical constraint errors increase significantly outside the training domain. The inclusion of Eq. (32) may hence improve the generalization capability in the whole parametric space.

In this work, the rotation matrix for $(\mathbb{C}_1, \mathbb{C}_2)$ of each of the $N = 2^L$ material nodes on the input layer is omitted compared to the original formulation [7, 9]. In Fig. 19, the training, validation and test errors are shown when they are instead included. Similarly, the training process is realized 20 times with random initialization and the model with the least final loss value is chosen. Compared to Fig. 16(a), the inclusion of such material rotations does not increase the expressive power of MipDMN. This could be partially due to the fact that the microstructure being considered does not contain local material orientation.

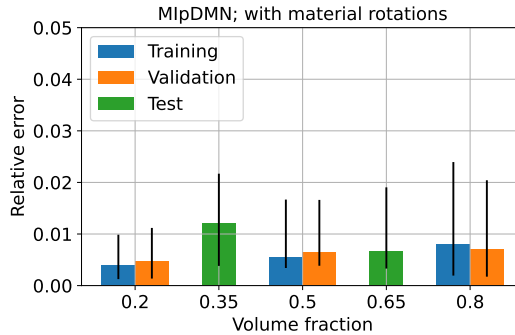


Figure 19: Training, validation and test errors of MipDMN with additional $N = 2^L = 32$ material rotations at different volume fractions.

Using the real linear elastic properties of the two phases in Tab. 4, the relative errors between the DMN predictions and the FE-RVE results for two parametric DMN architectures and with or without the physical constraints are shown in Fig. 20. The conclusions drawn from Fig. 16 and Fig. 17 are recovered. The MI architecture and the physical constraints both help to improve generalization ability of the parametric DMN model. Using MIpDMN with the physical constraints, the relative errors are less than 1.5% among the 5 microstructures being considered.

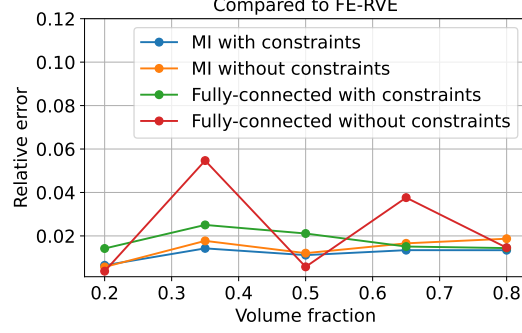


Figure 20: Relative error using the real properties between the DMN predictions and the FE-RVE results for two parametric DMN architectures with or without the physical constraints.

The homogenized elastic property prediction is presented in Fig. 21 in the *whole* parametric space. The general nonlinear influence of the volume fraction is well captured by these two parametric DMN models. The use of the MI architecture as well as the physical constraints helps to improve generalization capabilities.

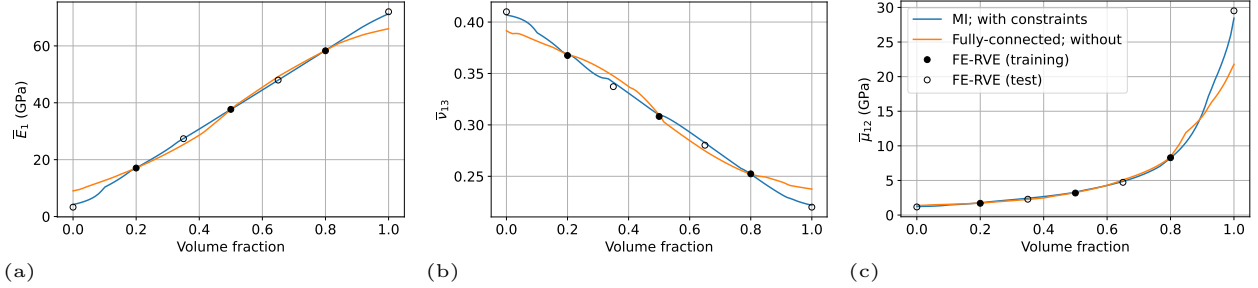


Figure 21: Homogenized elastic moduli prediction as a function of the fiber volume fraction obtained by parametric DMN models: (a) longitudinal Young’s modulus \bar{E}_1 ; (b) transverse Poisson ratio $\bar{\nu}_{13}$; (c) transverse shear modulus $\bar{\mu}_{12}$. The theoretical values at $vf = 0$ and $vf = 1$ are also indicated as test data.

In Fig. 22(a), the MI functional dependence of the DMN weights Eq. (22a) is illustrated. The MIpDMN architecture with the physical constraints is used. Globally, 22 of the 32 material nodes are active. With the increasing fiber volume fraction, DMN weights of the matrix decrease *gradually* and *individually*, while those of the fiber increase at the same time. This is also reflected in Fig. 22(b), where the ratio of the active DMN nodes is presented for both phases. Even though the ratios for both phases gradually increase or decrease, the total number of active nodes does not vary much. This ensures the expressive power of MIpDMN in the whole parametric space.

Using the proposed MIpDMN architecture and the fitted parameters trained previously on isothermal linear elastic data, we will now predict the effective thermal conductivity $\bar{\mathbf{k}}$ and the effective CTE $\bar{\boldsymbol{\alpha}}$ with the redefinitions of the laminate homogenization function Lam in Sect. 2.2. The constituent properties used in this online prediction test are provided in Tab. 4. In Fig. 23, the MIpDMN predictions are compared with the FE-RVE results as a function of the volume fraction. The nonlinear vf -dependences of $\bar{\mathbf{k}}$ and $\bar{\boldsymbol{\alpha}}$ are well captured by MIpDMN. An excellent agreement is obtained in the whole interval, even though the longitudinal $\bar{\alpha}_1$ is slightly underestimated near $vf \approx 0$. Using a similar relative Frobenius-norm error e_i in Eq. (34) for second-order tensors, the maximum error among these 5 FE-RVE data points is 0.85% for $\bar{\mathbf{k}}$ at $vf = 0.35$, and 4.5% for $\bar{\boldsymbol{\alpha}}$ also at $vf = 0.35$. This is a remarkable result knowing that our MIpDMN is only trained using isothermal linear elastic data. This demonstrates that DMN learns the microstructure *per se*, and not a particular physics property in particular.

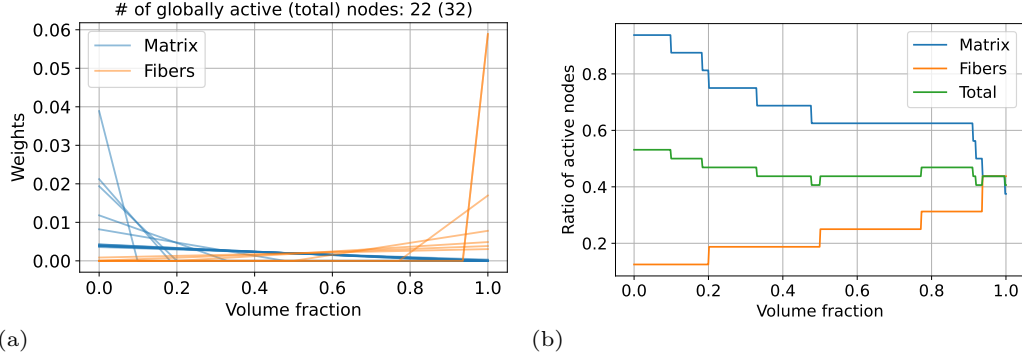


Figure 22: MIPDMN: (a) DMN weights evolution with the volume fraction; (b) variation of the ratio of active DMN nodes

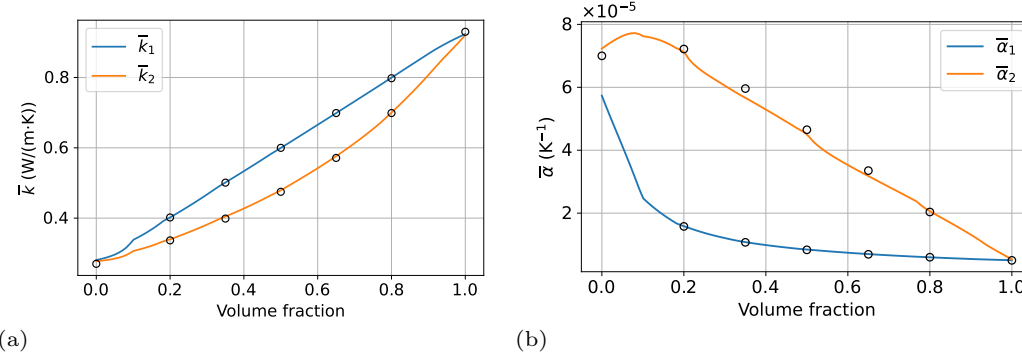


Figure 23: (a) Effective thermal conductivity prediction for the longitudinal \bar{k}_1 and transverse \bar{k}_2 components; (b) effective CTE prediction. The FE-RVE results and the theoretical values for $vf = 0$ and $vf = 1$ are indicated by circles.

Comparison with transfer-learning based interpolative DMN

We follow the procedure described in [26] and the microstructure with the least fiber volume fraction is used as the starting point in the training sequence $vf_0 = 0.2 \rightarrow vf_1 = 0.5 \rightarrow vf_2 = 0.8$. Subsequent training is initialized using Eq. (35) based on the previously trained DMN. As for our parametric DMN models, the transfer learning process is repeated 20 times with random initialization when training the base DMN model. The realization with the least final total loss value $\mathcal{L}_{\text{tot}} = \mathcal{L}_0 + \mathcal{L}_1 + \mathcal{L}_2$ is then chosen for further analysis. For comparison with previous MIPDMN results, 5 DMN layers are used.

In Fig. 24(a), loss histories obtained with transfer learning are presented. Transfer learning indeed accelerates convergence. For $vf = 0.5$ and $vf = 0.8$, convergence is achieved within the first 1000 epochs. However, the final loss value becomes an order of magnitude larger at $vf = 0.8$ compared to the base DMN. In [26], there is also a slight increase in the final loss values at the end of the transfer learning process. For $vf = 0.8$, DMN is also trained using random initialization. The loss histories are compared with that obtained from transfer learning. A reduction of the expressive power of DMN is hence observed with transfer learning.

The training, validation and test errors obtained with this transfer-learning based interpolative DMN model are presented in Fig. 25(a). Compared to the results obtained with MIPDMN (with the physical constraints) in Fig. 16(a), the errors at the training points $vf = 0.2$ and $vf = 0.5$ are lower with this transfer learning approach. However, due to the reduction of the expressive power, the errors at $vf = 0.8$ become comparable to those with MIPDMN. The test errors are significantly larger than the training and the validation ones, especially at $vf = 0.65$. The transfer-learning may hence tend to overfit at the data points.

Even though it is not proposed in [26], we can also use Eq. (35) to extrapolate DMN weights outside the training domain. In this case, the DMN weights at $vf = 0$ are hence extrapolated from the base DMN at $vf = 0.2$, while the DMN trained by transfer-learning at $vf = 0.8$ can be used to obtain the DMN weights at $vf = 1$. Since Eq. (35) respects the actual volume fraction Eq. (23), the volume fraction constraint $vf_{\text{DMN}} = vf_{\Omega}$ is satisfied at all vf when combined with piecewise linear interpolation. However, it is not the case for the orientation constraint. In Fig. 25(b), the error on the material frame orientation tensors is computed in the whole parametric space. Compared to MIPDMN, the error is much larger with the transfer-learning approach and also increases with interpolation. This

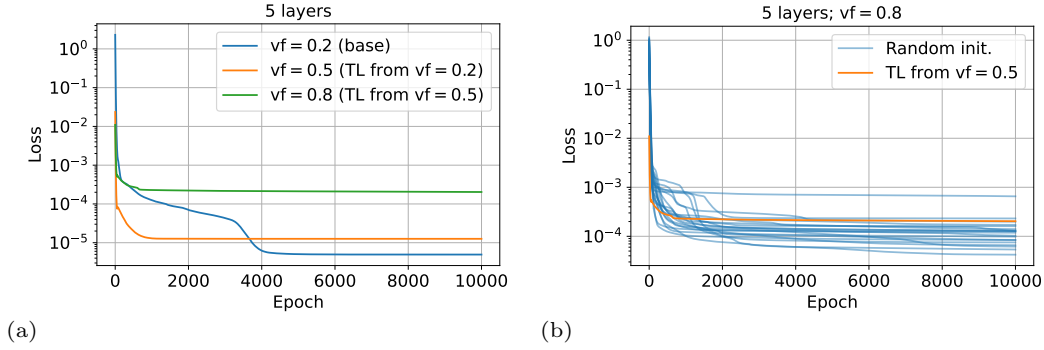


Figure 24: (a) Loss histories for different volume fractions using transfer learning; (b) loss history at $vf_2 = 0.8$ obtained using random initialization and from transfer learning.

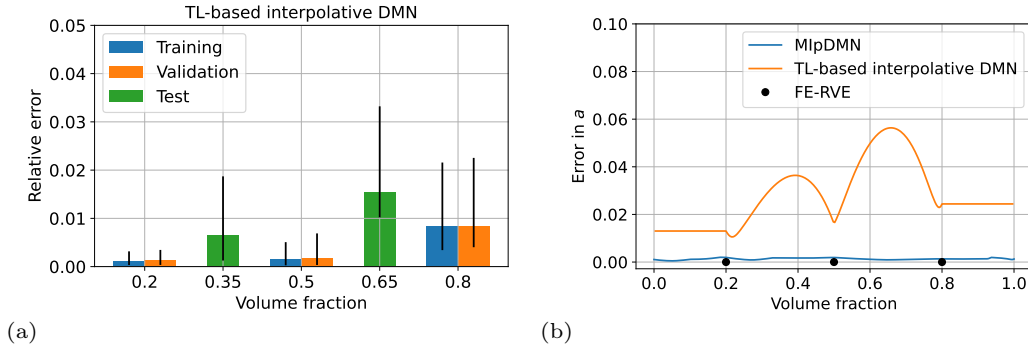


Figure 25: (a) Transfer-learning based interpolative DMN: training, validation and test errors at different volume fractions; (b) error on the material frame orientation tensors obtained by MIpDMN with the physical constraints and by the transfer-learning approach.

may explain the larger test errors in the transfer-learning model in Fig. 25(a).

The relative errors using the real linear elastic properties of the two phases in Tab. 4 are shown in Fig. 26 for the MIpDMN model and the transfer-learning approach. Even though the latter is more accurate for the first 3 volume fractions, it becomes less accurate for the last 2 volume fractions. The relative errors obtained with MIpDMN, on the contrary, are more uniform in the parametric space.

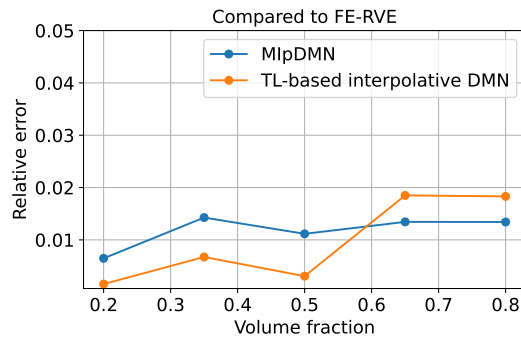


Figure 26: Relative error using real properties between the DMN predictions and the FE-RVE results obtained by MIpDMN with the physical constraints and by the transfer-learning (TL) approach.

The elastic moduli prediction is also compared in Fig. 27 between these two approaches. Both models are able to capture the nonlinear influence of the volume fraction, especially on the in-plane Poisson ratio ν_{23} . While a perfect agreement is found for \bar{E}_1 , the transfer-learning approach seems to slightly overfit the ν_{23} prediction between 0.5 and 1.0.

Similar to Fig. 22, the DMN weights evolution in the parametric space is also shown in Fig. 28 for the transfer-learning approach. Globally, 12 of the 32 material nodes are active. Due to transfer-learning and the ReLU

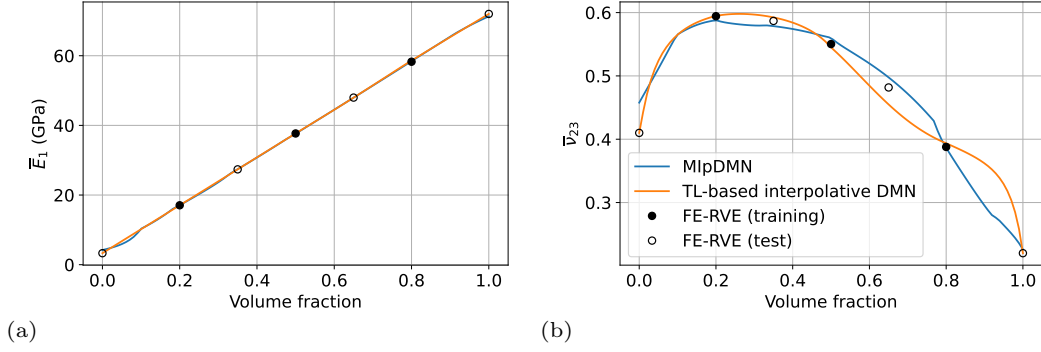


Figure 27: Homogenized elastic moduli prediction as a function of the fiber volume fraction obtained by MIpDMN and the transfer-learning (TL) approach: (a) longitudinal Young's modulus \bar{E}_1 ; (b) in-plane Poisson ratio $\bar{\nu}_{23}$.

activation function, the number of active nodes can only decrease [26]. While 12 material nodes are active for $vf = 0.2$ and $vf = 0.5$, only 9 active nodes are present for $vf = 0.8$. Compared to MIpDMN, this gradual decrease of the active DMN material nodes may lead to a reduced expressive power. The functional dependence on vf is mainly realized by adapting weights values at each transfer-learning interpolation point. There is no gradual change in the ratios of active nodes for intermediate volume fractions.

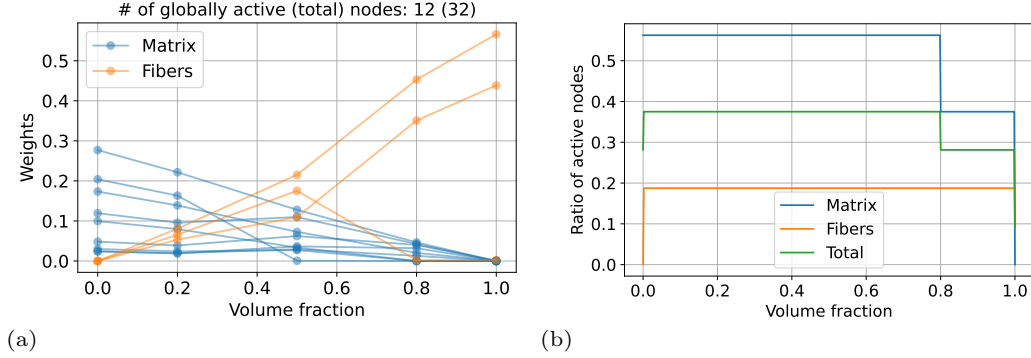


Figure 28: Transfer-learning based interpolative DMN: (a) DMN weights evolution with the volume fraction; (b) variation of the ratio of active DMN nodes.

4.2. Woven composite

We now consider the 2×2 twill woven composites with varying tow volume fractions. Compared to the unidirectional fiber composite, now local material orientation is present for the yarns, illustrated in Fig. 9. The 3-d finite element model is constructed using TexGen [52] with 4 volume fraction values for the tows, see Fig. 29. Three of them ($vf = 0.459$, $vf = 0.608$ and $vf = 0.729$) constitute the parametric sample set \mathbb{P} and are used to generate the training dataset, while $vf = 0.537$ is used to test interpolation accuracy. The FE-RVE model contains 75000 voxel elements and requires approximately 11 seconds for each run.

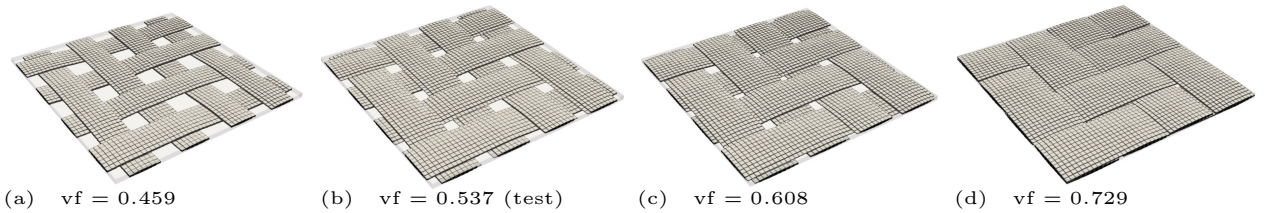


Figure 29: FE-RVE models for the woven composite with different tow volume fractions. The models with $vf = 0.459$, $vf = 0.608$ and $vf = 0.729$ are used to generate the training dataset, while $vf = 0.537$ is used to test interpolation accuracy.

The real linear elastic properties of the two phases are adapted from [9] and can be found in Tab. 5. The matrix is isotropic while the carbon fiber tows are assumed to be transversely isotropic in the local material frames. Compared

to [18], the transverse Young’s modulus E_2 of the tows is modified to satisfy transverse isotropy. A similar material sampling \mathbb{M} compared to Fig. 11 is performed to generate 500 input orthotropic material properties. Given the high moduli contrasts in the real properties, the synthetic yarn stiffnesses \mathbb{C}_2 are also generated to be statistically higher than the matrix ones \mathbb{C}_1 . For the training microstructures $\text{vf} = 0.459$, $\text{vf} = 0.608$ and $\text{vf} = 0.729$, 400 of the 500 samples are used as the training dataset, while the other 100 are reserved for validation. For $\text{vf} = 0.537$, all the 500 samples are used for testing the interpolation accuracy.

	E (GPa)	ν	μ (GPa)	k (W/(m·K))	α (K ⁻¹)
Matrix	3.8	0.387	1.37	0.3	7×10^{-5}
Tow (longitudinal)	78.8	0.35	2.39	1	4×10^{-6}
Tow (transverse)	6.24	0.6	1.95	0.4	4×10^{-5}

Table 5: Real linear elastic properties, thermal conductivity and CTE of the two phases for the woven composite.

MIpDMN is trained following Sect. 3.5 with the two physical constraints Eq. (25) and Eq. (32) and the ReLU activation function. In Eq. (32), the unidirectional material frame orientation tensor Eq. (29) is defined as the target for the matrix, since its material frame coincides with the global one. For the tows, we use the “planar isotropic” tensor Eq. (28) which characterizes the statistical spatial orientation of the yarns in the FE-RVE model. As before, the training process is realized 20 times with random initialization and the one with the least final loss value is chosen for further investigations. The median training time is only 800 s for 7 layers and 1176 s for 9 layers.

In Fig. 30, the loss histories are compared for 7 and 9 DMN layers. As for the unidirectional fiber composite, an increased expressive power is observed with more layers, leading to lower (final) loss values. The FE-RVE data and the physical constraints parts are monotonically decreasing. The physical constraints Eq. (25) and Eq. (32) are well satisfied since the corresponding loss value is approaching 10^{-5} with 9 layers. In the sequel, we will report the results using 7 layers which provides satisfactory accuracy.

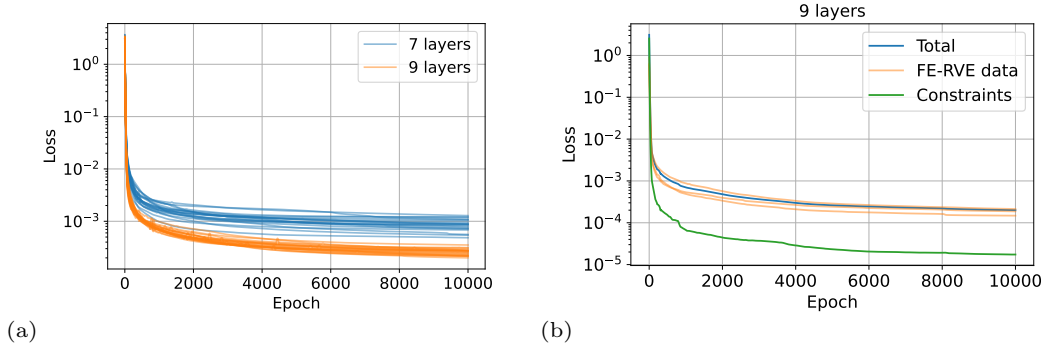


Figure 30: MIpDMN training for the woven composite: (a) loss histories with 7 and 9 DMN layers; (b) partition of the total loss into the FE-RVE data part and the physical constraints part.

The volume fraction and the material frame orientation tensors for the tows are compared with their prescribed values in Fig. 31. The DMN volume fraction matches the FE-RVE data points and agrees well with the theoretical straight line even when evaluated outside the training region. In Fig. 31(b), the components $a_{ii}^{(i)}$ of the DMN material orientation tensors are shown. An excellent agreement is also obtained. Even without the additional rotations on the input layer, our DMN architecture is capable of learning local material orientation present in the microstructure.

In total, 81 DMN material nodes are globally active in the parametric interval $[0.4, 0.8]$, as shown in Fig. 32. As in Fig. 22, the number of active nodes gradually decreases from 55% to 35% for the matrix while that of the tows increases from 50% to 65%. The overall ratio of the active nodes remains approximately 55% within the training region $[0.459, 0.729]$. This is believed to ensure sufficient expressive power for MIpDMN.

In Fig. 33(a), the training, validation and test errors are computed at different volume fractions for MIpDMN without material rotations. The median errors are approximately 2% for all the four microstructures. As in Fig. 16(a), the test error is only slightly larger compared to the training ones. In Fig. 33(b), the results are reported for MIpDMN with the additional $2^7 = 128$ material rotations. The training is also realized 20 times and the model with the least final loss value is shown. In the presence of local material orientation (tows), such material rotations do improve training and validation accuracy, but only marginally. Furthermore, the inclusion of these rotations

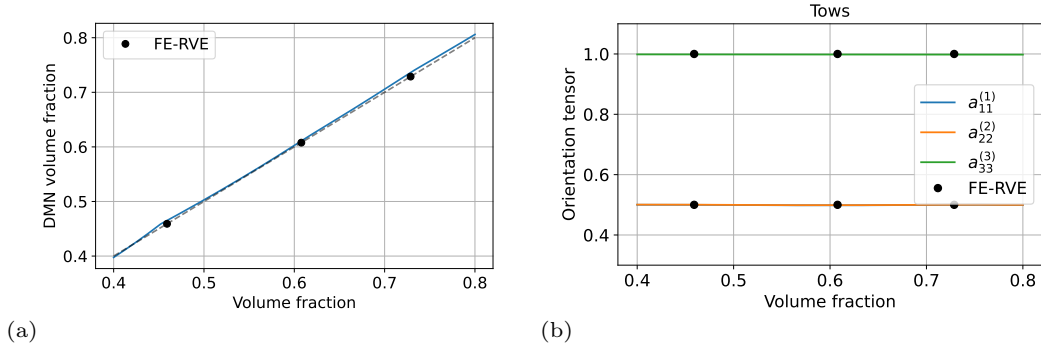


Figure 31: Verification of the physical constraints for the woven composite: (a) volume fraction; (b) material frame orientation tensors for the tows.

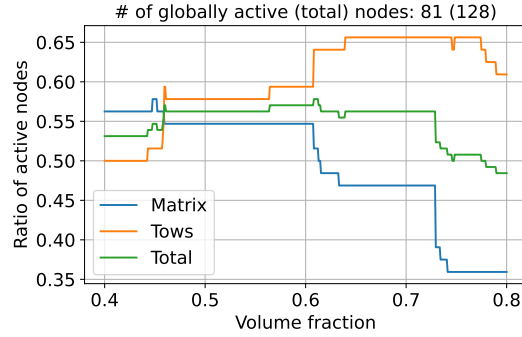


Figure 32: Variation of the ratio of active DMN nodes with varying volume fraction.

may increase the *risk* of overfitting and higher test interpolative errors, due to more fitting parameters in θ_0 of Eq. (22b). According to Fig. 33(b), the median and the 0.9-quantile values of the test error are now 4.6% and 12%. For the model with the *second* least final loss value, somehow less overfitting is present. The median test error becomes 3% and its 0.9-quantile is 6.7%, *i.e.*, still slightly higher than the case when such rotations are not introduced.

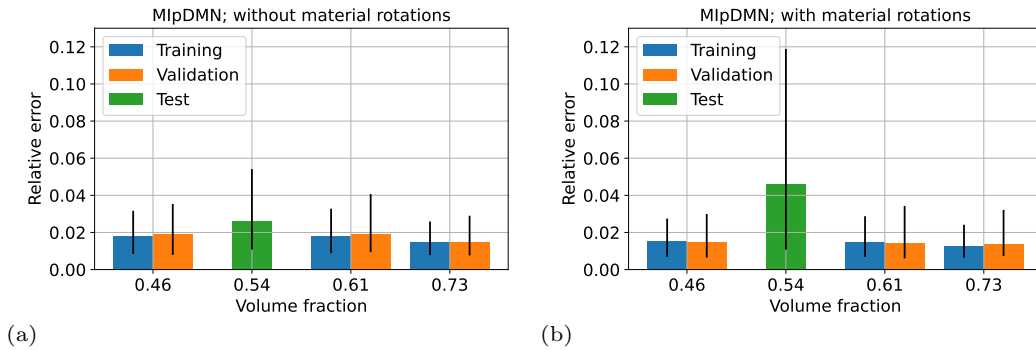


Figure 33: Training, validation and test errors of MIpDMN at different volume fractions: (a) without material rotations; (b) with additional $2^7 = 128$ material rotations.

With the real linear elastic properties provided in Tab. 5, the homogenized elastic moduli are computed in Fig. 34 with varying volume fraction. For comparison, the fully-connected architecture Eq. (20) is also tested with the physical constraints. Recall that Eq. (20) implies that the DMN rotations also become a function of the volume fraction. Both models capture well the nonlinear vf-dependence of these elastic moduli. The fully-connected architecture not only increases the number of fitting parameters (Θ_1 is now added), it does not improve prediction accuracy compared to the MI one Eq. (22). For MIpDMN, the maximum relative Frobenius-norm error e_i in Eq. (34) is less than 1.5% among the 4 FE-RVE data points.

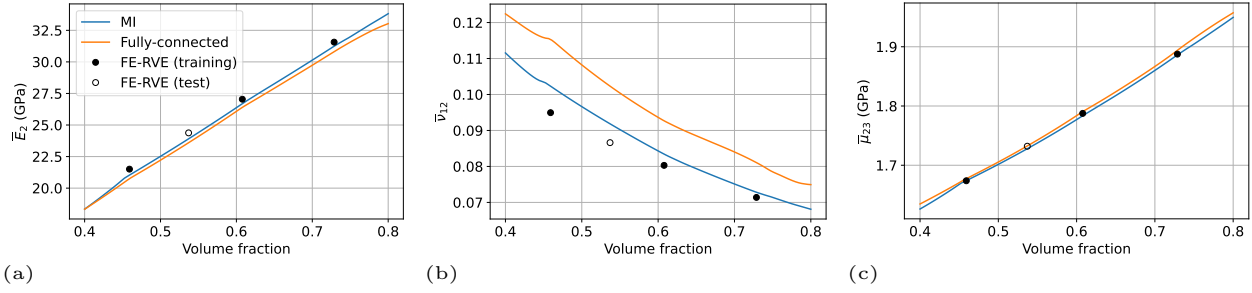


Figure 34: Homogenized elastic moduli prediction as a function of the tow volume fraction obtained by parametric DMN models: (a) in-plane Young's modulus \bar{E}_2 ; (b) in-plane Poisson ratio $\bar{\nu}_{12}$; (c) transverse shear modulus $\bar{\mu}_{23}$.

Using the previously trained 7-layer MIpDMN, we now predict the effective thermal conductivity and the effective CTE in Fig. 35 using the real properties of the constituents provided in Tab. 5. Due to the low CTE values of the tows (fibers) in the woven plane, the effective in-plane $\bar{\alpha}_1$ is well lower than the out-of-plane one. An excellent agreement is found with the FE-RVE results. Among the 4 data points, the maximum relative Frobenius-norm error is 0.25% for \bar{k} at $\text{vf} = 0.459$ and 0.98% for $\bar{\alpha}$ also at $\text{vf} = 0.459$. Even though MIpDMN is only trained using isothermal linear elastic data, it is capable of predicting other physical properties when the microstructure morphology varies.

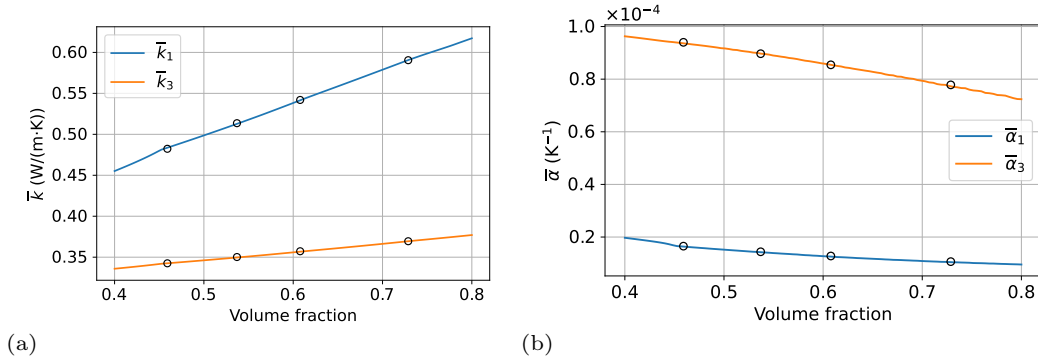


Figure 35: (a) Effective thermal conductivity prediction for the in-plane \bar{k}_1 and out-of-plane \bar{k}_3 components; (b) effective CTE prediction. The FE-RVE results are indicated by circles.

In this case, the second-order CTE tensor of the tows is transversely isotropic in its material frame, while the matrix remains isotropic. This implies that Eq. (19) must be applied as a neuron operation from the input layer to the output layer along with the effective stiffness tensor computation. If the laminate homogenization function of Eq. (19) for the effective CTE computation is applied directly to the obtained effective stiffness tensor, as shown in Fig. 36, the result would be incorrect.

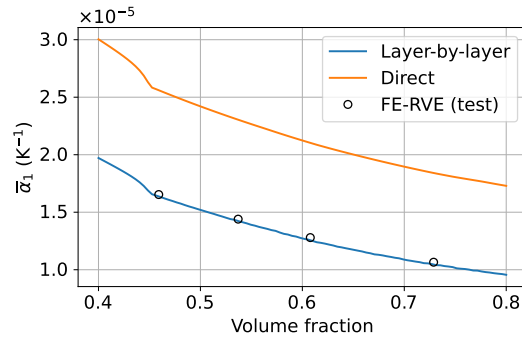


Figure 36: Comparison between a direct application and a DMN layer-by-layer application of the laminate homogenization function of Eq. (19) for the effective CTE computation.

The computational efficiency of MIpDMN in terms of computing the homogenized stiffness tensor given \mathbb{C}_1 and \mathbb{C}_2 is compared to FE-RVE simulations in Tab. 6. Thanks to this significant speed-up, MIpDMN can be further used for parametric analysis [26], uncertainty quantification [27] and material property calibration [17]. In particular, it can be employed to identify *simultaneously* the *material* and *microstructural* parameters in an inverse identification problem. A preliminary proof-of-concept study is provided in Appendix B.

	FE-RVE	7-layer MIpDMN
Wall time (speed-up)	11 s	6.62 ms (1662)

Table 6: Computational speed-up of MIpDMN compared to FE-RVE in terms of homogenized stiffness tensor prediction.

4.3. Ellipsoidal inclusion composite

Finally, we consider the ellipsoidal inclusion composite described by two parameters: volume fraction vf of the fibers and the aspect ratio ar (ratio between the length and the diameter of the fiber). The aspect ratio parameter is purely morphological and does not change vf . According to Eq. (21), we have thus $\mathbf{q} = (ar)$ in this case.

The two-dimensional parametric space (vf, ar) is sampled using Sobol low-discrepancy sequence which ensures an optimal space-filling. Since it is deterministic, additional sampling points can thus be easily included. The bounds for these two parameters are

- Volume fraction: $[0.05, 0.065]$. Higher volume fractions would lead to mesh generation issues with the body-centered fiber packing.
- Aspect ratio: $[1, 100]$. Both spherical inclusions $ar = 1$ and slender fibers $ar \gg 1$ are covered with such large variation of this ar parameter.

In this work, 20 samples of $\mathbf{p}_i = (vf, ar)_i$ are generated, see Fig. 37. The ar parameter is sampled in the log-scale. The first 10 points $\mathbf{p}_{0 \leq i < 10}$ constitute the parametric sample set \mathbb{P} and are used to generate training dataset. The other 10 points $\mathbf{p}_{10 \leq i < 20}$ are reserved to evaluate the generalization accuracy of MIpDMN.

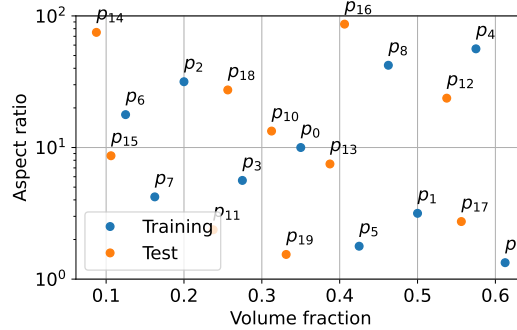


Figure 37: Sampling points in the parametric space for the ellipsoidal inclusion composite.

For each $\mathbf{p}_i = (vf, ar)_i$, a 3-d FE-RVE model is built using the body-centered fiber packing. In Fig. 38, two examples are given corresponding to the first two sampling points. The models contain approximately several hundreds of thousands of quadratic tetrahedral elements and may take up to 60 seconds for each run.

The real linear elastic properties of the two phases are the same as for the unidirectional fiber composite, given in Tab. 4. Given the high number of microstructures (10 for training, 10 for test), in total 150 input orthotropic material properties ($\mathbb{C}_1, \mathbb{C}_2$) are sampled to produce \mathbb{M} . The synthetic fiber stiffnesses \mathbb{C}_2 are also generated to be statistically higher than the matrix ones \mathbb{C}_1 . For the training microstructures $\mathbf{p}_{0 \leq i < 10}$, 100 of the 150 samples are used as training dataset, while the other 50 are reserved for validation. For the others $\mathbf{p}_{10 \leq i < 20}$, all the 150 material samples are used for testing the generalization accuracy.

MIpDMN is trained following Sect. 3.5 with the two physical constraints Eq. (25) and Eq. (32) and the ReLU activation function. Since local material orientation is absent, the unidirectional material frame orientation tensor Eq. (29) is defined as the target for both phases in Eq. (32). As before, the training process is realized 20 times with random initialization and the model with the least final loss value is chosen for further investigations. The median training time is only 800 s for 7 layers and 1198 s for 9 layers. The loss histories for 7 and 9 DMN layers

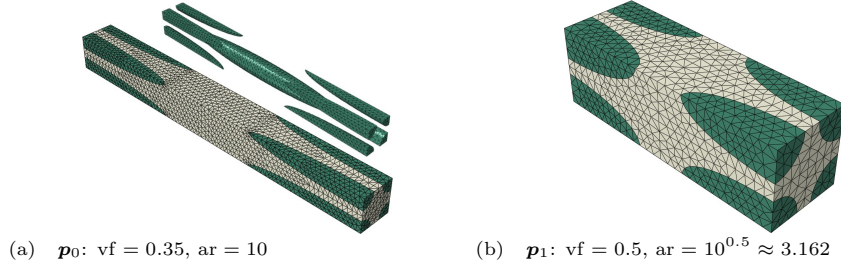


Figure 38: Example FE-RVE models for the ellipsoidal composite with different fiber volume fraction and aspect ratio values: (a) p_0 ; (b) p_1 .

are presented in Fig. 39. The final loss value is well decreasing with more layers. The FE-RVE data part and the physical constraints part are also monotonically decreasing.

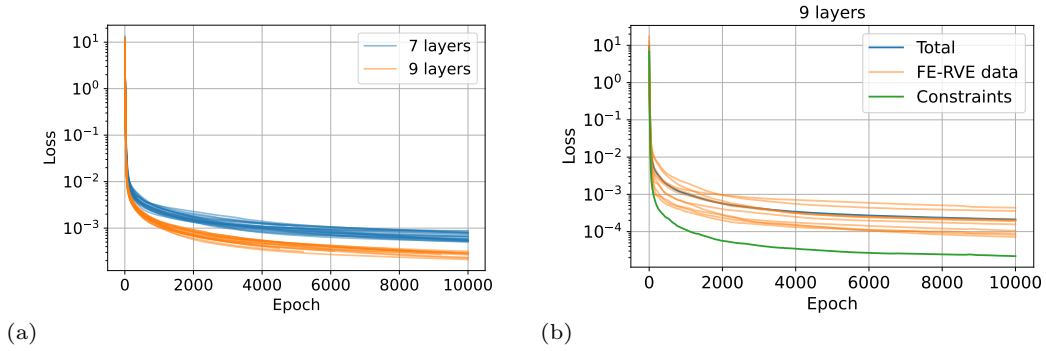


Figure 39: MIPDMN training for the ellipsoidal inclusion composite: (a) loss histories with 7 or 9 DMN layers; (b) partition of the total loss into the FE-RVE data part and the physical constraints part.

The variation of the number of active DMN nodes is presented separately for the matrix and for the fiber in Fig. 40. 7 DMN layers are used. In total, 73 out of the 128 DMN nodes are globally active in the parametric space $p = (vf, ar) \in [0, 0.7] \times [1, 100]$. Due to the MI architecture Eq. (22), the volume fraction parameter determines the number of active matrix or fiber nodes while the aspect ratio parameter has no influence.

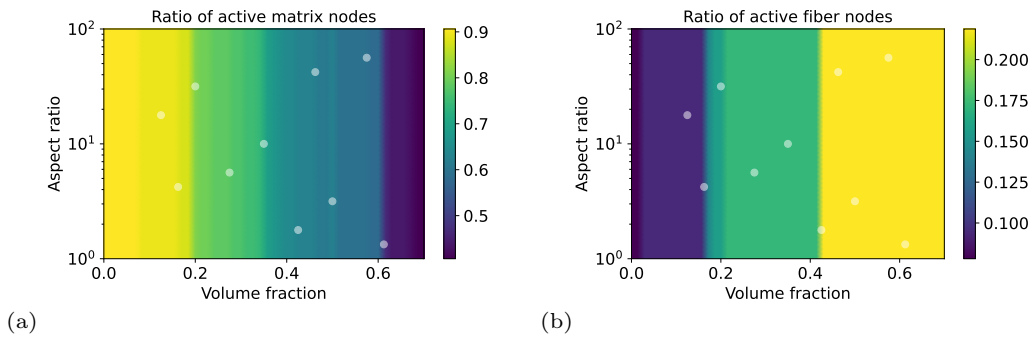


Figure 40: Variation of the ratio of active DMN nodes in the parametric space: (a) matrix; (b) fiber. The training points are indicated by circles.

In Fig. 41(a), the training, validation and test errors are presented for MIPDMN with 7 or 9 layers. These errors are first computed at each of the 20 sampling points, then aggregated using quantiles. All these errors become lower with more DMN layers. Overfitting is not observed since the validation errors are comparable with the training ones. For the training and validation errors, their statistical variations are limited. In the example of a 7-layer MIPDMN, the 0.9-quantile error is less than 4% while the median error value is less than 2%. However, the test errors present larger statistical dispersion since their 0.9-quantiles reach approximately 10%. Due to a large amount of test points in the parametric space (see Fig. 37), MIPDMN is being evaluated both with interpolation and

extrapolation. The 0.9-quantile of the test errors is mainly due to the presence of “outliers” at some particularly challenging test microstructures. The median test errors, nevertheless, are only slightly larger than the training and the validation errors, as before.

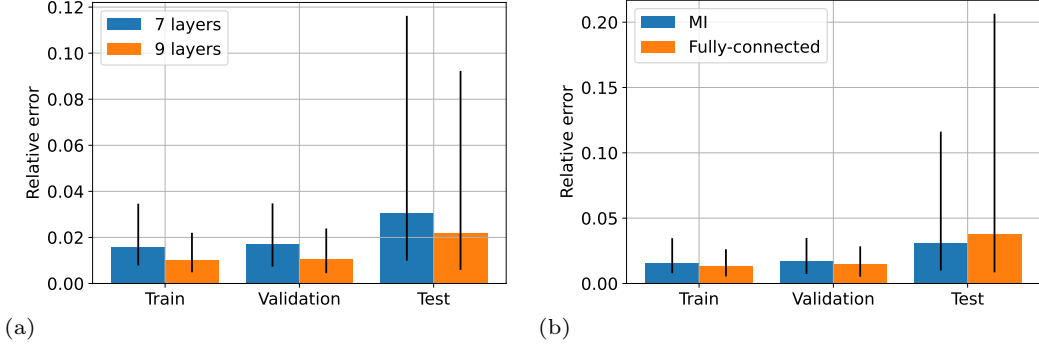


Figure 41: Training, validation and test errors of parametric DMN models: (a) comparison between 7 and 9 layers using the micromechanics-informed (MI) architecture; (b) comparison with the fully-connected architecture using 7 layers.

These errors are also computed for a fully-connected architecture Eq. (20) with 7 DMN layers. Compared to the MI one Eq. (22), now the DMN fitting parameters become function of both microstructural parameters $\mathbf{w} = \mathbf{w}(\text{vf}, \text{ar})$ and $\boldsymbol{\theta} = \boldsymbol{\theta}(\text{vf}, \text{ar})$. The physical constraints Eq. (25) and Eq. (32) are included. According to Fig. 41(b), not only the fully-connected architecture does not significantly reduce training and validation errors, it also nearly doubles the 0.9-quantile of the test errors. The MI architecture Eq. (22) ensures hence a comparable expressive power in the parametric space with certain generalization capability.

In Fig. 42, the median errors are first computed at each of the 20 training and test sampling points, then interpolated and extrapolated to the whole parametric space using Kriging implemented in the SMT library [53]. In Fig. 42(a), the maximum error (10%) is localized at the “outlier” microstructure \mathbf{p}_{16} with a very high aspect ratio (87) compared to the training domain. Except this point, the median errors are between 1% and 4% and are visually uniform in the parametric space. The fully-connected architecture, however, produces more error variations in the parametric space. This demonstrates again the satisfying interpolation and extrapolation generalization capability of the MI architecture.

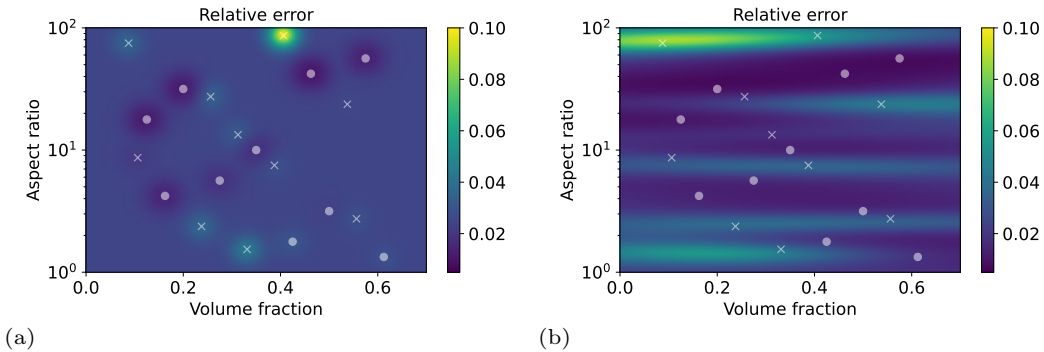


Figure 42: Median errors of parametric DMN models in the parametric space: (a) micromechanics-informed; (b) fully-connected. The training and test points are respectively indicated by circles and crosses.

With the real linear elastic properties in Tab. 4, the vf-dependence of the homogenized elastic moduli at fixed aspect ratio $\text{ar} = 20$ are illustrated in Fig. 43 using a 9-layer MIpDMN. A perfect agreement is found with the additional test FE-RVE simulation data, knowing that they all correspond to previously unseen microstructures during training. The microstructure $\text{vf} = 0.1$ lies even outside the training domain. Among the data points, the maximum relative Frobenius-norm error on $\bar{\mathbb{C}}$ is only 3.3% at $\text{vf} = 0.6$.

In Fig. 44, the influence of the aspect ratio parameter is analyzed with a fixed $\text{vf} = 0.4$ and compared with the FE-RVE results on these new microstructures. Not only MIpDMN is capable of predicting the increase of \bar{E}_1 with ar (at least in the training domain), it can also account for the more subtle variations of $\bar{\nu}_{12}$ and $\bar{\mu}_{23}$, which are all in good agreement with the FE-RVE data. Outside the training domain for ar , good predictions are obtained at

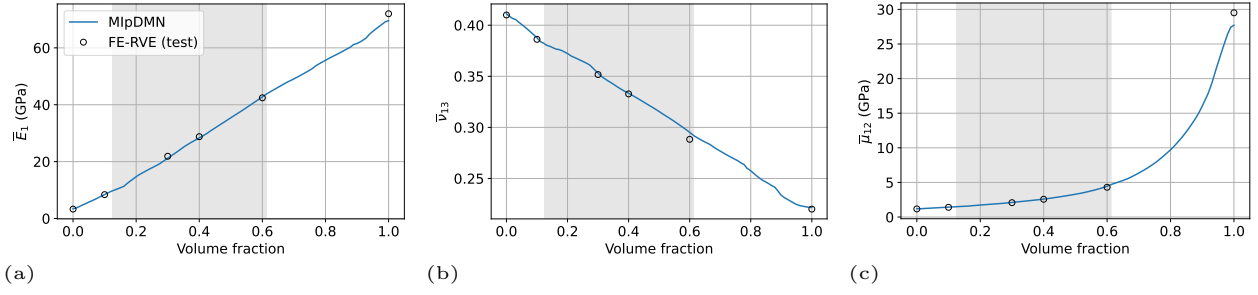


Figure 43: Homogenized elastic moduli prediction as a function of vf at fixed $ar = 20$ using a 9-layer MIPDMN: (a) longitudinal Young's modulus \bar{E}_1 ; (b) transverse Poisson ratio $\bar{\nu}_{13}$; (c) transverse shear modulus $\bar{\mu}_{12}$. The training domain is represented by a gray shadowed region. The theoretical values at $vf = 0$ and $vf = 1$ are indicated as test data.

$ar = 1$, while some deviations are found near $ar = 100$, where the relative Frobenius-norm error is 5.9%. While the physical constraints can indeed improve interpolative and extrapolative generalization capability of MIPDMN, they cannot replace the input training data. If the prediction at higher aspect ratios is important, more corresponding FE-RVE data could be included during training.

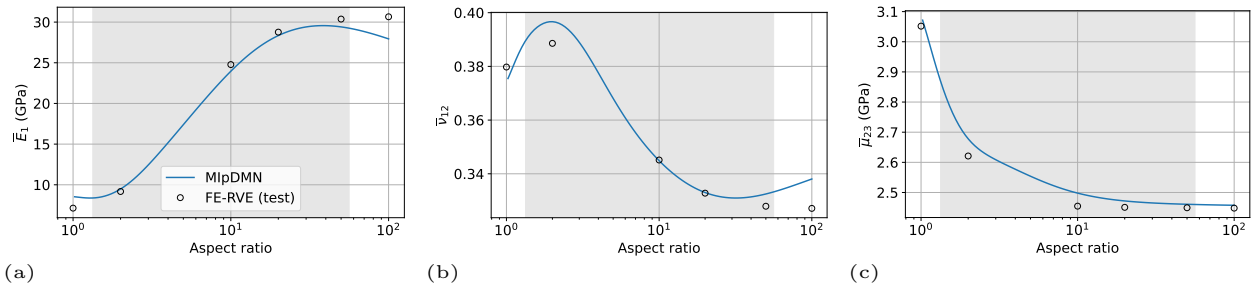


Figure 44: Homogenized elastic moduli prediction as a function of ar at fixed $vf = 0.4$ using a 9-layer MIPDMN: (a) longitudinal Young's modulus \bar{E}_1 ; (b) transverse Poisson ratio $\bar{\nu}_{12}$; (c) in-plane shear modulus $\bar{\mu}_{23}$. The training domain is represented by a gray shadowed region.

Using the real thermal conductivity and CTE properties of the constituents provided previously in Tab. 4, we will now use MIPDMN trained with isothermal linear elastic data to predict the effective thermal conductivity and the effective CTE. In Fig. 45, the nonlinear vf - and ar -dependences of \bar{k} are respectively well predicted. A satisfying agreement with the FE-RVE results is obtained for both the longitudinal and transverse components. Among the data points, the maximum relative Frobenius-norm error is 0.64% at $vf = 0.4$ with fixed $ar = 20$, and 1.1% at $ar = 1$ with fixed $vf = 0.4$. This remarkable result illustrates again the multiple physics property prediction capability of DMN, knowing that the thermal conductivity and the CTE data are not used as training data at all.

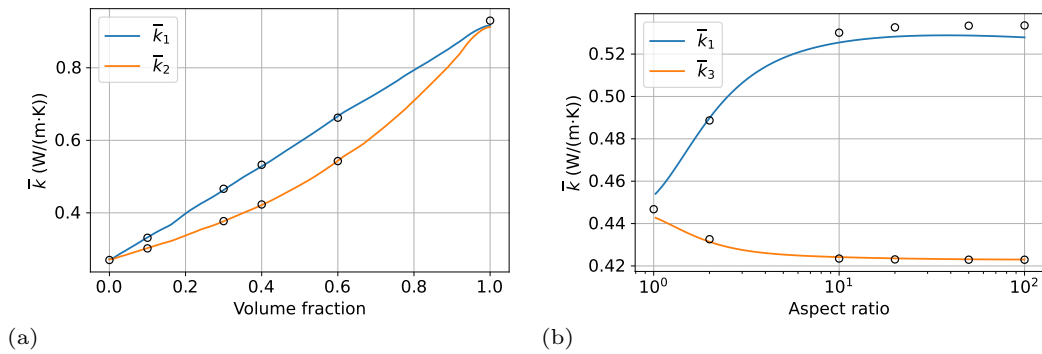


Figure 45: Effective thermal conductivity prediction using a 9-layer MIPDMN for the longitudinal \bar{k}_1 and transverse (\bar{k}_2, \bar{k}_3) components: (a) as a function of vf at fixed $ar = 20$; (b) as a function of ar at fixed $vf = 0.4$. The FE-RVE results and the theoretical values for $vf = 0$ and $vf = 1$ are indicated by circles.

In Fig. 46, the effective CTE is predicted while varying vf and ar . As before, a good agreement is found between

the MIPDMN prediction and the FE-RVE results. Among the data points, the maximum relative Frobenius-norm error is 2.2% at $vf = 0.6$ with fixed $ar = 20$, and 4.6% at $ar = 1$ with fixed $vf = 0.4$.

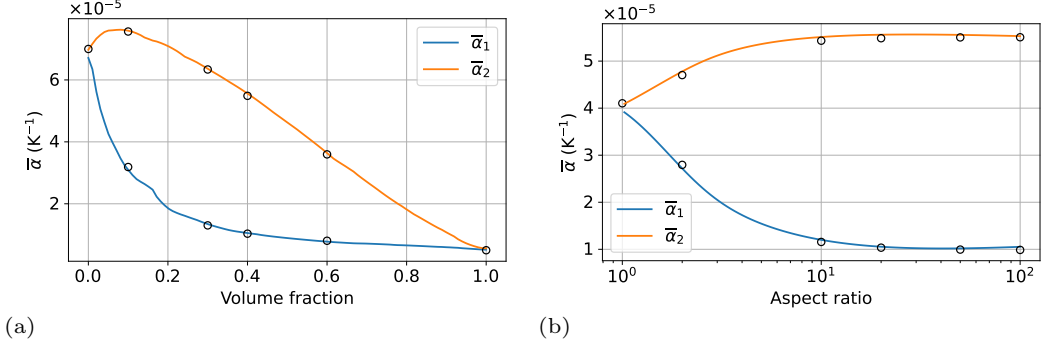


Figure 46: Effective CTE prediction using a 9-layer MIPDMN for the longitudinal $\bar{\alpha}_1$ and transverse $\bar{\alpha}_2$ components: (a) as a function of vf at fixed $ar = 20$; (b) as a function of ar at fixed $vf = 0.4$. The FE-RVE results and the theoretical values for $vf = 0$ and $vf = 1$ are indicated by circles.

Nonlinear mechanical simulations with an elasto-plastic matrix

Using the online prediction formulation described in Appendix A, we will now evaluate MIPDMN with nonlinear mechanical material behaviors while varying the microstructural parameters. We assume that the polypropylene matrix follows an elasto-plastic law with an isotropic power-law hardening

$$\sigma_Y = \sigma_0 + k(p + \epsilon)^n,$$

where σ_Y defines the yield surface, σ_0 is the yield stress, and (k, n) describes strain hardening with the equivalent plastic strain p . The following numerical values are used: $\sigma_0 = 30$ MPa, $k = 293$ MPa and $n = 0.34$. The small value $\epsilon = 10^{-6}$ is used to avoid infinite derivative $d\sigma_Y/dp$ at $p = 0$. The following proportional cyclic multiaxial loading–unloading path is considered in the strain space

$$\bar{\epsilon}(t) = f(t)\bar{\epsilon}_0, \quad \bar{\epsilon}_0 = (\bar{\epsilon}_{11}, \bar{\epsilon}_{22}, \bar{\epsilon}_{33}, \sqrt{2}\bar{\epsilon}_{12}, 0, 0), \quad \bar{\epsilon}_{11} = 1\%, \bar{\epsilon}_{22} = \bar{\epsilon}_{12} = 4\%, \bar{\epsilon}_{33} = -2\%. \quad (36)$$

The cyclic function $f(t)$ defined on $[0, T]$ is presented in Fig. 47. The time parameter t is assumed dimensionless since quasi-static loading is considered.

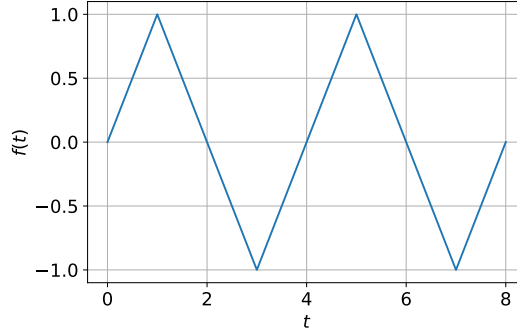


Figure 47: Cyclic function $f(t)$ used in Eq. (36).

In Fig. 48, the corresponding temporal stress responses are presented for different volume fraction values. The influence of the vf -parameter on the nonlinear effective responses is well captured by MIPDMN. With increasing volume fraction, the maximum value of $\bar{\sigma}_{22}$ increases from approximately 250 MPa to nearly 600 MPa. Furthermore, more plasticity effects are visible especially on the $\bar{\sigma}_{12}$ components. Good agreement between MIPDMN and the FE-RVE results is observed, even though these microstructures are previously unseen during offline training. The difference between them can be quantified with the following error measure on the full stress tensor with the Frobenius norm

$$e = \frac{\int_0^T \|y^{\text{DMN}}(t) - y^{\text{FE}}(t)\| dt}{\int_0^T \|y^{\text{FE}}(t)\| dt}. \quad (37)$$

For $vf = 0.1$ and $vf = 0.6$, the errors are only respectively 2.2% and 5.6%, considering that these two microstructures lie on the boundary of or even outside the parametric training domain.

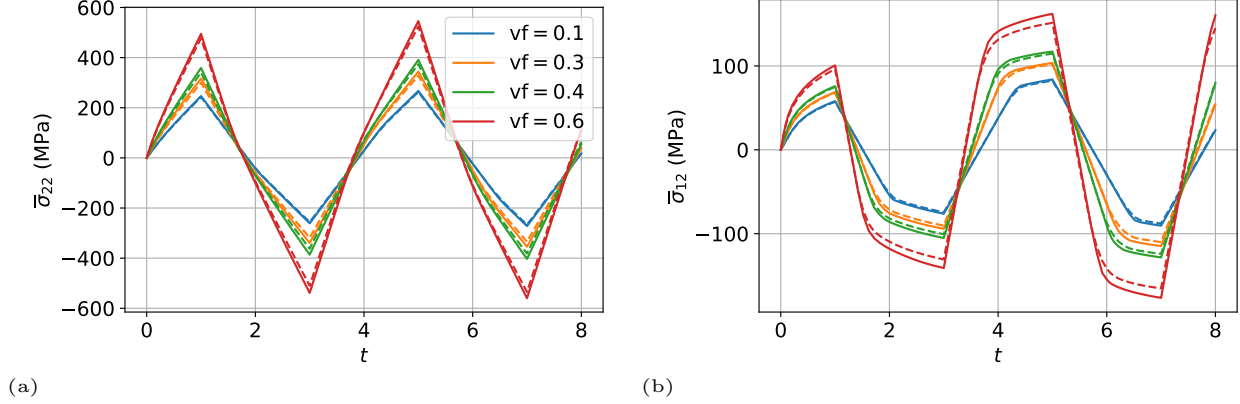


Figure 48: Effective nonlinear behavior of the ellipsoidal inclusion composite for different fiber volume fractions at fixed $ar = 20$ using a 9-layer MIpDMN: (a) $\bar{\sigma}_{22}$; (b) $\bar{\sigma}_{12}$. The DMN predictions are represented by solid lines, while the FE-RVE data are dashed lines.

The ar -dependence of the nonlinear effective behaviors is then analyzed in Fig. 49. The aspect ratio parameter has a bigger influence on the longitudinal effective stress, which increases from 300 MPa to nearly 500 MPa when ar varies from 1 (spherical inclusions) to 50 (slender fibers). The transverse shear stress $\bar{\sigma}_{12}$ seems to converge within the given ar range. Again, MIpDMN is in good agreement with the FE-RVE responses, even though a visible difference can be noticed on $\bar{\sigma}_{12}$ for $ar = 1$. For this microstructure outside the parametric training domain, the error is 8.6% using Eq. (37).

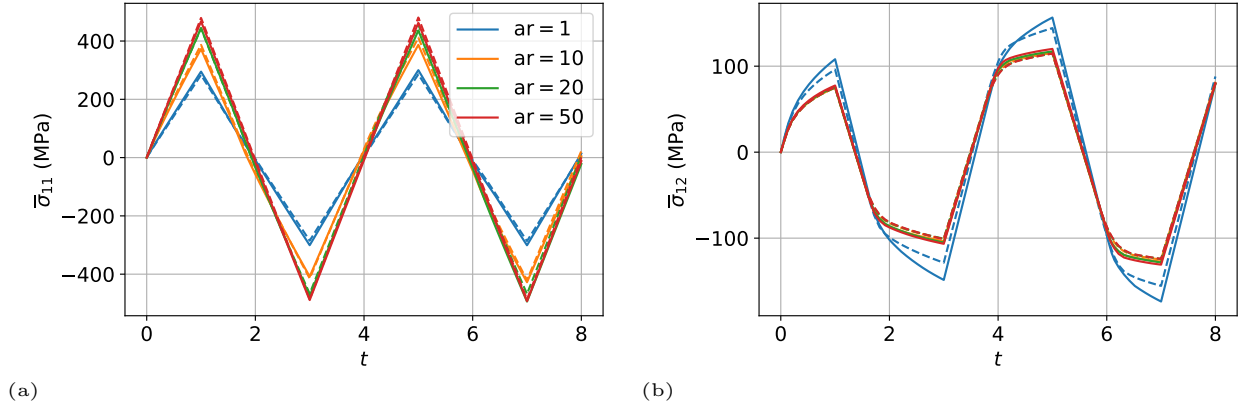


Figure 49: Effective nonlinear behavior of the ellipsoidal inclusion composite for different fiber aspect ratios at fixed $vf = 0.4$ using a 9-layer MIpDMN: (a) $\bar{\sigma}_{11}$; (b) $\bar{\sigma}_{12}$. The DMN predictions are represented by solid lines, while the FE-RVE data are dashed lines.

To further demonstrate the nonlinear prediction capability of MIpDMN, the volume-averaged equivalent plastic strain is computed and compared with the FE-RVE results in Fig. 50(a). Two microstructures outside the parametric training domain, *i.e.* extrapolating (vf, ar) , are considered. For DMN, volume-averaging is performed on p_i stored on the material nodes using their weights. Using Eq. (37) with the absolute-value norm, the errors are only 2.3% for $vf = 0.1$, $ar = 20$ and 3.1% for $vf = 0.4$, $ar = 100$.

Inspired by the work of [20], we investigate the energy balance and consistency of MIpDMN during nonlinear online prediction. The (average) mechanical power inside an increment $[t_n, t_{n+1}]$ of duration Δt can be computed by

$$\mathcal{P} = \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} \bar{\boldsymbol{\sigma}} \cdot \dot{\bar{\boldsymbol{\varepsilon}}} dt = \frac{1}{2\Delta t} ((\bar{\boldsymbol{\sigma}}_n + \bar{\boldsymbol{\sigma}}_{n+1}) \cdot \Delta \bar{\boldsymbol{\varepsilon}}). \quad (38)$$

In Fig. 50(b), The MIpDMN mechanical power evolution is compared with the FE-RVE one. A relatively good agreement is also found for these two extrapolating microstructures, since Eq. (37) give respectively 2.8% and 8.7%. For DMN, the effective mechanical power Eq. (38) can also be retrieved using the weighted-average of \mathcal{P}_i on the

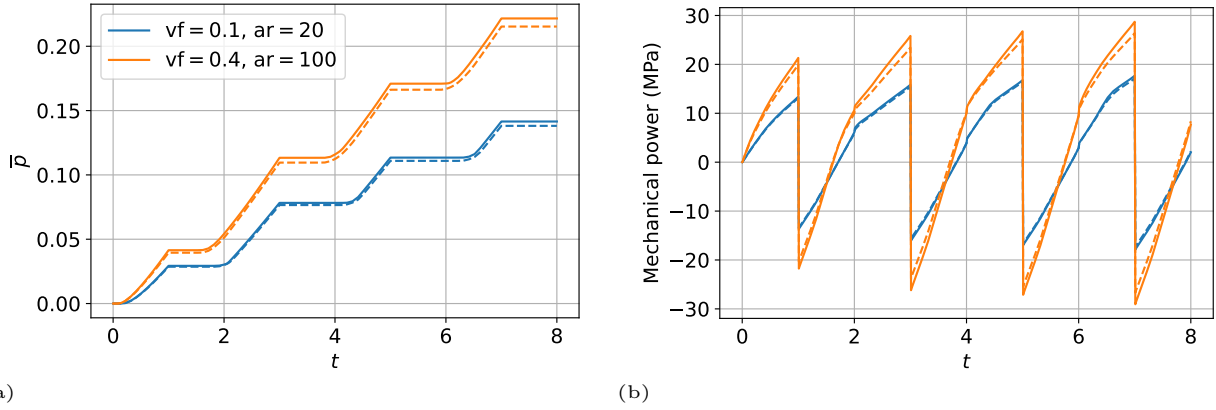


Figure 50: (a) Volume-averaged equivalent plastic strain; (b) Mechanical power. The DMN predictions are represented by solid lines, while the FE-RVE data are dashed lines.

material nodes with their local stress and strain states [20]. The errors between this material nodes-based approach and Eq. (38) are only 0.70% for $vf = 0.1, ar = 20$ and 1.1% for $vf = 0.4, ar = 100$. This demonstrates the energetic consistency of MIPDMN even when extrapolating the microstructural parameters.

Finally, the computational efficiency of MIPDMN is presented in Tab. 7 for the previous two microstructures. Even with a 9-layer architecture, a speed-up of approximately 700 is obtained compared to FE-RVE. Similar factors have also been reported for instance in [9, 10, 20, 16]. We also recall that the generation of the costly training data on new microstructures even outside the parametric training domain is also no longer needed with MIPDMN.

Wall time (speed-up)	FE-RVE	9-layer MIPDMN
$vf = 0.1, ar = 20$	3083 s	4.10 s (752)
$vf = 0.4, ar = 100$	3723 s	5.69 s (654)

Table 7: Computational speed-up of MIPDMN compared to FE-RVE for the two extrapolating microstructures during the nonlinear online prediction.

5. Conclusions

A novel micromechanics-informed parametric deep material network (MIPDMN) architecture is proposed for heterogeneous materials with a varying morphology described by multiple parameters. The dependence of the DMN fitting parameters on the microstructural ones is accounted for by a single layer feedforward neural network informed by micromechanical properties Eq. (22). While the DMN weights only vary with the scalar volume fraction parameter Eq. (22a), the DMN rotations are influenced by other purely morphological parameters that do not change the volume fraction Eq. (22b). Furthermore, two additional micromechanical constraints Eq. (25) and Eq. (32) are prescribed so that micromechanics-informed neural network recovers the volume fraction and the material frame orientation tensors of the actual parameterized microstructure. A unique offline training of MIPDMN is required, based on a total loss function Eq. (34) that aggregates the FE-RVE data at some given microstructural parameter values. The trained MIPDMN is then able to predict the material behaviors in the parametric space, *i.e.*, even for previously unseen microstructures during training. The time-consuming generation of the corresponding new computational homogenization data is hence no longer needed.

Based on the numerical results reported, the micromechanics-informed architecture as well as the constraints improve generalization capabilities in the parametric space while successfully capturing the structure-property relationships. The linear and nonlinear effective behaviors of the *parameterized* microstructure in question can hence be accurately predicted by MIPDMN. Besides, we show that the inclusion of initial material rotation matrices on the input layer does not significantly increase the expressive power even for microstructures with local material orientation.

MIPDMN is also recast in a multiple physics setting. Through the redefinition of the laminate homogenization function, other physical properties such as the thermal conductivity and the coefficient of thermal expansion can be accurately predicted at the online prediction stage. Based on the numerical simulations reported, the real relative

error on $\bar{\alpha}$ is of the same order of magnitude as $\bar{\mathbb{C}}$ (approximately 1% to 6%), while that on $\bar{\mathbf{k}}$ is in general near or less than 1%. This is a remarkable result since MIpDMN is trained using isothermal linear elastic data. It demonstrates that our MIpDMN learns the *parameterized* microstructure *per se*, and not a physical property in particular.

The limitations of this work as well as some possible future directions are summarized as follows:

- Even though Eq. (22) extends naturally to higher parametric dimensions, at most only two microstructural parameters are considered in the numerical simulations. The microstructures with more morphological parameters will be investigated in the future to demonstrate its full potential with respect to the existing approaches.
- A single-layer architecture is considered in Eq. (22). For more complex structure-property relationships, it can be expected that $\mathbf{q} \mapsto \boldsymbol{\theta}(\mathbf{q})$ in Eq. (22b) may become nonlinear and require hidden layers and weight regularization. It will be investigated in our future work.
- The MIpDMN formulation presented in this paper mainly targets two-phase parameterized microstructures. For polycrystalline materials, the partition Eq. (21) needs to be redefined knowing that now the 1-point correlation function becomes the crystallite orientation distribution function (CODF), see [41]. Hence, it suggests that in the MI architecture Eq. (22), now the DMN weights would become a function of CODF. Interestingly, it has been numerically found in [9] that the \mathbf{w} indeed remain uniformly distributed for a uniform (totally random) CODF and vary with the texture information.
- For multiphase materials, the non-parametric DMN architecture can be first generalized by considering multiphase laminates as building blocks [9, 10] or an interaction-based network [19]. Then, the partition Eq. (21) can be adapted by using correlation functions for multiphase materials. For instance, the single vf parameter would be replaced by $P - 1$ volume fractions for a P -phase microstructure. Also, the physical constraints Eq. (25) and Eq. (32) need to be accordingly extended for the additional constituents.
- In this work, we focused on the extension of the original DMN formulation to parameterized microstructures. Hence, much more complex constitutive behaviors such as viscoplasticity [16] are not considered. Furthermore, accounting for localized material failure such as fracture is also challenging especially for DMN. Compared to other reduced-order models such as the self-consistent clustering analysis [54], spatial fields in the RVE are volume-averaged to the DMN material nodes. In [15], a cell division scheme is proposed for consistent scale transition in DMN with promising results. It would be interesting in the future to couple it with non-local damage models such as phase-field fracture [55] to tackle multiscale strain-localization problems.
- We investigated the use of MIpDMN trained using stiffness tensors \mathbb{C} to predict other physical properties such as thermal conductivity \mathbf{k} . Our preliminary results not reported here indicate that MIpDMN trained with \mathbf{k} can also predict \mathbb{C} , with a slightly decreased accuracy. Future work will be devoted to the offline training of MIpDMN using *combined* multiple physics data.

Acknowledgments

The author thanks Rafael SALAZAR TIO for the fruitful discussions on using DMN in a multiple physics setting. During her internship, Maëlle PRAUD helped with investigating various optimization algorithms.

Appendix A. Nonlinear homogenization formulation for online prediction

Consider a time interval $[t_n, t_{n+1}]$ and an increment in the macroscopic strain from $\bar{\boldsymbol{\varepsilon}}_n$ to $\bar{\boldsymbol{\varepsilon}}_{n+1} = \bar{\boldsymbol{\varepsilon}}_n + \Delta\bar{\boldsymbol{\varepsilon}}$. As explained in Sect. 2.1, a material node is attached to each phase of the leaf laminates, see Fig. 2. At time t_n , we suppose that all their material state variables (strains, stresses and internal state variables) are known. The objective of the DMN surrogate during nonlinear online prediction is to compute at t_{n+1} the new homogenized stress tensor and the consistent tangent operator

$$\Delta\bar{\boldsymbol{\varepsilon}} \mapsto (\bar{\mathbb{C}}, \bar{\boldsymbol{\sigma}}_{n+1}). \quad (\text{A.1})$$

As for mean-field homogenization approaches, a linearization scheme is required [1] to linearize nonlinear constitutive laws. In [9], an incrementally affine formulation similar to [56] is proposed for DMN. In this work, we

continue to use this nonlinear homogenization formulation, since it leads to a fixed-point problem that can be efficiently solved by acceleration methods [30].

The incrementally affine formulation assumes that for each material node, the stress increment in this time interval can be computed as follows

$$\boldsymbol{\sigma}_{n+1} - \boldsymbol{\sigma}_n = \Delta\boldsymbol{\sigma} = \mathbb{C}\Delta\boldsymbol{\varepsilon} + \delta\boldsymbol{\sigma}, \quad (\text{A.2})$$

where $\Delta\boldsymbol{\varepsilon}$ is the strain increment for this material node, $\delta\boldsymbol{\sigma}$ the residual stress increment which measures material nonlinearity in this time interval and \mathbb{C} is the consistent tangent operator. Note that in Eq. (A.2), the strain increment $\Delta\boldsymbol{\varepsilon}$ is also unknown, since only the increment in the macroscopic strain tensor $\Delta\bar{\boldsymbol{\varepsilon}}$ is known. A consequence of Eq. (A.2) is that the effective stress tensor increment can also be put in this incrementally affine form

$$\bar{\boldsymbol{\sigma}}_{n+1} - \bar{\boldsymbol{\sigma}}_n = \Delta\bar{\boldsymbol{\sigma}} = \bar{\mathbb{C}}\Delta\bar{\boldsymbol{\varepsilon}} + \bar{\delta}\boldsymbol{\sigma}. \quad (\text{A.3})$$

The computation of Eq. (A.1) involves the following three *coupled* operations in the DMN architecture, see [9].

1. Forward homogenization: The consistent tangent operators \mathbb{C} and residual stresses $\delta\boldsymbol{\sigma}$ defined on each material node are homogenized from the leaf laminates to the root laminate, obtaining the homogenized tensors $\bar{\mathbb{C}}_l$ and $\bar{\delta}\boldsymbol{\sigma}_l$ for each laminate

$$\text{Forward} : (\mathbb{C}_i, \delta\boldsymbol{\sigma}_i)_{1 \leq i \leq N} \mapsto (\bar{\mathbb{C}}_l, \bar{\delta}\boldsymbol{\sigma}_l). \quad (\text{A.4})$$

2. Backward localization: The strain increment for each material node $\Delta\boldsymbol{\varepsilon}_i$ is computed based on the macroscopic strain increment $\Delta\bar{\boldsymbol{\varepsilon}}$ and the strain concentration tensors defined in each laminate obtained during forward homogenization, from the root laminate to the leaf laminates

$$\text{Backward} : (\bar{\mathbb{C}}_l, \bar{\delta}\boldsymbol{\sigma}_l; \Delta\bar{\boldsymbol{\varepsilon}}) \mapsto (\Delta\boldsymbol{\varepsilon}_i)_{1 \leq i \leq N}. \quad (\text{A.5})$$

3. Constitutive behavior integration: For each material node, \mathbb{C}_i and $\delta\boldsymbol{\sigma}_i$ are computed by integrating constitutive behaviors, given their respective strain increment $\Delta\boldsymbol{\varepsilon}_i$.

$$\text{MatInt} : (\Delta\boldsymbol{\varepsilon}_i)_{1 \leq i \leq N} \mapsto (\mathbb{C}_i, \delta\boldsymbol{\sigma}_i)_{1 \leq i \leq N}. \quad (\text{A.6})$$

These coupled equations Eq. (A.4), Eq. (A.5) and Eq. (A.6) naturally give rise to a *vectorial* fixed point problem $\underline{x} = \mathbf{F}(\underline{x})$ on the flattened vector of strain increments $\underline{x} = (\Delta\boldsymbol{\varepsilon}_i)_{1 \leq i \leq N}$ for the time interval $[t_n, t_{n+1}]$

$$\underline{x} = \underbrace{(\text{Backward} \circ \text{Forward} \circ \text{MatInt})}_{\mathbf{F}}(\underline{x}) \implies \underline{x}^{j+1} = \mathbf{F}(\underline{x}^j) \quad (\text{A.7})$$

where $j \geq 0$ is the fixed-point iteration number.

In this work, acceleration methods [30] are used to accelerate the convergence of the fixed-point problem Eq. (A.7). In particular, Aitken relaxation [57] is applied as a post-processing step in Eq. (A.7)

$$\begin{aligned} \underline{x}^{j+1} &= \mathbf{F}(\underline{x}^j) + (\omega^j - 1)\underline{r}^j, \quad \underline{r}^j = \mathbf{F}(\underline{x}^j) - \underline{x}^j \\ \omega^j &= \begin{cases} \omega^0 & j = 0 \\ -\omega^{j-1} \frac{(\underline{r}^{j-1}) \cdot (\underline{r}^j - \underline{r}^{j-1})}{\|\underline{r}^j - \underline{r}^{j-1}\|^2} & j \geq 1 \end{cases} \end{aligned} \quad (\text{A.8})$$

In Eq. (A.8), after updating the relaxation parameter ω^j , it is then numerically bounded by $[\omega_{\min}, \omega_{\max}]$. The algorithm that realizes nonlinear homogenization Eq. (A.1) in $[t_n, t_{n+1}]$ is summarized as follows.

- Given macroscopic strain increment $\Delta\bar{\boldsymbol{\varepsilon}}$ with $\|\Delta\bar{\boldsymbol{\varepsilon}}\| > \epsilon$. In this work we use $\epsilon = 10^{-8}$.
- Initialize the strain increments \underline{x}^0 on material nodes, by applying backward localization Eq. (A.5) with the strain concentration tensors computed at the last time increment t_n . For the first time increment t_0 , each material node and laminate is initialized using linear elastic properties.
- For the fixed-point iteration $j \geq 0$:
 1. Integrate constitutive behaviors for each material node Eq. (A.6), obtaining $(\mathbb{C}_i, \delta\boldsymbol{\sigma}_i)_{1 \leq i \leq N}$.
 2. Forward homogenization Eq. (A.4), obtaining $(\bar{\mathbb{C}}_l, \bar{\delta}\boldsymbol{\sigma}_l)$.
 3. Backward localization Eq. (A.5), obtaining $\mathbf{F}(\underline{x}^j)$.

4. Check convergence. In this work, the relative error in the L_2 norm with respect to the macroscopic strain increment is controlled

$$e_{\text{rel}} = \frac{\|\mathbf{F}(\underline{x}^j) - \underline{x}^j\|}{\|\Delta \bar{\boldsymbol{\varepsilon}}\|} < \text{rtol}. \quad (\text{A.9})$$

If convergence is reached, end the fixed-point iterations.

5. Apply Aitken relaxation Eq. (A.8), obtaining \underline{x}^{j+1} .

- Compute the effective stress tensor Eq. (A.3).

If all the material nodes remain linear elastic, only 1 fixed-point iteration is required, since $\|\mathbf{F}(\underline{x}^0) - \underline{x}^0\| = 0$ in Eq. (A.9).

We propose to numerically investigate the use of Aitken relaxation Eq. (A.8) on the convergence of the fixed-point problem Eq. (A.7), by using the cyclic multiaxial loading problem Eq. (36) on the ellipsoidal inclusion composite presented in Sect. 4.3. First, two tolerance values in Eq. (A.9) are compared in Fig. A.51, with $\text{vf} = 0.4$ and $\text{ar} = 2$. We observe that $\text{rtol} = 10^{-1}$ is sufficient to obtain converged nonlinear responses.

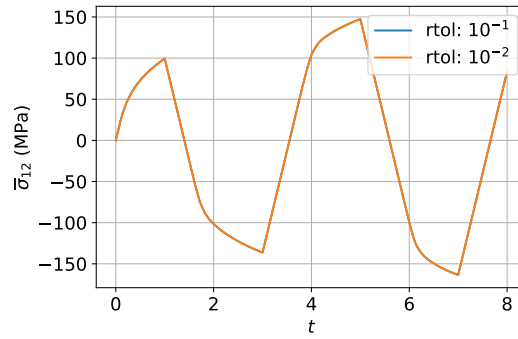


Figure A.51: Effect of the tolerance value on the obtained nonlinear effective behavior for the ellipsoidal inclusion composite.

In Fig. A.52(a), the number of fixed-point iterations is presented for each time increment. In Eq. (A.8), we use $\omega_0 = 1$, $\omega_{\min} = 1$ and $\omega_{\max} = 2$. This means we are over-relaxing the already converging fixed-point iterations. When the material behavior is linear, only one iteration is required. Otherwise, we observe an overall reduction in the number of iterations with Aitken relaxation. The total number of iterations is reduced by 15%, which leads to a reduction in the computational time by approximately 13%. Since Aitken relaxation essentially only involves a scalar product and a norm computation, it is hence much less costly than each fixed-point iteration.

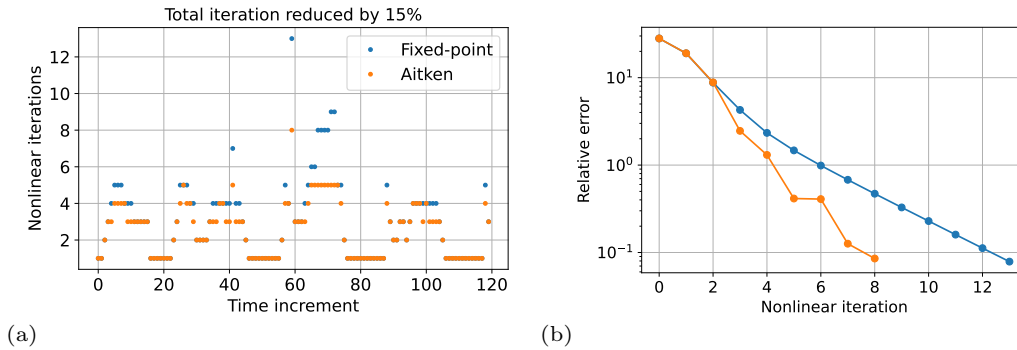


Figure A.52: Influence of Aitken relaxation on the convergence of the nonlinear fixed-point problem: (a) number of fixed-point iterations for each time increment; (b) relative error history for the increment 59.

The history of the relative error Eq. (A.9) at time increment 59 is used to illustrate the effect of Aitken relaxation, in Fig. A.52(b). Aitken relaxation accelerates convergence and the number of fixed-point iterations is reduced from 13 to 8.

Appendix B. Inverse identification of unknown material and microstructural parameters

Trained MIPDMN can serve as an accurate and computationally efficient surrogate of the parameterized microstructure. Not only it can be used in forward prediction, it can also be employed to identify *simultaneously* the *material* and *microstructural* properties in an inverse identification problem. Below, we provide a preliminary proof-of-concept study for the woven composite presented in Sect. 4.2. Since the objective here is to show that *microstructural* parameters can also be identified, only linear elastic behaviors are considered. For a more realistic inverse identification problem with complex nonlinear material responses, readers can refer to [17].

Now, the effective stiffness tensor $\bar{\mathbb{C}}$ is provided, and the elastic properties of the matrix \mathbb{C}_1 and the tows \mathbb{C}_2 as well as the volume fraction of the tows vf are to be identified. In practice $\bar{\mathbb{C}}$ can be measured experimentally. Here we use the FE-RVE simulation result $\bar{\mathbb{C}}^{\text{Data}}$ on the test microstructure $\text{vf} = 0.537$, obtained with the real properties in Tab. 5 which are now sought for.

Motivated by the fact that the gradients with respect to $(\mathbb{C}_1, \mathbb{C}_2, \text{vf})$ can be easily computed using automatic differentiation thanks to the MIPDMN architecture, in this work we adopt a gradient-based optimization approach based on a loss function Eq. (B.1) which compares DMN prediction with $\bar{\mathbb{C}}^{\text{Data}}$ using the Frobenius norm. Similar to the training strategy of MIPDMN, the Rprop optimizer is used for minimization.

$$\mathcal{L}_{\text{cal}} = \frac{\|\bar{\mathbb{C}}^{\text{DMN}} - \bar{\mathbb{C}}^{\text{Data}}\|^2}{\|\bar{\mathbb{C}}^{\text{Data}}\|^2}. \quad (\text{B.1})$$

The optimization iterations require the initial guess values for the unknowns $(\mathbb{C}_1, \mathbb{C}_2, \text{vf})$. Hence, the real properties in Tab. 5 as well as the true volume fraction $\text{vf} = 0.537$ are randomly perturbed using a normal distribution with a coefficient of variation equal to 20%. The initial values generated by two random realizations are indicated in Tab. B.8.

Matrix	E (MPa)	ν
Realization 1	4491	0.400
Realization 2	3908	0.437

Tow	vf	E_1 (GPa)	E_2 (GPa)	ν_{12}	ν_{23}	μ_{12} (GPa)
Realization 1	0.550	92.3	5.60	0.347	0.697	2.14
Realization 2	0.541	65.4	5.22	0.297	0.764	1.78

Table B.8: Initial linear elastic properties of the two phases for the inverse identification problem.

The loss histories corresponding to these two initial guess values are given in Fig. B.53. The loss function converges quickly and may reach 10^{-5} within 1000 optimization iterations (a few seconds). The effective stiffness tensor $\bar{\mathbb{C}}$ is well recovered. According to Tab. B.9, the relative error between the converged effective stiffness tensor and the $\bar{\mathbb{C}}$ data is 0.42% and 0.26% for these two sets of initial guess values.

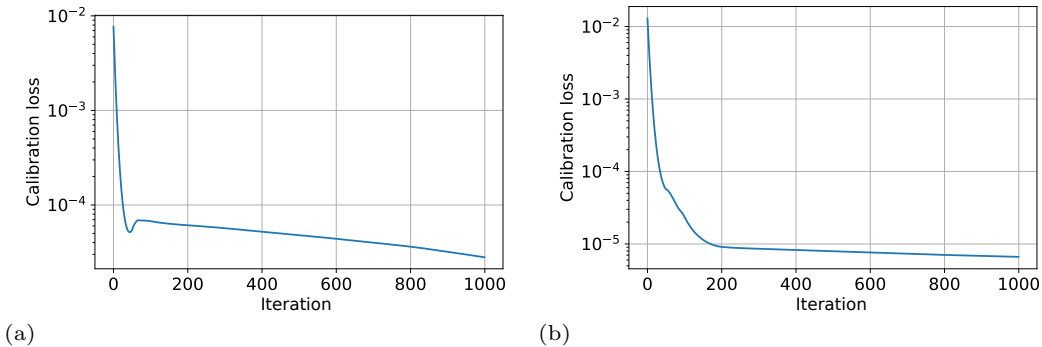


Figure B.53: Calibration loss histories using two initial guess values: (a) realization 1; (b) realization 2.

In Tab. B.10, the inversely identified input parameters $(\mathbb{C}_1, \mathbb{C}_2, \text{vf})$ are reported. The material and microstructural parameters found with the realization 1 are similar to the actual properties in Tab. 5. However, with the

$\bar{\mathbb{C}}$	$E_1 = E_2$ (GPa)	E_3 (GPa)	ν_{12}	$\nu_{13} = \nu_{23}$	μ_{12} (GPa)	$\mu_{13} = \mu_{23}$ (GPa)	Error
Data	24.4	7.03	0.0866	0.558	1.87	1.73	
Realization 1	24.5, 24.3	7.05	0.0879	0.556, 0.555	1.87	1.73	0.42%
Realization 2	24.4	7.02	0.0863	0.560, 0.558	1.86	1.74	0.26%

Table B.9: Converged homogenized stiffness tensor from two different realizations of initial values, compared with the true data.

realization 2, the identified volume fraction 0.667 is higher than the data 0.537. Meanwhile, the longitudinal Young’s modulus (64.6 GPa) of the tows is also smaller than the data (78.8 GPa). This illustrates the non-uniqueness of the inverse identification problem. Additional conditions or data (for instance, nonlinear responses) could be provided to further constrain the inverse problem.

		Matrix	E (MPa)	ν			
		Realization 1	4309	0.385			
		Realization 2	4624	0.375			

Tow	vf	E_1 (GPa)	E_2 (GPa)	ν_{12}	ν_{23}	μ_{12} (GPa)
Realization 1	0.533	80.7	5.53	0.340	0.614	2.09
Realization 2	0.667	64.6	5.27	0.364	0.617	1.90

Table B.10: Identified linear elastic properties of the two phases and the volume fraction for the inverse identification problem.

References

- [1] P. Kanouté, D. P. Boso, J. L. Chaboche, B. A. Schrefler, Multiscale methods for composites: A review, *Archives of Computational Methods in Engineering* 16 (1) (2009) 31–75. doi:10.1007/s11831-008-9028-8.
- [2] K. Matouš, M. G. Geers, V. G. Kouznetsova, A. Gillman, A review of predictive nonlinear theories for multiscale modeling of heterogeneous materials, *Journal of Computational Physics* 330 (2017) 192–220. doi:10.1016/j.jcp.2016.10.070.
- [3] J. Fish, G. J. Wagner, S. Keten, Mesoscopic and multiscale modelling in materials, *Nature Materials* 20 (6) (2021) 774–786. doi:10.1038/s41563-020-00913-0.
- [4] C. L. Tucker III, E. Liang, Stiffness predictions for unidirectional short-fiber composites: Review and evaluation, *Composites Science and Technology* 59 (5) (1999) 655–671. doi:10.1016/s0266-3538(98)00120-1.
- [5] O. Pierard, C. Friebel, I. Doghri, Mean-field homogenization of multi-phase thermo-elastic composites: a general framework and its validation, *Composites Science and Technology* 64 (10-11) (2004) 1587–1603. doi:10.1016/j.compscitech.2003.11.009.
- [6] M. Geers, V. Kouznetsova, W. Brekelmans, Multi-scale computational homogenization: Trends and challenges, *Journal of Computational and Applied Mathematics* 234 (7) (2010) 2175–2182. doi:10.1016/j.cam.2009.08.077.
- [7] Z. Liu, C. Wu, M. Koishi, A deep material network for multiscale topology learning and accelerated nonlinear modeling of heterogeneous materials, *Computer Methods in Applied Mechanics and Engineering* 345 (2019) 1138–1168. doi:10.1016/j.cma.2018.09.020.
- [8] S. Kumar, D. M. Kochmann, What machine learning can do for computational solid mechanics, in: *Current Trends and Open Problems in Computational Mechanics*, Springer International Publishing, 2022, pp. 275–285. doi:10.1007/978-3-030-87312-7_27.
- [9] Z. Liu, C. Wu, Exploring the 3d architectures of deep material network in data-driven multiscale mechanics, *Journal of the Mechanics and Physics of Solids* 127 (2019) 20–46. doi:10.1016/j.jmps.2019.03.004.

- [10] S. Gajek, M. Schneider, T. Böhlke, On the micromechanics of deep material networks, *Journal of the Mechanics and Physics of Solids* 142 (2020) 103984. doi:10.1016/j.jmps.2020.103984.
- [11] S. Gajek, M. Schneider, T. Böhlke, An FE-DMN method for the multiscale analysis of short fiber reinforced plastic components, *Computer Methods in Applied Mechanics and Engineering* 384 (2021) 113952. doi:10.1016/j.cma.2021.113952.
- [12] S. Gajek, M. Schneider, T. Böhlke, An FE-DMN method for the multiscale analysis of thermomechanical composites, *Computational Mechanics* 69 (5) (2022) 1087–1113. doi:10.1007/s00466-021-02131-0.
- [13] H. Wei, C. T. Wu, W. Hu, T.-H. Su, H. Oura, M. Nishi, T. Naito, S. Chung, L. Shen, LS-DYNA machine learning-based multiscale method for nonlinear modeling of short fiber-reinforced composites, *Journal of Engineering Mechanics* 149 (3) (mar 2023). doi:10.1061/jenmdt.emeng-6945.
- [14] Z. Liu, Deep material network with cohesive layers: Multi-stage training and interfacial failure analysis, *Computer Methods in Applied Mechanics and Engineering* 363 (2020) 112913. doi:10.1016/j.cma.2020.112913.
- [15] Z. Liu, Cell division in deep material networks applied to multiscale strain localization modeling, *Computer Methods in Applied Mechanics and Engineering* 384 (2021) 113914. doi:10.1016/j.cma.2021.113914.
- [16] A. P. Dey, F. Welschinger, M. Schneider, S. Gajek, T. Böhlke, Training deep material networks to reproduce creep loading of short fiber-reinforced thermoplastics with an inelastically-informed strategy, *Archive of Applied Mechanics* 92 (9) (2022) 2733–2755. doi:10.1007/s00419-022-02213-2.
- [17] A. P. Dey, F. Welschinger, M. Schneider, S. Gajek, T. Böhlke, Rapid inverse calibration of a multiscale model for the viscoplastic and creep behavior of short fiber-reinforced thermoplastics based on deep material networks, *International Journal of Plasticity* 160 (2023) 103484. doi:10.1016/j.ijplas.2022.103484.
- [18] L. Wu, L. Adam, L. Noels, Micro-mechanics and data-driven based reduced order models for multi-scale analyses of woven composites, *Composite Structures* 270 (2021) 114058. doi:10.1016/j.compstruct.2021.114058.
- [19] V. D. Nguyen, L. Noels, Micromechanics-based material networks revisited from the interaction viewpoint: Robust and efficient implementation for multi-phase composites, *European Journal of Mechanics - A/Solids* 91 (2022) 104384. doi:10.1016/j.euromechsol.2021.104384.
- [20] V. D. Nguyen, L. Noels, Interaction-based material network: A general framework for (porous) microstructured materials, *Computer Methods in Applied Mechanics and Engineering* 389 (2022) 114300. doi:10.1016/j.cma.2021.114300.
- [21] B. Halphen, Q. S. Nguyen, Sur les matériaux standard généralisés, *Journal de Mécanique* 14 (1) (1975) 39–63.
- [22] G. Allaire, *Shape Optimization by the Homogenization Method*, Springer New York, 2002. doi:10.1007/978-1-4684-9286-6.
- [23] G. W. Milton, *The Theory of Composites*, Cambridge University Press, 2002. doi:10.1017/cbo9780511613357.
- [24] Z. Hashin, S. Shtrikman, On some variational principles in anisotropic and nonhomogeneous elasticity, *Journal of the Mechanics and Physics of Solids* 10 (4) (1962) 335–342. doi:10.1016/0022-5096(62)90004-2.
- [25] G. A. Francfort, F. Murat, Homogenization and optimal bounds in linear elasticity, *Archive for Rational Mechanics and Analysis* 94 (4) (1986) 307–334. doi:10.1007/bf00280908.
- [26] Z. Liu, C. T. Wu, M. Koishi, Transfer learning of deep material network for seamless structure-property predictions, *Computational Mechanics* 64 (2) (2019) 451–465. doi:10.1007/s00466-019-01704-4.
- [27] T. Huang, Z. Liu, C. Wu, W. Chen, Microstructure-guided deep material network for rapid nonlinear material modeling and uncertainty quantification, *Computer Methods in Applied Mechanics and Engineering* 398 (2022) 115197. doi:10.1016/j.cma.2022.115197.
- [28] S. G. Advani, C. L. Tucker, The use of tensors to describe and predict fiber orientation in short fiber composites, *Journal of Rheology* 31 (8) (1987) 751–784. doi:10.1122/1.549945.

- [29] J. Diebel, Representing attitude: Euler angles, unit quaternions, and rotation vectors, *Matrix* 58 (15-16) (2006) 1–35.
- [30] I. Ramière, T. Helfer, Iterative residual-based vector methods to accelerate fixed point iterations, *Computers and Mathematics with Applications* 70 (9) (2015) 2210–2226. doi:10.1016/j.camwa.2015.08.025.
- [31] B. W. Rosen, Z. Hashin, Effective thermal expansion coefficients and specific heats of composite materials, *International Journal of Engineering Science* 8 (2) (1970) 157–173. doi:10.1016/0020-7225(70)90066-2.
- [32] Y. Benveniste, G. J. Dvorak, On a correspondence between mechanical and thermal effects in two-phase composites, in: *Micromechanics and Inhomogeneity*, Springer New York, 1990, pp. 65–81. doi:10.1007/978-1-4613-8919-4_4.
- [33] A. Gokhale, A. Tewari, H. Garmestani, Constraints on microstructural two-point correlation functions, *Scripta Materialia* 53 (8) (2005) 989–993. doi:10.1016/j.scriptamat.2005.06.013.
- [34] S. R. Niezgodna, Y. C. Yabansu, S. R. Kalidindi, Understanding and visualizing microstructure and microstructure variance as a stochastic process, *Acta Materialia* 59 (16) (2011) 6387–6400. doi:10.1016/j.actamat.2011.06.051.
- [35] M. V. Pathan, S. A. Ponnusami, J. Pathan, R. Pitisongsawat, B. Erice, N. Petrinic, V. L. Tagarielli, Predictions of the mechanical properties of unidirectional fibre composites by supervised machine learning, *Scientific Reports* 9 (1) (sep 2019). doi:10.1038/s41598-019-50144-w.
- [36] G. A. Sengodan, Prediction of two-phase composite microstructure properties through deep learning of reduced dimensional structure-response data, *Composites Part B: Engineering* 225 (2021) 109282. doi:10.1016/j.compositesb.2021.109282.
- [37] M. Raissi, P. Perdikaris, G. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *Journal of Computational Physics* 378 (2019) 686–707. doi:10.1016/j.jcp.2018.10.045.
- [38] J. K. Bauer, T. Böhlke, Variety of fiber orientation tensors, *Mathematics and Mechanics of Solids* 27 (7) (2021) 1185–1211. doi:10.1177/10812865211057602.
- [39] V. Müller, T. Böhlke, Prediction of effective elastic properties of fiber reinforced composites using fiber orientation tensors, *Composites Science and Technology* 130 (2016) 36–45. doi:10.1016/j.compscitech.2016.04.009.
- [40] J. K. Bauer, M. Schneider, T. Böhlke, On the phase space of fourth-order fiber-orientation tensors, *Journal of Elasticity* (jan 2023). doi:10.1007/s10659-022-09977-2.
- [41] T. Böhlke, Application of the maximum entropy method in texture analysis, *Computational Materials Science* 32 (3-4) (2005) 276–283. doi:10.1016/j.commatsci.2004.09.041.
- [42] T. Lee, M. Leok, N. H. McClamroch, Global symplectic uncertainty propagation on $SO(3)$, in: *2008 47th IEEE Conference on Decision and Control*, IEEE, 2008. doi:10.1109/cdc.2008.4739058.
- [43] M. D. McKay, R. J. Beckman, W. J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics* 42 (1) (2000) 55–61. doi:10.1080/00401706.2000.10485979.
- [44] I. M. Sobol’, On the distribution of points in a cube and the approximate evaluation of integrals, *USSR Computational Mathematics and Mathematical Physics* 7 (4) (1967) 86–112. doi:10.1016/0041-5553(67)90144-9.
- [45] A. Henkes, H. Wessels, R. Mahnken, Physics informed neural networks for continuum micromechanics, *Computer Methods in Applied Mechanics and Engineering* 393 (2022) 114790. doi:10.1016/j.cma.2022.114790.
- [46] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, *PyTorch: An Imperative Style, High-Performance Deep Learning Library*, Curran Associates Inc., Red Hook, NY, USA, 2019.

- [47] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, P. T. P. Tang, On large-batch training for deep learning: Generalization gap and sharp minima, in: 5th International Conference on Learning Representations (ICLR 2017), 2017.
- [48] M. Riedmiller, H. Braun, A direct adaptive method for faster backpropagation learning: the RPROP algorithm, in: IEEE International Conference on Neural Networks, IEEE, 1993. doi:10.1109/icnn.1993.298623.
- [49] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: 3rd International Conference on Learning Representations (ICLR 2015), 2015.
- [50] C. Florescu, C. Igel, Resilient backpropagation (Rprop) for batch-learning in TensorFlow, in: 6th International Conference on Learning Representations (ICLR 2018), 2018.
- [51] R. McLendon, Micromechanics Plugin for Abaqus, <https://www.linkedin.com/pulse/micromechanics-plugin-abaqus-ross-mclendon/>, accessed: 2023-02-13 (2017).
- [52] L. Brown, A. Long, Modeling the geometry of textile reinforcements for composites: TexGen, in: Composite Reinforcements for Optimum Performance, Elsevier, 2021, pp. 237–265. doi:10.1016/b978-0-12-819005-0.00008-3.
- [53] M. A. Bouhlef, J. T. Hwang, N. Bartoli, R. Lafage, J. Morlier, J. R. Martins, A python surrogate modeling framework with derivatives, Advances in Engineering Software 135 (2019) 102662. doi:10.1016/j.advengsoft.2019.03.005.
- [54] Z. Liu, M. Fleming, W. K. Liu, Microstructural material database for self-consistent clustering analysis of elastoplastic strain softening materials, Computer Methods in Applied Mechanics and Engineering 330 (2018) 547–577. doi:10.1016/j.cma.2017.11.005.
- [55] R. Bharali, F. Larsson, R. Jänicke, Computational homogenisation of phase-field fracture, European Journal of Mechanics - A/Solids 88 (2021) 104247. doi:10.1016/j.euromechsol.2021.104247.
- [56] I. Doghri, L. Adam, N. Bilger, Mean-field homogenization of elasto-viscoplastic composites based on a general incrementally affine linearization method, International Journal of Plasticity 26 (2) (2010) 219–238. doi:10.1016/j.ijplas.2009.06.003.
- [57] B. M. Irons, R. C. Tuck, A version of the Aitken accelerator for computer iteration, International Journal for Numerical Methods in Engineering 1 (3) (1969) 275–277. doi:10.1002/nme.1620010306.