Computer Methods in Applied Mechanics and Engineering 456 (2026) 118939

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

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

Multiscale analysis of woven composites using hierarchical physically recurrent neural networks

Ehsan Ghane [a,*], Marina A. Maia [b], Iuri B. C. M. Rocha [b], Martin Fagerström [c], Mohsen Mirkhalaf [a]

[a] Department of Physics, University of Gothenburg, Gothenburg, Sweden
[b] Delft University of Technology, Faculty of Civil Engineering and Geosciences, Delft, The Netherlands
[c] Department of Industrial and Materials Science, Chalmers University of Technology, Gothenburg, Sweden

ARTICLE INFO

Keywords:
Woven composites
Multiscale modeling
Physics-encoded neural networks
Surrogate modeling
Path dependency

ABSTRACT

Multiscale homogenization of woven composites requires detailed micromechanical evaluations, leading to high computational costs. Data-driven surrogate models based on neural networks address this challenge but often suffer from big data requirements, limited interpretability, and poor extrapolation capabilities. This study introduces a Hierarchical Physically Recurrent Neural Network (HPRNN) employing two levels of surrogate modeling. First, Physically Recurrent Neural Networks (PRNNs) are trained to capture the nonlinear elasto-plastic behavior of warp and weft yarns using micromechanical data. In a second scale transition, a physics-encoded meso-to-macroscale model integrates these yarn surrogates with the matrix constitutive model, embedding physical properties directly into the latent space. By adopting HPRNNs, nonphysical behavior often observed in predictions from pure data-driven recurrent neural networks and transformer networks can be avoided. This results in better generalization under complex cyclic loading conditions. The framework offers a computationally efficient and explainable solution for multiscale modeling of woven composites.

1. Introduction

Multiscale homogenization of woven composites presents challenges due to the complexity of their microstructural behavior and the transitions across micro-, meso-, and macroscales. At the microscale, the interactions between fibers and the matrix are critical, while the mesoscale considers bundles of fibers (yarns or tows) as a homogenized continuum, where the morphology of the woven structure is crucial. Finally, at the macroscale, the composite is studied at the lamina or laminate level as a homogeneous orthotropic continuum [1].

Some studies have proposed multiscale computational homogenization approaches to bridge microscale information to meso- and macroscale phenomena [2,3]. Techniques like FE^2 rely on detailed micromechanical evaluations at the lower scales for every quadrature (Gauss) point in the macroscale domain for every pseudo-time step, which leads to prohibitive computational costs [4,5]. This issue is taken to the extreme when modeling woven composites, as linking microscale information all the way to the macroscale would require two scale transitions (i.e., FE^3 approach). The nonlinear and path-dependent behavior of these materials further complicates the computational challenge. These factors underscore the need for computationally efficient alternatives to bypass the intensive costs associated with lower-scale homogenization while maintaining accuracy.

* Corresponding author.
E-mail address: ehsan.ghane@liu.se (E. Ghane).

https://doi.org/10.1016/j.cma.2026.118939
Received 17 December 2025; Received in revised form 25 February 2026; Accepted 20 March 2026
Available online 2 April 2026

Recent studies propose approximate methods, also known as *surrogate models*, for efficient computational experiments. The objective is to create surrogate models that can predict the multiaxial stress state from a given history of multi-dimensional deformation, thereby avoiding the high computational cost of traditional simulations. These methods offer great potential for reducing the computational burden of multiscale analyses and can be broadly categorized into multi-fidelity surrogates, data-driven surrogates and hybrid surrogates approaches. *Multi-fidelity surrogates* maintain a physics foundation but simplify the system using techniques such as coarser discretization, relaxed convergence criteria, or reduced physical details. Examples include reduced-order modeling methods, which are tied to the underlying physics. *Data-driven surrogates* construct approximations by fitting high-fidelity microscale data using interpolation or regression techniques without embedding physical principles. Common examples include polynomial response surfaces, kriging (Gaussian process regression) [6], neural networks (e.g., multi-layer perceptrons, also known as deep neural networks), radial basis functions [7], and splines [8].

Providing sufficient data for data-driven models in a supervised learning framework can be achieved via a well-planned design of experiment (DoE) [9] or on-the-fly data generation [10]. This approach enables the creation of surrogate models with precision in certain applications close to high-fidelity homogenizations [11]. For instance, *feed-forward neural networks* have proven effective as surrogates, learning the complex relationships between microscale input parameters and homogenized mesoscale responses in woven composites [12].

Regarding nonlinear path-dependent phenomena, analogies can be drawn from the field of natural language processing in AI. Specifically, the similarities between hidden states in *Recurrent Neural Networks (RNNs)* and the self-attention mechanism in *Transformers* with the internal variables governing thermodynamically irreversible processes have drawn significant attention [13–15]. These similarities highlight the potential of neural network architectures to model complex material behaviors. Two main engineered formats of RNNs, i.e. GRU and LSTM, and recently Transformers [16–18] have been demonstrated to have the ability to capture nonlinear constitutive behaviors by mapping input features, such as deformation tensors, to relevant outputs, such as stress tensors, at both the micro- [19] and mesoscale [20]. More studies on constitutive modeling of composites using data-driven surrogates are reviewed in [11,15].

One major limitation of conventional data-driven surrogate models is their heavy dependence on large and diverse datasets for accurate predictions [14]. Acquiring or generating these datasets, especially for high-fidelity applications, requires significant computational effort, making this approach impractical for large-scale or complex simulations. Data fusion approaches aim to address this issue by combining data from multiple sources, such as low-cost simulations and *few-shot* high-fidelity simulations, to reduce computational costs [21]. By leveraging knowledge from related datasets, transfer learning enables models to perform well on new datasets with limited high-fidelity data [22,23]. Transfer learning further enhances the applicability of surrogate models in multiscale woven composite modeling [20,24].

Despite these advancements, traditional data-driven approaches [25], often lack intrinsic knowledge of the underlying physics. As a result, they become prone to errors during extrapolation [9] and offer limited interpretability, leading to the so-called *black-box problem* [14,26]. These challenges highlight the need for models integrating physics-based information to enhance reliability.

Hybrid surrogate approaches combine data fitting with physical knowledge to address the limitations of exclusively data-driven neural network surrogate models. These methods integrate physics-based insights into neural networks and can be broadly classified into two categories:

Physics-informed Neural Networks (PiNNs). These models incorporate knowledge of physical laws such as conservation principles, boundary conditions, and kinematic relations by adding new terms related to these constraints in the neural network's loss function. Initially developed for solving Partial Differential Equations (PDEs) [27], PiNNs have since been applied to constitutive material modeling [28–31]. However, their effectiveness is highly sensitive to the relative weighting of loss terms [32].

Physics-encoded Neural Networks. Also known as physics-augmented [33], these models embed physics-based knowledge directly into the architecture, such as through custom neurons, layers, or constraints [34]. For example, Rao et al. [35] used element-wise feature multiplication to create a recurrent Π-block, mimicking terms governing PDEs for an optimization problem. Spatial dependencies are modeled using convolutional layers or finite-difference filters, while temporal evolution is handled via a forward Euler scheme. Other physics-encoded networks, like neural operators [36] and Fourier neural operators [37], integrate physics through engineered units combined with multi-layer perceptrons. Furthermore, Mostajeran et al. [38] introduced an elasto-plasticity-informed Chebyshev-based Kolmogorov-Arnold network to achieve accurate and generalizable function approximation for nonlinear stress-strain relationships while using fewer parameters than standard multi-layer perceptrons. Compared to physics-informed neural networks that use multi-layer perceptrons, Kolmogorov-Arnold networks require significantly less data for calibration and discovery.

While hybrid surrogate approaches enhance interpretability and generalization to unseen conditions, they are often constrained by large dimensionality and structured data requirements, making them less practical for complex multiscale heterogeneous materials [39]. Further research is needed to extend these methods to challenging multiscale composite modeling. At the same time, traditional multiscale modeling approaches often struggle to fully account for the complex interactions between different scales in woven composites [40]. Some models oversimplify the geometry of woven structures, neglecting critical details that influence the material's response under various loading conditions [41,42]. This gap highlights the need for a robust and efficient approach to address the hierarchical complexity of woven composites.

Motivated by these challenges, this study extends the Physically Recurrent Neural Network (PRNN) framework [43], which belongs to the family of physics-encoded hybrid neural networks. The PRNN embeds the constitutive relations of the microscale as a modified hidden layer in a multi-layer perceptron (a feed-forward neural network), creating physically recurrent blocks with internal variables tied to thermodynamic constraints. Each of these blocks, known as fictitious (microscopic) material points, have their own internal variables automatically updated by the embedded constitutive model. This allows path-dependent behaviors such as elasto-

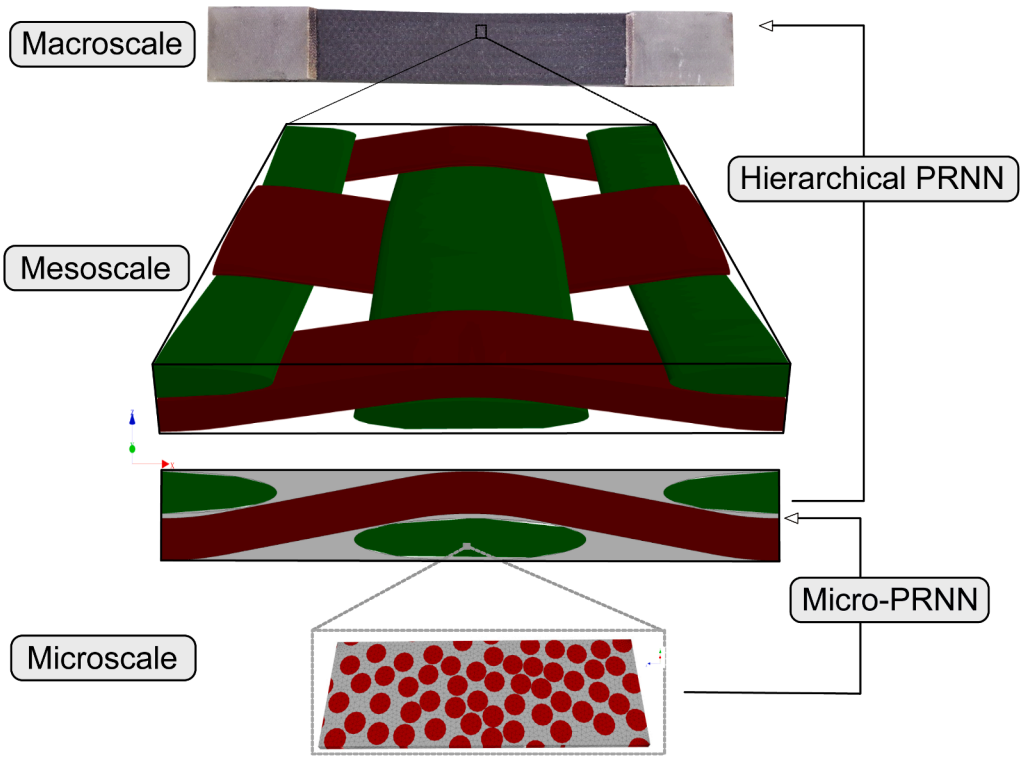


Fig. 1. Hierarchical structure of woven composites and the transition across two scales with PRNN and the proposed HPRNN. The macroscopic sample taken by E. Ghane [44] is included only to clarify the concept.

plasticity to appear naturally rather than relying on implicit and opaque memory mechanisms as in conventional RNNs. PRNNs have demonstrated strong performance in prior works in terms of reducing the training data requirements, extrapolation to non-monotonic scenarios, and significant computational efficiency in FE^2 applications on unidirectional composites [43]. Nevertheless, its application to complex hierarchical structures remains unexplored.

A key innovation in this work is extending PRNNs to perform transitions in two scales, effectively creating a surrogate model aiming for FE^3. PRNNs were originally designed for a single scale transition, from micro to macro, with applications so far focused on unidirectional composites. Here we leverage these developments to build a bottom-up approach to efficiently model woven composites, as illustrated in Fig. 1. At the lowest level, we train a PRNN to learn the homogenized constitutive response of unidirectional composites, which in this problem represents a (fictitious) material point at the mesoscale (see micro-PRNN in Fig. 1). At the upper level, this pre-trained PRNN acts as the building block of a larger network tasked to learn the second homogenization step from meso to macroscale, giving rise to the *Hierarchical* PRNN (HPRNN) in Fig. 1.

This study assumes that non-linearity at the micro- and mesoscales arises from plasticity in the matrix, while the fibers exhibit isotropic elastic behavior. The modular nature of the proposed framework facilitates the incorporation of further developments, such as extending to more complex material behaviors or incorporating additional mechanisms.

The rest of this paper is structured as follows. Section 2 briefly discusses the challenges of computational homogenization for woven composites. Section 3 describes the dataset used for training and evaluating the proposed models. Section 4 provides an overview of PRNN design, as introduced by Maia et al. [43]. Section 5 presents the HPRNN framework and its modular components. Section 6 evaluates HPRNN's performance against high-fidelity multiscale simulations and compares it with data-driven history-dependent neural networks, including GRUs and Transformers, under elasto-plastic conditions. Finally, Section 7 summarizes the key findings, highlights the advantages of HPRNN, and discusses its limitations and future research directions.

2. Problem overview

Woven fiber-reinforced polymer composites are specifically designed to conform more easily to complex curvatures than unidirectional laminates while maintaining balanced and desirable mechanical properties [40]. The plain woven fiber reinforcement pattern is tailored to provide balanced stiffness in both in-plane loading directions. However, this intricate morphology poses significant challenges when analyzing these materials. The mechanical response of woven composites varies depending on the loading direction. Under in-plane loading aligned with fiber bundles, the material exhibits high stiffness. In contrast, when the load is slightly misaligned

with the fiber direction, the material demonstrates pronounced nonlinear behavior, even in systems comprising stiff carbon fibers and high-performance epoxy resins [45].

Out-of-plane loading introduces additional complexities due to the interlaced fiber structure, especially in the absence of reinforcement fiber bundles in the out-of-plane direction. These conditions often lead to pronounced matrix-dominated behavior, including through-thickness deformation and potential delamination between yarn layers or fiber-matrix interfaces. The interfacial strength and the matrix's capacity to absorb and redistribute stresses significantly influence the overall material response. Consequently, woven composites exhibit lower stiffness and strength in the out-of-plane direction than their in-plane properties.

Material nonlinearity arises from mechanisms such as (visco-)plasticity in the matrix phase or damage initiation and propagation in the matrix, yarns, or their interfaces. This study focuses solely on plasticity in the matrix phase, excluding rate-dependency and damage mechanisms. While yarns exhibit transversely isotropic behavior at the mesoscale [46], we assume the fibers are elastic, homogeneous and isotropic at the microscale. Although this assumption may lack physical accuracy, particularly for carbon fibers, it is reasonable for glass fibers [47] and does not constrain the generality of our approach.

Understanding the material's loading history and constitutive behavior is essential to account for irreversible thermodynamic phenomena such as plasticity. This knowledge allows for the computation of internal variables, such as plastic strains. It also enables the prediction of plastic deformation using a yield surface and flow rule, both of which depend on the chosen plasticity model.

A common approach in modeling composites is to formulate constitutive models directly at the mesoscale and assemble them using lamination theories. However, this often relies on phenomenological laws to capture complex material behavior, which can overlook key microscopic internal variables that ultimately govern the material's response. To address this limitation, we employ a computational homogenization framework at two levels that explicitly account for the microscale properties while bridging the gap to the mesoscale.

2.1. Computational homogenization at two levels for woven composites

Assuming periodicity [4] at the subscale, the structural response is governed by effective properties derived from mesoscale analyses. Meanwhile, the microscale resolves the detailed behavior of the yarns' Representative Volume Element (RVE), capturing their intricate mechanical responses. The two levels of the material under study, the microscale and mesoscale, are illustrated in Fig. 2.

In first-order computational homogenization, scale separation is assumed, ensuring a uniform average deformation gradient across the RVE [4]. The solution typically involves iterative coupling between scales, with boundary conditions applied at the subscale to maintain consistency. However, this nested nature introduces significant computational challenges, especially when solving the microscale RVE problem at every integration point within the mesoscale model. This complexity arises from the need to account for two scale transitions. First, from the microscale to the mesoscale, where fiber-matrix interactions define the yarns' effective behavior. Second, from the mesoscale to the macroscale, where the homogenized yarn response determines the overall composite properties while preserving microscale mechanisms like plasticity and damage evolution.

Machine learning techniques, like neural network-based models, can help solve these computational challenges in the micro-to-meso and meso-to-macro transitions. Unlike conventional phenomenological approaches that impose constitutive assumptions at the mesoscale, data-driven methods can be used to efficiently homogenize microscale behavior without explicit mesoscale constitutive assumptions and without the need to run expensive microscale boundary value problems at every mesoscale material point. At the mesoscale, the constitutive responses of the yarns are not predefined. Instead, the mesoscopic behavior emerges naturally from solving the microscale boundary value problem. This eliminates the need for explicit assumptions regarding the constitutive behavior at the meso level.

2.2. Learning path-dependent material responses from data

Capturing history-dependent material behavior requires models that can account for the sequential nature of stress-strain evolution. The goal is to develop surrogate models for the heterogeneous composite that predict the stress tensor based on a given history of multi-dimensional strain or deformation tensors, eliminating the need for computationally intensive simulations. Conventional feed-forward networks struggle with long-term dependencies, necessitating the use of sequential architectures such as Recurrent Neural Networks (RNNs) and Transformers [20,24,48]. However, these models heavily rely on extensive datasets to generalize well [14]. This study extends the Physically Recurrent Neural Network (PRNN) framework [43] to overcome these limitations by incorporating physics-based constraints and explicit internal variables, ensuring robustness in extrapolation tasks.

3. Data-generation

In this study, we generated two distinct datasets for model development. The first dataset is obtained from a high-fidelity microscale model that captures the elasto-plastic behavior of UD composites. This dataset comprises 500 samples and is used for the micro- to mesoscale transition representing the warp and weft yarns. The second dataset with 400 samples is generated using Fast Fourier Transform (FFT) simulations for mesoscale homogenization of a woven composite RVE. Load generator algorithm is described in Section 3.1, and the material models at Section 3.2. Details on computational homogenization in both scales are described in Section 3.3.

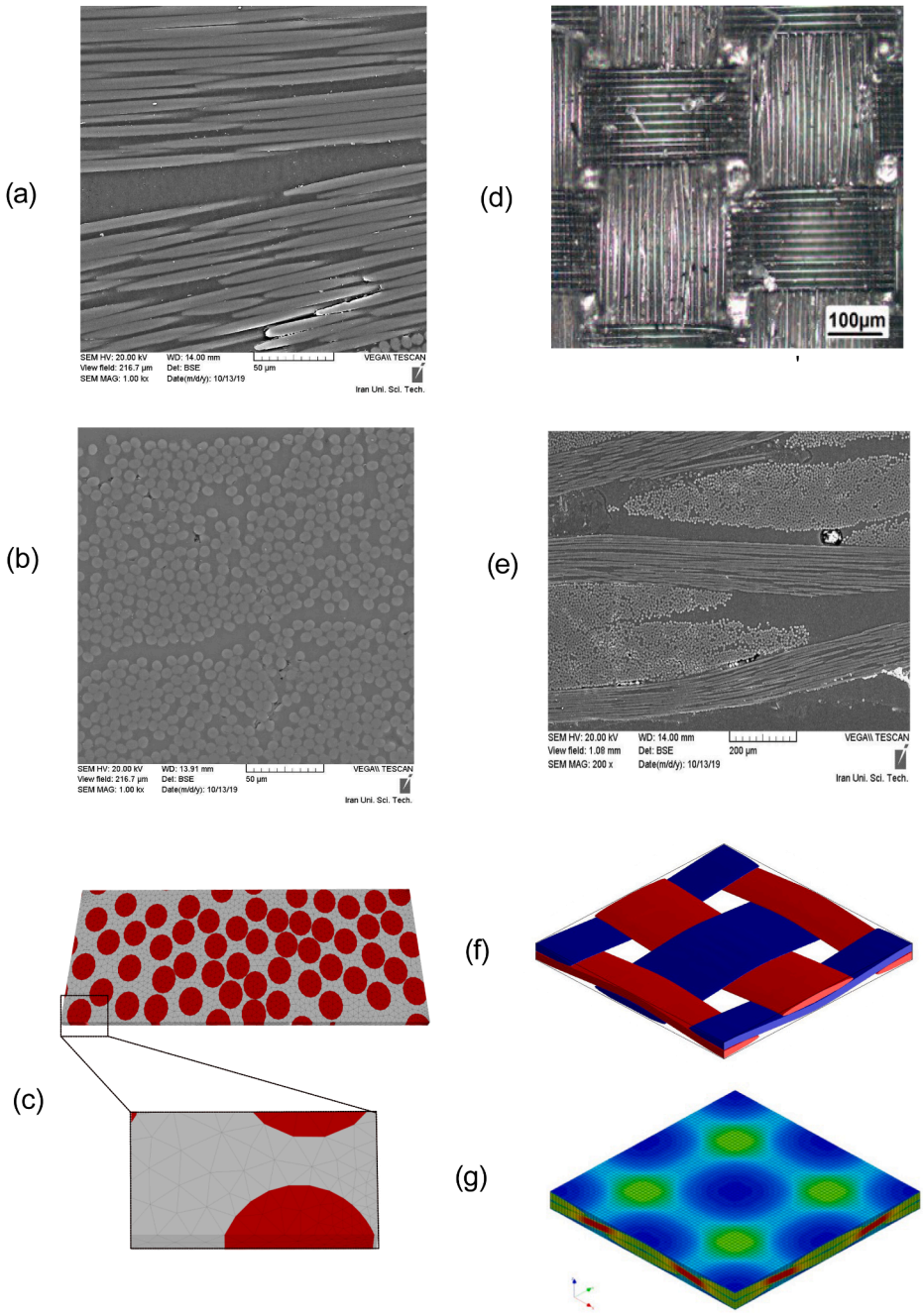


Fig. 2. Illustration of woven composite structures and associated simulations. Microscale images captured by E. Ghane from [44] alongside equivalent representative volume elements (RVEs) modeled using Digimat-FE. The figure includes: (a,b) Scanning electron microscopy of fibers in two orthogonal directions (warp and weft), (c) microscale simulation of unidirectional composites, (d) top view microscopic image of a carbon fiber woven composite, (e) SEM image showing the cross-section of the woven composite, (f) mesoscale RVE, and (g) FEM simulation results with a voxel-based mesh.

3.1. Load generator algorithm

A random-walk algorithm is utilized to generate six-dimensional input strain loading paths [49]. This algorithm updates the previous load step by adding two independent sources of white noise, both sampled from Gaussian noises with different scales. The first white noise is the primary driver on a larger scale (drift), while the second introduces small-scale variations (noise). Note that

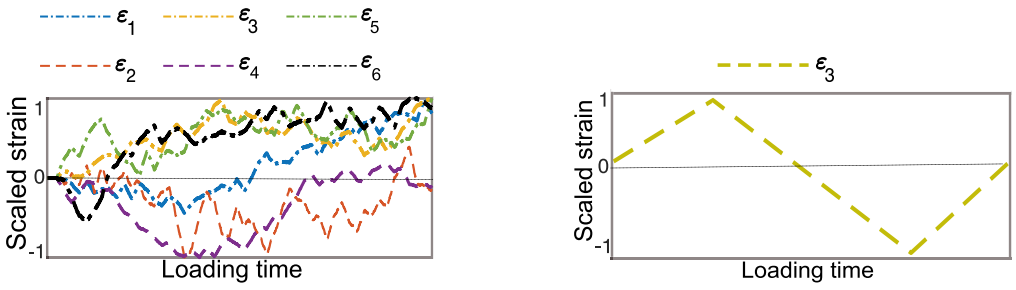


Fig. 3. Samples of scaled (normalized between 1 and -1) strain components ε_j generated in (a) random loading used for training and validation and (b) cyclic loading to be used in extrapolation.

Table 1
Material parameters for the elasto-plastic matrix and elastic fiber models.

Material	σ_y [MPa]	H [MPa]	H_∞ [MPa]	m [-]	E [GPa]	ν [-]
Matrix	18.00	10.00	65.00	180.00	3.7	0.35
Fiber	-	-	-	-	275	0.22

this algorithm differs from the load generator used in [43], as it generates random steps independently without introducing any correlation between load steps.

The proposed data generation approach enables the generation of multi-axial stress-strain histories under random-walk loading conditions. To assess the trained networks' ability to extrapolate to sparse regions in the input space[1], a second data generation strategy is also used [24]. This second strategy focuses on cyclic in-plane shear loading with different peak strain values and number of cycles, where significant plasticity occurs in woven composites. Unlike random-walk loading, cyclic loading exhibits a strong correlation between consecutive strain values in a sequence. Consequently, cyclic loading is used as an extrapolation test case. Examples of strain components generated by the random-walk algorithm and cyclic load generation are shown in Fig. 3.

3.2. Material models

The framework relies on two constitutive models: an elasto-plastic model for the matrix and a linear elastic model for the fibers. These models are used at the microscale during data generation and remain consistent across scales to ensure physical accuracy.

3.2.1. Elasto-plastic matrix model

The matrix material follows an elasto-plastic constitutive law based on J_2-plasticity, incorporating a linear-exponential hardening behavior as described in [50]. The yield function is defined as:

$$\Phi(\sigma, \bar{\varepsilon}^{\mathrm{p}}) = \sigma_{\mathrm{vM}} - \left(\sigma_{\mathrm{y}} + H\bar{\varepsilon}^{\mathrm{p}} + H_\infty\left(1 - e^{-m\bar{\varepsilon}^{\mathrm{p}}}\right)\right) \leq 0, \tag{1}$$

where σ_{vM} is von Mises stress, σ_{y} is the initial yield stress, H_∞ is the hardening modulus, H is the linear hardening modulus, $m > 0$ is the hardening exponent, and $\bar{\varepsilon}^{\mathrm{p}} \geq 0$ is the accumulated plastic strain.

3.2.2. Linear elastic fiber model

The fiber reinforcements are assumed to behave elastically, characterized by Young's modulus E and Poisson's ratio ν. The matrix and fibers' material parameters are presented in Table 1.

3.3. Computational homogenization of woven composites

Assuming no undulation through the yarns, microscale finite element simulations are performed in Digimat-FE using a three-dimensional unidirectional RVE as shown in Fig. 4(c). This assumption applies exclusively at the microscale, where each yarn is modeled as a straight unidirectional composite to characterize constituent level behavior. A similar scale separation strategy has been adopted in previous multiscale studies of woven composites, where fiber alignment is assumed at the microscale while yarn undulation is introduced at higher structural levels [51,52]. In contrast, yarn crimp is explicitly incorporated at the mesoscale in the present work through the geometric definition of the woven RVE. The microscale RVE consists of continuous cylindrical carbon

[1] The input space includes the six independent components of strain tensors. Sparsity refers to scenarios where specific strain components are zero, such as pure in-plane shear loading with only ε_3 being nonzero.

fibers embedded in an epoxy matrix with perfect fiber–matrix bonding. The fiber volume fraction is fixed for all microscale samples at $V_f = 54\%$, and the fiber aspect ratio is set to 10. Fiber orientation is fixed and aligned with the yarn direction.
The RVE geometry is defined explicitly with dimensions $300\,\mu\text{m} \times 100\,\mu\text{m} \times 5\,\mu\text{m}$ along the height, width, and fiber directions, respectively. Periodic boundary conditions are enforced at the RVE level using Digimat-FE's built-in periodic formulation, ensuring displacement compatibility and strain equivalence across opposite faces. Macroscopic strain-controlled loading is applied in all normal and shear components using user-defined strain histories.
The RVE is discretized using a conforming tetrahedral mesh with linear elements. The nominal element size is set to $4.0\,\mu$m, with a minimum element size of $0.2\,\mu$m. This mesh density is consistent with previous microscale homogenization study [12] and provides a reliable balance between accuracy and computational cost. Internal coarsening and curvature control are enabled to accurately resolve fiber–matrix interfaces. The resulting microscale discretization contains approximately $O(10^5)$ mechanical degrees of freedom. All microscale simulations are performed under quasi-static conditions using the Digimat-FE iterative solver.

For mesoscale simulations, both FE and FFT methods are explored. The RVE model used represents a balanced weave with 15 yarns per centimeter in the warp and weft directions. Key geometric parameters include a yarn spacing ratio of 0.1, a crimp factor of 0.5, and an ellipsoidal yarn cross-section measuring 0.05 mm in height and 0.5 mm in width. The FE simulations are performed on a three-dimensional voxel-based mesh with dimensions 64 × 64 × 32 (each voxel having dimensions of $21 \times 21 \times 1.7\ \mu m$), utilizing fully integrated 8-node elements. A convergence study verifies that further mesh refinement results in negligible changes to the stress-strain response, confirming the adequacy of this resolution. The FFT simulations employ a 64 × 64 × 64 voxel grid shown in Fig. 4(a,b). Sensitivity analyses with higher grid resolutions show marginal improvements in accuracy. Thus, this resolution is found to be optimal by balancing computational efficiency and accuracy. In [24], we compared the results from FE simulations with those from a voxel-based (non-conforming) mesh, and FFT shows strong agreement in the results, while significantly reducing the computational cost. For instance, one FE simulation requires approximately 7040 seconds to converge during a cyclic loading, while FFT completes the task in 440 seconds.

The RVE geometries (micro- and mesoscale) and the constitutive material models are kept fixed in both datasets. Simulations are orchestrated by Julia [53] script for writing input files and Digimat-FE running on a system equipped with a 16-core processor and an *NVIDIA® GeForce RTX™ 4090* GPU.

4. Infusing physics-based recurrence into neural networks

Developed by Maia et al. [43], the Physically Recurrent Neural Network (PRNN) draws inspiration for its formulation and interpretation from computational homogenization principles. In the full-order solution, the mesoscale strain state is mapped onto local material points at the microscale, where classical constitutive models compute the stress-strain response, which is then homogenized back to the mesoscale. In the network, shown in Fig. 5, an analogous process takes place through an *encoder-decoder* architecture in which the same constitutive models as in the full-order solution are embedded. The following section examines the role of each PRNN component in the first-scale transition, using terminology from computational homogenization. The second-scale transition is discussed in detail in Section 5.
Localization: The localization or down-scaling step (analogous to an encoder) maps the mesoscopic strain inputs $\boldsymbol{\varepsilon}^{\boldsymbol{\Omega}}$ to a set of fictitious material points (global-to-local mapping). This step serves to approximate the microscopic boundary-value problem typically solved in finite element-based homogenization. The mapping is expressed as:

$$\mathbf{a}_i = \phi(\mathbf{W}_i \mathbf{a}_{i-1} + \mathbf{b}_i), \quad i = 1, \dots, L \tag{2}$$

where $\mathbf{a}_0 = \boldsymbol{\varepsilon}^{\boldsymbol{\Omega}}$ is the input mesoscopic strain at the current step, $\mathbf{a}_i$ is the neurons states at layer i at the current step, $\mathbf{W}_i \in \mathfrak{R}^{n_i \times n_{i-1}}$ and $\mathbf{b}_i$ are the weight matrix and bias terms for layer i with n_i neurons, $\phi(\cdot)$ is the (non)linear activation function, such as ReLU, sigmoid or tanh, and L is the number of layers.

In this study, the encoder architecture consists of a linear layer with the identity function as activation function, resulting in a direct relationship between mesoscopic and local strains. While the encoder-generated strain paths do not inherently contain history effects, the material response (stress) after the material layer exhibits path dependency, as in actual RVEs where history-dependent effects emerge due to nonlinear constitutive behavior. This path dependency is explicitly captured in the PRNN through history variables updated within the material layer. It is important to note that the network does not store any strain field data from the microscale model; the encoder instead learns mappings based exclusively on snapshots of mesoscopic strains.
Microscale constitutive response at material layer: We modify how neurons are implemented in the material layer compared to standard dense layers to take full advantage of this layer's outputs. Instead of applying nonlinearities element-wise with scalar activation functions, neurons are grouped into sets matching the size of the PRNN input layer (see gray boxes in Fig. 5). Each group is treated as a fictitious material point and has the same length as the strain vector (six components in this study), with each neuron representing one strain-vector component. In this way, the material layer consists of units that act as fictitious material points, designed to replicate the process of solving the microscale boundary-value problem in computational homogenization.

In each group, a classical constitutive model $\mathcal{D}^{\omega}$ maps the fictitious microscale strain $\boldsymbol{\varepsilon}^{\boldsymbol{\omega}}$ to the corresponding fictitious microscale stress $\boldsymbol{\sigma}^{\boldsymbol{\omega}}$ and internal variables $\boldsymbol{\alpha}^{\omega}$ for that particular material point. This layer captures path-dependent behavior by updating the internal variables $\boldsymbol{\alpha}_j^{\omega}$ at each time step, replacing (in combination with the encoder) the iterative solution of local equilibrium problems typically required in finite element homogenization.

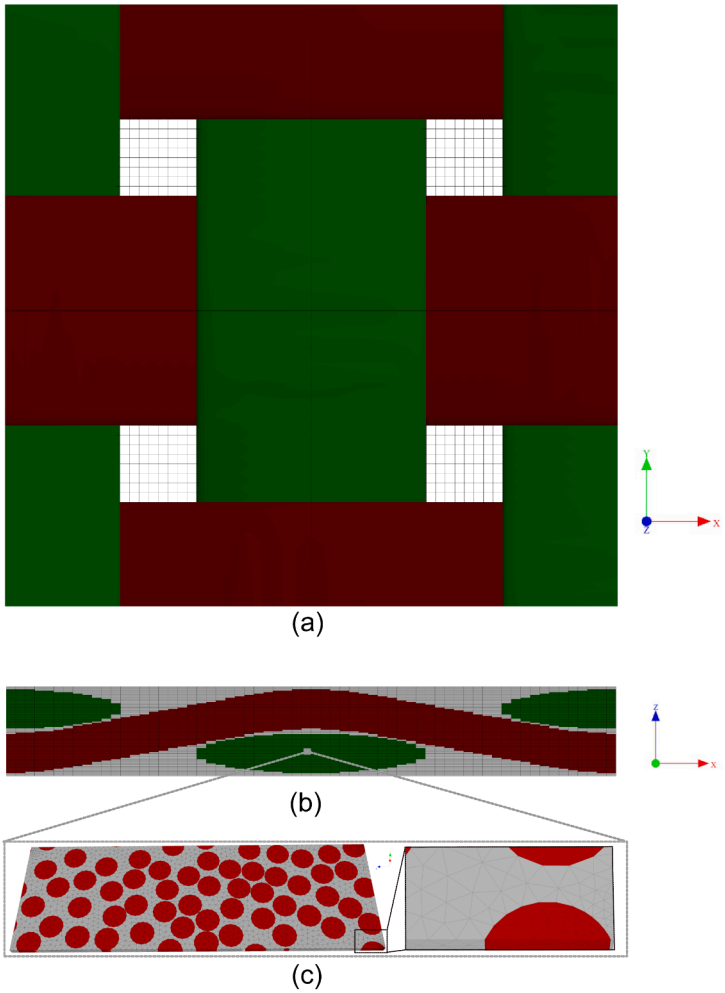


Fig. 4. Details of the computational model at micro- and mesoscale: (a) Top view of the mesoscale model, (b) X-Z cross-section of the mesoscale model and the mesoscale mesh, and (c) the microscale warp yarn and the details on it mesh.

For a given fictitious material point j, the response is computed as:

$$\boldsymbol{\sigma}_j^{\omega}, \boldsymbol{\alpha}_{j,t}^{\omega} = \mathcal{D}^{\omega}\left(\boldsymbol{\varepsilon}_j^{\omega}, \boldsymbol{\alpha}_{j,t-1}^{\omega}\right), \tag{3}$$

where $\boldsymbol{\varepsilon}_j^{\omega}$ is the local strain at material point j, $\boldsymbol{\alpha}_{j,t-1}^{\omega}$ are the internal state variables, i.e., plastic strain, from the previous time step $t-1$, and $\mathcal{D}^{\omega}$ represents the constitutive model.

The formulation allows the network to adapt to a variety of nonlinear material behaviors, including elasto-plastic models with isotropic hardening or combinations of multiple constitutive models. For simplicity, this study models only the elasto-plastic matrix using a single material model $\mathcal{D}^{\omega}$ (Section 3.2.1), while the elastic fiber described in Section 3.2.2 is not explicitly included in the network but is expected to be learned from the data. In other words, there is no explicitly defined linear elastic unit in the network architecture. Instead, the network infers and captures the fiber's elastic behavior based solely on the training data, as some training simulations will exhibit only elastic responses because they never reach the plastic regime. This constitutive model implemented in the micro-PRNNs is the same as that used for stress computations at the integration points of a full-order micro-model during the data generation phase.

While fewer in number than integration points in the finite element model counterpart, the fictitious material points in this network serve as an efficient, physics-encoded representation of the material response, capturing key nonlinear behaviors. In the present formulation, this physics encoding is partial, as the matrix response is explicitly governed by an embedded J_2-plasticity, while the elastic response of the fiber phase and scale-transition effects are learned from data. History dependency is inherently captured in the PRNN by storing and updating the stresses $\boldsymbol{\sigma}^{\boldsymbol{\omega}}$ and internal variables $\boldsymbol{\alpha}^{\omega}$ associated with each fictitious material point. The number of these fictitious material points is a hyperparameter and must be determined at each scale by a model selection procedure as explained in Section 6.1.

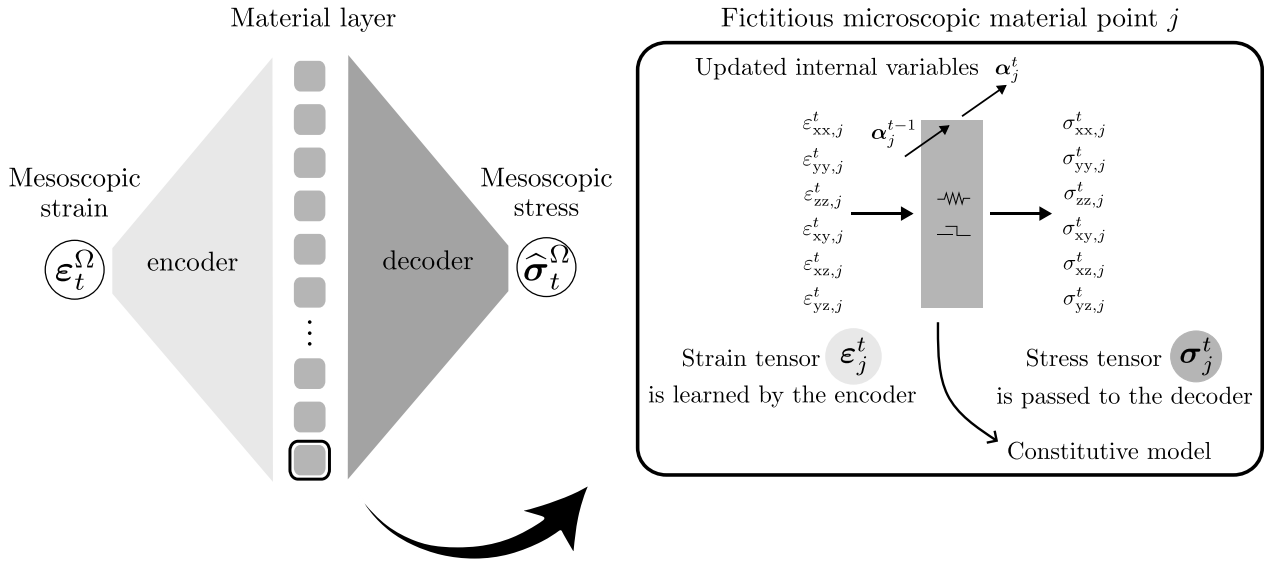


Fig. 5. Architecture of PRNN used as the transition between micro- to mesoscales. It consists of a feed-forward encoder for strain decomposition, a material layer representing the constitutive material at the microscale yarn of the woven RVE, and a feed-forward decoder for homogenization.

Homogenization: The decoder (analogous to the homogenization or up-scaling step) aggregates the stresses $\boldsymbol{\sigma}^{\omega}$ computed at all fictitious material points to predict the mesoscopic stress $\boldsymbol{\sigma}^{\boldsymbol{\Omega}}$ (local-to-global homogenization). This step mimics the homogenization operator in computational homogenization theory, where local fields are averaged to recover global responses. An important result of postulating the Hill-Mandel condition[2] for an RVE with kinematic boundary conditions (fully prescribed or periodically tied[3]) is that the macroscopic stress tensor $\boldsymbol{\sigma}^{\boldsymbol{\Omega}}$ equals the volume average of the microscopic stress tensors [4,5]. Similarly, in PRNN, the mesoscopic stress is computed as:

$$\hat{\boldsymbol{\sigma}}^{\Omega} = \sum_{j=1}^{m} \mathbf{w}_j \boldsymbol{\sigma}_j^{\omega}, \tag{4}$$

where $\mathbf{w}_j$ are the neural network weights representing the contributions of the m fictitious material points. To ensure non-negativity, the weights are defined using a *SoftPlus* function:

$$\mathbf{w}_j = \log(1 + e^{\bar{\mathbf{w}}_j}), \tag{5}$$

where $\bar{\mathbf{w}}_j$ are unconstrained trainable parameters. The weights in the output layer represent the relative contribution of each fictitious material point to the macroscopic stress. This step reconstructs the mesoscopic stress from the local responses, closing the computational homogenization loop.

Training Process. The network is trained to minimize the standard mean square error loss function for each training path as follows:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{2} \left\| \hat{\boldsymbol{\sigma}}_i^{\Omega}(\boldsymbol{\varepsilon}_i^{\Omega}) - \boldsymbol{\sigma}_i^{\Omega}(\boldsymbol{\varepsilon}_i^{\Omega}) \right\|^2, \tag{6}$$

where $\boldsymbol{\sigma}_i^{\Omega}$ are the target (ground truth) mesoscopic stresses, $\hat{\boldsymbol{\sigma}}_i^{\Omega}$ are the predicted mesoscopic stresses, $\boldsymbol{\varepsilon}_i^{\Omega}$ is the input strain at i^{th} pseudo-time step or load increment, and N is the number of pseudo-time steps (load increments) in the loading path.

Gradients are computed using backpropagation and optimized using a standard optimizer and automatic differentiation in PyTorch.

5. Hierarchical PRNN

The direct application of PRNNs to woven composites faces a fundamental challenge. These networks have been applied to bridge two-scale problems, while woven composites inherently exhibit a three-scale mechanical response as mentioned in Section 2: the microscale (fiber-matrix interactions), the mesoscale (yarn structure), and the macroscale (composite response).

[2] The volume average of the variation of work performed on the RVE is equal to the local variation of the work on the macroscale.

[3] Composite material behavior under various loading conditions can be modeled using periodic boundary conditions. As compared to prescribed displacement or traction boundary conditions, these conditions provide a better estimate of the overall properties [4,5].

To address this, we propose a hierarchical approach where PRNNs are first trained at the microscale to learn the nonlinear behavior of fiber-matrix interactions. It is anticipated that once trained, these microscale PRNNs can be frozen and employed as constitutive models at the mesoscale, enabling them to capture the inherent material behavior of yarns within the woven composite structure.

A key limitation of the frozen PRNN units is that they do not inherently account for yarns' orientations. Woven composites exhibit strong anisotropy due to their alternating warp and weft yarns. We introduce a rotation pre-processing and post-processing step around the PRNN modules to incorporate this directional dependence. This ensures the learned constitutive response is properly aligned with the local material orientation before homogenization.

With these modifications in place, we construct the Hierarchical PRNN (HPRNN), extending the standard PRNN framework to model woven composites. The architecture follows the principles of computational homogenization and consists of three key steps:

1. *Localization step (encoder)*. Maps macroscopic strains to fictitious mesoscopic strains that represent points in the yarns and in the embedding matrix.
2. *Material layer*. With embedded pre-trained PRNNs and conventional constitutive models, in this layer, we compute the stresses from fictitious mesoscopic material points based on the fictitious mesoscopic strains learned from the encoder.
3. *Homogenization step (decoder)*. Calculates the macroscopic stress for the woven composite based on the fictitious mesoscopic stresses computed at the material layer.

The material layer consists of three types of modules, shown in Fig. 6: one representing the nonlinear response of the pure matrix phase (*bulk material*), a *warp-PRNN* trained at the microscale to model yarns in the longitudinal direction, and a *weft-PRNN* trained at the microscale to model yarns in the transverse direction.

It is hypothesized that integrating these elements allows HPRNN to achieve an efficient yet physically consistent multiscale modeling strategy. The key hypothesis is that incorporating material-specific constitutive models into the material layer enhances the flexibility needed to capture the path-dependent behavior of woven composites while significantly reducing computational costs.

5.1. Module i: Constitutive material model

Module I represents the constitutive material model which accounts for the behavior of the pure matrix phase and is directly used at the mesoscale. The matrix follows the elasto-plastic law described in Section 3.2.1, while the fiber phase is assumed to be linearly elastic, as outlined in Section 3.2.2.

This constitutive material model is also present in the two remaining modules discussed in the following and remains the most computationally intensive part of the framework. This is due to the return-mapping problem that needs to be solved at every time step in each material call. Following the core idea of embedding the same constitutive models as in the lower scales in the network, the same implementation used in the full-order micro-model is also adopted for this second scale transition. The return mapping algorithm iteratively enforces the consistency condition, ensuring that the stress state remains on the yield surface.

5.2. Module II: Warp PRNNs

To model the behavior of the warp phase, we treat the warp yarns as analogous to a Unidirectional (UD) composite ply, ignoring yarn undulations at local material points. The frozen PRNN, trained to capture the nonlinear behavior of a UD ply driven by matrix plasticity, is directly used as a surrogate for warp yarn behavior.

5.3. Module III: Weft PRNNs

The orthogonality of warp and weft directions in a plain woven composite is handled through a standard tensor transformation. This allows the weft stress response to be directly derived from warp data, ensuring efficient characterization of the material's orthotropic behavior. The transformation of a 3D second-order stress or strain tensor $\mathcal{T}$ is achieved using the standard tensor rotation formula:

$$\mathcal{T}' = \mathcal{R}\,\mathcal{T}\,\mathcal{R}^T, \tag{7}$$

where $\mathcal{T}$ is the original stress or strain tensor in the warp reference frame, $\mathcal{T}'$ is the transformed tensor in the weft frame, and $\mathcal{R}$ is the rotation tensor. For a 90° rotation around the z-axis (common in woven composites), the matrix representation of the rotation tensor $\mathcal{R}$ is defined as:

$$\mathcal{R} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{8}$$

This tensor rotates the x and y components while leaving the z-axis component unchanged. Eq. (7) enables estimating the weft stress-strain behavior from the warp data. A synthetic weft dataset is generated from the high-fidelity warp dataset by rotating the input strain and stress tensors. This approach allows for leveraging existing computational resources to train a surrogate network for weft yarns.

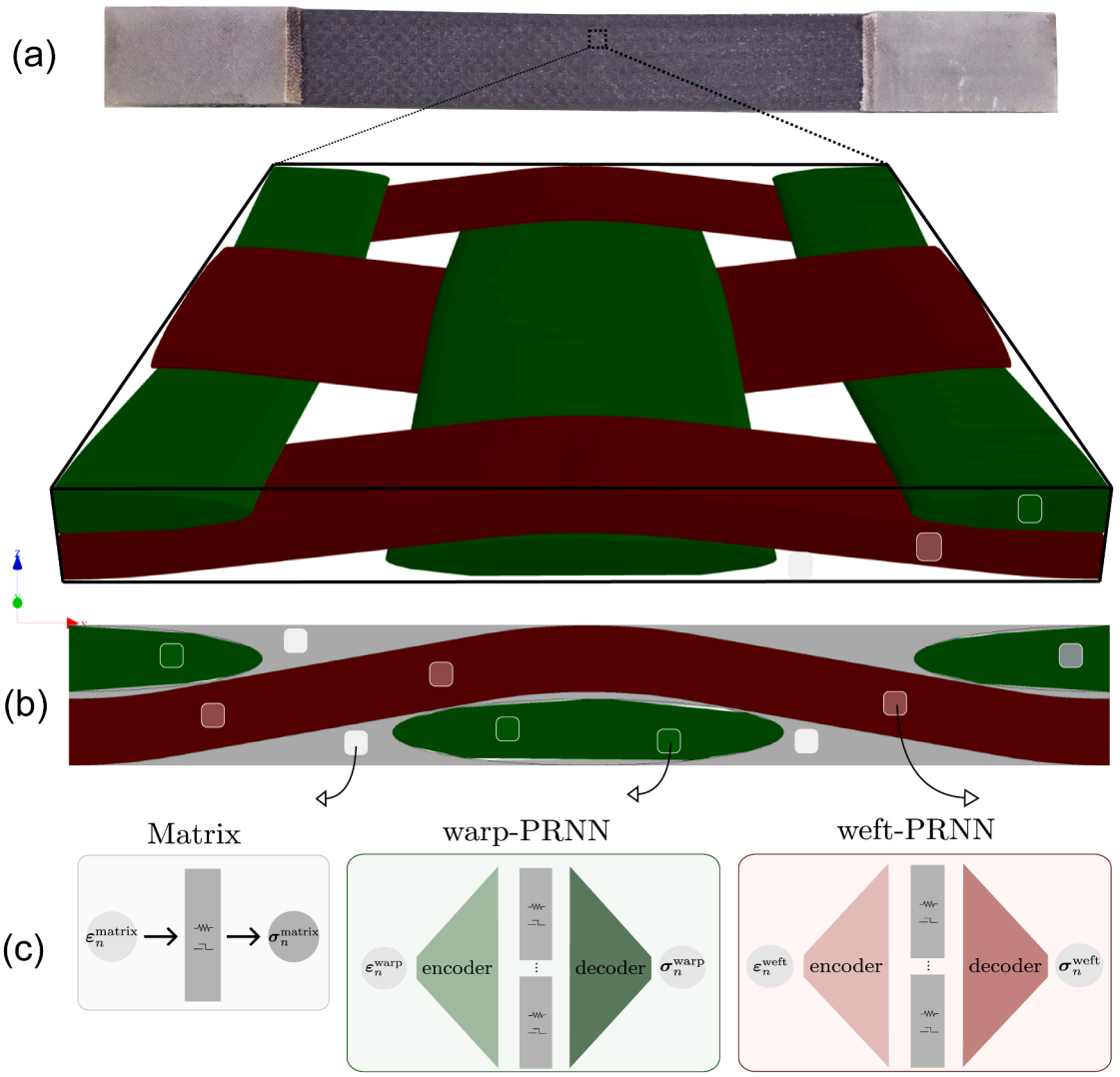


Fig. 6. Multi-scale structure of a woven composite. (a) Photograph of a carbon fiber woven composite taken by E. Ghane, adapted from [44], (b) Schematic representation of the mesoscale woven RVE, (c) The three fundamental components of a fictitious material point used in the HPRNN architecture: (I) matrix constitutive model, (II) weft-PRNN, and (III) warp-PRNN.

5.4. Full architecture summary

The HPRNN is designed to process six-dimensional strain input and predict six-dimensional stress outputs while incorporating material-specific microscale behavior through separate frozen PRNNs for warp and weft yarns. The architecture consists of three primary components according to Fig. 7: (1) an encoder that maps homogenized macroscale strain inputs ($\boldsymbol{\varepsilon}_t^{\Gamma}$) into latent representations, (2) a material layer that integrates the J_2 material model with warp and weft PRNN modules to capture the elasto-plastic behavior of yarns, and (3) a decoder that maps the mesoscale fictitious stress responses back to the macroscale output space ($\hat{\boldsymbol{\sigma}}_t^{\Gamma}$).

Pre-trained PRNN modules are incorporated in evaluation mode (frozen as mentioned in Section 5), ensuring their weights remain fixed while optimizing the meso-to-macroscale network. The input data, comprising strain and stress tensors, is formatted into time series datasets for training and validation. The datasets are divided into sequences to ensure compatibility with the time-dependent architecture. The model processes all curves in the mini-batch for all time steps in a vectorized format, improving computational efficiency.

Remark 1. In the data generation process, the FFT simulations used for the meso-to-macroscale transition in Digimat-FE inherently rely on a Mori-Tanaka-based Mean-Field Homogenization (MFH) for the microscale homogenization step. While MFH provides an efficient approximation, it does not fully capture the detailed stress and strain distributions within individual yarns, leading to potential inaccuracies when dealing with highly nonlinear material behaviors. In contrast, the HPRNN framework differs in treating the micro-to-mesoscale transition at the yarn level and learns a direct transition from the microscale to the mesoscale.

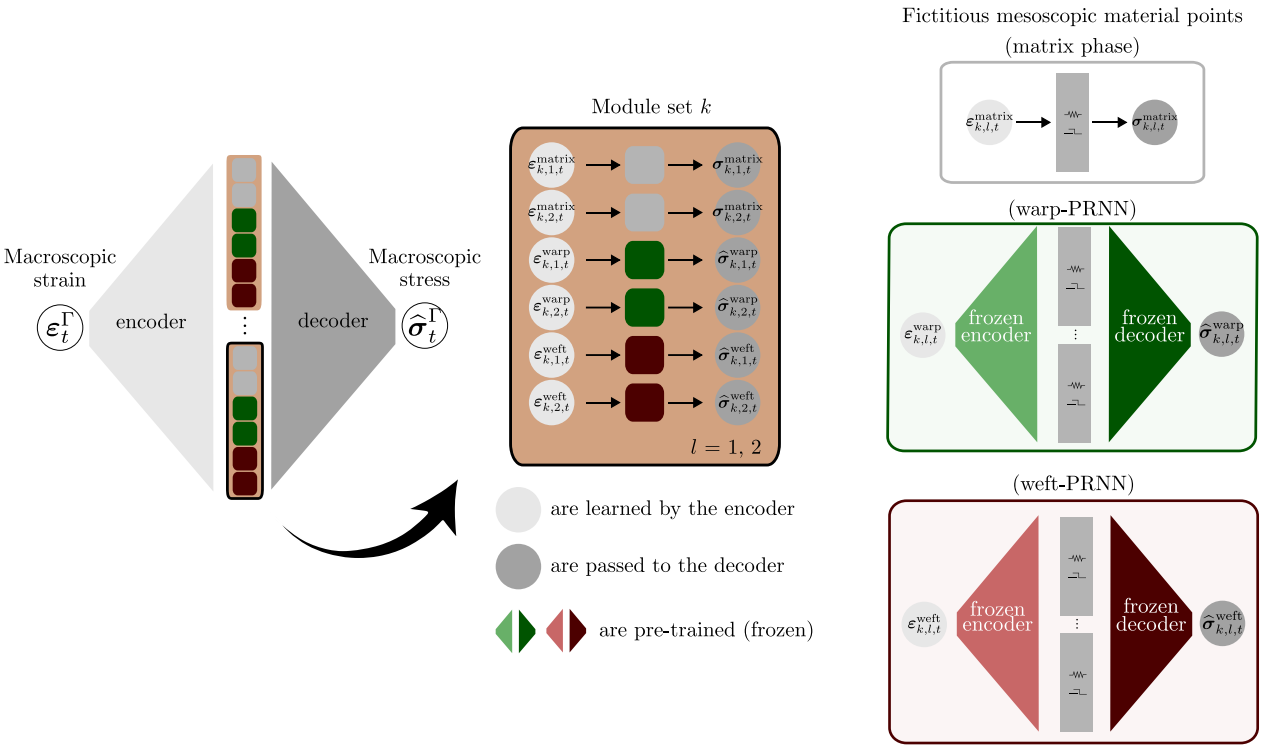


Fig. 7. HPRNN architecture, featuring a standard feed-forward encoder for strain decomposition, a material layer representing the three primary phases of the material at the mesoscale of the woven RVE, and a feed-forward decoder for homogenization.

This discrepancy introduces an interesting scenario (discussed further in Section 6.1): while we train HPRNN at the mesoscale using FFT simulations from Digimat-FE that depend on MFH for their lower-scale homogenization, while the PRNN models *within* HPRNN were trained on data originating from a much more rigorous two-scale transition. In essence, HPRNN is tackling a significantly more complex problem by directly encoding the micro-to-mesoscale transition through finite-element-based data. However, due to computational infeasibility, we lack an FE^2 analysis to fully justify this approach at the macroscale.

Despite this mismatch in training data fidelity, a plausible hypothesis is that the HPRNN is flexible enough to effectively learn the macroscale behavior by adjusting the pre-trained PRNNs' responses as needed to reproduce the FFT data. Consequently, the network predictions can still be meaningfully evaluated against FFT simulations at the macroscale.

6. Results and discussion

In this section, we evaluate the performance of the HPRNN architectures proposed in Section 5. The model selection process is examined by analyzing the network's performance across different training set and material layer sizes. The evaluation specifically focuses on the ability of the HPRNN to accurately capture the homogenized response of woven composites under diverse loading conditions while incorporating microscale constitutive properties.

The PyTorch machine learning library is used to implement the HPRNN. Various configurations are trained on a 16-core computer locally. The training process for all the network configurations uses the ADAM optimizer with a learning rate of 10×10^{-2}. To mitigate overfitting, an early stopping criterion with a patience of 10 epochs is applied, meaning that training ends if the validation error does not improve for 10 consecutive epochs.

Section 6.1 presents the results related to the performance of the original PRNN architecture in capturing the homogenized yarn properties at the microscale. Section 6.2 provides the results for the HPRNN architecture, which models the mesoscale behavior of woven composites. This is built upon the best-performing PRNNs selected in Section 6.1 for modeling the microscale yarns' behavior. This section also compares the effectiveness of encoding physics-based recurrence into feed-forward networks against a conventional path-dependent GRU network, as explored in our previous work [24] and a transformer-based network adopted from [16]. In Section 6.3, the ability of the HPRNN to extrapolate is tested using a cyclic loading scenario and compared with GRU and a Transformer. Finally, the discussion in Section 6.4 highlights the advantages and limitations of the HPRNN architecture in the multiscale analysis of woven composites, providing insights into its potential applications and areas for improvement.

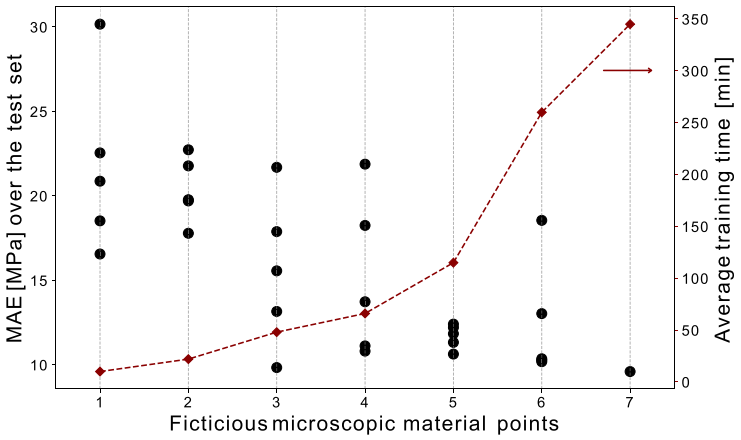


Fig. 8. Validation mean absolute error of the microscale PRNN as a function of the number of fictitious microscopic material points. Each individual black markers along the corresponding vertical line results from a different network initialization.

6.1. *Microscale yarn RVE performance*

We assess the performance of the original PRNN model in capturing the microscale elasto-plastic behavior of yarns using a newly generated random-walk dataset as described in Section 3.3. The generated dataset features non-monotonic and non-proportional strain paths based on FE computational homogenization, providing a comprehensive training and test set to evaluate the micro-PRNNs' ability to generalize to complex loading scenarios. Each micro-to-meso PRNN is trained on 200 strain paths drawn from a larger pool of loading paths, a dataset size previously suggested by Maia et al. [43] as sufficient for capturing elasto-plasticity in unidirectional composites, then evaluated on 20 unseen samples.

To evaluate the robustness of the PRNN architecture, a sensitivity analysis was performed across multiple configurations of fictitious microscopic material points and different networks random initializations. The results, summarized in Fig. 8, show a reduction in minimum mean absolute error as the number of fictitious microscopic material points increases from 1 to 7, corresponding to 6 to 42 neurons in the material layer as described in Section 4. Due to high computational cost only one network with 42 neurons is reported. The minimum mean absolute error reduction becomes less pronounced beyond moderate latent resolutions, indicating that a small material layer, combined with sufficient training repetitions, is adequate to capture the microscale RVE response. As a result, a PRNN with six fictitious material points equivalent to 36 neurons in the material layer is chosen for both yarn directions. The mean relative errors of the two networks are, as expected, quite similar, around 3.8% and 3.5% for warp and weft directions, respectively

The warp and weft networks exhibit comparable performance, confirming stable predictive capability across yarn directions. The right vertical axis of Fig. 8 also reports the average training time per configuration on an AMD Ryzen 7 processor with 8 cores. A substantial increase in computational cost is observed as the number of fictitious microscopic material points increases, particularly for configurations with six and seven units.

Since a similar analysis has already been performed by Maia et al. [43] on a different material system, we keep our discussion brief and do not include further details here. The accuracy of the chosen networks is illustrated in Fig. 9 using the weft-PRNN to predict the stress response[4] in one of the unseen random loading cases.

To further clarify the activation of plasticity within the physics-encoded material layer, Fig. 10 illustrates the evolution of equivalent plastic strain at the fictitious material points for a representative random loading case, obtained using a micro-PRNN with three fictitious material points. This quantity reflects the evolution of internal variables governed by the embedded J2 return-mapping algorithm within the recurrent structure. However, these plastic strain variables correspond to fictitious material points in the latent space and should not be interpreted as volume-averaged plastic strain or yielded fractions obtained from finite element analyses. Rather, they demonstrate that the PRNN follows the same constitutive update rule as classical plasticity models, while operating in a reduced, non-spatial representation. The purpose of this figure is therefore to illustrate the plastic evolution within the network, not to establish direct equivalence with FE-based field quantities.

Remark 2. As outlined in Section 5.4, Fig. 11 compares the micro-to-mesoscale transition for weft yarns under two standard loading scenarios that were not included in the training set. The results are presented for three different approaches:

- *Finite-Element-Based Full-order Homogenization.* Serves as the reference data for training warp-PRNN and weft-PRNN models, providing the *target* stress responses.

[4] The symmetric stress tensor in Voigt notation is represented as a six-component column vector as follows: $\tilde{\boldsymbol{\sigma}} = (\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{13}, \sigma_{23})^T \equiv (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6)^T$

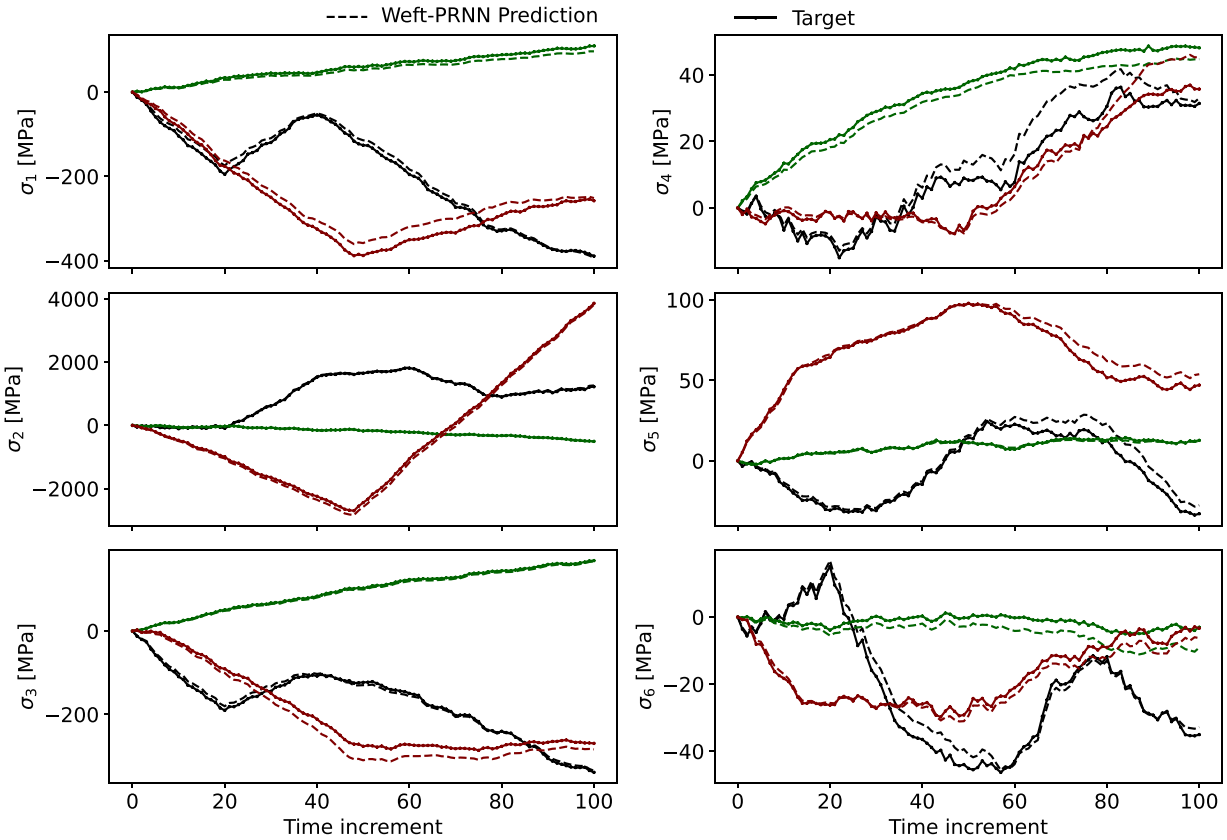


Fig. 9. Prediction of stress components by the weft-PRNN with six fictitious material points for three randomly selected samples from the test set under random loading conditions.

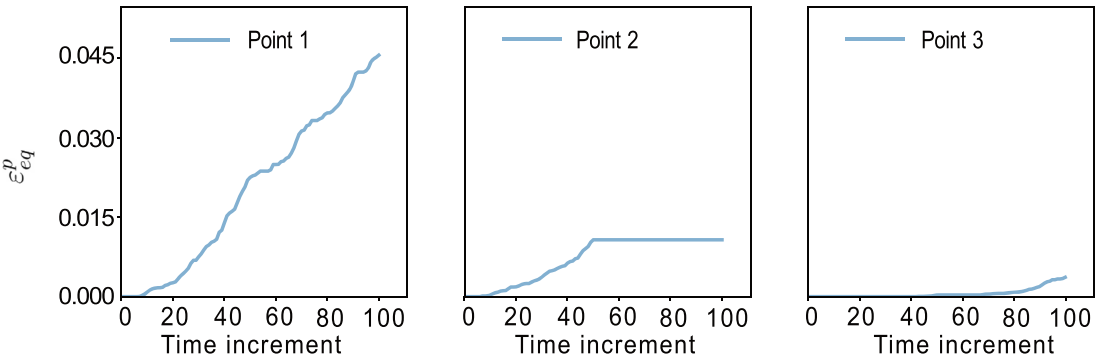
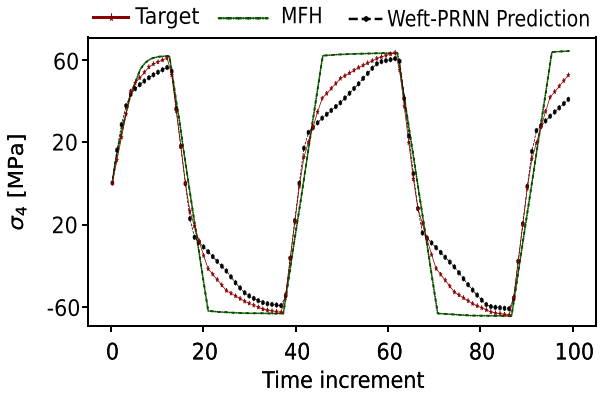


Fig. 10. Evolution of equivalent plastic strain in a micro-PRNN with three fictitious material points during a random loading..

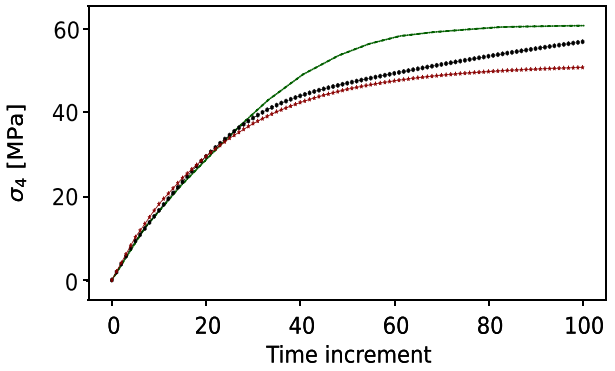
- *PRNN Prediction.* Represents the output of the trained PRNN models, capturing the nonlinear and path-dependent behavior of yarns.
- *Mean-Field Homogenization (MFH) in Digimat-MF.* Used to homogenize the lower scale during FFT/FE simulations at the macroscale.

The comparison shows that PRNN predictions align closely with FE results, accurately capturing the nonlinear and path-dependent behavior of yarns. PRNN successfully extrapolates from a random loading dataset to standard loading cases, demonstrating its robustness and generalization capability. However, the MFH approach shows larger deviations from FE results. This highlights the strength of HPRNN's physics-encoded architecture, which effectively learns full-order homogenization without relying on simplified mean-field assumptions.

That said, this also means that, in the next stage, when integrating all modules, the HPRNN is trained on "corrupted" data, as the homogenized mesoscale data relies on a different homogenization scheme (i.e. FFT based on MFH) than the one used to train the



(a) Cyclic in-plane shear loading.



(b) Uni-axial in-plane shear loading.

Fig. 11. Comparison of micro-to-mesoscale transition for weft yarns under two standard loading scenarios not included in the training set.

embedded warp and weft PRNNs, raising questions about the accuracy of results such as those in Section 6.3. While this limitation exists, we acknowledge it and proceed with caution in interpreting subsequent results.

6.2. Mesoscale woven RVE surrogate performance

In the next phase, the mesoscale dataset is used to train and evaluate different neural network architectures for predicting the homogenized behavior of woven composites. The optimum number of training set sizes is selected based on the minimum test error observed among the trained mesoscale networks according to Fig. 12. We have used the von Mises stress to incorporate all six stress components predicted by the network. This simplifies the assessment by consolidating multiple stress tensor components into a single scalar value with clear physical relevance. The Mean Absolute Error (MAE) measures the average magnitude of errors over the data sequence length (N_T) between the predicted and target values. MAE is calculated by taking the average of the absolute differences between each predicted and desired von Mises stress value, normalized by the number of tested samples from the unseen dataset (M) as:

$$\text{MAE} = \frac{1}{M N_T} \sum_{i=1}^{M} \sum_{t=1}^{N_T} \left| \hat{\sigma}_{vM,i}^{(t)} - \sigma_{vM,i}^{(t)} \right|, \tag{9}$$

where $\hat{\sigma}_{vM}^{(t)}$ is the predicted von Mises stress at time step t and $\sigma_{vM}^{(t)}$ is the desired von Mises stress at t. MAE indicates the average size of errors produced by the model.

In this case, the networks with 200 and 300 training set sizes exhibited the lowest error values. While increasing the training set size generally improves model performance, it also significantly impacts computational cost. On a CPU with 32 processors, training mesoscale networks with 300 samples takes approximately 7 to 13 hours, while training with 200 samples takes approximately 4 hours. Given this trade-off between computational time and model accuracy, here we opt for the model trained with 200 samples for its efficiency.

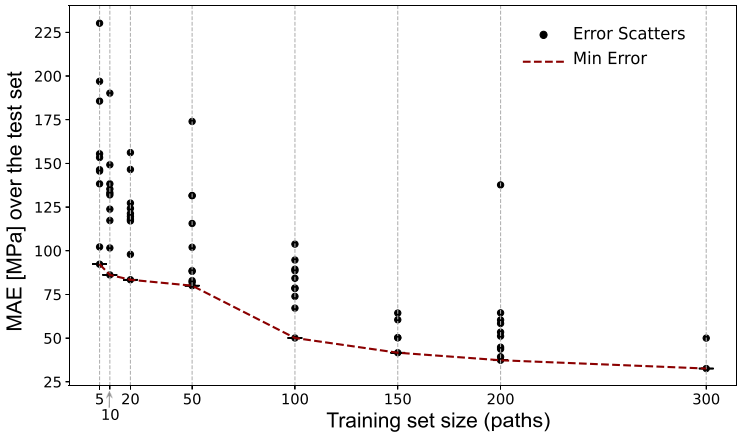


Fig. 12. Comparison of von Mises stress mean absolute errors for the mesoscale network (HPRNN) configured with three latent space module sets, trained across varying training set sizes. Each training size involves training multiple networks (with the same configuration) with different random initializations, evaluated on a fixed test set of 20 unseen simulations. The dashed line highlights the minimum errors achieved within each ensemble.

Using a fixed number of training samples generated from random-walk loadings, various HPRNN models are trained with different numbers of module sets. A module set consists of two bulk points, two warp-PRNNs, and two weft-PRNNs, amounting to six modules in total per set. Fig. 13 illustrates the effect of increasing the number of module sets from one to four. For each configuration, 10 networks with different initializations are trained and evaluated on the same test set. The results, shown in Fig. 13(b), indicate that the difference in errors between three and four module sets is minimal. Based on this observation, the configuration with three module sets (18 modules in total: 3×6 for each set) is selected as the optimal model.

In Fig. 14, a randomly selected simulation from the test set of the random loading dataset is compared with the predicted values from the proposed architecture. The results demonstrate that the HPRNN model effectively captures the sequential nonlinear elasto-plastic behavior of the woven composite.

6.3. Extrapolation capability: HPRNN vs. GRU and transformer

This section explores the extrapolation capabilities of HPRNN compared to GRU and transformer-based architectures. A GRU-based model, previously employed in [20,24], is used as a benchmark. The GRU model consists of three gated recurrent layers, each with a hidden size 512, and includes a dropout layer with a rate of 0.5 to reduce overfitting. The final fully connected layer maps the GRU output to the desired dimensions, processing input and output sequences of size 6. The model is trained with the AdamW optimizer (learning rate $= 0.001$, weight decay $= 1 \times 10^{-4}$) using a StepLR scheduler, which reduces the learning rate by 10% every five epochs.

In comparison, a transformer-based architecture adopted from [16] is implemented to evaluate its performance. The baseline transformer model includes three encoder and decoder layers, each with a hidden size of 40, and two attention heads. The feed-forward sub-layer in each block has a dimensionality of 120. A dropout rate of 0.1 is applied to prevent overfitting. The transformer is trained using the AdamW optimizer with a learning rate of 0.0005, combined with a warm-up cosine learning rate scheduler that ramps up over the first 40 epochs and decays thereafter.

To ensure a fair comparison in terms of effective capacity and training effort, an extended hyperparameter optimization was conducted for the transformer architecture. Approximately 60 transformer configurations were trained and evaluated using a structured greedy search over the embedding dimension, number of attention heads, number of encoder and decoder layers, feed-forward dimensionality, learning rate, warm-up duration, dropout rate, and early stopping patience. All models were trained using the same dataset splits, loss function, normalization strategy, maximum epoch budget of 3000 epochs, and early stopping criterion based on validation loss as the GRU and HPRNN models. The best-performing transformer configuration was found in a moderate-capacity regime with the embedding dimension 224, 8 attention heads, 2 encoder and 2 decoder layers, and a feed-forward dimension of 384, resulting in 1,908,454 trainable parameters. Increasing the transformer capacity beyond this range did not lead to improved predictive accuracy, and several substantially larger models exhibited degraded generalization performance. The MSE loss function on normalized data using RobustScaler is used to train GRU and transformer-based models as shown in Fig. 15.

Even when practically doubling the training set size from 200 to 437 samples compared to the GRU and HPRNN models, the transformer struggles to accurately predict random loading cases, as illustrated in Fig. 14 for three randomly selected samples from the random loading dataset. Furthermore, to evaluate the performance of the three networks across all random loading test samples, the error values relative to the target stress ($\hat{\sigma}^{(t)}_{vM,i} - \sigma^{(t)}_{vM,i}$) are plotted for all six stress components in Fig. 16. The shaded regions in the figure represent errors' uncertainty (standard deviation). The transformer shows the highest error standard deviation values across time increments, even in the initial states corresponding to zero stress values, while the GRU network and HPRNN exhibit similar error distributions. However, the HPRNN performs better, particularly for out-of-plane tensile (σ_6) and in-plane shear stress

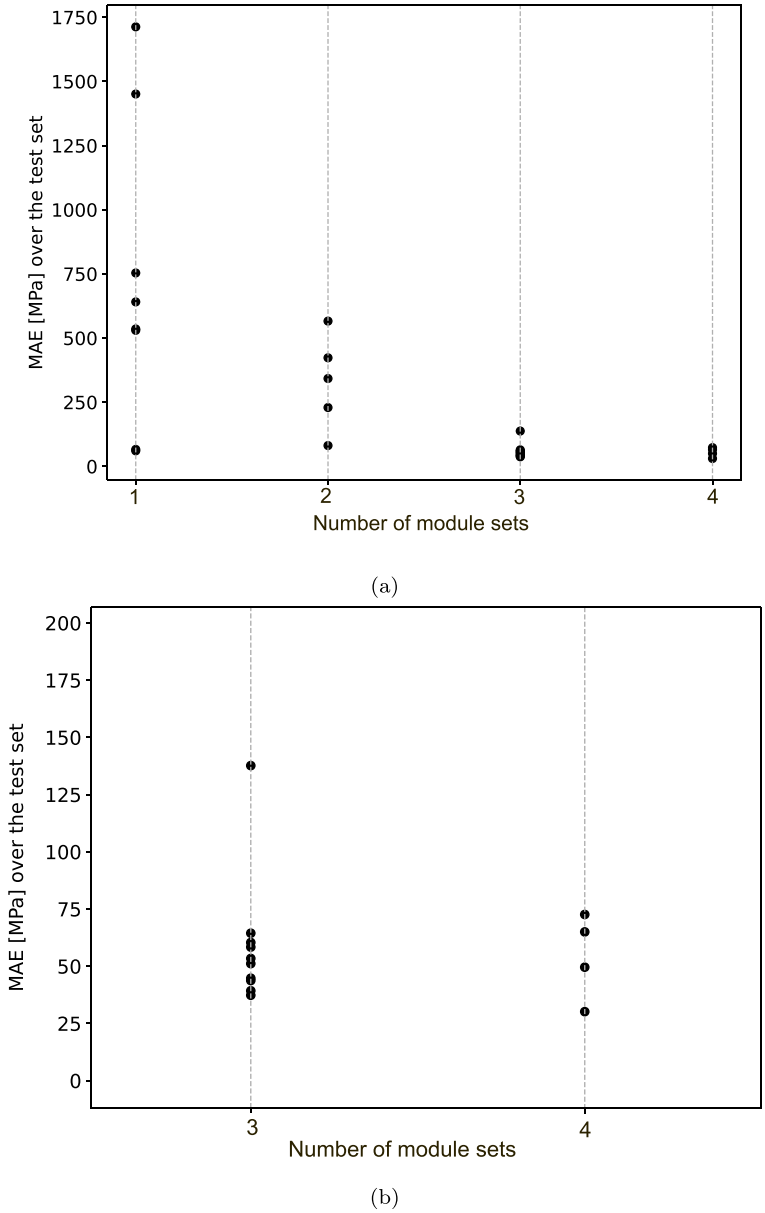


Fig. 13. (a) Comparison of validation mean absolute errors for the mesoscale network configured with one to four numbers of module sets in the latent space, trained across 200 training samples, and evaluated on a consistent set of 20 unseen simulations. (b) Zoomed in networks with three and four module sets.

components (σ_4 and σ_5), where the mean error over all test samples for σ_6 is 0.03 MPa versus 0.44 MPa for the GRU (with equal standard deviation). For σ_4, the errors are 0.15 MPa versus 0.35 MPa, and for σ_5 they are 1.46 MPa and 1.05 MPa, respectively.

To enable a consistent and magnitude-independent comparison across stress components with markedly different scales, such as normal and shear stresses, Fig. 16 is complemented by the corresponding relative error representation in Fig. 17. Fig. 17 illustrates the relative error distribution of GRU and HPRNN models to facilitate a balanced comparison across different stress components, which inherently exhibit varying magnitudes. The errors are normalized with respect to the absolute maximum target stress at each time increment, enabling a more objective assessment of predictive accuracy by accounting for differences in the order of magnitude among stress components.

Both models demonstrate a strong predictive performance for normal stress components, likely due to their lower sensitivity to noise introduced during the load generation process. Moreover, normal stresses exhibit lower variance compared to shear stresses.

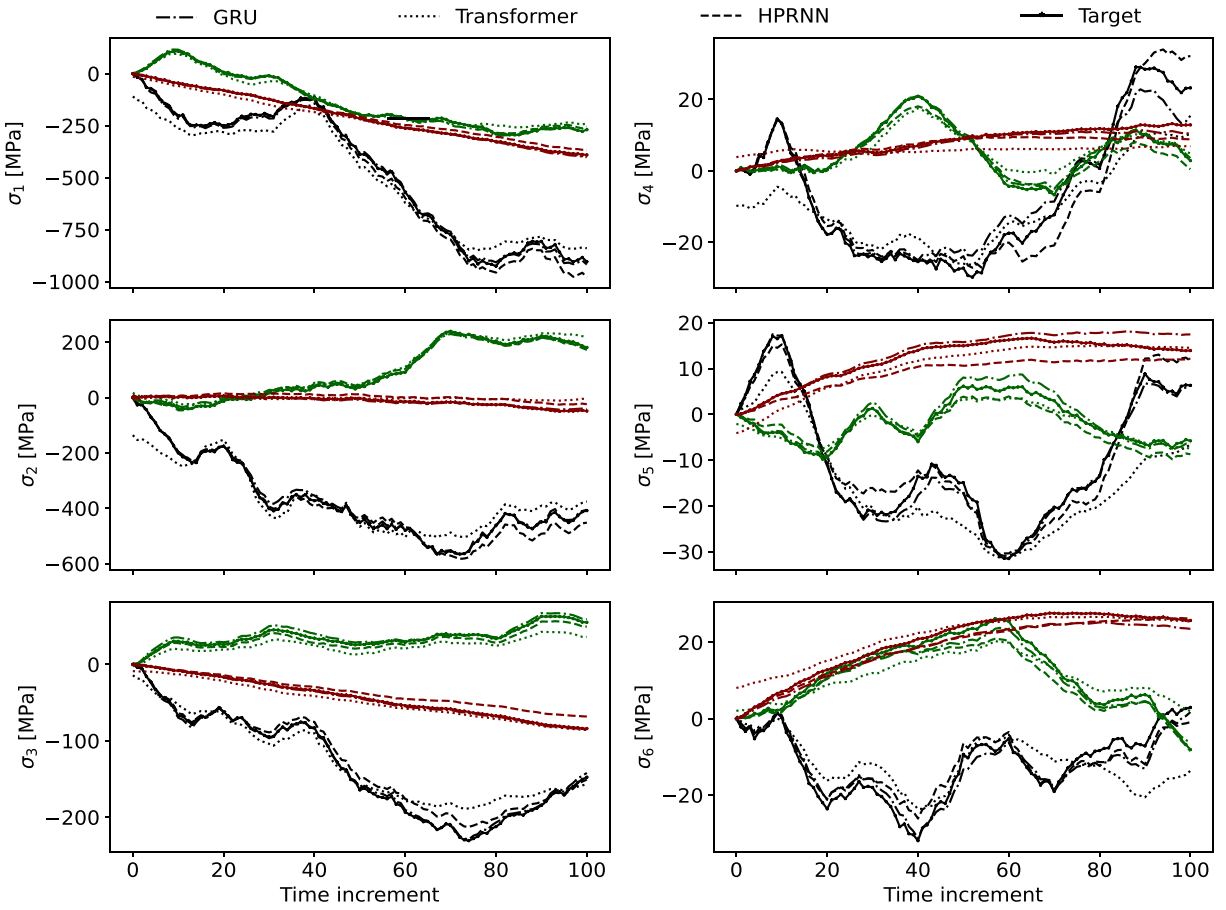


Fig. 14. Prediction of stress components by the mesoscale HPRNN, GRU, and Transformer models on three randomly selected unseen samples from random loading data, each represented by a different color. Networks predictions are plotted against simulations, with solid lines indicating the target values from high-fidelity simulations.

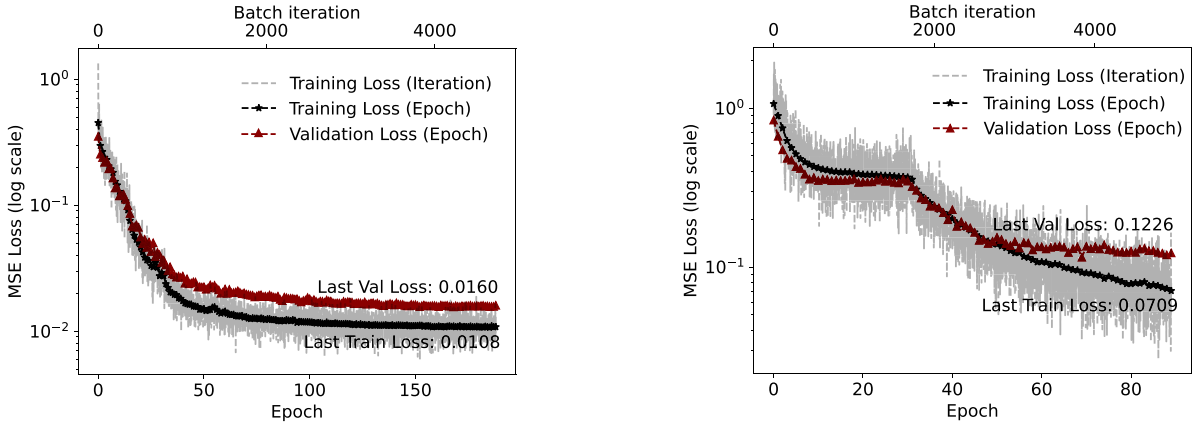


Fig. 15. (a) GRU and (b) transformer training curves over 200 and 437 random loading simulations, respectively. The GRU model with an optimized configuration is introduced in [24]. The transformer model is adopted from the introduced configuration in [16]. Both networks are trained with an early stopping criterion set to the patience of 20 epochs.

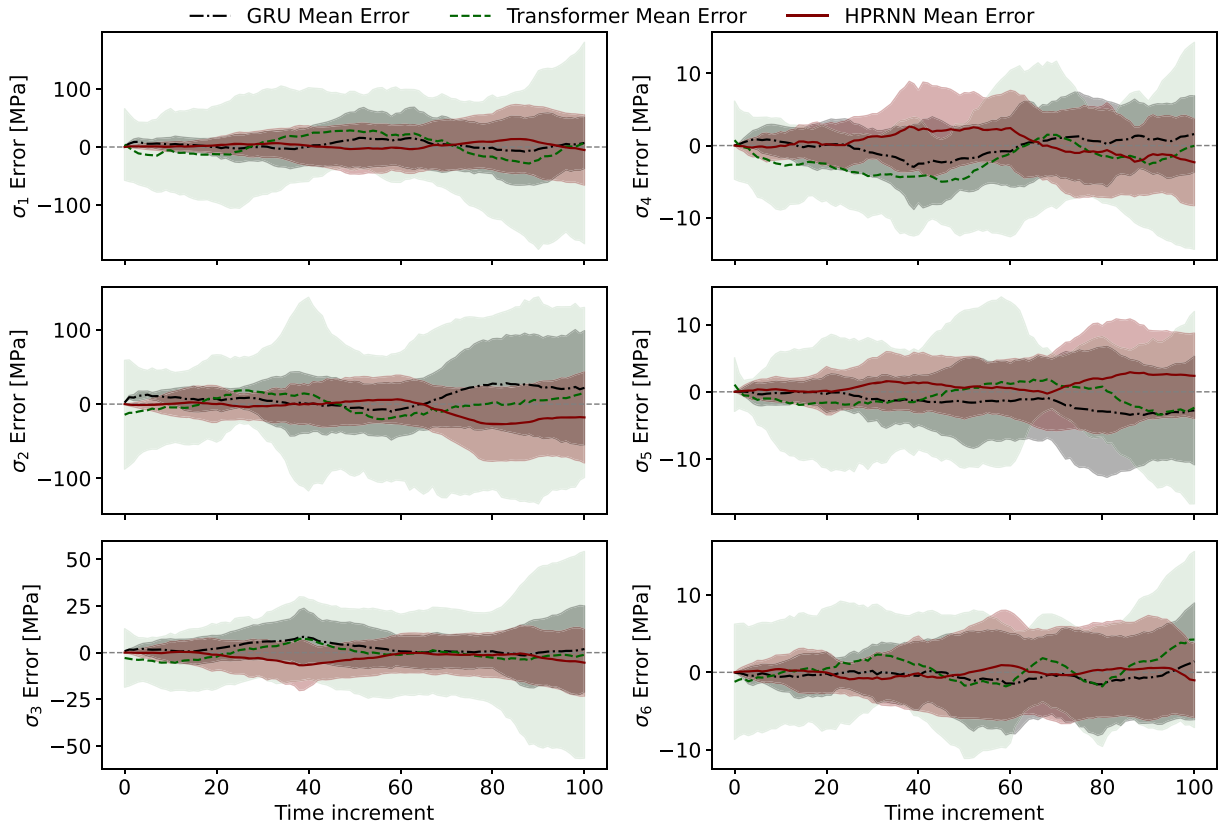


Fig. 16. Mean error and error standard deviation comparison for 20 random loading samples from the test set, evaluated across three networks: GRU recurrent neural network (gray), Transformer network (green), and HPRNN (red). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The shear stress components exhibit a relatively higher dispersion compared to the normal components. However, no systematic temporal drift or consistent degradation in prediction accuracy is observed within the evaluated loading sequences. Based on the present dataset, there is therefore no clear evidence of instability specific to shear-dominated loading scenarios.

The GRU model achieves a performance comparable to HPRNN in predicting random load responses. However, to further evaluate the performance of the HPRNN architecture, GRU and HPRNN are tested against cyclic loading samples described in Section 3.1 that were not part of the training set. As it was already introduced in [20], recurrent neural networks, including GRUs, struggle with extrapolation from random loads to such cyclic loading, particularly under unseen load patterns. In cyclic load cases, such as single and multiple cycles shown in Figs. 18b and 18c, the GRU model exhibits nonphysical softening behavior and inconsistent maximum and minimum stress values. This behavior persists despite normalization during training, indicating limitations in GRU's recurrent architecture, which relies on hidden states for recurrence.

The HPRNN, on the other hand, demonstrates superior performance in extrapolation tasks, avoiding nonphysical behaviors due to its physics-encoded recurrence mechanism. By embedding the material constitutive laws into the network, HPRNN accurately models the path-dependent elasto-plastic responses of woven composites. Fig. 19 highlights its ability to maintain physically consistent predictions even under complex cyclic loading scenarios, outperforming GRU and transformer-based models. While transformers are powerful for specific sequential tasks, their inability to generalize effectively to path-dependent behaviors in this study underscores their unsuitability for woven composite modeling.

6.4. Limitations and strengths

Despite the advancements of the HPRNN model and its success in capturing the complex behavior of woven composites, the architecture faces several challenges. HPRNN is trained to a specific geometrical system (e.g., volume fraction) and requires retraining for different material configurations. In contrast, data-driven models like GRUs and LSTMs [12] can generalize across various weave patterns by incorporating geometrical variables. With sufficient data, they can also capture geometric variations like fiber waviness or porosity. While HPRNN ensures physics-encoded learning, this lack of flexibility regarding geometrical features hinders its applicability to diverse material systems. Furthermore, HPRNN's complex architecture makes its implementation less straightforward than conventional time-dependent neural networks.

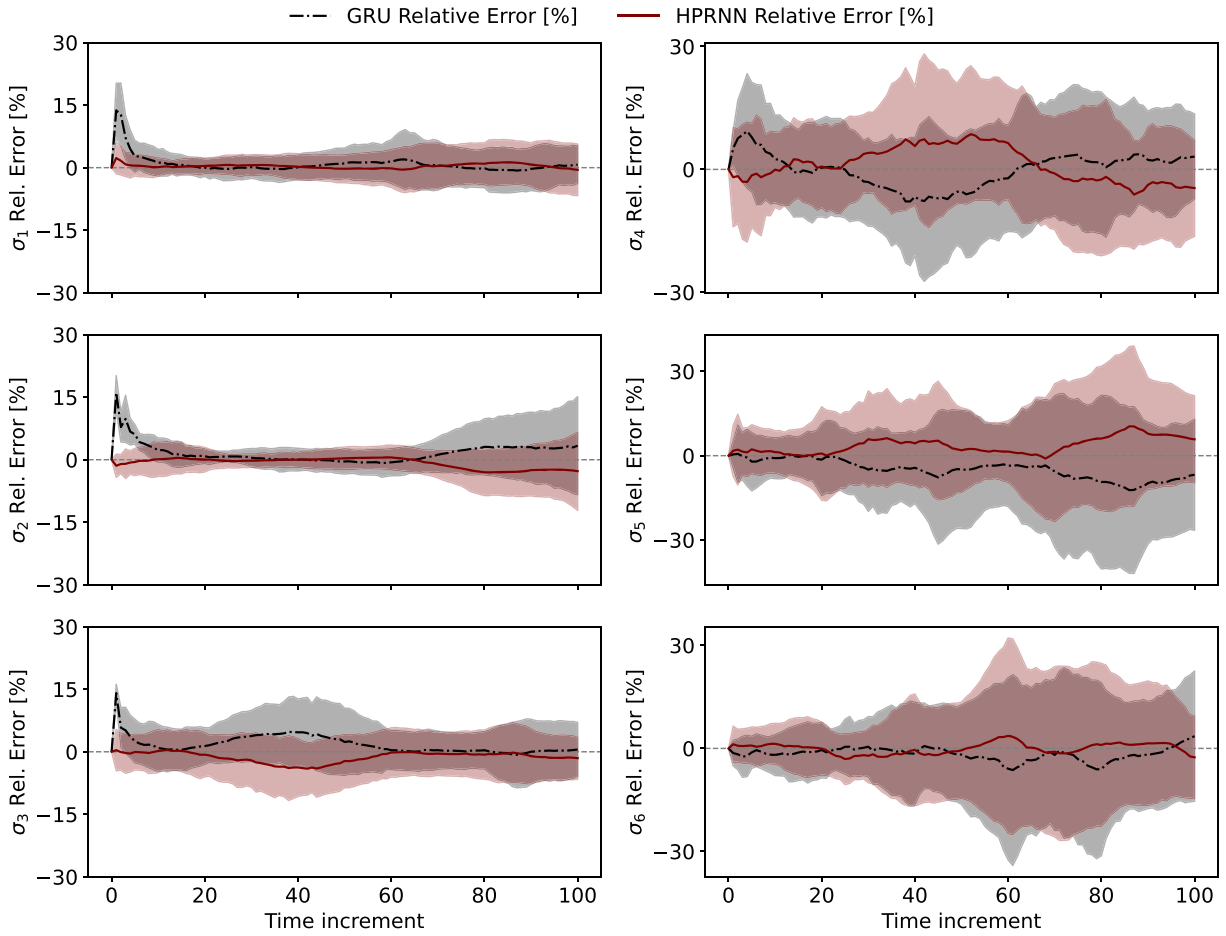


Fig. 17. The relative error distribution of GRU (black) and HPRNN (red) models is normalized with respect to the absolute maximum target stress at each time increment. The shaded regions represent the standard deviation of errors over the selected test samples. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Although the HPRNN architecture improves upon GRU networks by avoiding nonphysical behavior in cyclic loading extrapolation, its predictions are not entirely accurate for all cyclic load scenarios. These discrepancies suggest a potential limitation in the current data generation strategy, specifically in using MFH for subscale transition in FFT-based data generation.

A remaining limitation of the present study is the inability to quantitatively distinguish discrepancies originating from the mean field homogenization used in the FFT based data generation from intrinsic interpolation or extrapolation errors of the HPRNN, particularly under cyclic loading conditions. This limitation arises from the lack of a fully consistent FE^2 reference at the macroscale, which would be required to isolate MFH induced target bias from model related errors. Future work will therefore focus on developing a true FE^2 based validation framework to enable a rigorous and quantitative assessment of these error contributions and to further evaluate the predictive robustness of the proposed HPRNN under complex cyclic loading paths.

Despite these complexities, HPRNN's efficient algorithms in finite-dimensional settings, robustness in handling data scarcity, and strong generalizability offer significant potential for accelerating traditional scientific computing in applications such as computational solid mechanics.

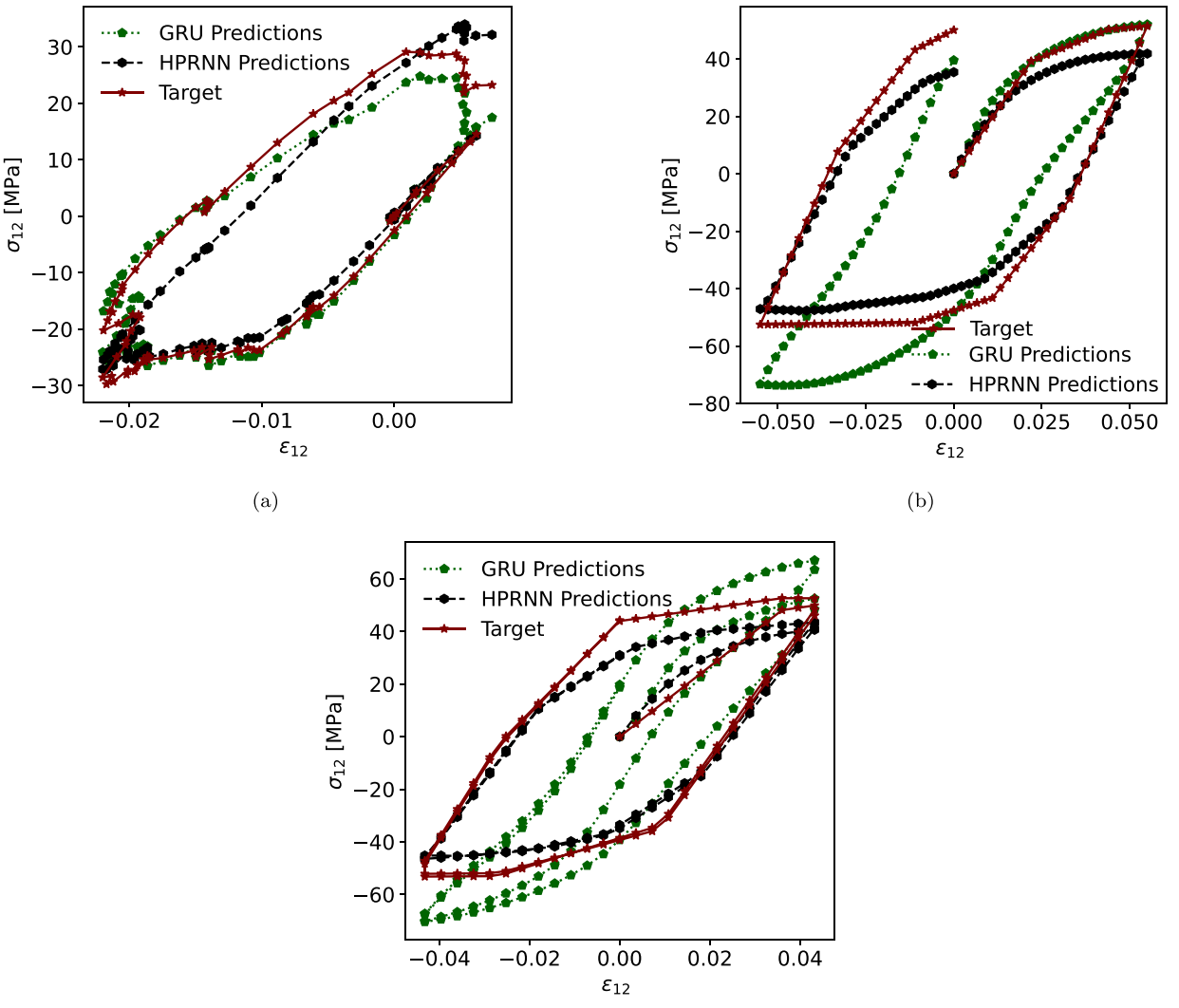


Fig. 18. In-plane stress vs. strain behavior in testing against (a) unseen random loading and extrapolation over unseen cyclic load cases. (b) one cycle, and (c) two cycles.

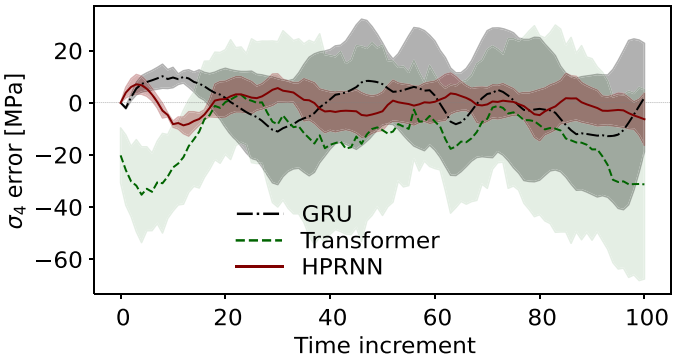


Fig. 19. Mean error and error standard deviation comparison for five in-plane shear cyclic loading samples from the test set, evaluated across three networks: GRU recurrent neural network (gray), Transformer network (green), and HPRNN (red). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

7. Conclusions

This study presents the hierarchical physically recurrent neural network (HPRNN) as an efficient and physics-encoded framework for modeling woven composites. HPRNN combines data-driven learning with physical principles to predict the elasto-plastic behavior of woven composites. By incorporating pretrained PRNNs for warp and weft yarns along with a matrix constitutive model, HPRNN captures detailed microscale material behavior and propagates it to the mesoscale. Furthermore, in contrast to conventional history-dependent neural networks, such as GRU- and transformer-based architectures, HPRNN mitigates nonphysical behaviors in extrapolation tasks by leveraging its physics-encoded recurrence design.

While transformers excel at capturing long-term dependencies in language-based data, their performance is limited when applied to path-dependent systems, such as the stress-strain behavior of the material system under study. Even with a doubled training set size, the transformer model failed to match GRU and HPRNN performance in random loading cases. Unlike RNNs or physics-encoded networks, transformers do not inherently maintain a notion of "memory" of previous states beyond what is encoded in the attention mechanism.

Key findings reveal that while the GRU network and HPRNN perform similarly under random loading conditions, their differences become evident in cyclic loading scenarios absent from the training data. HPRNN outperforms GRU by maintaining physically consistent stress-strain evolution, preventing nonphysical softening and unrealistic peak values. This advantage stems from HPRNN's physics-encoded recurrence, which captures constitutive behavior directly from material points, unlike GRU, which relies solely on hidden states.

Future research should focus on improving HPRNN's ability to generalize to different microscale geometries, accounting for aspects such as volume fraction and reinforcement morphology, as well as training and testing on other types of loading for improved performance in the cyclic extrapolation scenario. Additionally, expanding the hierarchical framework to incorporate damage mechanisms, dynamic effects, and anisotropic fiber behaviors could enhance its applicability to a broader range of materials and loading conditions. In particular, softening bulk damage up to the point prior to strain localization and cohesive interface behaviors, such as fiber–matrix debonding, can be incorporated by augmenting the material layer with additional internal variables and corresponding evolution laws within the recurrent structure [54]. Owing to its modular physics-encoded formulation, the HPRNN framework allows such extensions in a systematic manner without modifying the overall hierarchical architecture. Another key direction is optimizing training efficiency through parallel processing, advanced batch handling, and streamlined computation of physics-based activation functions to enable HPRNN's application to large-scale industrial problems.

The HPRNN framework provides a promising approach to accelerating multiscale modeling of woven composites, offering a unique advantage over conventional data-driven homogenization techniques by enabling a physics-consistent transition across micro, meso, and macro scales. This makes it a valuable tool for engineering design, material optimization, and the efficient prediction of composite behavior in real-world applications.

Declaration of generative AI and AI-assisted technologies in the manuscript preparation process

During the preparation of this work the authors used ChatGPT to improve readability and language. After using this tool/service, the author(s) reviewed and edited the content as needed and takes full responsibility for the content of the published article.

CRediT authorship contribution statement

Ehsan Ghane: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Marina A. Maia:** Writing – review & editing, Visualization, Validation, Software, Methodology, Investigation, Conceptualization; **Iuri B. C. M. Rocha:** Writing – review & editing, Supervision, Methodology, Investigation, Conceptualization; **Martin Fagerström:** Writing – review & editing, Supervision, Methodology, Investigation, Conceptualization; **Mohsen Mirkhalaf:** Writing – review & editing, Supervision, Funding acquisition.

Data availability

Data will be made available on request.

Declaration of competing interest

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

Acknowledgment

Ehsan Ghane and Mohsen Mirkhalaf gratefully acknowledge financial support from the Swedish Research Council (VR grant: 2019–04715) and the University of Gothenburg. Martin Fagerström is thankful for the support through Vinnova's strategic innovation programme LIGHTer, in particular via the project LIGHTer Academy Phase 4 (grant no. 2023-01937). Iuri Rocha and Marina Maia gratefully acknowledge support from the TU Delft AI Labs programme.

References

[1] T.-R. Liu, Y. Yang, O.R. Bacarreza, S. Tang, F.M. Aliabadi, An extended full field self-consistent cluster analysis framework for woven composite, Int. J. Solids Struct. 281 (2023) 112407.
[2] O. Erol, B.M. Powers, M. Keefe, Effects of weave architecture and mesoscale material properties on the macroscale mechanical response of advanced woven fabrics, Compos. Part A: Appl. Sci. Manuf. 101 (2017) 554–566.
[3] D.K. Patel, A.M. Waas, C.-F. Yen, Compressive response of hybrid 3D woven textile composites (H3DWTCs): an experimentally validated computational model, J. Mech. Phys. Solids 122 (2019) 381–405.
[4] M.G.D. Geers, V.G. Kouznetsova, W.A.M. Brekelmans, Multi-scale computational homogenization: trends and challenges, J. Comput. Appl. Math. 234 (7) (2010) 2175–2182.
[5] J. Schröder, A numerical two-scale homogenization scheme: the FE2-method, in: Plasticity and beyond: Microstructures, Crystal-plasticity and Phase Transitions, Springer, 2014, pp. 1–64.
[6] C. Mora, A. Yousefpour, S. Hosseinmardi, H. Owhadi, R. Bostanabad, Operator learning with gaussian processes, Comput. Methods Appl. Mech. Eng. 434 (2025) 117581.
[7] Y. Yamanaka, S. Matsubara, N. Hirayama, S. Moriguchi, K. Terada, Surrogate modeling for the homogenization of elastoplastic composites based on RBF interpolation, Comput. Methods Appl. Mech. Eng. 415 (2023) 116282.
[8] L. Herrmann, S. Kollmannsberger, Deep learning in computational mechanics: a review, Comput. Mech. (2024) 1–51.
[9] M.A. Bessa, R. Bostanabad, Z. Liu, A. Hu, D.W. Apley, C. Brinson, W. Chen, W.K. Liu, A framework for data-driven analysis of materials under uncertainty: countering the curse of dimensionality, Comput. Methods Appl. Mech. Eng. 320 (2017) 633–667.
[10] I.B.C.M. Rocha, P. Kerfriden, F.P. van Der Meer, On-the-fly construction of surrogate constitutive models for concurrent multiscale mechanical analysis through probabilistic machine learning, J. Comput. Phys.: X 9 (2021) 100083.
[11] Y. Liu, J. Zhang, L. Liu, Data-driven multiscale modeling of materials using neural networks: recent trends and perspectives, Appl. Mech. Rev. 73 (3) (2021) 030801.
[12] E. Ghane, M. Fagerström, M. Mirkhalaf, A multiscale deep learning model for elastic properties of woven composites, Int. J. Solids Struct. 282 (2023) 112452.
[13] M.J. Buehler, MechGPT, a language-Based strategy for mechanics and materials modeling that connects knowledge across scales, disciplines, and modalities, Appl. Mech. Rev. 76 (2) (2024) 021001.
[14] J.N. Fuhg, G.A. Padmanabha, N. Bouklas, B. Bahmani, W. Sun, N.N. Vlassis, M. Flaschel, P. Carrara, L. De Lorenzis, A review on data-driven constitutive laws for solids, (2024), arXiv preprint arXiv:2405.03658.
[15] M. Mirkhalaf, I. Rocha, Micromechanics-based deep-learning for composites: challenges and future perspectives, Eur. J. Mech.-A/Solids 105 (2024) 105242.
[16] Y. Zhongbo, P.L. Hien, Pre-trained transformer model as a surrogate in multiscale computational homogenization framework for elastoplastic composite materials subjected to generic loading paths, Comput. Methods Appl. Mech. Eng. 421 (2024) 116745.
[17] Y. Hu, M.J. Buehler, Deep language models for interpretative and predictive materials science, APL Mach. Learn. 1 (1) (2023).
[18] E. Pitz, K. Pochiraju, A neural network transformer model for composite microstructure homogenization, Eng. Appl. Artif. Intell. 134 (2024) 108622. https://doi.org/10.1016/j.engappai.2024.108622
[19] F. Ghavamian, A. Simone, Accelerating multiscale finite element simulations of history-dependent materials using a recurrent neural network, Comput. Methods Appl. Mech. Eng. 357 (2019) 112594.
[20] E. Ghane, M. Fagerström, M. Mirkhalaf, Recurrent neural networks and transfer learning for predicting elasto-plasticity in woven composites, Eur. J. Mech.-A/Solids (2024) 105378.
[21] Q.-M. Zhong, D.-C. Feng, S.-Z. Chen, Multi-fidelity enhanced few-shot time series prediction model for structural dynamics analysis, Comput. Methods Appl. Mech. Eng. 434 (2025) 117583.
[22] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, A comprehensive survey on transfer learning, Proc. IEEE 109 (1) (2021) 43–76.
[23] J. Yi, B.P. Ferreira, M.A. Bessa, Single-to-multi-fidelity history-dependent learning with uncertainty quantification and disentanglement: application to data-driven constitutive modeling, Comput. Methods Appl. Mech. Eng. 448 (2026) 118479.
[24] E. Ghane, M. Fagerström, M. Mirkhalaf, Multi-fidelity data fusion for inelastic woven composites: combining recurrent neural networks with transfer learning, Compos. Sci. Technol. 267 (2025) 111163.
[25] X. Xu, C. Liu, Physics-guided deep learning for damage detection in CFRP composite structures, Compos. Struct. 331 (2024) 117889.
[26] K. Willcox, O. Ghattas, C. Farhat, H. Narayanan, Machine learning and physics-based modeling: complementary approaches for data-driven science, Nature Comput. Sci. 1 (3) (2021) 164–176.
[27] M. Raissi, P. Perdikaris, N. Ahmadi, G.E. Karniadakis, Physics-informed neural networks and extensions, (2024), arXiv preprint arXiv:2408.16806.
[28] E. Haghighat, M. Raissi, A. Moure, H. Gomez, R. Juanes, A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics, Comput. Methods Appl. Mech. Eng. 379 (2021) 113741.
[29] S. Rezaei, A. Moeineddin, A. Harandi, Learning solutions of thermodynamics-based nonlinear constitutive material models using physics-informed neural networks, Comput. Mech. 74 (2) (2024) 333–366.
[30] A. Henkes, H. Wessels, R. Mahnken, Physics informed neural networks for continuum micromechanics, Comput. Methods Appl. Mech. Eng. 393 (2022) 114790.
[31] M.E.F. Idrissi, F. Praud, F. Meraghni, F. Chinesta, G. Chatzigeorgiou, Multiscale thermodynamics-Informed neural networks (muTINN) towards fast and frugal inelastic computation of woven composite structures, J. Mech. Phys. Solids 186 (2024) 105604.
[32] V. Dwivedi, N. Parashar, B. Srinivasan, Distributed learning machines for solving forward and inverse problems in partial differential equations, Neurocomputing 420 (2021) 299–316.
[33] M. Rosenkranz, K.A. Kalina, J. Brummund, W. Sun, M. Kästner, Viscoelasticty with physics-augmented neural networks: model formulation and training methods without prescribed internal variables, Comput. Mech. 74 (6) (2024) 1279–1301.
[34] K. Garanger, J. Kraus, J.J. Rimoli, Symmetry-enforcing neural networks with applications to constitutive modeling, Extreme Mech. Lett. 71 (2024) 102188.
[35] C. Rao, H. Sun, Y. Liu, Hard encoding of physics for learning spatiotemporal dynamics,(2021), arXiv preprint arXiv:2105.00557.
[36] R.T.Q. Chen, Y. Rubanova, J. Bettencourt, D.K. Duvenaud, Neural ordinary differential equations, Adv. Neural Inf. Process. Syst. 31 (2018).
[37] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, A. Anandkumar, Fourier neural operator for parametric partial differential equations,(2020), arXiv preprint arXiv:2010.08895.
[38] F. Mostajeran, S.A. Faroughi, Epi-ckans: Elasto-plasticity informed kolmogorov-arnold networks using chebyshev polynomials, (2024), arXiv preprint arXiv:2410.10897.
[39] S.A. Faroughi, N.M. Pawar, C. Fernandes, M. Raissi, S. Das, N.K. Kalantari, S. Kourosh Mahjour, Physics-guided, physics-informed, and physics-encoded neural networks and operators in scientific computing: fluid and solid mechanics, J. Comput. Inf. Sci. Eng. 24 (4) (2024) 040802.
[40] D.S. Ivanov, S.V. Lomov, Modeling of 2D and 3D woven composites, in: Polymer Composites in the Aerospace Industry, Elsevier, 2020, pp. 23–57.
[41] P. Couegnat, et al., Coupled multiscale analysis of woven composites: challenges and future directions, in: Proceedings of the 17Th International Conference on Composite Materials (ICCM), 2019. https://iccm-central.org/Proceedings/ICCM17proceedings/Themes/Materials/TEXTILE%20COMPOSITES/INT%20-%20TEXTILE%20COMPOSITES/ID11.6%20Couegnat.pdf.
[42] H. Wei, X. Huang, W. Xie, X. Jiang, G. Zhao, W. Zhang, Multiscale modeling for the impact behavior of 3D angle-interlock woven composites, Int. J. Mech. Sci. 276 (2024) 109382.
[43] M. Maia, I.B. Rocha, P. Kerfriden, F.P. van der Meer, Physically recurrent neural networks for path-dependent heterogeneous materials: embedding constitutive models in a data-driven surrogate, Comput. Methods Appl. Mech. Eng. 407 (2023) 115934.
[44] E. Ghane, B. Mohammadi, Entropy-damage mechanics for the failure investigation of plain weave fabric composites, Compos. Struct. 250 (2020) 112493.

[45] F. Taheri-Behrooz, H. Khayyam Rayeni, Strength prediction of woven composite rings using progressive damage analysis, J. Comput. Method. Eng. 37 (2) (2022) 17–39.
[46] A. Doitrand, C. Fagiano, F. Hild, V. Chiaruttini, A. Mavel, M. Hirsekorn, Mesoscale analysis of damage growth in woven composites, Compos. Part A: Appl. Sci. Manuf. 96 (2017) 77–88.
[47] G.-V. Săftoiu, C. Constantin, A.-I. Nicoară, G. Pelin, D. Ficai, A. Ficai, Glass fibre-reinforced composite materials used in the aeronautical transport sector: a critical circular economy point of view, Sustainability 16 (11) (2024) 4632.
[48] M.J. Buehler, MeLM, a generative pretrained language modeling framework that solves forward and inverse mechanics problems, J. Mech. Phys. Solids 181 (2023) 105454.
[49] J. Friemann, B. Dashtbozorg, M. Fagerström, S.M. Mirkhalaf, A micromechanics-based recurrent neural networks model for path-dependent cyclic deformation of short fiber composites, Int. J. Numer. Methods Eng. 124 (10) (2023) 2292–2314.
[50] J.C. Simo, T.J.R. Hughes, Computational inelasticity, 7, Springer Science & Business Media, 2006.
[51] K. Spilker, V.-D. Nguyen, L. Wu, L. Noels, Three-scale bridging for woven composites using homogenization techniques, Eur. J. Mech.-A/Solids 100 (2023) 104974.
[52] X. Han, C. Xu, W. Xie, S. Meng, Multiscale computational homogenization of woven composites from microscale to mesoscale using data-driven self-consistent clustering analysis, Compos. Struct. 220 (2019) 760–768.
[53] J. Bezanson, A. Edelman, S. Karpinski, V.B. Shah, Julia: a fresh approach to numerical computing, SIAM Rev. 59 (1) (2017) 65–98. https://doi.org/10.1137/141000671.
[54] N. Kovács, M.A. Maia, I.B. Rocha, C. Furtado, P.P. Camanho, F.P. van der Meer, Physically recurrent neural networks for computational homogenization of composite materials with microscale debonding, Eur. J. Mech.-A/Solids 112 (2025) 105668.