Hybrid machine learning based scale bridging framework for permeability prediction of fibrous structures

Denis Korolev^a, Tim Schmidt^b, Dinesh K. Natarajan^c, Stefano Cassola^b, David May^d, Miro Duhovic^b, Michael Hintermüller^{a,e}

^a Weierstrass-Institute for Applied Analysis and Stochastics, Mohrenstr.
 39, Berlin, 10117, Germany
 ^b Leibniz-Institut für Verbundwerkstoffe GmbH, Erwin-Schrödinger-Str.
 58, Kaiserslautern, 67663, Germany
 ^c German Research Center for Artificial Intelligence GmbH (DFKI), Trippstadter Str.
 122, Kaiserslautern, 67663, Germany
 ^d Faserinstitut Bremen e.V. (FIBRE) University of Bremen, Am Biologischen Garten 2, Geb. IW 3, Bremen, 28359, Germany
 ^e Institute for Mathematics, Humbolt-Universität zu Berlin, Unter den Linden 6, Berlin, 10099, Germany

Abstract

This study introduces a hybrid machine learning-based scale-bridging framework for predicting the permeability of fibrous textile structures. By addressing the computational challenges inherent to multiscale modeling, the proposed approach evaluates the efficiency and accuracy of different scale-bridging methodologies combining traditional surrogate models and even integrating physicsinformed neural networks (PINNs) with numerical solvers, enabling accurate permeability predictions across micro- and mesoscales. Four methodologies were evaluated: Single Scale Method (SSM), Simple Upscaling Method (SUM), Scale-Bridging Method (SBM), and Fully Resolved Model (FRM). SSM, the simplest method, neglects microscale permeability and exhibited permeability values deviating by up to 150% of the FRM model, which was taken as ground truth at an equivalent lower fiber volume content. SUM improved predictions by considering uniform microscale permeability, yielding closer values under similar conditions, but still lacked structural variability. The SBM method, incorporating segment-based microscale permeability assignments, showed significant enhancements, achieving almost equivalent values while maintaining computational efficiency and modeling runtimes of 45 minutes per simulation. In contrast, FRM, which provides the highest fidelity by fully resolving microscale and mesoscale geometries, required up to 270 times more computational time than SSM, with model files exceeding 300 GB. Additionally, a hybrid dual-scale solver incorporating PINNs has been developed and shows the potential to overcome generalization errors and the problem of data scarcity of the data-driven surrogate approaches. The hybrid framework advances permeability modelling by balancing computational cost and prediction reliability, laying the foundation for further applications in fibrous composite manufacturing.

Keywords: A. Fabrics/textiles, A. Tow, B. Permeability, C. Computational modelling, E. Resin flow

1. Introduction

The manufacturing of fiber-reinforced polymers (FRP) entails combining a fiber reinforcement structure with a matrix polymer, typically by infiltrating the fibers with a resin system driven by a pressure difference. Predicting resin flow within this structure requires considering multiple scales: on the microscale, flow between impermeable fibers is simulated; on the mesoscale, flow between permeable rovings becomes relevant; and on the macroscale, mold filling is considered [1]. Since simulating multiple scales simultaneously requires immense computing resources, they are typically treated separately. However, data must be exchanged between scales: structural information from meso- to microscale (referred to as downscaling) and, conversly, the assignment of microscale permeability to the permeable rovings at the mesoscale (referred to as upscaling), as shown in Fig.1. This combination of down- and upscaling, referred as scale bridging [2], requires the homogenization of data for the exchange between scales. In addition, scale separation and simulation lead to simplifications in the underlying physics [3]. These simplifications, however, are necessary to reduce the computational costs. Still, state-of-the-art permeability determination methods require multiple computationally expensive simulations, and assigning permeability to rovings at the higher scale is complex due to significant local variations in fiber structure, such as fiber orientation, which only increases the effort [4].

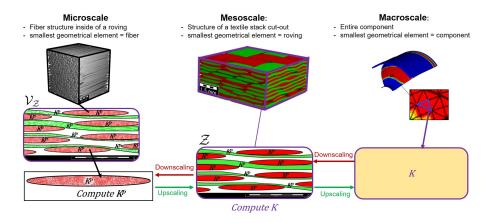


Figure 1: Relevant scale levels for permeability prediction.

Scientific machine learning (SciML) appears capable of aiding the scale bridging process, but its success depends on overcoming the problem that most of the surrogate modelling approaches struggle with: to effectively bridge scales while incorporating physical laws, data-driven insights from experimental observations, and scale relations into a unifying framework. In this work, Section 2 first presents the mathematical basis for upscaling and downscaling. In Sections 3 and 4, data-driven scale-bridging methods for permeability prediction are presented, although their accuracy is fundamentally constrained by generalization error, limited training data, and simplifications in flow models and homogenization techniques for data generation. To overcome these limitations, following the multiscale approach [5], Section 5 introduces a dual-scale solver that combines PINNs [6] for the microscale and a robust numerical solver for the mesoscale. The dual-scale solver integrates data from both scales, including available permeability prediction values, into a multi-fidelity framework to further improve the predictions through physics-informed regularization.

2. Upscaling and Downscaling

For the textile stack model $\Omega_{\mathrm{Me}} \subset \mathbb{R}^d$ and its microscopic representation $\Omega_{\mathrm{Mi}} \subset \mathbb{R}^d$, let $\mathcal{M}_{\mathrm{Mi}} = \{\mathcal{V} \mid \mathcal{V} \subseteq \Omega_{\mathrm{Mi}}\}$ and $\mathcal{M}_{\mathrm{Me}} = \{\mathcal{Z} \mid \mathcal{Z} \subseteq \Omega_{\mathrm{Me}}\}$ denote the sets of micro- and mesoscale geometry models, respectively. In mesomodels $\mathcal{Z} \in \mathcal{M}_{\mathrm{Me}}$, represented as mesoscale statistical volume elements (SVE) in the form of textile stack cut-outs, porous rovings are modeled as a continuum, and the SVEs are divided into fluid parts \mathcal{Z}^F and porous parts \mathcal{Z}^P , such that $\mathcal{Z} = \mathcal{Z}^P \cup \mathcal{Z}^F$. The resolution operator

$$\mathcal{F}_{\downarrow}: \mathcal{M}_{Me} \to \mathcal{M}_{Mi} \quad \text{with} \quad \mathcal{Z} \mapsto \mathcal{V}_{\mathcal{Z}} := \mathcal{F}_{\downarrow}(\mathcal{Z}),$$
 (1)

transfers mesomodels into their respective micromodels $\mathcal{V}_{\mathcal{Z}} \in \mathcal{M}_{\text{Mi}}$, corresponding to fully resolved mesoscale SVEs, as in Fig. 1. The micromodels are divided into fluid $\mathcal{V}_{\mathcal{Z}}^F$ and solid $\mathcal{V}_{\mathcal{Z}}^S$ parts. The microstructures $\mathcal{V}_{\mathcal{Z}}^S$ (bundles of impermeable fibers) are derived from the mesostructure of \mathcal{Z} . The data derived from \mathcal{Z} include the local orientation of the roving, which roughly corresponds to the fiber angle, and the fiber volume content (FVC) in the roving, which is calculated from the roving cross-sectional area, the number of fibers and the fiber diameter. With these features, suitable microscale SVEs as statistical approximations of resolved mesomodels can be generated as well, reducing simulation costs. This transfer of information from the mesoscale to the microscale is referred to as downscaling.

The flow of information from the micro- to the mesoscale is called **upscaling**. For its realization, we define the upscaling operator

$$\mathcal{F}_{\uparrow}: \ \mathcal{M}_{\mathrm{Mi}} \to \mathbb{R}^{d \times d} \quad \text{with} \quad \mathcal{V}_{\mathcal{Z}} \mapsto \mathbf{K}^{p}[\mathcal{V}_{\mathcal{Z}}],$$
 (2)

which extracts the geometric information about the microstructures of $\mathcal{V}_{\mathcal{Z}}$ in the form of the permeability tensor $\mathbf{K}^p[\mathcal{V}_{\mathcal{Z}}]$ (also referred to as micropermeability) to characterize the porous part \mathcal{Z}^P of \mathcal{Z} . Computing $\mathbf{K}^p[\mathcal{V}_{\mathcal{Z}}]$ involves solving the Darcy's law algebraic system:

$$U = -\frac{1}{\mu} K^p[\mathcal{V}_{\mathcal{Z}}] P D, \tag{3}$$

where μ is the dynamic viscosity of the fluid, $U \in \mathbb{R}^{d \times d}$ and $PD \in \mathbb{R}^{d \times d}$ are the volume-averaged flow velocity and pressure drop matrices with

$$\boldsymbol{U}_{k,j} = \frac{1}{|\mathcal{V}_{\mathcal{Z}}^F|} \int_{\mathcal{V}_{\mathcal{Z}}^F} u_j^{(k)}(x) \ dx, \quad \boldsymbol{P}\boldsymbol{D}_{k,j} = \frac{1}{|\mathcal{V}_{\mathcal{Z}}^F|} \int_{\mathcal{V}_{\mathcal{Z}}^F} \frac{\partial p^{(k)}}{\partial x_j}(x) \ dx. \tag{4}$$

The respective velocity fields $\mathbf{u}^{(k)} = [u_1^{(k)}, u_2^{(k)}, u_3^{(k)}]$ and pressures $p^{(k)}$ are obtained by solving three Stokes equations (for d = 3) at the microscale¹:

$$-\mu \Delta \boldsymbol{u}^{(k)} + \nabla p^{(k)} = \boldsymbol{f}^{(k)} \qquad \text{in } \mathcal{V}_{\mathcal{Z}}^{F},$$

$$\nabla \cdot \boldsymbol{u}^{(k)} = 0 \qquad \text{in } \mathcal{V}_{\mathcal{Z}}^{F},$$

$$\boldsymbol{u}^{(k)} = 0 \qquad \text{on } \partial \mathcal{V}_{\mathcal{Z}}^{S},$$
(5)

where $\partial \mathcal{V}_{\mathcal{Z}}^{S}$ denotes the (no-slip) boundary of $\mathcal{V}_{\mathcal{Z}}^{S}$ and $\boldsymbol{f}^{(k)}$ is a volume force. For example, one applies $\boldsymbol{f}^{(k)} = \boldsymbol{e}^{k}$, where \boldsymbol{e}^{k} is the unit vector in k-th direction, and uses periodic boundary conditions for $\boldsymbol{u}^{(k)}$ and $p^{(k)}$ to get such three solutions; cf. [7] and references therein. Solving (5) on a fully resolved model $\mathcal{V}_{\mathcal{Z}}$ of the mesogeometry \mathcal{Z} is computationally expensive or even prohibitive. Therefore, the SVE approximation of $\mathcal{V}_{\mathcal{Z}}$ is often used instead.

At the mesoscale, flow is modeled by the Stokes-Brinkman equation

$$-\tilde{\mu}\Delta \boldsymbol{u}^{SB(k)} + \mu \left(\boldsymbol{K}_{Me}[\mathcal{V}_{\mathcal{Z}}]\right)^{-1}\boldsymbol{u}^{SB(k)} + \nabla p^{SB(k)} = \boldsymbol{f}^{(k)} \quad \text{in } \mathcal{Z},$$

$$\nabla \cdot \boldsymbol{u}^{SB(k)} = 0 \quad \text{in } \mathcal{Z}.$$
(6)

where $\tilde{\mu}$ is the effective Brinkman viscosity, and $\mathbf{K}_{\mathrm{Me}}[\mathcal{V}_{\mathcal{Z}}] = \chi_{\mathcal{Z}^F} \cdot \infty + \chi_{\mathcal{Z}^P} \cdot \mathbf{K}^P[\mathcal{V}_{\mathcal{Z}}]$ ($\chi_{\mathcal{Z}^F}$ and $\chi_{\mathcal{Z}^P}$ are the indicator functions of \mathcal{Z}^F and \mathcal{Z}^P) represents the permeability properties of different domain regions. For $1 \leq k \leq d$, $\mathbf{f}^{(k)} = \mathbf{e}^k$ is applied, along with periodic boundary conditions for $\mathbf{u}^{SB(k)}$ and $p^{SB(k)}$, to obtain three solutions of (6). The macroscale permeability $\mathbf{K}[\mathcal{Z}]$ is then computed using (3) by averaging these solutions over \mathcal{Z} .

 $^{^{1}}$ The equation is often written in the dimensionless form with the Reynolds number.

3. Methods for dual scale permeability prediction

According to the state of the art, the micro- and mesoscale are modeled and simulated separately in order to numerically determine the dual-scale permeability of a textile stack [8, 9, 10]. For this, geometry models are required that represent the fiber structure. To ensure that the models account for all relevant structural properties, including realistic variations, studies are conducted in advance with the aim of determining SVEs for the fiber structure, see Fig. 3 [11]. In addition to the structural features, suitable model resolution, model size, and the number of models must be identified during SVE development to achieve a distribution of permeabilities similar to the experiments [12].



Figure 2: Visualisation of the statistical representative volume elements for micro- and mesoscale and the fully resolved model

In our numerical realization, 3D micro- and mesoscale models were generated in the GeoDict[®] software and are voxel-based, with voxels (equilateral hexahedra akin to image pixels) classified as solid, fluid, or porous. The microscale SVE contains several hundred fibers [13], and the mesoscale SVE contains several textile layers, which are modeled using rovings, see Fig. 2. The rovings are taken to represent fiber bundles as a continuum, which allows the use of a coarser resolution. To create the model, individual textile layers with varying roving cross-sections are generated and positioned randomly on top of each other within a defined range, followed by a virtual compaction of the textile stack until the desired stack height or FVC is achieved. This causes the rovings to deform

and the textile layers to nest into each other, as in reality. To account for the flow within the rovings, the anisotropic micropermeability must be assigned. In GeoDict, each voxel has a material ID that allows properties such as anisotropic permeability to be defined. All methods are shown in Fig. 3. For the single scale method (SSM) and the simple upscaling method (SUM), 10 models for each of five compaction levels with a voxel resolution of $6^3 \mu m^3$ and six textile layers were generated. For the scale bridging method (SBM), 12 models were generated for each of the four compaction levels using the same modelling parameters and model sizes as for SSM and SUM. Due to the tremendous effort involved, one multifilament model (FRM) for each of the three compaction levels was generated and numerically calculated. In principle, it applies to all models that the model parameters are target values for modelling and are not always met exactly due to randomization. An overview of the resulting models for the individual methods can be found in Fig. 2. All numerical flow simulations were carried out in the GeoDict module FlowDict. For the microscale, the FRM, and the mesoscale models (without considering micropermeability), the LIR solver was used for (5), and for SUM and SBM, the SimpleFFT solver was used for (6) [14]. Periodic boundary conditions were applied in tangential direction and in flow direction, with additional empty inflow and outflow regions of 40 voxels. An error bound of 0.01, corresponding to a change in permeability of less than 1\%, was chosen as the stopping criterion [14].

3.1. Single Scale Method - SSM

Our dual-scale methods are compared with the SSM, which neglects micropermeability, to evaluate whether the increased complexity of dual-scale methods is justified.

3.2. Simple Upscaling Method - SUM

For the SUM, all warp and weft rovings have the same material ID and, therefore, the same permeability tensor with identical orientation. In the roving direction, assumed to align with the respective model axis, the rovings are

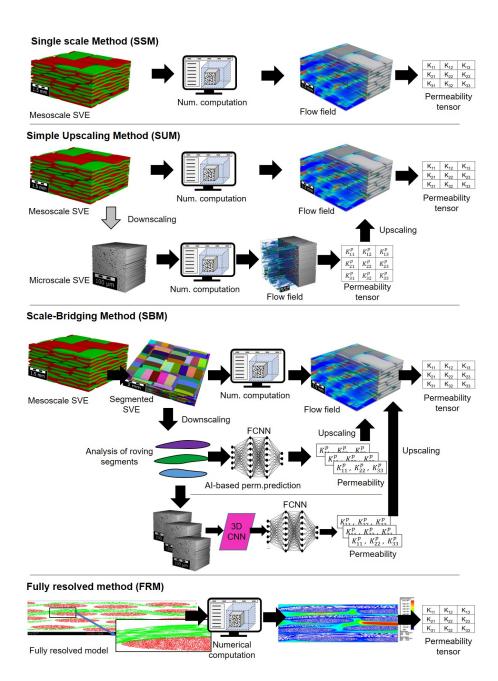


Figure 3: Schematic illustrations of the various single and dual scale methods

assigned the fiber-direction micropermeability. Perpendicular directions are assigned the transverse micropermeability. It implies that local variations within the roving structure, such as those induced by the weave structure or deformations, are neglected. However, this approach represents an improvement over the SSM, which completely neglects the micropermeability.

3.3. Scale-Bridging Method - SBM

The SBM approach was developed to reduce the inaccuracies of the SUM approach. In this method, the rovings are decomposed into several segments, see Fig. 3, each with individual material IDs before compaction. Downscaling is performed for each segment: the FVC and orientation of the roving segment, which approximately correspond to the fiber orientation within the roving, are determined. However, this method also represents a simplification of the local structure, as homogeneous properties are assumed for each segment. Therefore, the segment size affects the modelling accuracy. Based on the structural parameters, FVC, fiber orientation, and fiber diameter, the micropermeability is determined using the data-driven methods described in Section 3.3.1 and subsequently assigned to the material ID of the corresponding roving segment. This method can be used to assign up to 255 different anisotropic permeability values (due to the limited number of material IDs in GeoDict), resulting in more accurate micropermeabilities in the mesomodel compared to the SSM and SUM.

3.3.1. Feature-based and geometry-based emulator for permeability prediction

Several recent studies have proposed ML methods as surrogate models (or) emulators for the permeability prediction of microstructures, i.e., on a single scale. Based on the input features used for prediction, these methods can be categorized into two classes: (1) feature-based methods and (2) geometry-based method, as shown in Fig. 4. Various modelling approaches for micropermeability prediction have been investigated in [13, 15, 16] and references therein.

The main advantage of such ML emulators is their significant speed-up in inference times compared to numerical simulations, albeit with a trade-off in accuracy. [15] showed that feature-based emulators achieved an inference speed-up of 10^6 with a relative error of 11.35%, whereas geometry-based emulators achieved an inference speed-up of 10^4 with a relative error of 8.33%. Depending on the available features describing the microstructure, either feature-based or geometry-based emulators can be employed for micropermeability prediction.

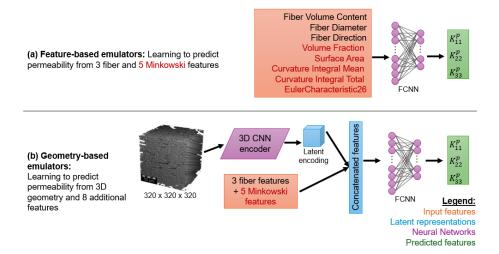


Figure 4: Two classes of modeling approaches for the ML emulators for permeability prediction. The approaches differ based on the input features used to predict the permeabilities. The emulators are constructed using fully connected neural networks and 3D convolutional neural networks. The architecture of the 3D CNN encoder is based on [17].

3.4. Fully resolved Model - FRM

The method using FRM is theoretically the most precise compared to SSM, SUM, and SBM. Scales are not separated, and no microscale homogenization takes place. A sufficiently small voxel size must be chosen for the entire model to resolve the flow between the fibers ($1^3 \ \mu \text{m}^3/\text{voxel}$). This results in large geometry models and flow simulations, which can only be computed on high-performance clusters. The model generation was comparable to the modeling of SUM and SBM, with the only difference being that multifilament rovings (3000 fibers with a diameter of 7 μ m) were created. However, using an appropriate resolution ($0.5^3 \ \mu \text{m}^3/\text{voxel}$) for accurate microscale representation at a

mesoscale SVE size of 6 layers with 3x3 rovings per layer exceeded 2 TB of RAM during the flow simulatio, which was the maximum available. As a compromise between size and resolution, models were created with a size of 5.100 x 5.100 x 1.120 voxels (4 layers with 2.4 x 2.4 rovings per layer at 10% compaction) and a voxel size of $1^3 \mu m^3$. Result files totaling approx. 310 GB were generated for each model. This fully resolved high-fidelity model offers the fewest simplifications and can therefore be considered the reference. However, due to the slightly reduced size and resolution, the results must be critically scrutinized. Nevertheless, the method highlights the need for advanced and efficient scale-bridging methods due to the vast computational cost of fully resolved models.

4. Comparison of dual scale methods for permeability prediction

The results of the previously presented approaches are shown in Table 1, and Fig. 5 and 6. The average run time of the presented methods increases from SSM over SUM and SBM to FRM by more than two orders of magnitude. This shows the high effort required for the FRM and the advantage of dual scale approaches. No significant difference between SUM and SBM can be determined, as the run times vary too much from model to model. The effort for modelling, structural analysis and micropermeability prediction for SBM takes approximately 45 min which is about twice as long as the modelling for SUM and SSM. Except for SSM, the run time of the flow simulation is multiple times longer in comparison to the model creation run time. The run time for SSM is much faster because of the significantly faster solver (LIR solver) that can be used. But the LIR solver is not suitable for SUM and SBM, as it is not applicable for porous media with anisotropic permeability. Despite the use of the LIR solver, the run time for the FRM is extremely high (approx. factor 270 compared to SSM and 23 compared to SUM/SBM).

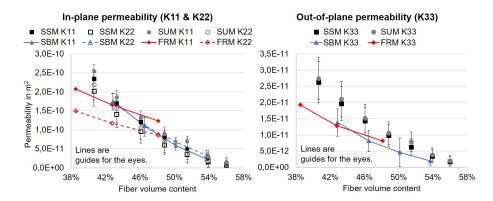


Figure 5: Diagrams of the in-plane and out-of-plane permeability results plotted over FVC of SSM, SUM, SBM and FRM

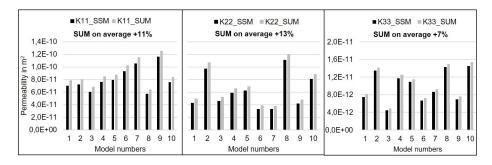


Figure 6: Comparison of the permeability results (K11, K22, K33) from SSM and SUM for ten models at 15 % compaction

Comparing the results of SSM and SUM, which were both carried out with the same models, shows that the usage of the SUM yields consistently higher permeability values for all models as can be seen in Fig. 6. This diagram shows the comparison of all SSM and SUM models at 15% compaction. The differences between the permeability determined using SSM and SUM vary between 3% and 92% for the different models. The difference tends to increase with increasing FVC from 7% on average at 40% FVC to 25% on average at 56% FVC. This trend can be explained by the increasing influence of the micropermeability with increasing FVC since meso flow channels are reduced with higher compaction of the textile stack. While the coefficient of variation (CV) for SSM and SUM

7 1 1 1 1 1		0	$_{ m SVP}$	FVC	Model size in voxel	oxel		K11 in m ²		$_{ m K22~in~m}^2$		$K33 \text{ in m}^2$		Run time in h
Method	Compaction	voxet size in μin	MV	MV	X-axis (MV)	Y-axis (MV)	Z-axis (MV)	MV	CV	MV	CV	MV	CV	MV
	%0	9	67,3%	40,6%	1109	1094	305	2,34E-10	%8	2,02E-10	20%	2,63E-11	24%	0,39
	22%	9	71,0%	43,3%	1109	1094	289	1,70E-10	11%	1,41E-10	25%	1,97E-11	27%	0,39
	10%	9	74,9%	46,1%	1109	1094	274	1,20E-10	15%	9,56E-11	32%	1,43E-11	30%	0,41
SSM	15%	9	78,8%	48,8%	1109	1094	260	8,07E-11	22%	6,07E-11	43%	9,92E-12	34%	0,44
	20%	9	82,5%	51,5%	1109	1094	246	4,87E-11	36%	3,45E-11	62%	6,24E-12	43%	0,72
	25%	9	85,9%	53,9%	1109	1094	233	2,54E-11	61%	1,64E-11	95%	3,48E-12	21%	1,00
	30%	9	88,8%	26,0%	1109	1094	221	1,16E-11	106%	6,90E-12	145%	1,81E-12	81%	1,10
	%0	9	67,3%	40,6%	1109	1094	305	2,53E-10	2%	2,19E-10	18%	2,74E-11	23%	11,81
	22%	9	71,0%	43,3%	1109	1094	289	1,86E-10	10%	1,57E-10	24%	2,11E-11	26%	11,94
	10%	9	74,9%	46,1%	1109	1094	274	1,32E-10	14%	1,06E-10	30%	1,52E-11	29%	9,00
SUM	15%	9	78,8%	48,8%	1109	1094	260	8,91E-11	21%	6,76E-11	40%	1,06E-11	33%	7,55
	20%	9	82,5%	51,5%	1109	1094	246	6,96E-11	20%	4,96E-11	26%	7,91E-12	38%	6,60
	25%	9	85,9%	53,9%	1109	1094	233	2,93E-11	22%	1,92E-11	%68	3,86E-12	54%	3,96
	30%	9	88,8%	26,0%	1109	1094	221	1,38E-11	%86	8,46E-12	133%	2,05E-12	77%	0,00
	SVP 70%	9	70,4%	42,8%	1168	1144	286	1,76E-10	11%	1,74E-10	12%	1,39E-11	30%	9,41
, and a	SVP 75%	9	75,5%	46,5%	1145	1171	268	1,10E-10	24%	1,12E-10	19%	8,19E-12	40%	7,89
MGC	SVP 80%	9	80,6%	50,1%	1180	1144	249	5,86E-11	21%	6,84E-11	17%	4,68E-12	95%	6,29
	SVP 85%	9	85,6%	53,7%	1195	1183	230	2,34E-11	65%	3,29E-11	38%	2,00E-12	78%	4,87
	10%	1	38,5%	38,5%	5100	5100	1119	2,08E-10		1,50E-10		1,93E-11		148,91
FRM	20%	1	42,8%	42,8%	5100	5100	1007	1,66E-10		1,18E-10		1,28E-11		89,76
	30%	1	48,1%	48,1%	5100	5100	895	1,24E-10		8,56E-11		8,20E-12		114,18

Table 1: Results, model size and run time of SSM, SUM, SBM and FRM $\,$

is relatively high, the CV of SUM (42.7 %) is on average 3.7 lower compared to the CV of SSM (46.4 %). These differences results from the randomized model creation and thus variation in the structure, which leads to significantly different permeability values. The CV of the SBM (37.6 %) is on average 5.1 percentage points lower compared to the CV of SUM. Fig. 6 shows exemplarily the wide range of results that can be achieved with models created based on the same modelling parameters. The SBM and FRM approaches provide comparable values for the out-of-plane direction (K33), while considerably higher K33 values are obtained from SSM and SUM. However, the FRM seems to be less sensitive to an increasing FVC compared to the other methods. Compared to the SBM this results in similar K33 values at a median FVC of 42.8 % but higher values at 48.1 %. Similarly the K11 in-plane permeability value around 38.5 % for all methods is comparable but values diverge when changing the FVC. Generally, in-plane permeabilities obtained from SSM, SUM and SBM are comparable in value. The methods presented clearly show the trade off between modeling complexity with regards to physical and geometric accuracy and computational efficiency of dual-scale structures.

5. Hybrid physics-informed dual scale framework for upscaling

In Section 3, ML-based surrogate models were used to replace the standard numerical methods for permeability determination. In addition to the data-driven methods, we consider rewriting the microscale PDE solver as a PINN optimization problem, as it offers several advantages. Unlike standard numerical methods, PINNs are meshless and can incorporate experimental data and cross-scale information via optimization, enabling the inference of unknown physics and multiscale modelling [18]. They leverage transfer learning to reuse prior simulations [19, 20] and support efficient distributed GPU implementation due to excellent model and data parallelization [21, 22], making them potentially suitable for demanding SciML tasks as upscaling. However, the nonlinear and non-convex nature of PINN optimization makes it sensitive to the network's

(hyper-)parameter initialization and prone to local minima, with additional challenges stemming from the PDE structure. These issues make the application of PINNs to complex 3D (fibrous) geometries quite challenging. Thus, we focus here on 2D periodic microscale geometries with periodic fiber arrangements only.

Let $\Phi(\cdot;\Theta)$ denote a permeability prediction network with parameters $\Theta \in \mathbb{R}^s$, trained, for example, using the methods from Section 3. To enhance $\Phi(\cdot;\Theta)$ using a hybrid approach, we define the *learning-informed* upscaling operator:

$$\widehat{\mathcal{F}_{\uparrow}}(\cdot;\theta,\Theta):\mathcal{M}_{\mathrm{Mi}}\to\mathbb{R}^{d\times d},\ \mathcal{V}_{\mathcal{Z}}\mapsto\widehat{\boldsymbol{K}^{p}}[\mathcal{V}_{\mathcal{Z}}](\theta,\Theta)=:\widehat{\mathcal{F}_{\uparrow}}(\mathcal{V}_{\mathcal{Z}};\theta,\Theta). \tag{7}$$

We evaluate (7) by generating a guess for $\widehat{K^p}[\mathcal{V}_{\mathcal{Z}}](\theta,\Theta)$ using $\Phi(\cdot;\Theta)$, refined by our hybrid approach, which adds dependence on the PINN parameters $\theta \in \mathbb{R}^{n_p}$. To achieve this, we introduce a special coarse-scale prior that bridges two scales and uses the permeability guess to effectively inform PINNs, which approximate the microscale problems. This coupling, combined with the physics-informed regularization, enhances the reliability of the permeability prediction process.

For periodic fiber arrangements in $\mathcal{V}_{\mathcal{Z}} \subset \mathbb{R}^2$, $\mathbf{K}^p[\mathcal{V}_{\mathcal{Z}}]$ satisfies $\mathbf{K}^p[\mathcal{V}_{\mathcal{Z}}] = \mathbf{K}_{11}^p[\mathcal{V}_{\mathcal{Z}}]\mathcal{I}$, where $\mathcal{I} \in \mathbb{R}^{2\times 2}$ is the identity matrix and $\mathbf{K}_{11}^p[\mathcal{V}_{\mathcal{Z}}]$ is given by:

$$\boldsymbol{K}_{11}^{p}[\mathcal{V}_{\mathcal{Z}}] = -\frac{1}{\mu} \boldsymbol{P} \boldsymbol{D}^{-1} \boldsymbol{U}, \tag{8}$$

where $U := U_{11}$ and $PD := PD_{11}$ according to (4). Therefore, one problem (5) is solved for $u := u^{(1)}$, $p := p^{(1)}$. For this, we use the boundary conditions:

$$\boldsymbol{u} = (1,0)^T$$
 on $\Gamma_{\text{in}} \cup \Gamma_{\text{wall}}$, $\frac{\partial \boldsymbol{u}}{\partial \eta} = 0$ on Γ_{out} , $p = 0$ on Γ_{out} , (9)

where $\Gamma_{\rm in}$ and $\Gamma_{\rm out}$ are the inlet (left) and outlet (right) sides of $\mathcal{V}_{\mathcal{Z}}$, and $\frac{\partial \boldsymbol{u}}{\partial \eta}$ is the directional derivative of \boldsymbol{u} in the outward normal direction η . With this 2D setup, we further present the PINN ansatz, followed by the hybrid one.

5.1. Physics-informed neural networks as microscale surrogates

Following [6], we aim to find a neural network \widehat{NN}_{θ} that learns the map from spatial coordinates to an approximation of the solution to (5):

$$\mathbf{x} \mapsto \widehat{NN}_{\theta}(\mathbf{x}) := [\widehat{u}_{\theta,1}(\mathbf{x}), \ \widehat{u}_{\theta,2}(\mathbf{x}), \ \widehat{p}_{\theta}(\mathbf{x})].$$
 (10)

The formulation (10) is then used to compute $\widehat{K}_{11}^p[\mathcal{V}_{\mathcal{Z}}](\theta) \approx K_{11}^p[\mathcal{V}_{\mathcal{Z}}]$ from (8). The parameters θ are determined by minimizing the PINN objective:

$$\mathcal{J}(\theta) := \sum_{i=1}^{2} \lambda_{i}^{r} \mathcal{J}_{i}^{r}(\theta) + \lambda_{i}^{b} \mathcal{J}_{i}^{b}(\theta) + \lambda_{i}^{\text{out}} \mathcal{J}_{i}^{\text{out}}(\theta) + \lambda^{\text{div}} \mathcal{J}^{\text{div}}(\theta), \tag{11}$$

where $\lambda_i^r,~\lambda_i^b,~\lambda_i^{\text{out}},\lambda^{\text{div}}$ are positive weights for the penalty terms

$$\mathcal{J}_{i}^{r}(\theta) = \frac{1}{N_{r}} \sum_{j=1}^{N_{r}} \left| \frac{\partial \widehat{p}_{\theta}}{\partial x_{i}}(\boldsymbol{x}_{j}^{r}) - \mu \Delta \widehat{u}_{\theta,i}(\boldsymbol{x}_{j}^{r}) \right|^{2}, \quad \mathcal{J}^{\text{div}}(\theta) = \frac{1}{N_{r}} \sum_{j=1}^{N_{r}} \left| \nabla \cdot \widehat{u}_{\theta}(\boldsymbol{x}_{j}^{r}) \right|^{2}, \\
\mathcal{J}_{i}^{b}(\theta) = \frac{1}{N_{b}} \sum_{j=1}^{N_{b}} \left| \widehat{u}_{\theta,i}(\boldsymbol{x}_{j}^{b}) \right|^{2}, \quad \mathcal{J}_{i}^{\text{out}}(\theta) = \frac{1}{N_{\text{out}}} \sum_{j=1}^{N_{\text{out}}} \left| \frac{\partial \widehat{u}_{\theta,i}}{\partial x_{1}}(\boldsymbol{x}_{j}^{\text{out}}) \right|^{2}. \tag{12}$$

The collocation points $\{\boldsymbol{x}_{j}^{r}\}_{j=1}^{N_{r}} \subset \mathcal{V}_{Z}^{F}, \{\boldsymbol{x}_{j}^{b}\}_{j=1}^{N_{b}} \subset \partial \mathcal{V}_{Z}^{S}, \{\boldsymbol{x}_{j}^{\text{out}}\}_{j=1}^{N_{\text{out}}} \subset \Gamma_{\text{out}}$ are sampled uniformly at random in our case, and the spatial derivatives in (12) are computed using automatic differentiation, as it is usual for PINNs.

The adaptive modified Fourier feature network is employed for neural networkbased approximation; cf. [23]. The network follows the ansatz:

$$f_{NN}(\boldsymbol{x};\theta) = \boldsymbol{W}^{L_f}(\phi_{L_f-1} \circ \phi_{L_f-2}... \circ \phi_{L_1} \circ \phi_E)(\boldsymbol{x}) + \boldsymbol{b}^{L_f}.$$
(13)

For $1 \le l \le L_f - 1$ in (13), the *l*-th hidden layer is defined as:

$$\phi_{L_l}(\mathbf{h}^{l-1}) = \mathbf{h}^l := [1 - \sigma(\mathbf{W}^l \mathbf{h}^{l-1} + \mathbf{b}^l)] \odot \mathbf{T}_1 + \sigma(\mathbf{W}^l \mathbf{h}^{l-1} + \mathbf{b}^l) \odot \mathbf{T}_2,$$

$$\mathbf{T}_1 = \mathbf{W}_T^1 \phi_E(\mathbf{x}) + \mathbf{b}_T^1, \quad \mathbf{T}_2 = \mathbf{W}_T^2 \phi_E(\mathbf{x}) + \mathbf{b}_T^2.$$

where \odot is element-wise multiplication, $\boldsymbol{W}_{T}^{1}, \boldsymbol{W}_{T}^{2} \in \mathbb{R}^{n_{1} \times d_{E}}, \ \boldsymbol{W}^{l} \in \mathbb{R}^{n_{l} \times n_{l-1}}$ and $\boldsymbol{b}_{T}^{1}, \boldsymbol{b}_{T}^{2} \in \mathbb{R}^{n_{l}}, \ \boldsymbol{b}^{l} \in \mathbb{R}^{n_{l}}$ are trainable weights and biases, $\sigma : \mathbb{R}^{n_{l}} \to \mathbb{R}^{n_{l}}$ are activation functions, and the hidden states are $\boldsymbol{h}^{l} \in \mathbb{R}^{n_{l}}$, with $\boldsymbol{h}^{0} := \phi_{E}(\boldsymbol{x}) = [\sin(2\pi\boldsymbol{E}\boldsymbol{x}), \cos(2\pi\boldsymbol{E}\boldsymbol{x})] \in \mathbb{R}^{d_{E} \times d}$. Choosing large d_{E} in the embeddings ϕ_{E} helps mitigate the spectral bias [24] toward learning low-frequencies by better capturing high-frequencies of the solution [25, 26]. Trainable $\boldsymbol{E} \in \mathbb{R}^{\frac{d_{E}}{2} \times d}$ is useful as the spectral properties of (5) are not known a priori. The Dirichlet boundary condition \boldsymbol{u}_{D} on $\Gamma_{\text{in}} \cup \Gamma_{\text{wall}}$ and p = 0 on Γ_{out} are imposed penalty-free by modifying (13) as follows: $[\widehat{NN}_{\theta}(\boldsymbol{x})]_{i} = g_{i}(\boldsymbol{x}) + s_{i}(\boldsymbol{x})[f_{NN}(\boldsymbol{x};\theta)]_{i}$, where $g_{1}(\boldsymbol{x}) = 1, \ g_{2}(\boldsymbol{x}) = 0$ and $s_{i}(\boldsymbol{x}) = 4x_{1}x_{2}(1 - x_{2})$ for $\widehat{u}_{\theta,1}$ and $\widehat{u}_{\theta,2}$, and $g_{3} = 0$, $s_{3} = 1 - \exp(10x_{1} - 10)$ for \widehat{p}_{θ} .

5.2. Hybrid physics-informed neural network based dual scale solver

For multiscale PDEs, incorporating a prior into the PINN objective (11) that well-informs the coarse-scale component of the microscale PDE solution can improve neural network training [5]. To this end, we adopt a mesoscale model governed by the learning-informed Stokes-Brinkman equation:

$$-\tilde{\mu}\Delta \boldsymbol{u}_{\theta}^{SB} + \mu \left(\widehat{\boldsymbol{K}}_{Me}[\mathcal{V}_{\mathcal{Z}}](\theta, \Theta)\right)^{-1} \boldsymbol{u}_{\theta}^{SB} + \nabla p_{\theta}^{SB} = \boldsymbol{f} \quad \text{in } \mathcal{Z},$$

$$\nabla \cdot \boldsymbol{u}_{\theta}^{SB} = 0 \quad \text{in } \mathcal{Z},$$
(14)

where $\widehat{\boldsymbol{K}}_{\text{Me}}[\mathcal{V}_{\mathcal{Z}}](\theta,\Theta) = \chi_{\mathcal{Z}^F} \cdot \infty + \chi_{\mathcal{Z}^P} \cdot \widehat{\boldsymbol{K}_{11}^p}[\mathcal{V}_{\mathcal{Z}}](\theta,\Theta)$ and $\widehat{\boldsymbol{K}_{11}^p}[\mathcal{V}_{\mathcal{Z}}](\theta,\Theta)$ is computed via (8) using the PINN ansatz. We define the coarse-scale prior

$$\mathcal{R}(\theta; \theta_s) := \sum_{i=1}^{2} \lambda_i^{SB} \mathcal{R}_i(\theta; \Theta), \tag{15}$$

where $\mathcal{R}_i(\theta;\Theta) = \frac{1}{N_{SB}} \sum_{j=1}^{N_{SB}} |\widehat{u}_{\theta,i}(\boldsymbol{x}_j^c) - u_{\theta,i}^{SB}(\boldsymbol{x}_j^c;\Theta)|^2$ with $\{\boldsymbol{x}_j^c\}_{j=1}^{N_{SB}} \subset \mathcal{Z}_F$, and $\lambda_i^{SB} > 0$ are the prior weights. Using (11) with (15), our hybrid problem reads:

$$\min_{\theta \in \mathbb{R}^{n_p}} J_{\lambda}(\theta; \Theta) := \mathcal{J}(\theta) + \mathcal{R}(\theta; \Theta) \quad \text{subject to the constraints (14)}.$$
 (16)

Evaluating J_{λ} (where the subscript λ marks the dependence on the loss weights) requires (numerically) solving (14), resulting in a PDE-constrained optimization problem. Since coarse-scale geometries are easier to mesh in general, we apply a standard numerical solver at the mesoscale. Enabling (14) with the boundary conditions (9) and using Taylor-Hood finite elements [27], we approximate $\boldsymbol{u}_{\theta}^{SB}$ and p_{θ}^{SB} with $\boldsymbol{u}_{h,\theta}^{SB}$ and $p_{h,\theta}^{SB}$ (the subscript h>0 denotes mesh resolution), forming the dual-scale solver shown in Fig. 7, realized through optimization.

The hybrid objective (16) is optimized using a variant of the gradient descent algorithm with the following generic form

$$\theta_{k+1} = \theta_k - l_k(\nabla_{\theta} \mathcal{J}(\theta_k) + \nabla_{\theta} \mathcal{R}(\theta_k; \Theta)), \tag{17}$$

where $l_k > 0$ is the learning rate. The discrete gradient flow (17) is known for its stiffness due to the variation in the magnitudes of the involved gradients, which causes convergence issues [28]. Indeed, the histograms of back-propagated gradients for two momentum equations and two prior terms, shown in Fig. 8(a), reveal

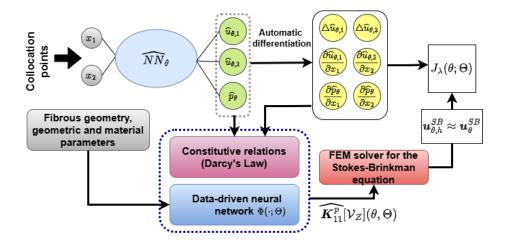


Figure 7: The schematic workflow for the hybrid neural network based dual scale solver.

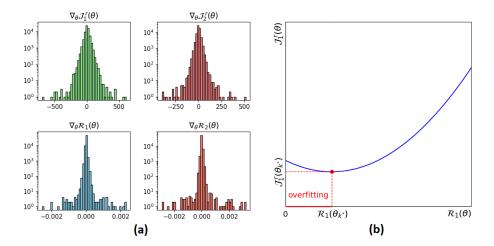


Figure 8: (a) Histogram of back-propagated gradients for the momentum equations (top) and prior terms (bottom). (b) Schematic minimization of the physics-informed term $\mathcal{J}_1(\theta)$ and the coarse-scale prior $\mathcal{R}_1(\theta;\Theta)$: overfitting $\mathcal{R}_1(\theta;\Theta)$ perturbs $\mathcal{J}_1(\theta)$.

that prior gradients are concentrated near zero, while momentum gradients have much wider ranges. No-slip penalty gradients in (12) also similarly concentrate near zero, hence stiffness increases with the number of fibers. To help mitigate this, we scale all the gradients using weights λ and following [23, Algorithm 1(c)]. Besides, overfitting discrepancies between the micro- and meso-models in (15) distort the PDE residuals in (12), hindering convergence at both scales. Poor scale bridging further exacerbates it, increasing the red gaps in Fig. 8(b) by causing greater model mismatches. To partially mitigate it, weights in (15) are updated via $\lambda_{i,k+1}^{SB} = \lambda_{i,k}^{SB} \exp(-\alpha_i(k-k^*))$, where $\alpha_i > 0$ are hyperparameters and k^* marks the iteration, where the coarse component of (5) is rather wellapproximated. By default, we set $k^* = k_{\text{max}}/2$ and $\alpha_i = 5 \times 10^{-4}$. For $k < k^*$, [23, Algorithm 1(c)] is applied. To update (15), we numerically solve (14) every T iterations. The hybrid optimization, outlined in Algorithm 1, requires initializing θ_0 , typically using Glorot initialization [29] or transfer learning, where θ_0 is set to θ from a similar problem [19, 20]. Algorithm 1 also requires $\sigma_{\Phi} > 0$ as a deviation constraint on the desired permeability to enhance training robustness; σ_{Φ} is often available from experimental and theoretical estimates.

We set $\mathcal{Z}_i = (0,1)^2$ with $\mathcal{Z}_{P_i} = (0.25,0.75)^2$ $(1 \leq i \leq 2)$ as the mesoscale models. For downscaling, we formally specify the number of fibers for $\mathcal{V}_1 := \mathcal{F}_{\downarrow}(\mathcal{Z}_1)$ (16 fibers) and $\mathcal{V}_2 := \mathcal{F}_{\downarrow}(\mathcal{Z}_2)$ (25 fibers) to resolve the square rovings \mathcal{Z}_{P_1} and \mathcal{Z}_{P_2} , see Fig. 9. The fiber obstacles correspond to domain perforations with no-slip boundary conditions. We set Algorithm 1 as follows. The architecture from Section 5.1 (with $d_E = 100$, 4 hidden layers, each with 100 neurons) and Adam optimizer [30] are used for hybrid and PINN approaches. For both approaches, we use a full batch of collocation points and an exponential learning rate schedule. The initial learning rate is set to $l_0 = 1 \times 10^{-3}$, with a decay rate of 0.9. A subset of finite element mesh points $\{\boldsymbol{x}_j^c\}_{j=1}^{N_{SB}} \subset \mathcal{Z}_F$ from discretization of (14) is used in (15). We use T = 250 to reduce computational costs. Using finite elements with a very fine mesh, we computed permeabilities $\boldsymbol{K}^p = 1.4 \times 10^{-3}$ for \mathcal{V}_1 and $\boldsymbol{K}^p = 6.4 \times 10^{-4}$ for \mathcal{V}_2 as our references. We set $\widetilde{\boldsymbol{K}}_{11} = 9 \times 10^{-4}$ as our initial guess for both geometries, with $\sigma_{\Phi} = 0.5 \times \boldsymbol{K}^p$. We note that

Algorithm 1 Hybrid dual scale solver

```
Input: \widetilde{K}_{11}[\mathcal{V}_{\mathcal{Z}}](\Theta) := \Phi(\mathcal{V}_{\mathcal{Z}};\Theta), \ \theta_0, \ \sigma_{\Phi}, \text{ maximal iteration number } k_{\max}.
Output: \theta and \widehat{K_{11}^p}[\mathcal{V}_{\mathcal{Z}}](\theta;\Theta).
  1: while 0 \le k \le k_{\text{max}} - 1 do
             Compute \nabla_{\theta} J_{\lambda_k}(\theta_k) using automatic differentiation
             if k \mod T = 0 then
  3:
                 Update: \lambda_k \leftarrow \text{Weights scaling}(\nabla_{\theta} J_{\lambda_k}(\theta_k), \lambda_k).
  4:
                 \text{if } \widetilde{\boldsymbol{K}_{11}^{p}}[\mathcal{V}_{\mathcal{Z}}](\Theta) - \sigma_{\Phi} < \widehat{\boldsymbol{K}_{11}^{p}}[\mathcal{V}_{\mathcal{Z}}](\theta_{k};\Theta) < \widetilde{\boldsymbol{K}_{11}^{p}}[\mathcal{V}_{\mathcal{Z}}](\Theta) + \sigma_{\Phi} \text{ then }
  5:
                      Update: \widehat{\boldsymbol{K}_{11}^p}[\mathcal{V}_{\mathcal{Z}}](\theta_k) \leftarrow \text{Darcy's Law}(\widehat{NN}_{\theta_k}).
  6:
                      Compute FEM solution to (14) and update the prior (15).
  7:
                 end if
  8:
             end if
  9:
             Update: \theta_{k+1} \leftarrow \text{Optimizer}(\nabla_{\theta} J_{\lambda_k}(\theta_k), \text{ optimizer hyperparameters})
10:
             k \leftarrow k+1
11:
12: end while
```

the boundary conditions in (9), which induce a strong boundary layer, are not well-suited for permeability computation, resulting in suboptimal scale bridging. Therefore, we treat the Stokes-Brinkman viscosity $\tilde{\mu}$ in (14) as a free parameter to improve scale bridging. For \mathcal{V}_1 , $\tilde{\mu}=0.175$ (upscaling-consistent case) achieves a good match between flows at the microscale and mesoscale, whereas $\tilde{\mu}=0.45$ leads to significant discrepancies; see (a) and (b) in Fig. 9. For \mathcal{V}_2 , we use $\tilde{\mu}=0.25$ to achieve good flow matching; see Fig. 9(c).

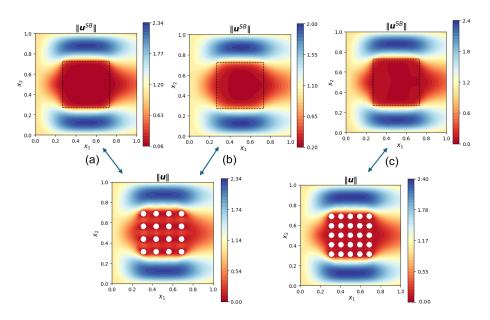


Figure 9: Geometries and the respective velocity magnitudes at the microscale and mesoscale (computed with FEM) for (a): V_1 , $\tilde{\mu} = 0.175$, (b): V_1 , $\tilde{\mu} = 0.45$, (c): V_2 , $\tilde{\mu} = 0.25$. Here, $\tilde{\mu}$ stands for the Stokes-Brinkman viscosity.

Method	Relative l^2 error (u_1)	Relative l^2 error (u_2)	Relative l^2 error (p)
Hybrid $_{\mathcal{V}_1}^G$	6.78E - $03 \pm 1.50\text{E}$ - 03	$1.95\text{E-}02 \pm 2.11\text{E-}03$	$6.80\text{E-}03 \pm 2.51\text{E-}03$
$\operatorname{PINN}_{\mathcal{V}_1}^G$	$1.13\text{E-}02\pm9.18\text{E-}04$	$6.20\text{E-}02\pm5.61\text{E-}03$	$8.06\text{E-}03\pm9.91\text{E-}04$
$\mathrm{Hybrid}_{\mathcal{V}_2}^G$	$9.00\text{E-}03 \pm 1.14\text{E-}03$	$2.78\text{E-}02\pm5.07\text{E-}03$	$4.10\text{E-}02\pm9.02\text{E-}03$
$\mathrm{Hybrid}_{\mathcal{V}_2}^{\mathrm{TL}}$	$7.64\text{E-}03\pm5.13\text{E-}04$	$2.18\text{E-}02\pm1.41\text{E-}03$	$2.43\text{E-}02\pm1.27\text{E-}03$
$\mathrm{PINN}_{\mathcal{V}_2}^G$	$2.76\text{E-}02\pm5.54\text{E-}04$	$8.32\text{E-}02\pm6.50\text{E-}03$	$8.06\text{E-}03 \pm 1.86\text{E-}03$
$PINN_{\mathcal{V}_2}^{TL}$	$1.60\text{E-}02\pm2.28\text{E-}04$	$8.34\text{E-}02\pm1.81\text{E-}03$	$2.85\text{E-}03 \pm 1.85\text{E-}03$

Table 2: Comparison for PINN-based approaches with relative errors are averaged across 5 runs with different random seeds. The superscript G and TL in the method indicates default Glorot or transfer learning initializations. The subscript \mathcal{V}_i indicates the geometry used. The results for $\text{PINN}_{\mathcal{V}_1}^G$ are shown for twice more iteration (40000), compared to $\text{Hybrid}_{\mathcal{V}_1}^G$.

For V_1 , we use $N_r = 27500$, $N_{\text{out}} = 1000$ collocation points and 250 collocation points per obstacle, totaling $N_b = 4000$. In Table 2, we observe that despite requiring twice as many (40000 against 20000) iterations, the accuracy of PINNs is still slightly lower for the velocity components, compared to the

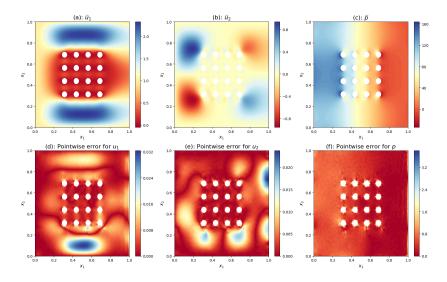


Figure 10: The hybrid solver results for the microscale PDE and pointwise errors (compared with the FEM solution) for V_1 using Glorot initialization and scale bridging with $\tilde{\mu} = 0.175$.

hybrid approach. The results in Table 2 are presented for proper scale bridging with $\tilde{\mu}=0.175$, and, exemplarily, the pointwise errors of the hybrid approach are shown in Fig. 10. However, Fig. 11 shows, among other cases, the relative error curves for $\tilde{\mu}=0.25$, where the PINN method and the hybrid solver achieve comparable results. This supports our prior expectations as shown in Fig. 8(b). Notably, the hybrid solver exhibits faster initial error decay due to the embedded mesoscale information. This advantage, observed in both upscaling-consistent (Fig. 11(b)) and non-upscaling-consistent (Fig. 11(c)) cases, helps yield better permeability values faster, as shown in Fig. 12(a).

For V_2 , we use $N_r = 45,000$, $N_{\rm out} = 1,000$, and 250 collocation points per obstacle, totaling $N_b = 6,250$, and $k_{\rm max} = 25000$. Table 2 compares the hybrid solver and PINNs using Glorot initialization and transfer learning (TL) initialization $\theta_0 := \tilde{\theta}$ ($\tilde{\mu} = 0.25$ in the hybrid case), where $\tilde{\theta}$ is obtained via the hybrid approach on V_1 . Fig.13 shows pointwise errors for the hybrid solver with TL. TL consistently improves accuracy and yields the best permeability value, as seen in Fig. 12(b), demonstrating its potential for multi-query SVE

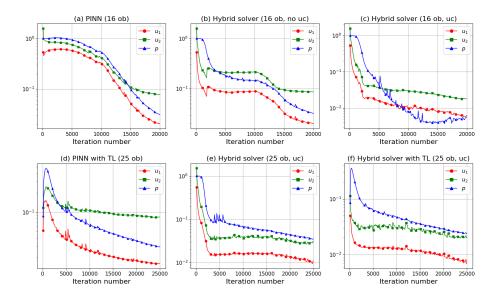


Figure 11: Relative l_2 errors for u_1, u_2 and p vs iterations. Abbreviations '16 ob" and '25 ob" stand for V_1 and V_2 , respectively.

computations. However, the PINN with TL and the hybrid solver with Glorot initialization yield similar results. Notably, Fig.11(e) shows that the hybrid solver refines permeability rapidly even without TL, as upscaling partially averages out neural network approximation error at the microscale. However, a significant limitation of the current PINN-based approach is its training speed — a well-known issue in the scientific community [31]. PINNs take 15 to 30 minutes to train (on A100 GPUs with 40 GB RAM) for \mathcal{V}_1 and \mathcal{V}_2 , respectively. The hybrid approach is about 20-30% more computationally expensive if used for $k:=k_{\text{max}}$. Thus, substantial mathematical and algorithmic development is required, and more powerful hardware must be utilized.

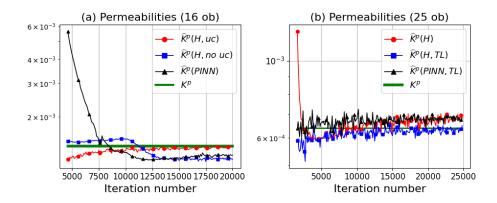


Figure 12: Physics-informed permeabilities obtained using the PINN and hybrid (H) apporaches from the microscale geometries (a): V_1 , where abbreviation 'uc' and 'no uc' stands for consistent ($\tilde{\mu} = 0.175$) and non consistent ($\tilde{\mu} = 0.45$) scale bridging, and (b): V_2 , where TL stands for transfer learning.

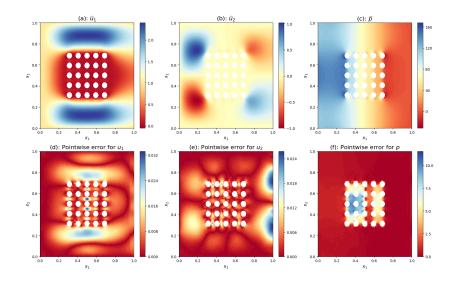


Figure 13: The hybrid solver results for the microscale PDE and pointwise errors (compared with the FEM solution) for V_2 using transfer learning initialization and scale bridging with $\tilde{\mu}=0.25$.

6. Conclusion

This work presents several data-driven approaches for predicting the permeability of mesoscale textile models, based on various permeability determination methods: SSM, SUM, SBM, and FRM. The SSM is the simplest, ignoring microscale permeabilities, while the SUM represents the state of the art. The SBM improves upon the SUM by accounting for the influence of microscale permeability on mesoscale permeability. Despite its increased complexity, the SBM maintains computational efficiency due to the use of data-driven surrogates. Although permeability values vary across methods, as noted in [12, 32], the SBM shows potential to enhance both efficiency and accuracy. The FRM is theoretically the most accurate but faces prohibitive computational costs, emphasizing the need for efficient scale-bridging methods. Therefore, we propose the hybrid PINN-numerical dual-scale solver to further assist the data-driven upscaling process. Using simplified 2D models, we demonstrate that hybridization with properly embedded upscaling process enhances microscale PINN approximations and improves the micropermeabilities via physics-informed regularization. The hybrid ansatz serves as a first step not only in merging various approaches to assist scale bridging, but also in capturing the inherent reciprocal dependency between scales, such as those arising in unsaturated flows. However, advanced applications will require progress in optimization to achieve faster and more robust PINN training, as this remains a key limitation.

7. Acknowledgments

All authors acknowledge the support of the Leibniz Collaborative Excellence Cluster under project ML4Sim (funding reference: K377/2021).

References

S. V. Lomov, G. Huysmans, Y. Luo, R. Parnas, A. Prodromou, I. Verpoest,
 F. Phelan, Textile composites: modelling strategies, Composites Part A: applied science and manufacturing 32 (10) (2001) 1379–1394.

- [2] N. Lubbers, A. Agarwal, Y. Chen, S. Son, M. Mehana, Q. Kang, S. Karra, C. Junghans, T. C. Germann, H. S. Viswanathan, Modeling and scalebridging using machine learning: Nanoconfinement effects in porous media, Scientific reports 10 (1) (2020) 13312.
- [3] R. Gorguluarslan, S.-K. Choi, A simulation-based upscaling technique for multiscale modeling of engineering systems under uncertainty, International Journal for Multiscale Computational Engineering 12 (6) (2014).
- [4] J. Seuffert, L. Bittrich, L. Cardoso de Oliveira, A. Spickenheuer, L. Kärger, Micro-scale permeability characterization of carbon fiber composites using micrograph volume elements, Frontiers in Materials 8 (2021) 745084.
- [5] M. Hintermüller, D. Korolev, A hybrid physics-informed neural network based multiscale solver as a partial differential equation constrained optimization problem, arXiv preprint arXiv:2309.04439 (2023).
- [6] M. Raissi, P. Perdikaris, G. E. 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.
- [7] M. Griebel, M. Klitz, Homogenization and numerical simulation of flow in geometries with textile microstructures, Multiscale Modeling & Simulation 8 (4) (2010) 1439–1460.
- [8] T. Schmidt, F. Schimmer, A. Widera, D. May, N. Motsch, C. Bauer, A novel simulative-experimental approach to determine the permeability of technical textiles, Key Engineering Materials 809 (2019) 487–492.
- [9] P. B. Nedanov, S. G. Advani, Numerical computation of the fiber preform permeability tensor by the homogenization method, Polymer composites 23 (5) (2002) 758–770.

- [10] E. Syerko, C. Binetruy, S. Comas-Cardona, A. Leygue, A numerical approach to design dual-scale porosity composite reinforcements with enhanced permeability, Materials & Design 131 (2017) 307–322.
- [11] X. Du, M. Ostoja-Starzewski, On the size of representative volume element for darcy law in random media, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 462 (2074) (2006) 2949–2963.
- [12] E. Syerko, T. Schmidt, D. May, C. Binetruy, S. G. Advani, S. Lomov, L. Silva, S. Abaimov, N. Aissa, I. Akhatov, et al., Benchmark exercise on image-based permeability determination of engineering textiles: Microscale predictions, Composites Part A: Applied Science and Manufacturing 167 (2023) 107397.
- [13] T. Schmidt, D. K. Natarajan, M. Duhovic, S. Cassola, M. Nuske, D. May, Geometric model dataset-fiberstructureperm-and machine learning models for permeability prediction of fibrous structures, Available at SSRN 4872087 (2024).
- [14] J. Hilden, S. Linden, B. Planas, Flowdict user guide geodict release 2024, GeoDict 2024 User Guide. Math2Market GmbH (2024).
- [15] D. K. Natarajan, T. Schmidt, S. Cassola, M. Nuske, M. Duhovic, D. May, A. Dengel, Data-driven permeability prediction of 3D fibrous microstructures, Proceedings of ECCOMAS Congress 2024 - 9th European Congress on Computational Methods in Applied Sciences and Engineering (Sep. 2024).
- [16] B. Caglar, G. Broggi, M. A. Ali, L. Orgéas, V. Michaud, Deep learning accelerated prediction of the permeability of fibrous microstructures, Composites Part A: Applied Science and Manufacturing 158 (2022) 106973.
- [17] M. Elmorsy, W. El-Dakhakhni, B. Zhao, Generalizable Permeability Prediction of Digital Porous Media via a Novel Multi-Scale 3D Convolutional

- Neural Network, Water Resources Research 58 (3) (2022) e2021WR031454. doi:10.1029/2021WR031454.
- [18] M. Alber, A. Buganza Tepole, W. R. Cannon, S. De, S. Dura-Bernal, K. Garikipati, G. Karniadakis, W. W. Lytton, P. Perdikaris, L. Petzold, et al., Integrating machine learning and multiscale modeling—perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences, NPJ digital medicine 2 (1) (2019) 115.
- [19] K. Prantikos, S. Chatzidakis, L. H. Tsoukalas, A. Heifetz, Physics-informed neural network with transfer learning (tl-pinn) based on domain similarity measure for prediction of nuclear reactor transients, Scientific Reports 13 (1) (2023) 16840.
- [20] C. Xu, B. T. Cao, Y. Yuan, G. Meschke, Transfer learning based physicsinformed neural networks for solving inverse problems in engineering structures under different loading scenarios, Computer Methods in Applied Mechanics and Engineering 405 (2023) 115852.
- [21] K. Shukla, A. D. Jagtap, G. E. Karniadakis, Parallel physics-informed neural networks via domain decomposition, Journal of Computational Physics 447 (2021) 110683.
- [22] P. Escapil-Inchauspé, G. A. Ruz, h-analysis and data-parallel physicsinformed neural networks, Scientific Reports 13 (1) (2023) 17562.
- [23] S. Wang, S. Sankaran, H. Wang, P. Perdikaris, An expert's guide to training physics-informed neural networks, arXiv preprint arXiv:2308.08468 (2023).
- [24] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, A. Courville, On the spectral bias of neural networks, in: International conference on machine learning, PMLR, 2019, pp. 5301–5310.
- [25] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, R. Ng, Fourier features let net-

- works learn high frequency functions in low dimensional domains, Advances in neural information processing systems 33 (2020) 7537–7547.
- [26] S. Wang, H. Wang, P. Perdikaris, On the eigenvector bias of fourier feature networks: From regression to solving multi-scale pdes with physics-informed neural networks, Computer Methods in Applied Mechanics and Engineering 384 (2021) 113938.
- [27] D. Boffi, F. Brezzi, M. Fortin, et al., Mixed finite element methods and applications, Vol. 44, Springer, 2013.
- [28] S. Wang, Y. Teng, P. Perdikaris, Understanding and mitigating gradient flow pathologies in physics-informed neural networks, SIAM Journal on Scientific Computing 43 (5) (2021) A3055–A3081.
- [29] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the thirteenth international conference on artificial intelligence and statistics, JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.
- [30] D. P. Kingma, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).
- [31] N. McGreivy, A. Hakim, Weak baselines and reporting biases lead to overoptimism in machine learning for fluid-related partial differential equations, Nature Machine Intelligence (2024) 1–14.
- [32] E. Syerko, T. Schmidt, D. May, C. Binetruy, S. V. Lomov, S. Advani, L. Silva, Mesoscale results from the virtual permeability benchmark on engineering textiles, in: 21st European Conference on Composite Materials, Vol. 8 - Special Sessions, The European Society for Composite Materials (ESCM) and the Ecole Centrale de Nantes, 2024, pp. 1312–1320.