

Predicting heat transfer in fibre-reinforced anisotropic composite materials using physics-informed neural networks (PINNs)

Bernabe Lorenzo Avila, Dietmar Koch, Nils Meyer

Angaben zur Veröffentlichung / Publication details:

Lorenzo Avila, Bernabe, Dietmar Koch, and Nils Meyer. 2026. "Predicting heat transfer in fibre-reinforced anisotropic composite materials using physics-informed neural networks (PINNs)." *Materials Today Communications* 56: 116177.
<https://doi.org/10.1016/j.mtcomm.2026.116177>.

Nutzungsbedingungen / Terms of use:

CC BY 4.0





Predicting heat transfer in fibre-reinforced anisotropic composite materials using physics-informed neural networks (PINNs)

Bernabe Lorenzo Avila^{a,*}, Dietmar Koch^{a,b}, Nils Meyer^{a,b}

^a University of Augsburg, Institute of Materials Resource Management, Augsburg, Germany

^b University of Augsburg, Centre for Advanced Analytics and Predictive Sciences, Augsburg, Germany

ARTICLE INFO

Dataset link: [Dataset for PINN-based heat transfer in anisotropic fibre-reinforced composites \(Original data\)](#)

Keywords:

Physics-informed neural network
Heat transfer
Composite materials

ABSTRACT

Accurate prediction of heat transfer in ceramic matrix composites (CMCs) is essential for designing components subjected to high thermal loads, where fibre orientation strongly influences anisotropic heat conduction. Conventional finite element method (FEM) analyses require the temperature field to be recomputed for each configuration, resulting in high computational cost. This work develops a physics-informed neural network (PINN) to solve the parametric heat transfer partial differential equation, introducing the fibre orientation angle as a direct input variable to enable predictions for arbitrary orientations. The proposed framework is validated against FEM solutions for two-dimensional anisotropic heat conduction, achieving a mean relative error below 2%, a training time of 0.3 h, and a mean inference time of 1.478 ms for 10,000 points. To establish an effective training methodology, three strategies are investigated: random-angle, sequential-angle, and batched-angle training. Random-angle training assigns a random fibre orientation to each collocation point, introducing angle-dependent bias, while sequential-angle training optimises the network successively on individual sample domains, leading to catastrophic forgetting. Batched-angle training optimises multiple sample domains simultaneously, providing stable convergence. Building on this approach, a dynamic batched-angle strategy generates new sets of random fibre orientations at each training epoch to increase training diversity and improve convergence. The PINN maintains a mean relative error below 2% for thermal diffusivity ratios between 0.1 and 1. Within the investigated geometry and boundary conditions, the trained PINN is applicable to materials with the same thermal diffusivity ratio; changes in anisotropy ratio, geometry, or boundary conditions require retraining.

1. Introduction

The prediction of the temperature distribution in composite materials plays a crucial role in the design of thermal protection systems, critical components in the aerospace sector responsible for maintaining spacecraft performance and safety. Thermal loads constitute one of the main stresses acting on these systems, making accurate modelling of heat transfer and temperature distribution indispensable during the design of thermal barriers. One of the primary materials employed for thermal protection is ceramic matrix composites (CMCs), as they combine high-temperature resistance of ceramics with enhanced damage tolerance provided by the fibre reinforcement. Due to the presence of continuous fibres, these materials exhibit anisotropic thermal properties, with a thermal conductivity along the fibre direction that is up to three times greater than that in the perpendicular direction [1]. Consequently, different fibre orientations result in different temperature distributions.

During design, this anisotropy is exploited to rapidly dissipate the heat across the material, preventing localised overheating, or to isolate specific regions, thereby reaching an optimal temperature distribution. To determine which fibre orientation achieves this optimum, it is necessary to iteratively evaluate the temperature field for many orientations to best meet the thermal requirements.

The temperature distribution can be determined using the heat transfer equation, a linear partial differential equation (PDE) that describes the time-dependent heat diffusion. Traditionally, finite element solvers are employed to solve this problem [2,3]. However, a new calculation is required for each orientation configuration explored in the design phase, making this step of the design potentially computationally expensive and time-consuming. An alternative to FEM is the use of deep neural networks (DNNs), which are capable of rapidly predicting the temperature field for arbitrary fibre orientations, enabling efficient determination of the most favourable configuration for the design, once they are trained.

* Corresponding author.

Email address: bernabe.lorenzo.avila@uni-a.de (B. Lorenzo Avila).

<https://doi.org/10.1016/j.mtcomm.2026.116177>

Received 23 June 2026; Received in revised form 28 August 2026; Accepted 14 September 2026

Available online 16 September 2026

2352-4928/© 2026 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Open access funding enabled and organized by Projekt DEAL.

The principal benefits of using DNNs to solve PDEs, in this instance the heat equation, include their capability to approximate any non-linear function, their rapid and efficient evaluation of forward problems once trained and their analytical differentiability, which permits derivatives to be computed easily [4]. One of the main drawbacks of DNNs is the large amount of labelled data they require, which in the initial design step is usually limited or unavailable. However, in cases where the governing PDE is known, its solution can be learnt under physics-constraints with little data [5,6] or, as in this study, no real data [7,8]. Since the output of a DNN is a continuously differentiable function, all terms in the PDE can be calculated using automatic differentiation [9,10]. These networks, known as physics-informed neural networks (PINNs), incorporate the PDE and other governing physical constraints into the loss function of the DNN during training, thereby leveraging prior physical knowledge and creating a bridge between deep learning and classical physics-based modelling. PINNs can also infer unseen values within the training range with high accuracy, reduce overfitting and improve generalisation. In addition, they exhibit robustness to noisy inputs [10].

The heat equation has been solved using PINNs for isotropic and anisotropic problems [11,12], as well as for material stacks [13]. However, in these cases, the properties of the materials were kept constant during training, meaning the models could only predict the temperature distributions for a single, fixed material configuration. This work goes further by aiming to predict the temperature distribution for arbitrary fibre orientations, treating the fibre angle as a parameter. The parametric nature of the heat equation further emphasises this point, as it produces distinct solutions depending on varying material properties, fibre orientations, and geometries. The strategy and effectiveness of using regular PINNs to enable parametric solutions are subject to discussion, with some supporting their efficiency [14] and others doubting their performance and generalisation capabilities [15].

Learning parametric PDEs with PINNs can be divided into two main approaches: The first approach learns a mapping between the parameters of interest and the trained weights of the PINN [16,17]. Although this method accelerates the training, it requires re-optimisation for every new parameter, making it impractical to use in optimisation problems. The second approach treats the parameters as additional inputs to the PINN, alongside the conventional PDE variables [18]. However, the typical optimisation difficulties of PINNs are amplified in this approach, proving to be particularly challenging for non-linear PDEs [19]. It is emphasised that the solution of parametric PDEs has been successfully addressed using more complex network architectures, such as Physics-Informed DeepONets [20–22].

In the present study, the fibre angle is incorporated directly as an input parameter, allowing the standard PINN formulation to be retained within a simple feedforward network. A central aim is to demonstrate the capability of this approach to represent the temperature response over a continuous range of fibre orientations while maintaining computational efficiency. The novelty of this work lies in investigating the suitability of different strategies - random-angle, sequential-angle and batched-angle training - for exposing the network to the input parameter space and in developing an improved training procedure based on the most effective strategy. Rather than introducing a more complex neural architecture, the proposed methodology focuses on the application of a simple PINN to a parametric anisotropic materials problem. In this way, it addresses the practical challenge of predicting orientation dependent temperature distributions when the material configuration itself constitutes a key parameter of the governing physical response.

2. Methodology

2.1. Problem definition

Heat transfer in a two-dimensional square composite domain with side length L , which maps to the unit square in dimensionless coordinates, as shown in Fig. 1, is considered in this work. Non-dimensionalisation is a key pre-processing step in both traditional deep

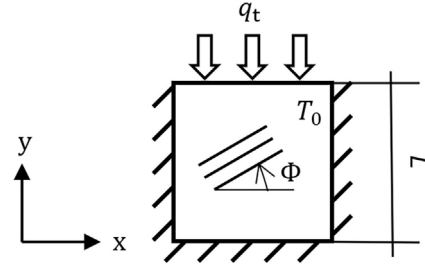


Fig. 1. Schematic representation of a 2D piece of material exposed to an external heat flux on the upper edge.

learning and physics-informed neural networks (PINNs), as it improves training stability and convergence by bringing inputs and outputs to comparable scales, thereby stabilising gradient propagation and facilitating more efficient optimisation [23,24]. In this context, the governing transient heat transfer equation is formulated in dimensionless form as

$$\frac{\partial T}{\partial t} = \nabla \cdot (\alpha \nabla T), \quad (x, y, t) \in [0, 1] \times [0, 1] \times [0, t_{\max}] \quad (1)$$

where $t_{\max}$ is the dimensionless duration. The dimensionless anisotropic thermal diffusivity tensor α is computed for a given fibre orientation angle φ via

$$\alpha = \mathbf{R}(\varphi) \alpha^{\text{UD}} \mathbf{R}^T(\varphi), \quad (2)$$

where $\alpha^{\text{UD}} = \text{diag}(\alpha_{\parallel}, \alpha_{\perp})$ denotes the dimensionless thermal diffusivity tensor in the fibre direction, with parallel and transverse diffusivities $\alpha_{\parallel}$ and $\alpha_{\perp} = \frac{1}{2} \alpha_{\parallel}$, respectively. The rotator tensor

$$\mathbf{R}(\varphi) = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{bmatrix} \quad (3)$$

transforms the material properties to any given fibre orientation, with $\mathbf{R}^T$ denoting its transpose.

All quantities are dimensionless: lengths are scaled by L , time by $\rho c_p L^2 / k_{\parallel}$, temperature by ΔT about T_{ref} , and heat flux by $k_{\parallel} \Delta T / L$, so that $\alpha = \mathbf{K} / k_{\parallel}$ with $\mathbf{K}$ being the thermal conductivity tensor. For the CMC considered this gives $\alpha_{\parallel} = 1$, $\alpha_{\perp} = 0.5$, $T_0 = -0.4$, $t_{\max} = 0.286$ and $q = 1$ at the heated edge.

The system is initialised with a uniform dimensionless temperature field $T(x, y, 0) = T_0$. Neumann boundary conditions are applied according to

$$\mathbf{n} \cdot (\alpha \nabla T) = q, \quad (4)$$

where $\mathbf{n}$ is the outward unit normal and q the prescribed dimensionless normal heat flux, taken positive for heating. The domain is heated along the top edge with $q(y = 1) = 1$, while all the remaining edges (left, bottom, right) are adiabatically insulated with $q(x = 0) = q(x = 1) = q(y = 0) = 0$.

Under these boundary conditions, a temperature gradient is induced from the top to the bottom of the anisotropic material. Based on this setup, the goal is to develop surrogate models using deep learning neural networks that enable rapid and accurate prediction of the temperature field for any fibre orientation, at any point in the domain, and at any instant, while adhering to these physical expressions.

2.2. Physics-informed neural networks (PINNs)

Fig. 2 illustrates the architecture of the PINN employed in this work. A fully connected feedforward neural network is used. It comprises an input layer, a random Fourier embedding layer (see Section 2.2.2), and six hidden layers with 60 neurons each using the activation function σ . An output layer approximates the solution $T(x, y, t, \phi)$.

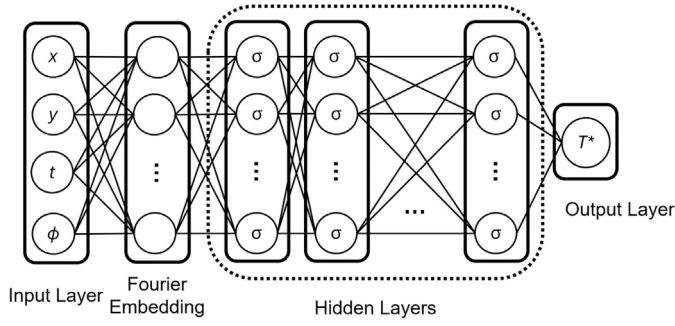


Fig. 2. Schematic representation of the parametric neural network used to solve the heat transfer PDE.

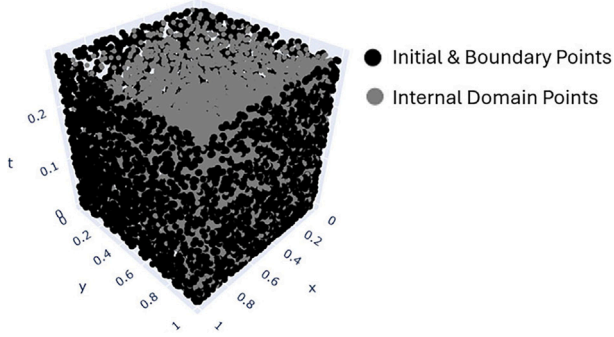


Fig. 3. Visualisation of a random sample domain for training.

The PINN receives the spatial coordinates x and y , the temporal variable t , and the fibre orientation ϕ as dimensionless inputs. The dimensionless fibre orientation $\phi \in [0, 1]$ is related to the fibre orientation angle via $\varphi = \pi\phi$. Due to symmetry, any orientation outside ϕ 's interval can be transformed into an equivalent orientation within it. The output of the PINN is the predicted dimensionless temperature field, indicated as $T^*(x_i, y_i, t_i, \phi_i; \theta) = T_i^*$, where θ represents the weights W_{kl} and biases b_k associated with the transmission between neurons.

To train the model and solve the equation, no mesh is required. Spatial and temporal collocation points are sampled from the space $[0, 1] \times [0, 1] \times [0, t_{\max}] \subset \mathbb{R}^3$, corresponding to a physical duration of 100 s in a CMC material, as shown in Fig. 3. For the training loss, points are classified into three sets: set $\mathcal{I}$, containing N_{ic} points corresponding to the initial condition, where the target values are known; set $\mathcal{N}$, containing N_{neu} points for the Neumann boundary conditions, for which the derivatives are known; and set $\mathcal{R}$, containing N_r points corresponding to the internal domain, where the target values are unknown and for which the predicted values of the network are substituted into the governing Eq. (1). Consequently, the loss function used for training is

$$\mathcal{L} = \lambda_{ic}\mathcal{L}_{ic} + \lambda_{neu}\mathcal{L}_{neu} + \lambda_r\mathcal{L}_r \quad (5)$$

with the loss of the initial condition

$$\mathcal{L}_{ic} = \frac{1}{N_{ic}} \sum_{i \in \mathcal{I}} (T_i^* - T_i)^2, \quad (6)$$

the loss of the Neumann boundary condition

$$\mathcal{L}_{neu} = \frac{1}{N_{neu}} \sum_{i \in \mathcal{N}} (\mathbf{n} \cdot (\alpha \nabla T_i^*) - q)^2, \quad (7)$$

and the loss of the PDE

$$\mathcal{L}_r = \frac{1}{N_r} \sum_{i \in \mathcal{R}} r^2(T_i^*), \quad (8)$$

where r denotes the residual of Eq. (1). These loss terms are balanced by weight factors λ_i , as explained in Section 2.2.2.

2.2.1. Orientation sampling strategy

The PINN proposed in this work learns the parametric solution by incorporating a dimensionless fibre orientation as parametric input.

Although the architecture remains largely identical to that of a standard PINN, the presence of this additional independent parameter introduces the issue of determining an appropriate sampling and training strategy. Therefore, three training methodologies are analysed and discussed:

Training with random angles over the sample domain: This approach follows the idea of non-parametric PINNs, where random collocation points are created over the sample domain. Each point is assigned not only a random spatial location and time, but also a random fibre orientation, which changes with every new epoch. The independently assigned fibre orientations constitute a mathematical sampling strategy for exploring the material parameter space and are not intended to represent physically discontinuous or randomly oriented fibres. Instead, the underlying material model remains a homogenised anisotropic continuum, with the fibre orientation treated as an independent material parameter. PINN training proceeds with sample domains as illustrated in Fig. 4 and according to Eq. (5).

Training with sequential sample domains: This approach first generates a random sample domain with a fixed, randomly selected angle ϕ_i , as shown in Fig. 5(a). The PINN is then trained on this domain according to Eq. (5). Once training on that sample domain is completed, a new random sample domain with a different fixed angle is generated, as shown in Fig. 5(b) and training continues with this new sample domain.

Training with batched sample domains: This approach begins by generating N_ϕ random sample domains, each with a different, fixed, and randomly selected angle ϕ_i . Instead of training the PINN on each domain separately, the loss for each domain is first calculated as in Eq. (5), is subsequently summed as

$$\mathcal{L} = \frac{1}{N_\phi} \sum_{i=1}^{N_\phi} \mathcal{L}_{\phi_i} \quad (9)$$

and is then used collectively for training. This work also extends previous studies [25] by introducing the concept of generating a new set of N_ϕ sample domains with new random angles ϕ_i at the start of every epoch, as shown in Fig. 6.

2.2.2. Techniques for enhancing accuracy and training efficiency

This section presents five techniques incorporated into the PINN to boost performance and accelerate the learning process.

Sampling distribution: PINNs are typically trained with randomly generated sample points across the domain. This approach can lead to poor model performance since some regions may be more difficult to learn than others. Different collocation points contribute different amounts of information, which implies that redistributing them, such that more points lie in the harder-to-learn regions, improves training efficiency and reduces the likelihood of the network getting trapped in local minima [26]. In this case, the region near the top side is expected to be more challenging to learn, as it presents sharper gradients. Therefore, the y-component of the randomly generated internal points is transformed into an exponential distribution,

$$y_{r,exp} = \frac{e^{-3}y_r - 1}{e^{-3} - 1} \quad (10)$$

concentrating the points on the top side.

Learning rate scheduler: The learning rate η determines the magnitude of weight updates throughout training and is a critical parameter for controlling both the speed of convergence and the final accuracy of the model. An excessively large initial η can hinder convergence, particularly in the final steps of training [27]. To address this, a learning rate scheduler is introduced, which reduces η exponentially every M steps

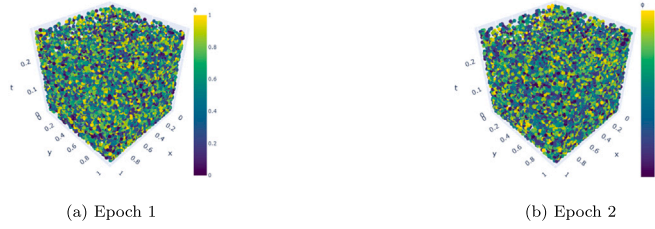


Fig. 4. Exemplary sample domains in different training epochs for training with random angles.



Fig. 5. Exemplary sample domains with fixed angles ϕ_i for sequential training.

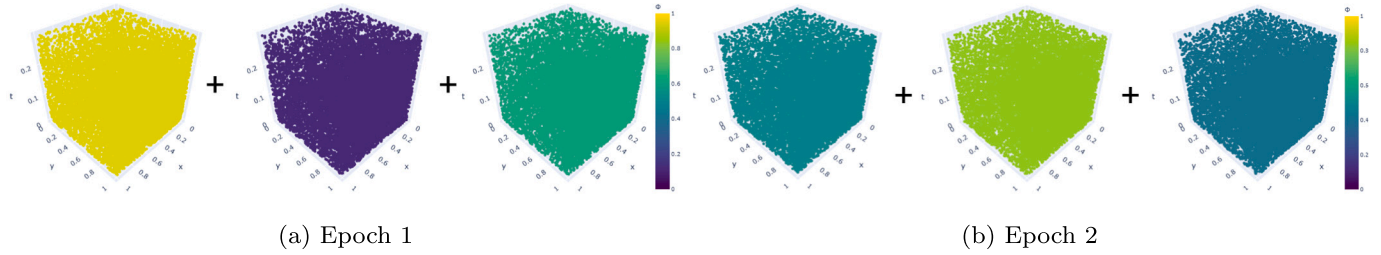


Fig. 6. Exemplary sample domains with $N_\phi = 3$ fixed angles ϕ_i for batched training.

according to:

$$\eta(t) = \eta_0 \cdot \gamma^{\frac{t}{M}}, \quad \text{for } t = n \cdot M, \quad n = 1, 2, 3, \dots \quad (11)$$

Random Fourier Features: Although PINNs can approximate any function, they tend to favour low-frequency components, exhibiting a bias towards smooth functions, an effect known as spectral bias [28]. Random Fourier Features mitigate this spectral bias [29] by mapping the input coordinates into frequencies

$$\gamma(\mathbf{v}) = \begin{bmatrix} \cos(2\pi \mathbf{B}\mathbf{v}) \\ \sin(2\pi \mathbf{B}\mathbf{v}) \end{bmatrix}, \quad (12)$$

where $\mathbf{v} = (x, y, t, \phi) \in \mathbb{R}^4$ represents the input vector of the PINN, and $\mathbf{B} \in \mathbb{R}^{m \times 4}$ is a random matrix sampled from a Gaussian distribution $\mathcal{N}(0, \sigma^2)$.

Adaptive loss balancing: As previously shown in Eq. (5), the loss function in a PINN usually consists of three terms: the loss of the initial condition, the boundary conditions and the residual of the physical equation. During training, each term might behave differently, causing the optimizer's priority in minimising them to change due to their different natures, potentially reducing the final accuracy [30]. The adaptive loss balancing method addresses this issue by weighting the loss terms with factors λ_i , which equalise the gradient norms across the components. These weights are calculated and automatically updated according to [30] as

$$\hat{\lambda}_{ic} = \frac{\|\nabla_{\theta} \mathcal{L}_{ic}(\theta)\| + \|\nabla_{\theta} \mathcal{L}_{Neu}(\theta)\| + \|\nabla_{\theta} \mathcal{L}_r(\theta)\|}{\|\nabla_{\theta} \mathcal{L}_{ic}(\theta)\|}$$

$$\hat{\lambda}_{Neu} = \frac{\|\nabla_{\theta} \mathcal{L}_{ic}(\theta)\| + \|\nabla_{\theta} \mathcal{L}_{Neu}(\theta)\| + \|\nabla_{\theta} \mathcal{L}_r(\theta)\|}{\|\nabla_{\theta} \mathcal{L}_{Neu}(\theta)\|}$$

$$\hat{\lambda}_r = \frac{\|\nabla_{\theta} \mathcal{L}_{ic}(\theta)\| + \|\nabla_{\theta} \mathcal{L}_{Neu}(\theta)\| + \|\nabla_{\theta} \mathcal{L}_r(\theta)\|}{\|\nabla_{\theta} \mathcal{L}_r(\theta)\|}$$

$$\lambda_{new} = \beta \lambda_{old} + (1 - \beta) \hat{\lambda}_{new} \quad (13)$$

every M_λ steps and are thus referred to as adaptive weights.

Temporal loss balancing: PINNs exhibit a tendency to first minimise the PDE residuals at later times before adequately fitting the initial data, thereby neglecting the temporal causality inherent in transient problems [31]. This effect leads to an accumulation of inaccuracies and becomes more pronounced the further into the future the prediction extends. To address this issue, the causal training algorithm proposed in [31] employs a temporal weighting of N_t chronologically sorted time segments via

$$\mathcal{L}(\theta) = \frac{1}{N_t} \sum_{i=1}^{N_t} w_i \mathcal{L}_r(t_i, \theta) \quad (14)$$

with weights

$$w_i = \exp\left(-\epsilon \sum_{k=1}^{i-1} \mathcal{L}_r(t_k, \theta)\right) \quad \text{for } i = 2, 3, \dots, N_t. \quad (15)$$

These temporal weights depend on the losses of the previous time frames, preventing the PINN from learning later times until losses at the earlier times are sufficiently low.

2.2.3. Implementation, evaluation, and validation of the PINN

All training is performed on an NVIDIA GeForce RTX 4060 Ti Graphics Processing Unit (GPU). The PINN was built using PyTorch with the following fixed settings: A fully connected neural network with six hidden layers and 60 neurons each is employed. The activation function σ is selected as the hyperbolic tangent (tanh) due to its differentiability at higher orders. The models are trained with $N_t = 10,000$ internal points and $N_{\text{Neu}} = N_{\text{ic}} = 1000$ points on each boundary and initial time. To ensure a fair comparison, all three training methodologies were trained using the same total number of optimization steps. Consequently, the random-angle and batched-angle training strategies were trained for 500 epochs, whereas the sequential-angle training strategy, which is performed over 20 sequential domains, was trained for 25 epochs per domain. In all cases, training was performed using the Adam optimizer with an initial learning rate of $\eta_0 = 0.01$, and a learning rate scheduler applying a decay factor of $\gamma = 0.9$ every $M = 200$ optimization steps. Since the optimal weighting of the loss is initially unknown, the adaptive loss balancing factors λ_i are set to 1 at the beginning and dynamically updated every $M_\lambda = 2000$ steps using an update factor $\beta = 0.9$. For temporal loss balancing, the domain is divided into $N_t = 5$ time frames with $\epsilon = 1$. The hyperparameters of the random Fourier features are set to $m = 40$ and $\sigma = 1$. The resulting 80 input features from the sine and cosine mappings yield a total of 23,221 trainable parameters in the fully connected network. Unless otherwise stated, the results reported in this work correspond to a single training run. Whenever the neural network is newly trained, a different random seed is used, and consequently, some differences in the resulting predictions may occur between independent training runs. Finally, the PINN results are validated by comparison with

FEM models generated in COMSOL Multiphysics. This is carried out by calculating the relative error

$$\text{RE} = \frac{T_{\text{PINN}} - T_{\text{FEM}}}{T_{\text{FEM}}} \quad (16)$$

at each point N_p and the maximum absolute value of the relative error of all points $|\text{RE}|_{\text{max}}$ as well as the mean relative error

$$\text{MRE} = \frac{1}{N_p} \sum_{i=1}^{N_p} |\text{RE}_i| \quad (17)$$

over the entire domain for the fibre orientations $\varphi = [0, 30, 45, 90, 120, 135, 150, 180]^\circ$, thereby providing a comprehensive view of the error distribution across all angles. For simplicity and clarity, only errors at the final time t_{max} are shown, and only the angles with the highest and lowest MRE are plotted.

3. Results and discussion

The transient heat transfer problem for an anisotropic 2D material domain at an arbitrary angle, presented in Section 2.1, is analysed in this section using the three methodologies described in Section 2.2.1.

3.1. Training with random angles over the sample domain

The results of this training methodology are presented in Figs. 7 and 8. As shown in Fig. 7, both MRE and $|\text{RE}|_{\text{max}}$ are minimal at 0° , whereas 90° exhibits the highest MRE, revealing a clear trend in which errors

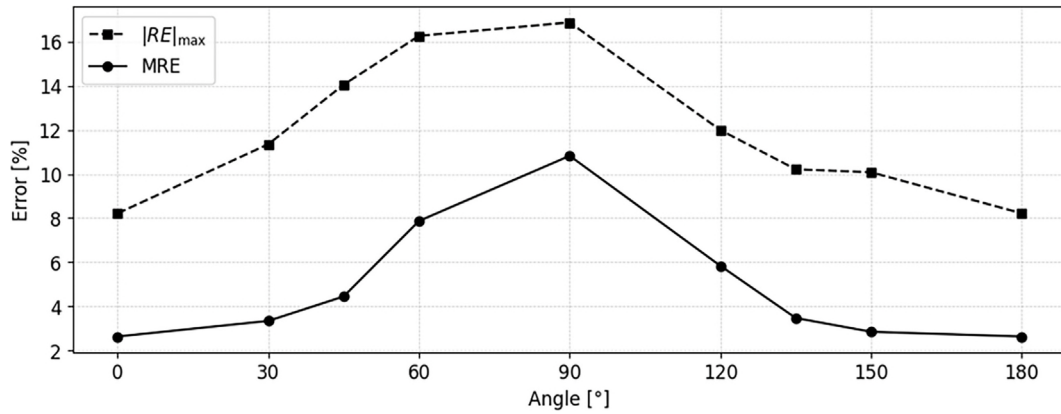
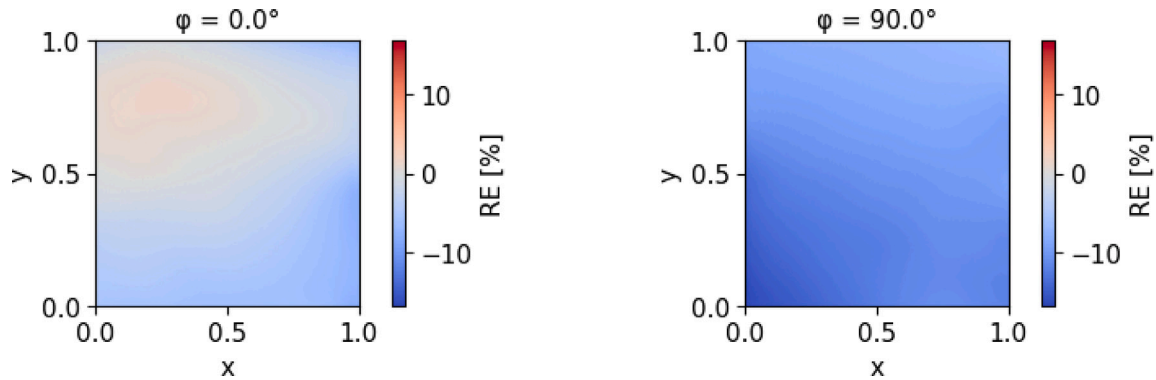


Fig. 7. Mean relative error (MRE) and maximum absolute value of the relative error ($|\text{RE}|_{\text{max}}$) for all angles at t_{max} during training with random angles over the sample domain.



(a) RE of the angle with the lowest MRE.

(b) RE of the angle with the highest MRE.

Fig. 8. RE distribution for the angles with the highest and lowest MRE at t_{max} during training with random angles across the sample domains.

decrease towards 0° and its analogue 180° and increase towards 90° . This behaviour can be attributed to the characteristics of the training domain, which consists of points with randomly assigned angles. This assignment is not intended to represent the physical geometry of individual fibres or discontinuous randomly oriented fibres within the sample. Instead, it is a mathematical sampling strategy that exposes the parametric PINN to different configurations of the homogenised anisotropic material model during training. At each collocation point, the assigned fibre orientation determines the corresponding components of the effective thermal diffusivity tensor, while the collocation point itself does not represent an individual fibre or fibre segment. Consequently, the PINN is required to satisfy PDE residuals associated with different material orientations at independently sampled collocation points. This can introduce inconsistent constraints between neighbouring collocation points assigned different orientations, as the local PDE residuals are evaluated using different thermal diffusivity tensors. This inconsistency can hinder the PINN from learning a coherent spatial temperature field across the domain and can contribute to the poor prediction performance observed with the random-angle training strategy.

The observed angle-dependent error can therefore be interpreted as a consequence of the sampling strategy rather than as a physical effect associated with variations in fibre length or morphology. The angles 0° and 180° both fall within the training range, represent the same orientation, and lie at opposite ends of the domain. This results in a higher representation of orientations equivalent or close to 0° within the training data, biasing the PINN towards learning the corresponding solutions. Consequently, as the orientation deviates from $0^\circ/180^\circ$, the prediction error increases. On the other hand, Fig. 8 shows no clear spatial error pattern. Nevertheless, the spatial covariance is $6.176\%^2$ for 0° and $4.260\%^2$ for 90° , indicating that the prediction errors exhibit spatial correlation. The higher covariance at 0° suggests that the errors are more strongly clustered into localised regions compared

with 90° , although no consistent spatial pattern is observed across the investigated fibre orientations. Ultimately, the training took 0.92 h, while the mean inference time for predicting the temperature field at $t = t_{\max}$ over 10,000 spatial points for each of the considered fibre orientations was 2.77 ms per orientation. Because of insufficient accuracy and bias towards certain angles, this training methodology has proven to be unsuitable for learning the parametric solution of the heat transfer equation.

3.2. Training with sequential sample domains

The PINN was trained over 20 sequential sample domains, each with a random fixed orientation ϕ_i . The corresponding loss function throughout the training is shown in Fig. 9, where the start of each new domain appears as a peak. This occurs because the PINN initially fails to represent the new domain with the weights and biases obtained from the previous epoch, resulting in a high loss that is then minimised as the weights and biases are adjusted for the new domain. The peaks grow larger than the loss reduction achieved during training in that domain, leading to an overall increase in the loss function. This indicates that instead of generalising the problem across new domains with angles ϕ_i , the PINN learns each new domain by forgetting the previously learnt ones. This highlights a key problem with training on sequential sample domains, since each new domain introduces noise, leading to unstable updates that hinder the model's convergence to the true minimum across all domains, as reflected in the high loss in Fig. 9 and inaccurate temperature predictions shown in Figs. 10 and 11. On the other hand, neither Fig. 10 nor Fig. 11 reveals a consistent error pattern across the fibre orientations or within the spatial domain. Nevertheless, the spatial covariance is $13.885\%^2$ for 0° and $27.456\%^2$ for 150° , indicating that the prediction errors are strongly spatially correlated despite the absence of a consistent spatial bias.

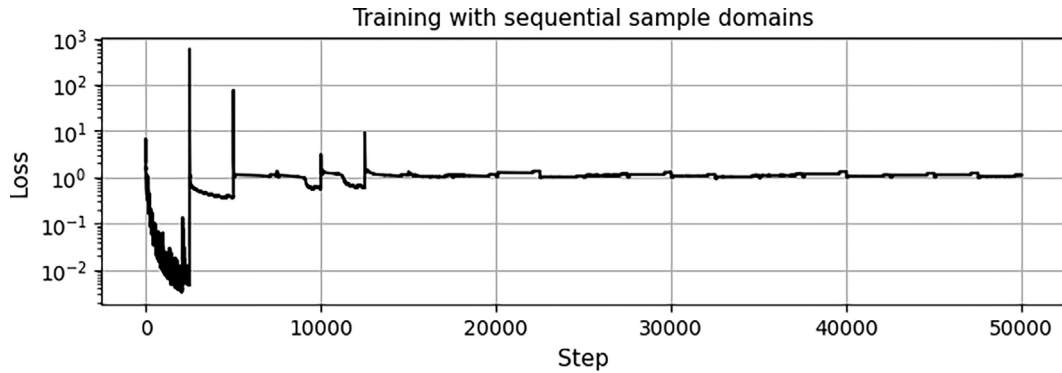


Fig. 9. Training loss with sequential sample domains.

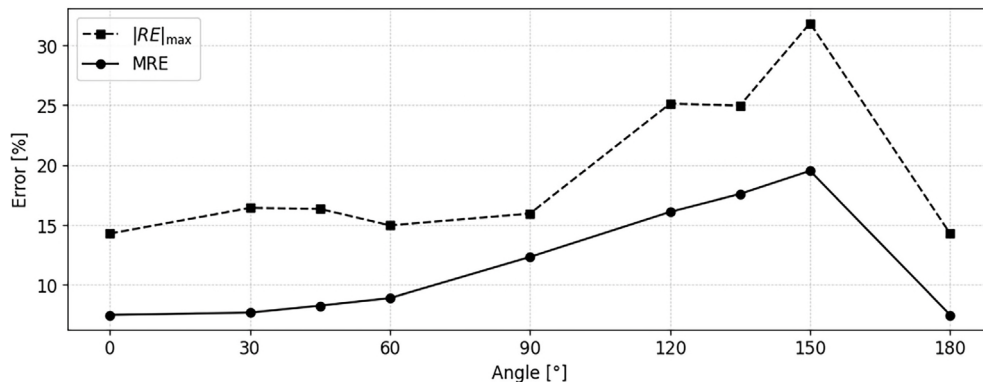


Fig. 10. Mean relative error (MRE) and maximum absolute value of the relative error ($|RE|_{\max}$) for all angles at $t_{\max}$ during training with sequential sample domains.

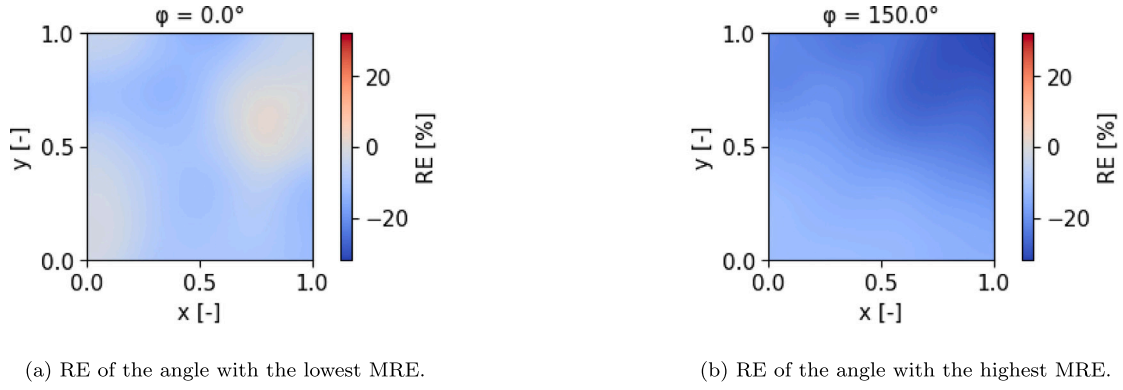


Fig. 11. RE distribution for the angles with the highest and lowest MRE at $t_{\max}$ during training with sequential sample domains.

Ultimately, this training approach is more time-consuming than the previous strategy, as each model must be trained sequentially before the next can begin, preventing the use of GPU parallelization. To ensure a fair comparison with the random-angle sampling strategy, the total number of optimization steps was kept the same (50,000). Consequently, the number of epochs per domain was reduced from 500 to 25, resulting in the same total training time of 0.92 h as in Section 3.1. Had each of the 20 sequential domains been trained for 500 epochs, the total training time would have been approximately 20 times longer. The mean inference time for predicting the temperature field at $t = t_{\max}$ over 10,000 spatial points for each of the considered fibre orientations averaged 3.283 ms per orientation. For all these reasons, this training methodology has proven

to be unsuitable for learning the parametric solution of the heat transfer equation.

3.3. Training with batched sample domains

The PINN was trained with the combined loss from $N_{\phi}=5$ sample domains, each with a randomly fixed angle ϕ_i . Fig. 12 shows that the mean relative error across all angles is below 2%, demonstrating significantly better performance than the previous strategies. Some angles present lower errors than others, which is expected, as certain orientations may be more difficult to learn. Importantly, in contrast to Fig. 7, there is no clear tendency towards any angle, thus proving that the PINN generalises across all angles. Furthermore, Fig. 13 shows no clear spatial

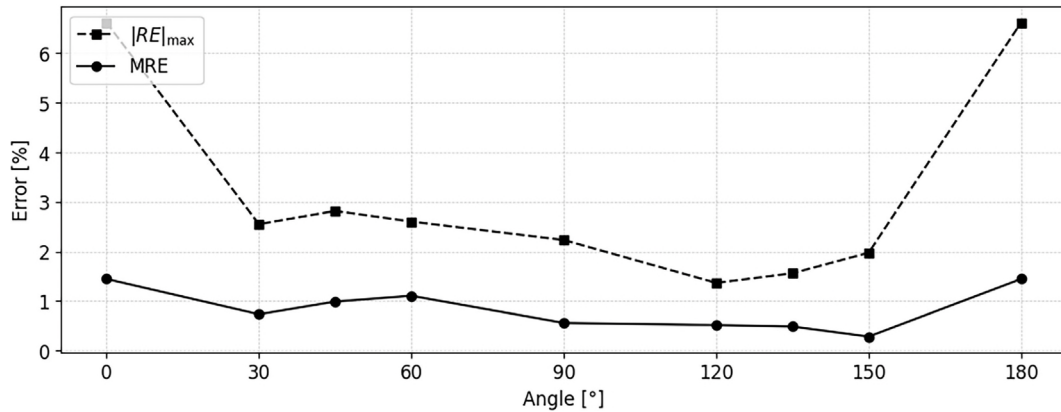


Fig. 12. Mean relative error (MRE) and maximum absolute value of the relative error ($|RE|_{\max}$) for all angles at $t_{\max}$ during training with batched sample domains.

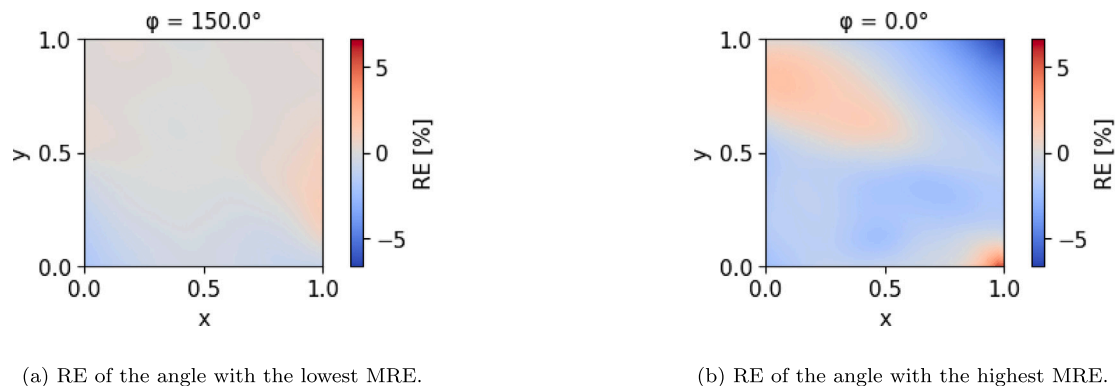
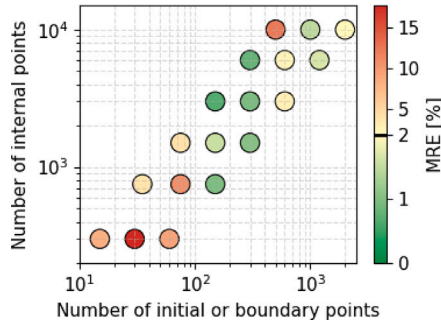
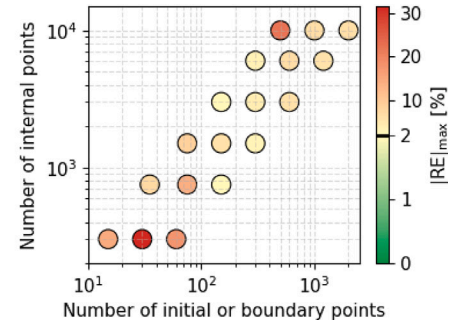


Fig. 13. RE distribution for the angles with the highest and lowest MRE at $t_{\max}$ during training with batched sample domains.



(a) Maximum values of the mean relative error (MRE)



(b) Maximum values of the maximum absolute relative error ($|RE|_{\max}$)

Fig. 14. Maximum values of the mean relative error (MRE) and of the maximum absolute relative error ($|RE|_{\max}$) among the evaluated domains for different combinations of internal and initial/boundary collocation points.

error pattern and, consequently, no noticeable bias across the domain. This observation is further supported by the spatial covariance, which is $0.171\%^2$ for 150° and $1.875\%^2$ for 0° . The former indicates minimal spatial clustering of the errors, whereas the latter suggests only weak spatial correlation, with small errors forming locally coherent regions. This outcome can be explained by the fact that batching the sample domains during training, analogous to batching internal domain points within a single domain, reduces stochastic diffusion noise by decreasing the variance of gradient estimates, leading to more stable updates and consequently smoother and faster convergence. Ultimately, this method allows the parallel processing of operations across the batched domains on the GPU, resulting in a training time of 4.4 h, while the mean inference time for predicting the temperature field at $t = t_{\max}$ over 10,000 spatial points for each of the considered fibre orientations averaging 2.719 ms per orientation. Because of high accuracy and the lack of bias towards any particular angle or across the domain, this training methodology has proven to be suitable for learning the parametric solution of the heat transfer equation.

3.4. Additional considerations

As only the “Training with batched sample domains” strategy proved to be effective, it was used to systematically investigate whether the number of collocation points could be reduced while maintaining a maximum MRE below 2%. The number of internal points N_r , as well as the number of boundary and initial points, $N_{\text{Neu}} = N_{\text{ic}}$, were varied over six representative levels using three ratios, $\frac{N_r}{N_{\text{Neu}}} = \{5, 10, 20\}$ resulting in 18 different combinations. The resulting maximum values of MRE and $|RE|_{\max}$ among the evaluated domains are shown in Fig. 14. The prediction accuracy depends on both sampling densities and does not follow a simple proportional relationship between N_r and $N_{\text{Neu}} = N_{\text{ic}}$. Nevertheless, several configurations achieved MRE < 2%. The empirical reduction resulted in $N_r = 750$ and $N_{\text{Neu}} = N_{\text{ic}} = 150$, which provided the lowest number of points while satisfying the accuracy criterion. With the reduced sampling density, the total number of optimization steps decreases to 3500, while the adaptive loss balancing factors λ_i are dynamically updated every 140 steps.

In addition, the mean training times for the different numbers of internal points are summarised in Table 1, as no noticeable differences were observed between the tested ratios of boundary and initial points to internal points for a given N_r . The training time is reduced from the original TT = 4.4 h to TT = 0.3 h for the resulting configuration. The mean inference time over 10,000 points was 1.478 ms. By comparison, the FEM models used to validate the PINN contained 3821 degrees of freedom (DOFs) plus 220 internal DOFs and required 8 s per calculation. Based on these computational costs, the break-even point of the PINN is

Table 1

Mean training time (TT) for different numbers of internal points (N_r).

N_r	Mean TT [h]
10,000	4.40
6000	3.49
3000	1.29
1500	0.65
750	0.30
300	0.13

reached after 135 temperature-field evaluations. This threshold is comparatively high: despite the advanced training strategy employed here, the upfront training effort is not recovered until well over a hundred FEM evaluations have been replaced. Accordingly, we regard the primary value of this study as methodological rather than as evidence that the PINN is a cost optimal surrogate. The batched parametric training clarifies how conventional feedforward PINNs can be trained effectively for parametric heat-transfer prediction. Where the dominant objective is to minimise the cost of many repeated evaluations, other reduced-order modelling approaches, such as POD/Galerkin methods, are likely better suited.

To assess the reliability of the model with $N_r = 750$ and $N_{\text{Neu}} = N_{\text{ic}} = 150$, the PINN was trained three times using the same number of collocation points. For each training, the maximum values of $|RE|_{\max}$ and MRE over all fibre orientations were selected for comparison, as these maxima can occur at different fibre orientations in each training. The results, summarised in Table 2, show a mean MRE of 1.319% and mean $|RE|_{\max}$ of 4.804%, with a range of 0.958% and 4.476% respectively, demonstrating the reliability of the model.

Table 2

Maximum values of the mean relative error (MRE) and maximum values of the maximum absolute relative error ($|RE|_{\max}$) for three independent training runs, along with their mean values and ranges.

Training	MRE [%]	$ RE _{\max}$ [%]
1	0.963	1.972
2	1.921	6.448
3	1.074	5.993
Mean	1.319	4.804
Range	0.958	4.476

As discussed previously, the training was performed using angles $\varphi_i \in [0, 180]^\circ$, since any fibre orientation outside this range can be substituted with an equivalent one within it. Although this relationship is commonly understood, the PINN does not have such knowledge. Providing an input angle outside the trained domain results in an erroneous temperature prediction, which also reflects a poor extrapolation capability of the PINN. Therefore, angles outside the training range must be converted to their equivalent within the range before being used as input to the PINN, which guarantees accurate and reliable predictions.

Training was conducted using a dimensionless thermal diffusivity formulation for a unidirectional material characterised by a thermal diffusivity ratio $\frac{\alpha_\perp}{\alpha_\parallel} = 0.5$, as this is the value exhibited by the ceramic matrix composite material that motivated this study. Since the PINN is trained using dimensionless parameters, the trained network can be applied to other anisotropic composite materials with the same thermal diffusivity ratio without requiring retraining. For such materials, the corresponding dimensional properties only need to be converted into the appropriate dimensionless form before being provided as input to the PINN, and the predicted dimensionless results can subsequently be transformed back into dimensional quantities using the corresponding dimensionalisation procedure. However, if the thermal diffusivity ratio of the material differs from 0.5 or if the boundary conditions are qualitatively modified, the PINN must be retrained for the new problem configuration. To demonstrate the applicability of the PINN to different levels of material anisotropy, three independent training runs were performed using the selected reduced collocation and boundary/initial point configuration for the thermal diffusivity ratios $\frac{\alpha_\perp}{\alpha_\parallel} = \{0.1, 0.5, 1.0\}$, representing a CFRP (Carbon Fibre-Reinforced Polymer), a CMC, and an isotropic material, respectively. The temperature predictions obtained with FEM and the PINN for an exemplary fibre orientation of 45° are shown in Fig. 15, while the maximum errors across all fibre orientations are summarised in Table 3. The results demonstrate that the PINN is able to keep the mean relative error below 2% across the different

Table 3

Maximum values of the mean relative error (MRE) and maximum values of the maximum absolute relative error ($|RE|_{\max}$) for different $\frac{\alpha_\perp}{\alpha_\parallel}$.

$\frac{\alpha_\perp}{\alpha_\parallel}$	MRE [%]	$ RE _{\max}$ [%]
0.1	1.952	6.229
0.5	0.744	3.961
1	1.244	3.486

anisotropy ratios considered. Some variation in prediction accuracy can be attributed to the random initialisation of the network weights and biases. Furthermore, cases with stronger anisotropy can result in steeper temperature gradients, which may require a higher sampling density to adequately resolve these gradients.

4. Conclusion

In this work, a PINN with a simple feedforward architecture was developed to obtain the parametric solution of the heat transfer equation for an anisotropic composite material by directly incorporating the fibre orientation as an input parameter.

Three training strategies were analysed and compared against FEM solutions generated using COMSOL Multiphysics, with the objective of identifying an effective training approach for solving the parametric heat transfer problem. The strategies *Training with random angles over the sample domain* and *Training with sequential sample domains* were both found to be unsuitable for the considered problem. The former did not reach the required accuracy and showed a bias towards the angle 90° , which corresponds to the mean of the training domain $[0, 180]^\circ$, reaching $|RE|_{\max}$ of over 16%. The latter exhibited catastrophic forgetting of previously learnt solutions, which limited its ability to generalise across

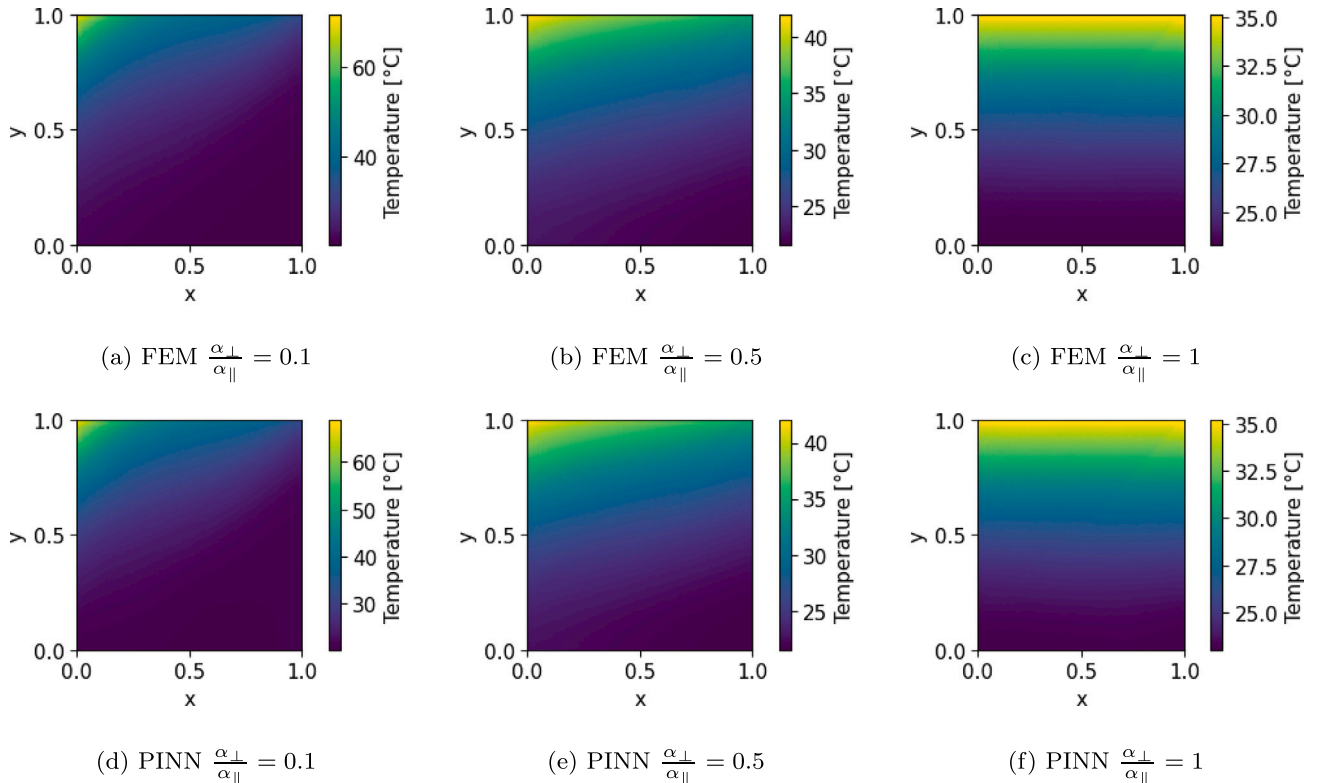


Fig. 15. Temperature distributions at an exemplary fibre orientation of 45° obtained using FEM (a–c) and PINN (d–f) for different thermal diffusivity ratios.

the investigated fibre orientations. Consequently, the $|RE|_{\max}$ across the different angles exceeded 30%, with strongly spatially correlated errors. The strategy *Training with batched sample domains* exhibited both high accuracy and robustness compared with the other training methodologies. It achieved a $|RE|_{\max}$ of approximately 6% and an MRE below 2% across the angles. Furthermore, no systematic bias was observed across the angles or within the spatial domain, as reflected by the low spatial covariance values of $0.171\%{}^2$ for 90° and $1.875\%{}^2$ for 0° . These results demonstrate the feasibility of parametrically learning the temperature distribution of an anisotropic material using a simple PINN for arbitrary fibre orientations. Subsequently, the number of collocation points was empirically reduced to 150 initial and boundary points and 750 internal points while maintaining the established error level. This reduced the training time to 0.3 h and yielded a mean inference time of 1.478 ms over 10,000 points. It should be noted that even this batched parametric strategy, designed to exploit the full potential of the PINN by amortising training across many orientations, only reaches break-even with FEM after more than 100 evaluations.

Ultimately, the PINN was trained and validated primarily for a specific geometry, boundary-condition configuration, and thermal diffusivity ratio. Within this defined problem setting, the proposed framework is applicable to arbitrary fibre orientations within the considered parameter range. The dimensionless formulation also enables the trained PINN to be applied to other anisotropic composite materials with the same thermal diffusivity ratio without modifying the trained network. For these materials, the corresponding physical properties are used to convert the dimensional input parameters into the dimensionless form required by the PINN before prediction, and the resulting dimensionless temperature fields are subsequently converted back into dimensional quantities through the appropriate dimensionalisation process. In addition to the considered reference ratio, the applicability of the approach to different levels of anisotropy was verified through three independent training runs with thermal diffusivity ratios $\alpha_\perp/\alpha_\parallel = \{0.1, 0.5, 1.0\}$, for which mean relative errors below 2% were achieved in all cases. Thus, the approach can be adapted to different material properties; however, changes in the thermal diffusivity ratio, geometry, or boundary conditions require retraining for the corresponding problem setting.

CRedit authorship contribution statement

Bernabe Lorenzo Avila: Writing – original draft, Software, Methodology, Investigation. **Dietmar Koch:** Writing – review & editing, Supervision, Funding acquisition. **Nils Meyer:** Writing – review & editing, Supervision, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

In the preparation of this work, the authors used GitHub Copilot and ChatGPT for the efficient programming of helper functions, as well as for rephrasing individual text sections. After using these tools, we have reviewed and modified the content as necessary and take full responsibility for the content of the published article.

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.

Acknowledgements

The research documented in this manuscript has been supported by the High-Tech Agenda Plus. We would like to thank the Bavarian State Ministry of Science and the Arts for funding the Augsburg AI Production Network as part of the High-Tech Agenda Plus.

Data availability

The code and necessary data required to reproduce the results presented in this study are openly available at the following GitHub repository: <https://github.com/lorenzo-bernabe/anisotropic-heat-transfer-pinns>.

References

- [1] R. Brandt, M. Frieß, G. Neuer, Thermal conductivity, specific heat capacity, and emissivity of ceramic matrix composites at high temperatures, *High Temp.-High Press.* 35/36 (2) (2003) 169–177, <https://doi.org/10.1068/htj105>
- [2] Z. Tu, J. Mao, H. Jiang, X. Han, Z. He, Numerical method for the thermal analysis of a ceramic matrix composite turbine vane considering the spatial variation of the anisotropic thermal conductivity, *Appl. Therm. Eng.* 127 (2017) 436–452, <https://doi.org/10.1016/j.applthermaleng.2017.07.204>
- [3] C. Zhao, Z. Tu, J. Mao, Investigation of the film-cooling performance of 2.5D braided ceramic matrix composite plates with preformed hole, *Aerospace* 8 (4) (2021) 116, <https://doi.org/10.3390/aerospace8040116>
- [4] H. Gao, L. Sun, J.-X. Wang, PhyGeoNet: physics-informed geometry-adaptive convolutional neural networks for solving parameterized steady-state PDEs on irregular domain, *J. Comput. Phys.* 428 (2021) 110079, <https://doi.org/10.1016/j.jcp.2020.110079>
- [5] R. Zhang, Y. Liu, H. Sun, Physics-guided convolutional neural network (PhyCNN) for data-driven seismic response modeling, *Eng. Struct.* 215 (2020) 110704, <https://doi.org/10.1016/j.engstruct.2020.110704>
- [6] L. Sun, J.-X. Wang, Physics-constrained Bayesian neural network for fluid flow reconstruction with sparse and noisy data, *Theor. Appl. Mech. Lett.* 10 (3) (2020) 161–169, <https://doi.org/10.1016/j.taml.2020.01.031>
- [7] Y. Zhu, N. Zabarar, P.-S. Koutsourelakis, P. Perdikaris, Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data, *J. Comput. Phys.* 394 (2019) 56–81, <https://doi.org/10.1016/j.jcp.2019.05.024>
- [8] L. Sun, H. Gao, S. Pan, J.-X. Wang, Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data, *Comput. Methods Appl. Mech. Eng.* 361 (2020) 112732, <https://doi.org/10.1016/j.cma.2019.112732>
- [9] A. Baydin, B. Pearlmutter, A. Radul, J. Siskind, Automatic differentiation in machine learning: a survey, *J. Mach. Learn. Res.* 18 (153) (2015) 1–43, <https://doi.org/10.48550/arXiv.1502.05767>
- [10] M. Raissi, P. Perdikaris, G. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* 378 (2019) 686–707, <https://doi.org/10.1016/j.jcp.2018.10.045>
- [11] M. Billah, A. Khan, J. Liu, P. Dutta, Physics-informed deep neural network for inverse heat transfer problems in materials, *Mater. Today Commun.* 35 (2023) 106336, <https://doi.org/10.1016/j.mtcomm.2023.106336>
- [12] Z. Xing, H. Cheng, J. Cheng, Deep learning method based on physics-informed neural network for 3D anisotropic steady-state heat conduction problems, *Mathematics* 11 (19) (2023) 4049, <https://doi.org/10.3390/math11194049>
- [13] B. Bowman, C. Oian, J. Kurz, T. Khan, E. Gil, N. Gamez, Physics-informed neural networks for the heat equation with source term under various boundary conditions, *Algorithms* 16 (9) (2023) 428, <https://doi.org/10.3390/a16090428>
- [14] R. Zhang, N. Xu, K. Zhang, L. Wang, G. Lu, A parametric physics-informed deep learning method for probabilistic design of thermal protection systems, *Energies* 16 (9) (2023) 3820, <https://doi.org/10.3390/en16093820>
- [15] R. Asl, Y. Yamazaki, K. Taghikhani, M. Muramatsu, M. Apel, S. Rezaei, A physics-informed meta-learning framework for the continuous solution of parametric PDEs on arbitrary geometries, preprint [arXiv:2504.02459](https://arxiv.org/abs/2504.02459), 2025, <https://doi.org/10.48550/arXiv.2504.02459>
- [16] M. Penwarden, A. Jagtap, S. Zhe, G. Karniadakis, R. Kirby, A unified scalable framework for causal sweeping strategies for physics-informed neural networks (PINNs) and their temporal decompositions, *J. Comput. Phys.* 493 (2023) 112464, <https://doi.org/10.1016/j.jcp.2023.112464>
- [17] Z. Hao, S. Liu, Y. Zhang, C. Ying, Y. Feng, H. Su, J. Zhu, Physics-informed machine learning: a survey on problems, methods and applications, 2022, <http://arxiv.org/pdf/2211.08064v2>, preprint.
- [18] O. Hennigh, S. Narasimhan, M. Nabian, A. Subramaniam, K. Tangsali, Z. Fang, M. Rietmann, W. Byeon, S. Choudhry, NVIDIA SimNet: an AI-accelerated multi-physics simulation framework, in: M. Paszynski, D. Kranzlmüller, V. Krzhizhanovskaya, J. Dongarra, P. Sloot (Eds.), *Computational Science – ICCS 2021*, vol. 12746, Springer International Publishing, Cham, 2021, pp. 447–461, https://doi.org/10.1007/978-3-030-77977-1_36 Lecture Notes in Computer Science.
- [19] M. Naderibeni, M. Reinders, L. Wu, D. Tax, Learning solutions of parametric navier-Stokes with physics-informed neural networks, preprint [arXiv:2402.03153](https://arxiv.org/abs/2402.03153), 2024, <https://doi.org/10.48550/arXiv.2402.03153>
- [20] S. Wang, H. Wang, P. Perdikaris, Learning the solution operator of parametric partial differential equations with physics-informed DeepONets, *Sci. Adv.* 7 (40) (2021) eabi8605, <https://doi.org/10.1126/sciadv.abi8605>
- [21] L. Lu, P. Jin, G. Karniadakis, DeepONet: learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators, *Nat. Mach. Intell.* 3 (3) (2021) 218–229, <https://doi.org/10.1038/s42256-021-00302-5>
- [22] Z. Li, H. Zheng, N. Kovachki, D. Jin, H. Chen, B. Liu, K. Azizzadenesheli, A. Anandkumar, Physics-informed neural operator for learning partial differential equations, *ACM/IMS J. Data Sci.* 1 (3) (2024) 1–27, <https://doi.org/10.1145/3648506>

- [23] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Y. Teh, M. Titterton (Eds.), Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, vol. 9 Proceedings of Machine Learning Research, PMLR, Chia Laguna Resort, Sardinia, Italy, 2010, pp. 249–256, <https://proceedings.mlr.press/v9/glorot10a.html>.
- [24] S. Wang, S. Sankaran, H. Wang, P. Perdikaris, An expert's guide to training physics-informed neural networks, preprint [arXiv:2308.08468](https://arxiv.org/abs/2308.08468), 2023, <https://doi.org/10.48550/arXiv.2308.08468>.
- [25] N. Demo, M. Strazzullo, G. Rozza, An extended physics-informed neural network for preliminary analysis of parametric optimal control problems, *Comput. Math. Appl.* 143 (2023) 383–396, <https://doi.org/10.1016/j.camwa.2023.05.004>.
- [26] J. Florido, H. Wang, A. Khan, P. Jimack, Investigating guiding information for adaptive collocation point sampling in PINNs, in: *Lecture Notes in Computer Science*, 14834, 2024, pp. 323–337, https://doi.org/10.1007/978-3-031-63759-9_36 Lecture Notes in Computer Science.
- [27] T. Yu, H. Zhu, Hyper-parameter optimization: a review of algorithms and applications, preprint [arXiv:2003.05689](https://arxiv.org/abs/2003.05689), 2020, <https://doi.org/10.48550/arXiv.2003.05689>.
- [28] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, A. Courville, On the spectral bias of neural networks, preprint [arXiv:1806.08734](https://arxiv.org/abs/1806.08734), 2018, <https://arxiv.org/pdf/1806.08734>.
- [29] 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, *Comput. Methods Appl. Mech. Eng.* 384 (1) (2021) 113938, <https://doi.org/10.1016/j.cma.2021.113938>.
- [30] S. Wang, X. Yu, P. Perdikaris, When and why PINNs fail to train: a neural tangent kernel perspective, *J. Comput. Phys.* 449 (2022) 110768, <https://doi.org/10.1016/j.jcp.2021.110768>.
- [31] S. Wang, S. Sankaran, P. Perdikaris, Respecting causality for training physics-informed neural networks, *Comput. Methods Appl. Mech. Eng.* 421 (2024) 116813, <https://doi.org/10.1016/j.cma.2024.116813>.