

International Journal of Hydromechatronics

ISSN online: 2515-0472 - ISSN print: 2515-0464

<https://www.inderscience.com/ijhm>

Surrogate models in machine learning for computational stochastic multi-scale modelling in composite materials design

Bokai Liu, Weizhuo Lu

DOI: [10.1504/IJHM.2022.10047396](https://doi.org/10.1504/IJHM.2022.10047396)

Article History:

Received: 26 January 2022

Accepted: 11 March 2022

Published online: 18 November 2022

Surrogate models in machine learning for computational stochastic multi-scale modelling in composite materials design

Bokai Liu* and Weizhuo Lu

Department of Applied Physics and Electronics,
Umeå University,
90187 Umeå, Sweden
Email: mrbokailiu@gmail.com
Email: weizhuo.lu@umu.se
*Corresponding author

Abstract: We propose a computational framework using surrogate models through five steps, which can systematically and comprehensively address a number of related stochastic multi-scale issues in composites design. We then used this framework to conduct an implementation in nano-composite. Uncertain input parameters at different scales are propagated within a bottom-up multi-scale framework. Representative volume elements in the context of finite element modelling (RVE-FEM) are used to finally obtain the homogenised thermal conductivity. The input parameters are selected by a top-down scanning method and subsequently are converted as uncertain inputs. Machine learning approaches are exploited for computational efficiency, where particle swarm optimisation (PSO) and ten-fold cross validation (CV) are employed for hyper-parameter tuning. Our machine learning prediction results agree well with published experimental data, which proves our computational framework can be a versatile and efficient method to design new complex nano-composites.

Keywords: surrogate models; data-driven modelling; DDM; machine learning; stochastic multi-scale modelling; polymeric nanotube composites; PNCs.

Reference to this paper should be made as follows: Liu, B. and Lu, W. (2022) 'Surrogate models in machine learning for computational stochastic multi-scale modelling in composite materials design', *Int. J. Hydromechatronics*, Vol. 5, No. 4, pp.336–365.

Biographical notes: Bokai Liu received his PhD (Dr-Ing) in 2022 from the Institute of Structural Mechanics (ISM), Bauhaus-Universität Weimar, Germany. He is currently working as a researcher in the Intelligent Human-Buildings Interactions Lab (IHBI), Department of Applied Physics and Electronics, Umeå University, Sweden. His research areas include data-driven modeling, machine learning, materials informatics, computational mechanics, multiscale stochastic modeling, and uncertainty analysis.

Weizhuo Lu received his PhD in 2008 from the Harbin Institute of Technology, China. He holds a Full Professor position at the Department of Applied Physics and Electronics, Umeå University, Sweden and the Director

of Intelligent Human-Buildings Interaction Lab (IHBI). His main research interests include building information modelling, occupant-centric energy efficient and big-data driven modelling.

1 Introduction

The excellent and outstanding mechanical, thermal, and electrical properties of carbon nanotube (CNT) make it an ideal filler to strengthen polymer materials' comparable properties (Budarapu et al., 2019a). Some extraordinary characters, such as high-performance load, lightweight design, excellent chemical resistance, easy processing, and heat transfer, make the design of polymeric nanotube composites (PNCs) a hot topic in recent research (Baughman et al., 2002; Budarapu et al., 2021; Kumar et al., 2019). Due to the reinforcing effects with different fillers on composite materials, compared with traditional materials, it has a higher degree of freedom and can be designed for the structure according to detailed applications' needs (Dusane et al., 2021).

The heat transfer of composite materials has very promising engineering applications in many fields, such as automobile engineering, aerospace engineering, electronic devices, and energy storage equipment. PNCs are used as heat sinks as a component of the structure to dissipate the heat generated under the current load (Siruvuri and Budarapu, 2020). There are many common carbon nanofillers, such as single-walled carbon nanotubes (SWCNT), multi-walled carbon nanotubes (MWCNT), carbon nanobuds (CNB), fullerenes, and graphene (Thostenson et al., 2001; Javvaji et al., 2018). SWCNT is widely used in various structures as the most commonly added filler among those proper candidates. SWCNT is a tubular structure formed by many carbon atoms with a diameter of about or less than 1 nm, which has an excellent thermal conductivity that can be regarded as an ideal additive to be designed into the polymer matrix.

Many studies have shown that the shape of the inclusions, the proportion of the additives in the composites, the fillers' thermal properties, and the interface layer between the fillers and the matrix have significant impacts on the thermal conductivity. These influencing factors have different performances at different scales and have different contributions to the overall structural materials. At present, many researchers have conducted systematic studies on these effects through theory and experimental methods (Francis et al., 2022; Siruvuri et al., 2021). However, the experimental approaches have imposed limitations, such as unavoidable randomness and many resource costs. There are many common carbon-based fillers such as SWCNTs, MWCNTs, CNBs, fullerene, and graphene (Thostenson et al., 2001). As already stated, the focus will be on PNCs. Since experiments are time-consuming, sometimes expensive and cannot shed light into phenomena taking place for instance at the interfaces/interphases of composites, they are often complemented through theoretical and computational analysis (Budarapu et al., 2018a; Ojo et al., 2017; Budarapu et al., 2019b; Sutrarakar et al., 2021).

Shokrieh and Rafiee (2010) conduct a hierarchical stochastic multi-scale modelling from nano to macro-scale to predict mechanical properties in carbon nanotube reinforced polymers (CNTRP) where random uncertainties are considered in all scales. Their study applies nano-scale continuum modelling at nano and micro scales and continuum modelling with finite element analysis (FEA) in RVE at meso and macroscopic scales.

Vu-Bac et al. (2014a, 2014b, 2015a) propose a stochastic multi-scale method to quantify the different input parameters influencing the mechanical properties of PNCs. Rafiee and Zehtabzadeh (2020) use stochastic multi-scale modelling with nano-scale continuum modelling and FEM to predict Young's modulus of graphene/polymer composites. Jeong et al. (2019) present a 3D stochastic computational homogenisation model for carbon fibre-reinforced (CFR) CNT/epoxy matrix composites, which can be a basis for hierarchical multi-scale material design of nanocomposite materials.

Although stochastic modelling can well reflect the inherent mechanism of materials related to thermal properties, the modelling is too complicated and time-consuming, significantly requiring computational resources and cost (Pinsky and Karlin, 2010). Thus, data-driven techniques, i.e., machine learning approaches, are applied to deal with these complex internal mapping relationships caused by extensive computation (Mortazavi et al., 2020, 2021). As a salient component of artificial intelligence, machine learning has become a versatile and full-featured modelling tool due to the powerful computing ability and application prospects (Alpaydin, 2020; Sharma et al., 2021).

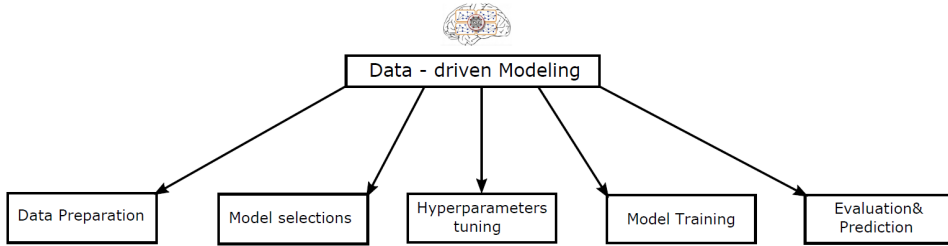
This method can be used in many practical ways. To be specific, Rong et al. (2019) conduct a data-driven technique, a deep learning tool – convolutional neural networks (CNNs) to predict the effective thermal conductivity of the particle-reinforced composite with image recognition approach. Goswami et al. (2020) propose a hybrid algorithm for solving brittle fracture problems, where physics informed neural network (PINN) is efficiently enhanced by transfer learning. Matos et al. (2019) present a mixed architecture wherein an artificial neural network (ANN) is applied to train the physical data at micro-scale. Subsequently, the electro-mechanical response at macro-scale can be predicted. Zhuang and Zhou (2019) provide an intelligent integrated method where the particle swarm optimisation (PSO) is an ideal optimiser for hyper-parameter tuning, which can vastly improve the predictive performance of crack closure percentage (CCP) in bacteria-based self-healing concrete. Mortazavi et al. (2020) apply machine-learning interatomic potentials (MLIPs) as an enhanced data-driven approach to enable first-principles multi-scale modelling via hierarchical employment of DFT/CMD/FEM simulations, which can efficiently predict the lattice thermal conductivity of graphene and borophene pristine phases. Guo et al. (2022) present a stochastic deep collocation method (DCM) based on neural architecture search (NAS) and transfer learning for heterogeneous porous media. In our previous works (Liu et al., 2020, 2021, 2022), we also applied surrogate models for data-driven modelling (DDM) to simplify the computational cost of stochastic multi-scale modelling.

In this manuscript, we present a computational framework based on surrogate models via machine learning approach accounting for uncertainties to assist stochastic multi-scale modelling. Then we implement an example under this framework to predict the macroscopic thermal conductivity of CNT-PCs based on numerous fine scale features including the Kapitza resistance. This article is organised as follows. In the next section, we describe the entire architecture of this five-steps computational framework via DDM. We introduce our implementation based on this overall framework in Section 3, which is a stochastic multi-scale modelling in thermal conduction of nano-composites. Subsequently, we discuss the results before we conclude our manuscript in Section 4.

2 Computational framework of surrogate models in machine learning method

In this section, we briefly introduce our proposed computational framework using surrogate models. Generally speaking, we can form a framework through five steps: data preparation, model selections, hyper-parameter tuning, model training, evaluation and prediction, shown as Figure 1. This five-step framework addresses a number of related problems, and each of the components is described later.

Figure 1 Five steps computational framework (see online version for colours)



2.1 Dataset preparation

The ‘raw data’ is obtained directly from physical and engineering models. It is divided into two sets: the training set, which accounts for 80% of the overall proportion, and the test set, which accounts for the remaining 20%. Subsequently, we normalise the data ensuring that the data of different attributes are at the same scale, which reduces the computational cost. After the dataset is collected, we also perform a test for missing data and outliers.

In data-driven technology, the establishment of datasets is significant. The choice of subsequent machine learning training algorithms and the strategy of hyper-parameter tuning all depend on the size of the dataset and the degree of data completeness.

2.2 Ensemble learning approaches

The tree-based model is non-parametric modelling, which can establish a connection between inputs and outputs by dividing the data multiple times. The structure of this model is illustrated in Figure 2, where it is mainly composed of one or more if-then judgments named divide-and-conquer strategy. Each judgment corresponds to an if-then, and those data can be divided more times. The following three significant elements are applied to tree-based modelling: the depth and complexity of the tree, the segmentation points, and the prediction equations at the final node.

The model’s composition structure is prone to insufficient fitting when running a regression model with a solitary regression tree. Therefore in this study, we employ two integrated tree-based models, namely random forest (RF) and gradient boost machine (GBM). Both methods are based on tree architecture, so the basic structure is similar, including leaf, nodes, and discrimination conditions. The only difference is the contribution ratio of the results obtained by different branches to the total result,

whether there are weights to consider. Besides, there are several newly developed robust gradient boosting algorithms, including XGBoost, LightGBM, and CatBoost (Chen and Guestrin, 2016; Ke et al., 2017; Dorogush et al., 2018).

Figure 2 The structure of regression tree (see online version for colours)

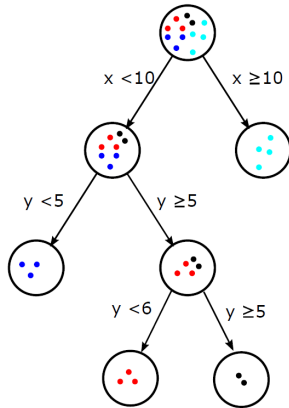
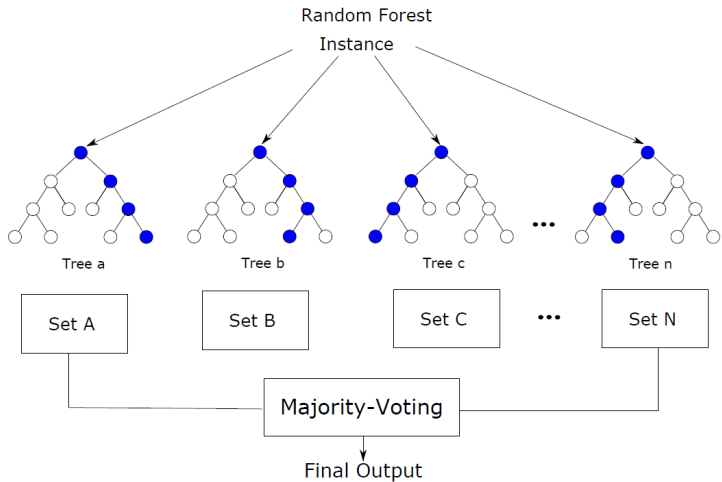


Figure 3 The architecture of RF (see online version for colours)



2.2.1 Random forest

RF is a variation of the bagging method, proposed by Breiman (2001). The architecture of this tree-based model is illustrated in Figure 3. The RF introduces randomness in constructing trees, thereby reducing the correlation between predictors. Let us assume the attribute set of the current node consists of d attributes. For each node of the basic regression tree in the RF, a subset of attributes containing k is randomly selected from the attribute set of the node, and then an optimal attribute is chosen from the sub-set for partitioning. This integrated approach adds some self-attribute disturbances based on the sample disturbance to increase the component learners' diversity, i.e., the

regression trees. Algorithm 1 presents the RF algorithm, which is distinct from the original regression tree in the way of choosing and separating attributes at the current node. The RF has the following advantages (Kuhn et al., 2013):

- 1 lower possibility of over-fitting
- 2 noise resistance in response variables
- 3 high efficiency
- 4 low sensitivity w.r.t parameter tuning.

Algorithm 1 RF algorithm

Require: The number of predictors k ;
The number of trees t ;
The number of split points s ;
Ensure: Predictive value $Y(x)$;
1: Choose the number of models m
2: **For** $t = 1, 2, \dots, m$ **do**
3: Generate a bootstrap sample from the original data
4: Train a tree model on this sample
5: **For** $s = 1, 2, \dots, m$ **do**
6: Randomly extract $k < d$ attributes as predictor
7: Choose the optimal variable among k attributes
8: **end for**
9: Tree model rule termination conditions take effect
10: **end for**
11: **return** $Y(x)$;

2.2.2 Gradient boosting machine

The boosting is an ensemble method based on a series of weak learners, through this compelling combination to improve the model's predictive performance, proposed by Friedman (2001). The implement of this integration requires the following steps: initially, a component learner is trained from the original training set, and the training samples are adjusted and refined based on this trained component. The adjustment of this sample pays more attention to the analysis of the wrong sample, that is, the wrong sample is assigned a more considerable weight. When this component is ready, the adjusted sample is entered into the next epoch of training. Repeat the above process, epoch after epoch until the preset convergence conditions are met, so all the trained trainers are integrated simultaneously. The above algorithm flow is illustrated by the pseudo code of Algorithm 2 as follows. From this algorithm, we can find that the most original principle is to try to find a cumulative model that can minimise the loss function's value under a determined loss function and given the decision tree model. In addition, the residual gradient also needs to be initialised by the predicted value that is most suitable for the predictors, and the loss function is minimised by modelling the residual.

Compared with the previous RF, GBM has certain similarities since both have the same integrated idea to improve the overall performance. In terms of specific details, the individual components in the RF are trained simultaneously and contribute equally to the complete model. However, GBM is inconsistent with others due to

chain relationships between the neighboring components. The late component’s result is highly related to the previous one, and the weight is assigned to each component during integration.

Algorithm 2 Gradient boosting machine

Require: The depth of trees D ;
 The number of iterate K ;

Ensure: Predictive value $Y(x)$

- 1: Calculate the mean of the response variable as the initial predicted value of each sample
- 2: **For** $k = 1, 2, \dots, K$ **do**
- 3: Calculate the residual between observed value and predictive value;
- 4: Use residuals as response variables to fit a tree with depth D ;
- 5: Use the previous steps to get the regression tree to predict each sample;
- 6: Update each sample and add the obtained in the previous step;
- 7: **end for**
- 8: **return** $Y(x)$;

Figure 4 The architecture of neural network

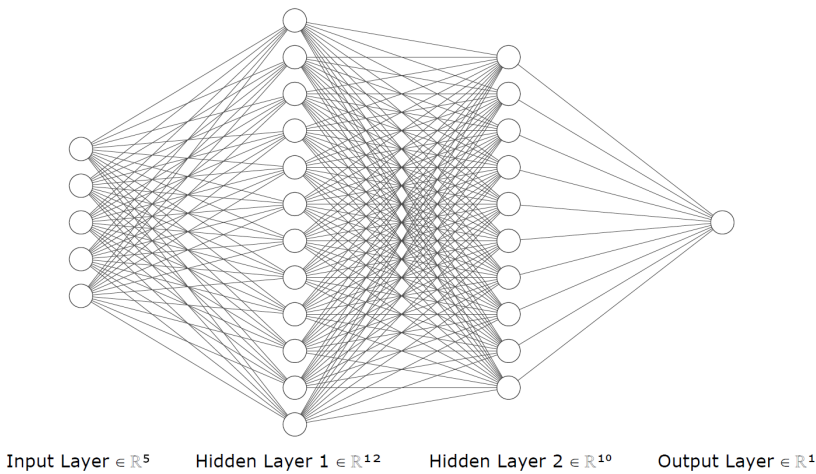
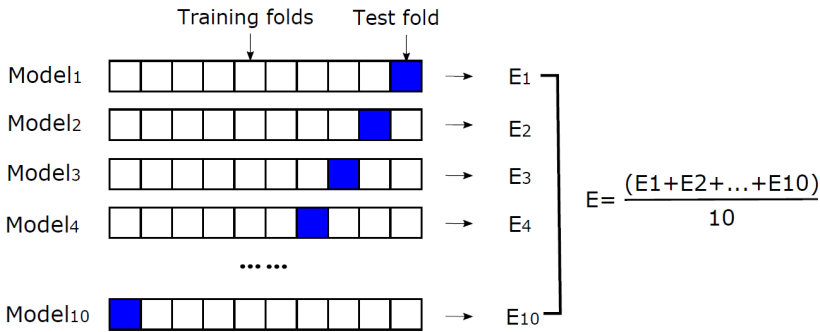


Figure 5 K-fold CV (see online version for colours)



2.3 Neural-network-based approach

ANNs are based on neurons and try to mimic the architecture of the human brain. We use a multi-layer neural network. The initial layer – or input layer – is connected to one or more hidden layers whose main function is to process data through the activation function between the layers up to the last neural layer, the output layer. Needless to say that there is no connection between neurons in the same layer. Figure 4 illustrated an example of a neural network with architecture of [5, 12, 10, 1].

We consider a network with l hidden layer, in which the 0^{th} layer is the input layer and the $(l + 1)^{\text{th}}$ is the output layer, respectively. The output at a layer l can be computed by:

$$Y_i^l = F_{l-1} \left(\sum_{j=1}^{m_{l-1}} (X_t^{l-1} \omega_{ij}^l) + b_i^l \right) \quad (1)$$

where Y_i^l is the weighted input into a i^{th} neuron on the l^{th} layer, W_{ij}^l is a weight and b_i^l a bias; F_{l-1} denotes the activation function in the l^{th} layer and m_{l-1} is the number of neurons in the $(l - 1)^{\text{th}}$ layer. We employ ReLU activation functions in both, input layer and hidden layer 1 and hidden layer 1 and hidden layer 2 given by:

$$F_{ReLU}(x) = \max(0, x) \quad (2)$$

The back propagation method (Algorithm 3) is applied, in which the initial weights and biases are self-adjusted after the computational error is minimised. The stochastic gradient descent (SGD) method is used in the training process (to minimise the loss function). The BP algorithm starts with neurons in the input layer, and then the data is transmitted layer by layer until the output layer. After the error between the target value and results generated by the output layer is calculated, the error is backpropagated to the hidden layers for comparison, adjusting the weights and biases. One round of this process is called an epoch, and then the epochs are repeated many times until a stopping condition is met. We take advantage of the TensorFlow r2.0 platform with Python/Rstudio using the RMSProp optimiser, an extension of the SGD and define an improved L_2 -loss root mean square error (RMSE) as the loss function.

Algorithm 3 Back propagation method

Require: The training set $D = (\mathbf{x}_k, \mathbf{y}_k)_{k=1}^m$;
The learning rate η ;

Ensure: Minimise the cumulative error on D

- 1: Randomly initialise weights and deviations in the range of (0, 1)
 - 2: **For all** $(\mathbf{x}_k, \mathbf{y}_k) \in D$ **do**
 - 3: Calculate the output of the current sample $\hat{\mathbf{y}}_k$;
 - 4: Calculate the gradient of the neurons in the output layer g_j ;
 - 5: Calculate the gradient of hidden layer neurons e_h ;
 - 6: Update connection weight w_{hj} , v_{ih} and bias θ_j , γ_h
 - 7: **end for**
 - 8: **return** Neural network \mathbf{Y} with determined weights and biases
-

2.4 Cross validation and hyper-parameters tuning

Cross validation (CV) is commonly used in machine learning method to estimate hyper-parameters and build integral models. CV reuses data, dividing the existing data into a training set and a testing or validation set. By putting different data segments into different sets, all data can be effectively employed in the data learning to avoid over-fitting the data in the training process. Among the most common CV approaches are the holdout C-V, K-fold C-V, and leave-one-out C-V (Kohavi et al., 1995). Since the computational cost of LOOCV is high and the holdout CV's accuracy is insufficient, we will use K-fold CV in our study. K-fold CV separates the original dataset into two parts, a training set and a testing set. Both are divided into several segments. In each training process, the testing set does not participate in the training, but it is only used to verify the models' performance and determine whether there is insufficient fitting. If the test set's model performance is much smaller than the training set, it indicates over-fitting of the model, otherwise an under-fitting.

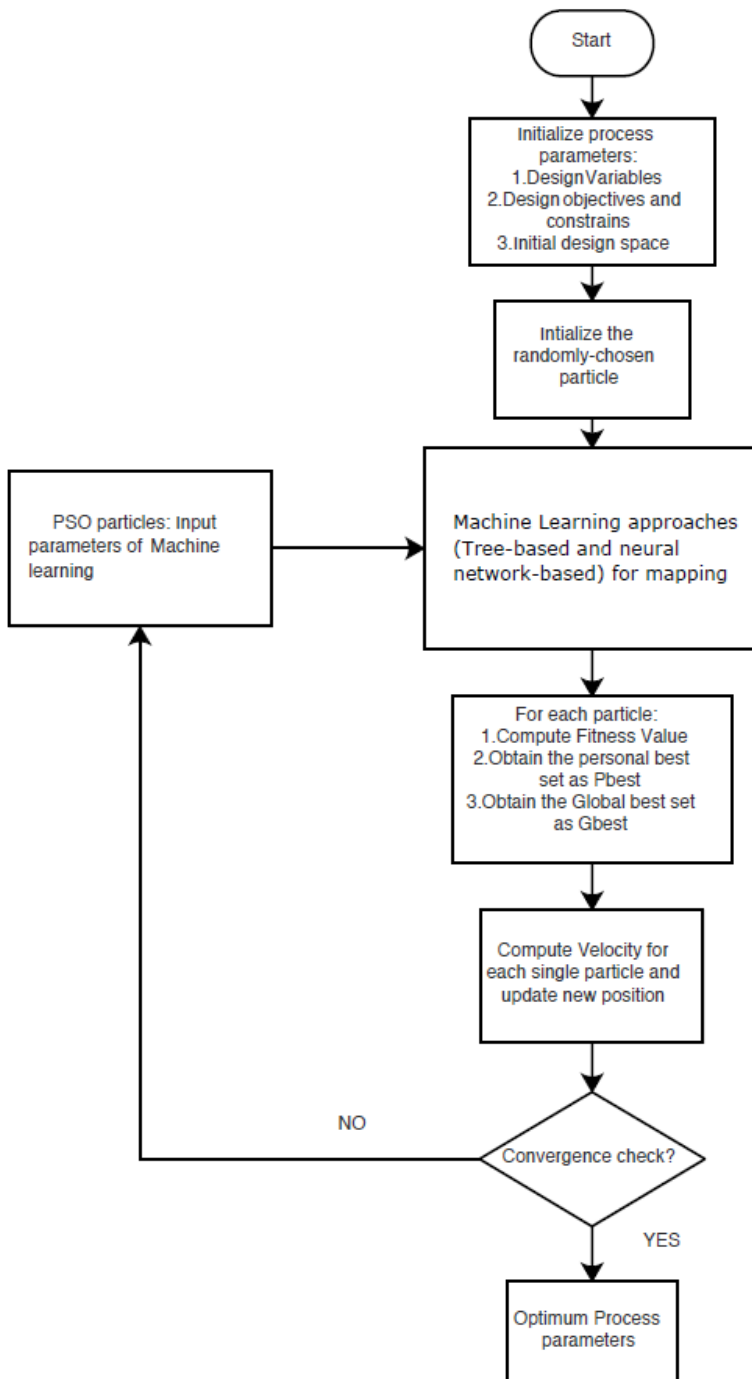
Figure 5 illustrates how K-fold CV separates the initial data into K groups (K-Fold) and generates a validation set while the remaining K-1 group subsets are used as the training set during the learning procedure. The obtained K models assess the outcome in the validation sets before the final indicators like RMSE and R^2 are averaged to construct the K-fold CV indicators. Through the use of limited data, each dataset can be a test set, which maximises the diversity of the current data. K-fold CV effectively reduces the variance by averaging results and easily meets the requirement for data division selection. It is suitable for applications with 'medium' amount of data. The K's value is flexible enough and only determined by the complexity of the available data. When the training set has enough data, it is commonly set to 5 in order to reduce the extra training time; Otherwise, it is set to 10 increasing the accuracy in the results when the dataset is not large (Kuhn et al., 2013). In our study, we use a ten-fold CV according to our original training data.

A 'proper' selection of hyper-parameters is crucial for the performance of machine learning methods. In order to avoid over-fitting and obtain a better prediction result, hyper-parameters need to be determined before constructing the machine learning model. Both, the selection of hyper-parameters and the determination of interval values, are commonly based on experience and trial and error approaches. A more 'systematic' way is to apply optimisation – such as PSO – to tune the hyper-parameters. The PSO algorithm, which has been proven successful also in our previous studies (Liu et al., 2021), is shown in Algorithm 4. The SSE is chosen as fitness function from the 10-fold CV; it is minimised continuously during the PSO process. It is given by

$$SSE = \frac{1}{N} \sum_{i=1}^N (Y_{ri} - Y_{pi})^2 \quad (3)$$

where Y_{ri} and Y_{pi} are the required and predicted i^{th} output parameters, respectively; N is the number of output parameters. A swarm size of 450 is chosen, ω , c_1 and c_2 are 1 and (2.0, 2.0), respectively (Liu et al., 2021). A flowchart illustrating the hyper-parameters tuning process can be found in Figure 6.

Figure 6 Flowchart of PSO algorithm in hyper-parameters tuning



Algorithm 4 Particle swarm optimisation

Require: i, V_i, X_i, N
Ensure: $gBest$

for each particle i **do**
 Initialise velocity V_i and position X_i for particle i ;
 Evaluate particle i and set the $pBest_i = X_i$;
end for
 $gBest = \min(pBest_i)$;
while Not Stop Running **do**
 for $i = 1$ to N **do**
 Update the velocity and position of particle i ;
 Evaluate particle i ;
 if $\text{fit}(X_i) < pBest_i$ **then**
 $pBest_i = X_i$;
 end if
 if $pBest_i < \text{fit}(gBest)$ **then**
 $gBest = pBest_i$;
 end if
 end for
end while
Print $gBest$;

2.5 Evaluation and prediction

The accuracy and predictability is measured by standard metrics such as the coefficient of determination (R^2) and the RMSE, which is a function of the model's residual. The RMSE can also be regarded as the 'average distance' between the predicted and observed values. The RMSE and R^2 values can be computed by

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (Y_{ri} - Y_{pi})^2} \quad (4)$$

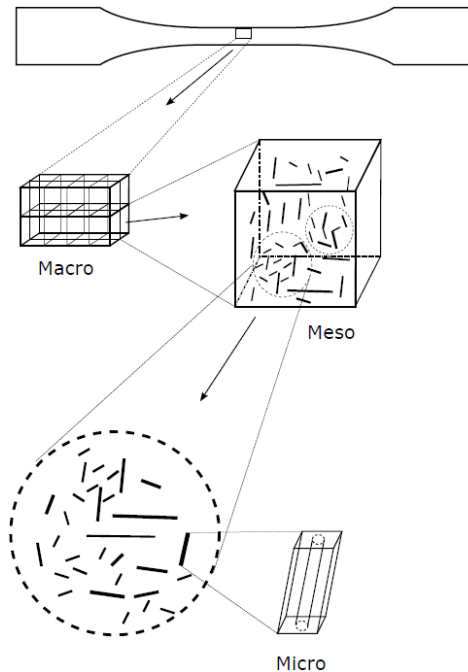
$$R^2 = 1 - \frac{\sum_{j=1}^N (Y_{ri} - Y_{pi})^2}{\sum_{j=1}^N (Y_{ri} - Y_{mean})^2} \quad (5)$$

3 Implementation: stochastic multi-scale modelling in composites materials

In this section, we briefly introduce the implementation of a five-step computational framework to design and predict composite materials. We present our stochastic multi-scale modelling in detail and apply it for a specific PNC built by using chemical vapour deposition (CVD). The epoxy resin is cycloaliphatic polyamine (Araldite LY564/Aradur HY2954), and the catalyst is zeolite-supported cobalt and iron (HSZ-390HUA) (Maruyama et al., 2002; Harish et al., 2012; Vavouliotis et al., 2010; Fiamegkou et al., 2014).

Our modelling strategy is hierarchical multi-scale model bridging different length scales, the micro-scale to the macro-scale, see Figure 7. A bottom-up approach then is employed to transfer ‘information’ through the length scales accounting for uncertainties. In this hierarchical framework, the output of the finer scale is the input of the next coarser scale (Korupolu et al., 2002). More details about materials model can be observed on our precious work (Liu et al., 2020, 2021).

Figure 7 Multi-scale modelling scheme



3.1 Multi-scale modelling

The models for PNCs at different length scales are based on representative volume elements (RVEs) and can be found in Figure 7. For each length scale, we first need to identify a suitable RVE size to subsequently upscale the ‘relevant’ material parameters to the next higher length scale (Yang et al., 2015; Budarapu et al., 2017). All effective parameters are summarised in Table 1.

We employ a bottom-up approach, i.e., a hierarchical multi-scale method where information is transferred only from the fine scale to the next coarser scale (Budarapu et al., 2018b, 2014). Figure 8 shows the associated flowchart. The models at different length scales will be described subsequently.

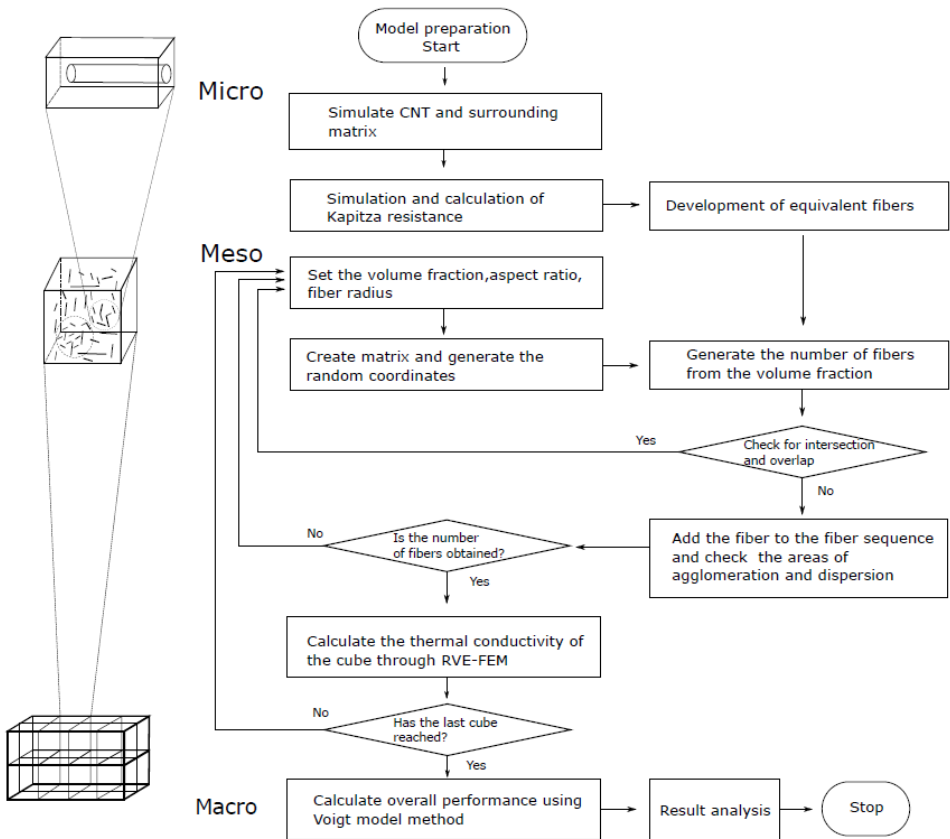
A continuum model is employed at the meso-scale, which consists of the equivalent fibres embedded in the polymer matrix. A Python script to generate the microstructure of the RVE can be downloaded from <https://github.com/jackylbk/Generation-Algorithm-of-PNCs.git>. The algorithm has been described in detail elsewhere (Liu et al., 2020, 2021; Mortazavi et al., 2013c; He et al., 2016; Mortazavi

et al., 2013a, 2013b). It avoids overlapping of the equivalent fibres and ensures the imposition of periodicity. The commercial FE package ABAQUS is used in all simulations. A typical discretisation of the RVE with quadratic tetrahedra elements is illustrated in Figure 11.

Table 1 Relational effective parameters for each scale

<i>Effective scale</i>	<i>Effective length</i>	<i>Parameters</i>
Micro	nm and μm	Interface resistance between CNT and matrix, thermal conductivity of epoxy
Meso	μm	Thermal conductivity of fibre, aspect ratio, agglomeration, dispersion
Macro	mm	Volume fraction, homogeneous thermal properties

Figure 8 Multi-scale modelling scheme



After the corresponding fibre generation is completed, we also need to consider fibre agglomeration and dispersion for some RVE models with high aspect ratio and high volume fraction. In terms of agglomeration and dispersion, we use the

two-parameter method, which is a simple and widely used approach to quantify the degree of agglomeration. Let us simplify the agglomeration model first, in which using a simplified sphere to represent the gathered CNT area. Figure 9 illustrates this spherical zone that is considered as ‘inclusions’. We define the total volume of CNTs is V_{CNT} , and it can be classified into two parts, namely $V_{CNT}^{inclusion}$ and V_{CNT}^{matrix} .

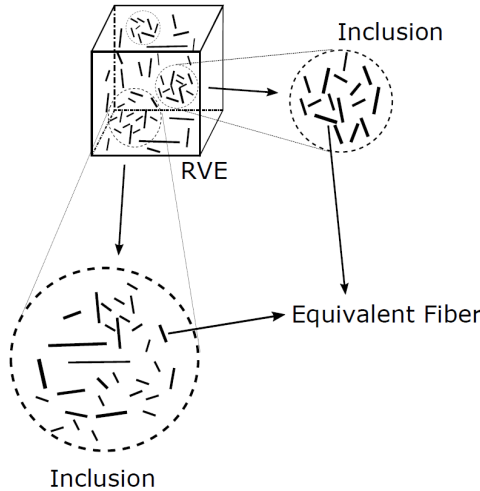
$$V_{CNT} = V_{CNT}^{inclusion} + V_{CNT}^{matrix} \quad (6)$$

where the $V_{CNT}^{inclusion}$ and V_{CNT}^{matrix} indicate the CNTs are located in the inclusions and matrix, respectively. Based on this definition, we introduce two parameters, which are agglomeration index ξ and dispersion index ζ . They are shown as:

$$\xi = \frac{V_{inclusion}}{V}, \quad \zeta = \frac{V_{CNT}^{inclusion}}{V_{CNT}} \quad (7)$$

where $V_{inclusion}$ is the entire spherical volume of inclusions in the RVE volume V . ξ indicates the volume fraction of inclusions w.r.t the entire cubic volume of RVE as well as ζ shows the inner volume fraction – that is, the volume of CNTs in inclusion w.r.t the total volume of CNTs. Usually, when $\xi = \zeta$, we define that there is no agglomeration, CNTs are uniformly distributed in the cube. When $\xi > \zeta$, it shows that CNTs are unevenly located in various positions in the inner RVE.

Figure 9 The agglomeration and dispersion in the cubic RVE



We use the selected cubes to maximise the smallest possible volume to contain more material information, including micro-scale CNTs and the surrounding matrix. Hence, a RVE of suitable size needs to be determined. We use the sample enlargement method (SEM) to specific this appropriate size, so the initial model should be relatively small enough to evaluate the material properties of interest, i.e., thermal conductivity in this case. Figure 10 indicates the SEM method, where we can find the size is increasing if the condition is not met. That means we enlarge the size of RVE continuously until the predictive property converges into a nearly stable value. Here we should note that our results require an average ensemble method because of statistical scatter:

$$\langle R \rangle = \frac{1}{M} \sum_{K=1}^M R^{(K)} \quad (8)$$

where $R^{(k)}$ is the current value in the k^{th} RVE, and M is the total members of the entire RVEs. After the ensemble is averaged, a convergence criterion must be satisfied subsequently, which states that the predicted value is allowed to meet convergence when it is less than a tolerance value:

$$\left| \frac{\langle R^{(K+1)} \rangle - \langle R^{(K)} \rangle}{\langle R^{(K)} \rangle} \right| < Tol = 1\% \quad (9)$$

where $R^{(k)}$ is the current value in the k^{th} RVE, and $R^{(k+1)}$ is the following $k + 1^{\text{th}}$ RVE.

Figure 10 Sample enlargement method

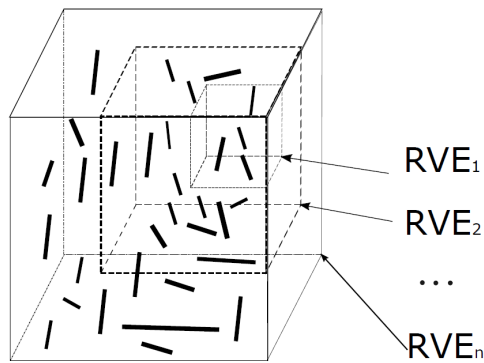
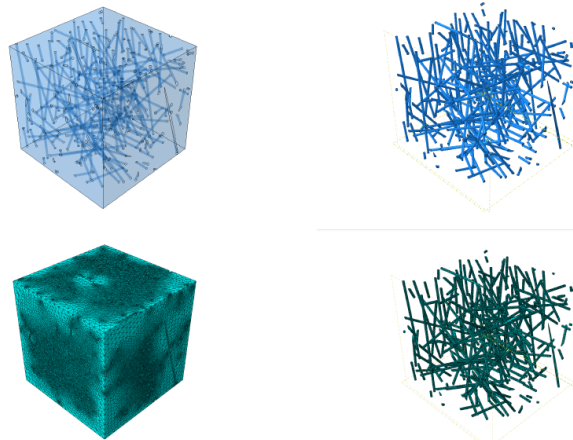


Figure 11 3D cubic RVE meshing (see online version for colours)



After the generation of RVE at meso-scale, we prepare to apply the heat load. The governing equation of the underlying problem is the constitutive equation, balance equation, and boundary conditions. Our heat transfer problem is steady conduction

wherein the classical formulation of the general governing equations for the linear thermal problem can be employed on a domain Ω :

$$C_f \frac{\partial \theta}{\partial t} + \nabla \cdot \mathbf{q} - Q = 0 \quad (10)$$

where C_f is the heat capacity. Q is the heat source in our simulation, and \mathbf{q} is the heat flux vector with the unit of (J/m^2s) . θ is the absolute temperature, and ∇ is an operator to show the divergence of \mathbf{q} . According to quasi-static theorem, the first item $C_f \frac{\partial \theta}{\partial t}$ is 0.

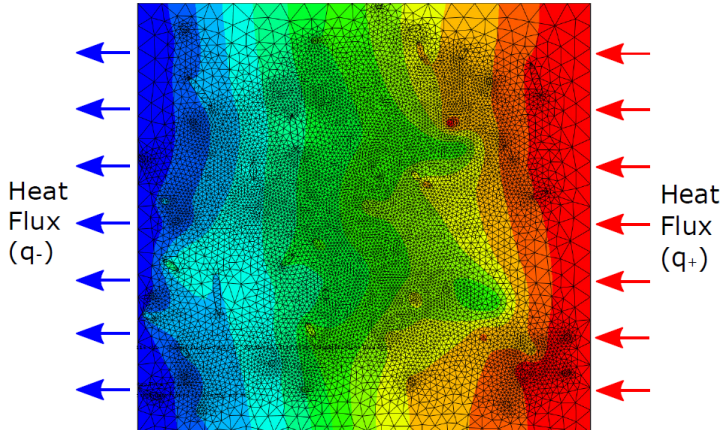
We use the direct method to calculate the thermal conductivity, by establishing two different heat flux at both ends of the RVE, that is, one surface applies a heat flux $+q$, and the opposite surface applies other heat flux $-q$, respectively as Figure 12 shown. It will generate a continuous heat flow through the cube and produce a different temperature distribution to meet the energy redistribution. The generated temperature gradient can be brought into Fourier's law to obtain the thermal conductivity, which is given as:

$$\mathbf{q} = -\kappa \nabla T \quad (11)$$

$$\text{with } \kappa = \begin{Bmatrix} \kappa_{xx} & 0 & 0 \\ 0 & \kappa_{yy} & 0 \\ 0 & 0 & \kappa_{zz} \end{Bmatrix} \quad (12)$$

where κ is the conductivity in composite. We assumed isotropy with $\kappa_{xx} = \kappa_{yy} = \kappa_{zz}$, which can be confirmed by applying boundary conditions at different RVE edges. Our priority is to obtain effective thermal conductivity (ETC), so each direction's value needs to be averaged.

Figure 12 Applying heat flux in both sides (see online version for colours)



The Fourier law is inserted into the governing equation. The formulations can be shown as follows:

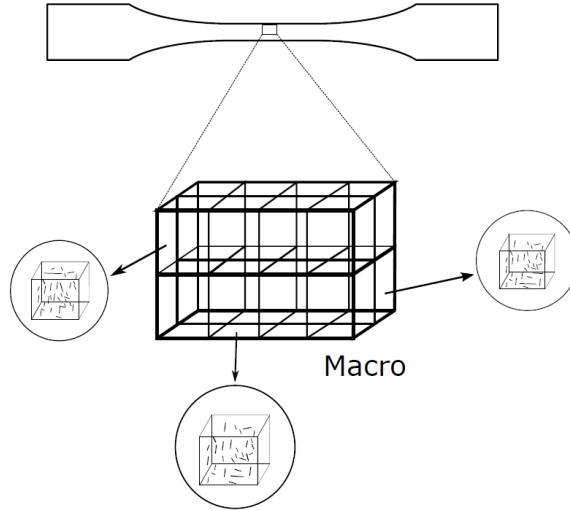
$$\text{div}(\kappa \nabla \theta) + Q = 0 \text{ in } \Omega \quad (13)$$

$$q_n|_{\Gamma_q} = \mathbf{q} \cdot \mathbf{n}|_{\Gamma_q} = \bar{q}_n \text{ on } \Gamma_q \quad (14)$$

$$q_n = -\mathbf{q} \cdot \mathbf{n} = \bar{q} \text{ on } \Gamma_q \quad (15)$$

where \mathbf{n} we call it normal vector, \bar{q} is flux value on the boundary Γ_q .

Figure 13 The material region in macro-scale modelling



We need to discrete the above formulation, so the weak form is applied for this FEM. The weak form for the steady-state solution in the Ω can be designed as: find $\theta \in \nu$ such that:

$$\int_{\Omega} \kappa \nabla \theta \cdot \nabla \delta \theta d\Omega = - \int_{\Gamma_q} \delta \theta \bar{q} d\Gamma - \int_{\Gamma_{\theta}} \delta \theta q_n d\Gamma + \int \delta \theta Q d\Omega \quad \forall \delta \theta \in \nu_0 \quad (16)$$

From a macroscopic perspective, CNT reinforced materials are composed of small mesoscopic cubes with different thermal properties. The distribution of this model simulates the material as heterogeneous due to uncertainty. Each small cube RVE results from the mesoscopic modelling and is used as an input parameter in the macro-scale. Therefore, ‘cubes’ with different thermal properties extracting from simulations at the meso-scale are randomly distributed in the macroscale, see Figure 13. Though it is principally possible to use FEM at the macro-scale, we employ the rule of mixtures for computational efficiency, which is given by

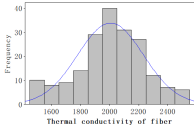
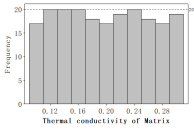
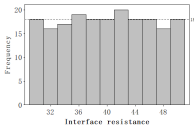
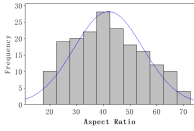
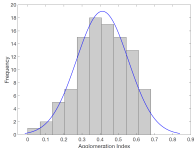
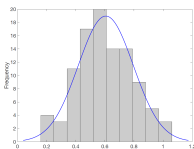
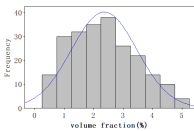
$$\bar{X} = \frac{\sum_i X_i P_i}{\sum_i P_i} \quad (17)$$

where X_i is the thermal properties of the i^{th} RVE cube in this case. P_i as a weighting factor is expressed in volume fraction. This model considers that the inhomogeneous distribution of SWCNTs inside the CNT reinforcement material makes the volume fractions in different cubes inconsistent. We can observe that this form of conversion can reasonably consider the uncertainty of volume fraction, agglomeration, and dispersion caused by the irregular distribution of CNTs in macroscopic modelling.

3.2 Stochastic modelling

We account for uncertainties at different scales, i.e., uncertainties in the material including the interfaces between phases, the micro-structure and uneven distribution of the filler inside the material, see Table 2. The boundary conditions have been assumed as deterministic.

Table 2 Model uncertainties (see online version for colours)

Scale	Inputs	Mean	Standard deviation	Type of distribution	Sources
Micro	Thermal conductivity of fibre	2006	227.2		Ghosh et al. (2010)
	Thermal conductivity of matrix	0.20	0.013		Moisala et al. (2006)
	Interface resistance	35	8.66		Freitag et al. (2009)
Meso	Aspect ratio	58	1.8		Bui et al. (2011)
	Agglomeration index	0.4	0.15		Vu-Bac et al. (2015b)
	Dispersion index	0.58	0.18		Vu-Bac et al. (2015b)
Macro	Volume fraction	0.0235	0.01		Shokrieh and Rafiee (2010)

After defining a probability density function (PDF) including variance and mean value of all uncertain input parameters, we use Latin hypercube sampling (LHS) to design experiments for specific stochastic modelling (Iman and Conover, 1980). This method can determine the mean, standard deviation, and variance of the actual physical model's output parameter through the PDF of the various input parameters. During the implementation of LHS, we can obtain a $N \times m$ design matrix, where m is the input parameters and N is the number of intervals where the cumulative probability curve is equally divided. Subsequently, we map the obtained design matrix to the real physical model to calculate the actual target output value (Novák et al., 1998). LHS optimises the sampling cost by randomly generating multiple input parameters, which can greatly reduce the calculation cost and reduce the sampling time compared with Monte Carlo sampling (MCS) (Keitel et al., 2012). The surrogate models are applied for the uncertainty analysis to largely reduce the computational cost.

4 Numerical results and discussion

The model output is the macroscopic thermal conductivity of the composite. The input parameters are the thermal conductivity of matrix, fibre and interface (Kapitza resistance), the aspect ratio, the agglomeration index, the dispersion index and volume fraction.

4.1 Multi-scale modelling results

The output parameter at the micro-scale is the interface resistance. The outcomes at meso and macro scales are homogenised thermal conductivity.

Table 3 The distribution of properties in material region

Numbers	V_f (%)	Thermal conductivity (W/mk)	Numbers	V_f (%)	Thermal conductivity (W/mk)
1	0.99	0.314063	9	3.85	0.735386
2	1.32	0.219208	10	0.95	0.375841
3	3.60	0.474656	11	1.08	0.484727
4	2.86	0.558519	12	2.07	0.548841
5	2.14	1.001029	13	4.23	0.487791
6	3.68	0.407369	14	2.63	0.423529
7	4.61	0.968785	15	1.99	0.433960
8	3.31	0.483864	16	4.14	0.794539

Notes: Thermal conductivity in Voigt model: 0.59363663 (W/mk) (volume fraction: 2.71%).

Figures 14 and 15 show the temperature distribution inside the RVE. Figure 16 shows the predicted macroscopic thermal conductivity w.r.t the RVE size, which converges to a stationary size at some point.

Figure 17 shows a principal macroscopic model. The thermal conductivity of each cube with different V_f and aspect ratio extracted from the meso-scale and presented in Table 3 yields a final thermal conductivity of 0.59363663 (W/mk). This value is compared to values in the literature in Table 4, in which the FEM-RVE values are obtained through stochastic multi-scale modelling.

Figure 14 Temperature distribution of composites within RVEs (see online version for colours)

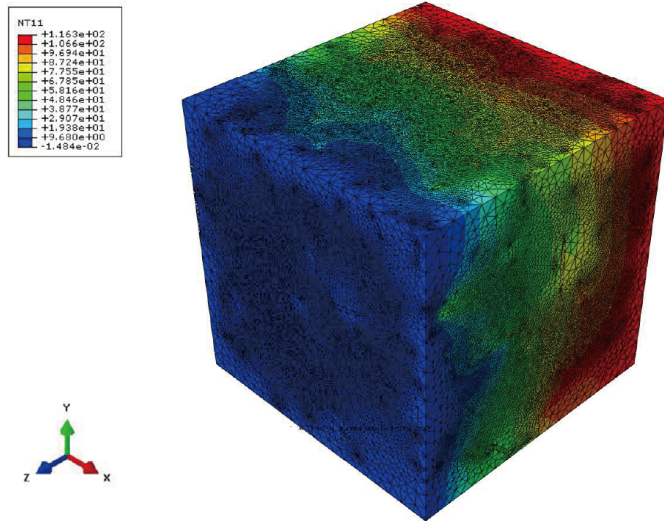
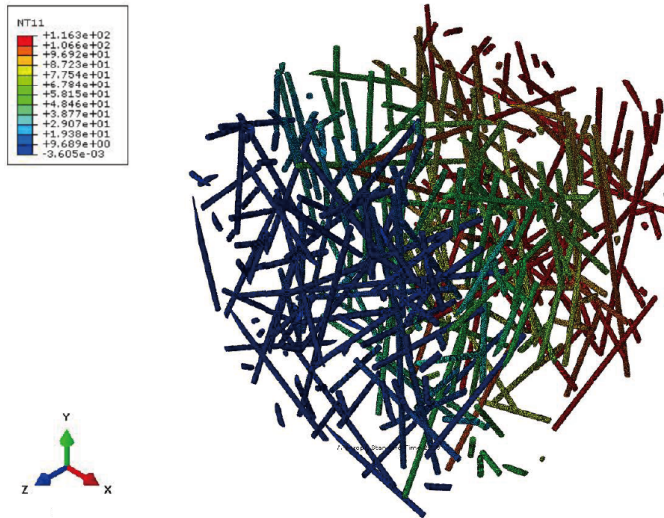


Figure 15 Temperature distribution of inner fibres within RVEs (see online version for colours)



4.2 Machine learning modelling results

We first focus on the PSO hyper-parameter tuning. There is an overview of the selected hyper-parameters summarised in Table 5. Besides, Table 6 illustrates the results of all models, in which Deep neural network with two hidden layers yields the best performance among all the machine learning models though all of them are reliable enough to predict the thermal conductivity performance of composites.

Table 4 The comparison between experimental value and FEM-RVE predictive value

<i>Literature</i>	κ_m	<i>Materials properties</i>	<i>Thermal conductivity (W/mk)</i>			
			V_f	<i>Experiment value</i>	<i>RVE-FEM value</i>	<i>Percent error</i>
Yang et al. (2010)	Diglycidyl ether of bisphenol A (0.13 W/mk)	1%		0.2375	0.2192	7.70%
		3%		0.3506	0.3189	9.04%
		5%		0.4479	0.4238	5.36%
Kapadia et al. (2014)	Ethylene terpolymer (RET: Elvaloy 4170 0.32 W/mk)	0.5%		0.35	0.3214	8.17%
		1%		0.3619	0.3683	1.73%
		1.5%		0.3496	0.3633	3.77%
		2%		0.3616	0.3938	8.17%
		2.5%		0.3956	0.4291	7.80%
		3.4%		0.4015	0.4091	1.85%

Table 5 Hyper-parameters tuning

<i>ML method</i>	<i>Hyper-parameters</i>	<i>Definition</i>	<i>Interval</i>	<i>Optimum value</i>
Random forest	C_p	Complexity parameter	[0, 1]	0
	Min-sample-split	The minimum samples for split	[2, 5]	2
	M	The maximum depth of tree	[1, 10]	5
	Max-DT regression tree	The maximum numbers of	[1, 6,000]	4,500
Gradient boosting machine	λ	The learning rate	[0.001, 0.99]	0.01
	N	The maximum number of regression tree	[100, 6,000]	4,500
	D_{ia}	Interaction depth	[1, 10]	4
Artificial neural network	N_{hn}	The number of neurons in hidden layer	[1–100]	26
	λ	The learning rate	[0.001–1]	0.01
Deep neural network (2 hidden layers)	N_{hn}	The number of neurons in hidden layer	[1–100]	[64, 64]
	λ	The learning rate	[0.001–1]	0.01
	$Epoch_{max}$	The maximum epoch	[1–1,0000]	2,000

Table 6 Predictive performance in ML models (see online version for colours)

<i>ML method</i>	<i>Training</i>		<i>Test</i>	
	R^2	$RMSE$	R^2	$RMSE$
Random forest	0.8912	0.1742142	0.8431	0.1429053
Gradient boosting machine	0.9231	0.0918721	0.9012	0.1145233
Artificial neural network	0.9253	0.0832968	0.8187	0.1646913
Deep neural network	0.9998	0.0678237	0.9319	0.0579143

Figure 18 shows the convergence on training and validation datasets for deep neural network models. Figure 19 illustrates the scatter plot of the RF in test set; the majority of the predictions are distributed on the $Y = T$ line. The points above the line indicate

over-predicted data. Figure 20 shows the predictive performance of the GBM while Figures 21 and 22 refer to the results obtained by the neural network with a single hidden layer (ANN) and the deep neural network with two hidden layers (DNN), respectively. DNN expectedly outperforms not only the ANN but all other ML models which for instance the R^2 value suggests. The predicted performance is more accurate when the thermal conductivity is below 0.6 since the training set's data distribution is more concentrated in the interval [0–0.6].

Figure 16 Thermal conductivity versus RVE size (see online version for colours)

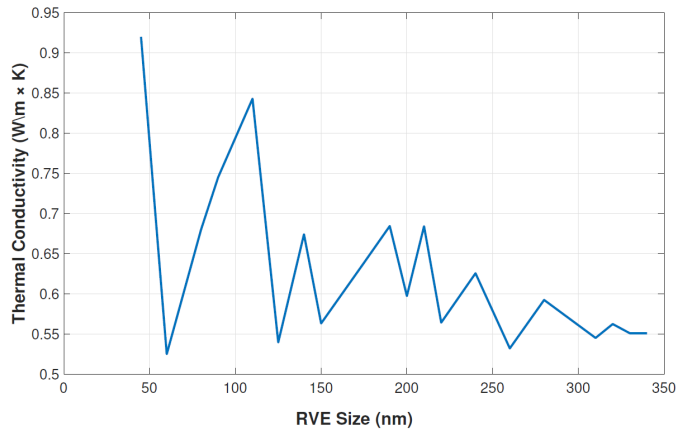


Figure 17 Macro-scale modelling of material region

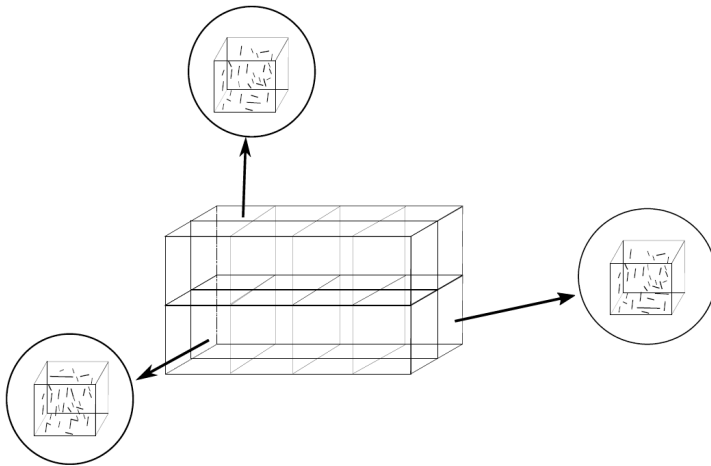


Table 7 indicates the computational cost of different ML approaches. The RF and GBM are the least costly ones while the DNN is computationally expensive. However, compared to the computational cost of the stochastic multi-scale model, all ML models are more efficient. For example, the multi-scale model simulation in Table 3 takes 80.5 hours on the cluster with 16 cores at 2 sockets (Intel Xeon e5-2650 v2) – (32 cores with multi-threading enabled) in simulation at micro and meso-scales. It also takes 71.2 hours with the macroscopic 2.71% volume fraction in continuum mechanics modelling on the

cluster with the same computational cores. This sample runs for a total of 151.7 hours on those scales. On the contrary, the training times of machine learning models are all less than an hour on the same computing configuration. As the number of samples increases, the training time will also grow. Figure 23 displays how the computational time increases with enlargement of sample size. RF is apparently least sensitive and due to its highest complexity, the DNN has the most significant increase in time.

Figure 18 The loss function w.r.t epoch in training with TensorFlow (see online version for colours)

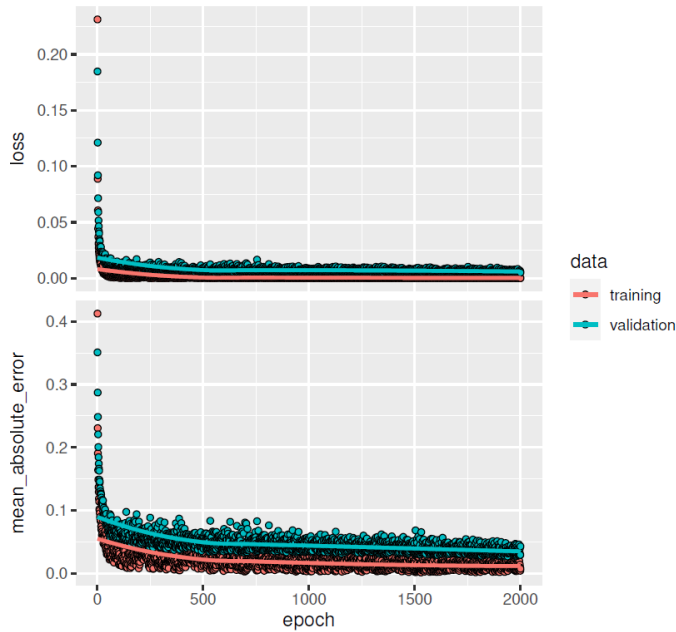


Figure 19 The predictive performance of RF in test set (see online version for colours)

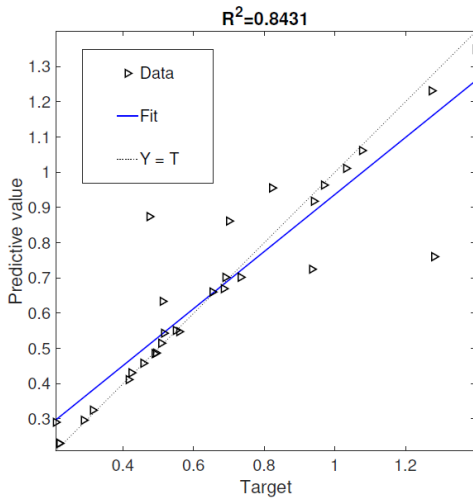


Figure 20 The predictive performance of GBM in test set (see online version for colours)

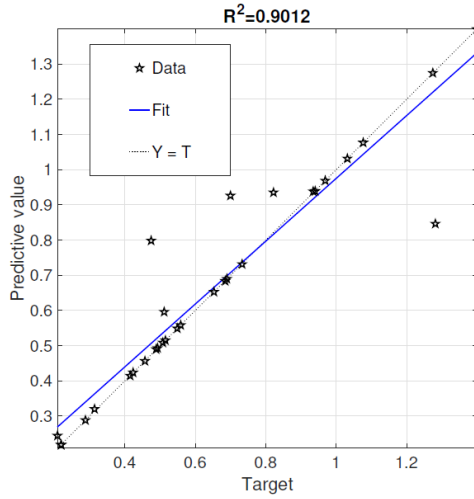


Figure 21 The predictive performance of ANN in test set (see online version for colours)

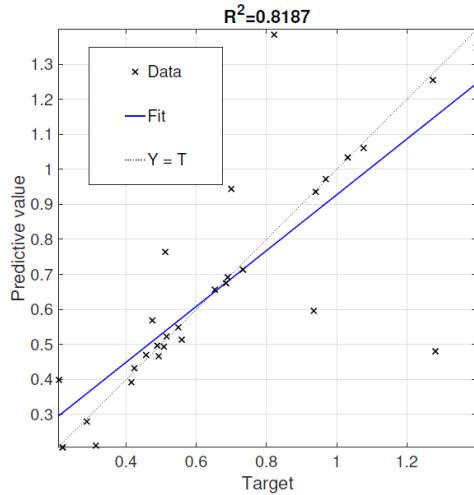


Table 7 Time complexity in ML models

<i>ML method</i>	<i>Time complexity</i>	<i>Computation time</i>
Regression tree	$O(\log_2 n)$	12.63 s
Random forest	$O(m(\log n))$	30.21 s
Gradient boosting machine	$O(mn \log n)$	226.13 s
Artificial neural network	$O(mn^2)$	1,896.75 s
Deep neural network	$O(n^3)$	1,531 s (25 m 31 s)

Note: n is the samples number.

Figure 22 The predictive performance of DNN in test set (see online version for colours)

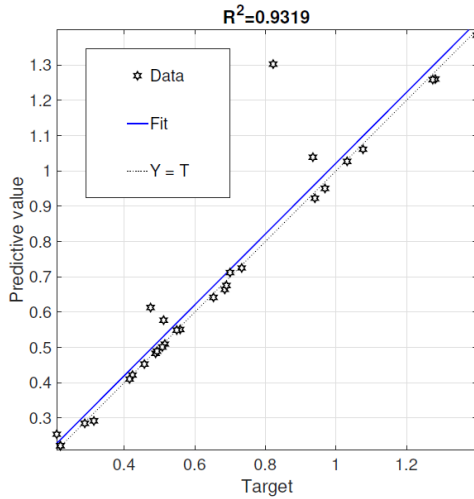
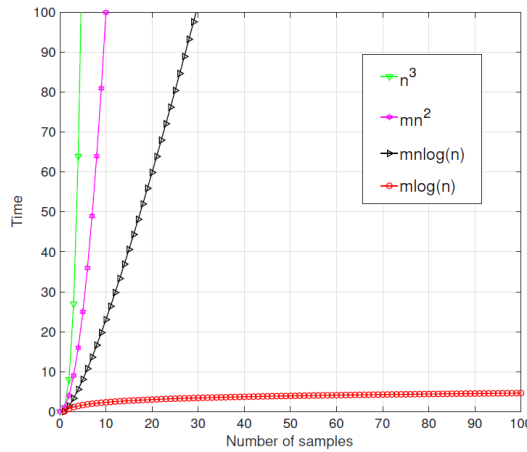


Figure 23 The time complexity among ML models (see online version for colours)



5 Conclusions

We proposed a computational framework using surrogate models through five steps: data preparation, model selections, hyper-parameter tuning, model training, evaluation and prediction. This five-step framework can systematically and comprehensively address a number of related problems in composites design.

We then used this framework to conduct an implementation: nano-composite thermal conductivity. The thermal conductivity of PNCs has been predicted by a data-driven enhanced hierarchical multi-scale method. We conducted a multi-scale stochastic analysis from micro-, meso- to macro-scale via three processes: bottom-up modelling, stochastic modelling, and machine learning models. At each scale, the effective parameters are extracted and transferred to the next higher length scale. The

interfacial resistance is modelled at the microscopic scale, where the SWCNT and its surrounding matrix are simplified into an equivalent fibre. The equivalent fibres are used at the meso-scale, where the agglomeration and dispersion are accounted for. At the macro-scale, the material consists of cubes with different material properties.

In addition, we performed a multi-scale stochastic analysis quantifying uncertainties at different length scales. Seven parameters are selected as random parameters, namely the thermal conductivity of fibre, the thermal conductivity of matrix, interface resistance, aspect ratio, agglomeration index, dispersion index and volume fraction. Machine learning methods have been used subsequently to structure those surrogate models. We exploited tree-based and neural network-based models. All surrogate models yield reasonable results. This data-driven approach can greatly reduce the requirements for multi-scale stochastic modelling in engineering materials design, reducing computational time and computational cost. In summary, the following conclusion can be summarised:

- 1 The five-steps computational framework using surrogate models enables efficient DDM to design complex materials.
- 2 Every machine learning model has a good predictive ability; the predictive performance of Deep neural networks (DNN) is the best but also computationally most expensive.
- 3 RF yields the lowest computational cost but of the cost of a lower accuracy.
- 4 PSO with ten-fold CV has a significant impact on hyper-parameter tuning.
- 5 The neural network-based models incur more computational cost for the increase of samples according to the computational complexity.

Acknowledgements

We gratefully acknowledge the support of the China Scholarship Council (CSC) and the DAAD for supporting the latter stages of this work through the STIBET Doctoral Degree Completion Scholarship.

References

- Alpaydin, E. (2020) *Introduction to Machine Learning*, MIT Press, Cambridge, MA.
- Baughman, R.H., Zakhidov, A.A. and De Heer, W.A. (2002) 'Carbon nanotubes – the route toward applications', *Science*, Vol. 297, No. 5582, pp.787–792.
- Breiman, L. (2001) 'Random forests', *Machine Learning*, October, Vol. 45, pp.5–32.
- Budarapu, P.R., Gracie, R., Bordas, S. and Rabczuk, T. (2014) 'An adaptive multiscale method for quasi-static crack growth', *Computational Mechanics*, June, Vol. 53, pp.1129–1148.
- Budarapu, P.R., Reinoso, J. and Paggi, M. (2017) 'Concurrently coupled solid shell-based adaptive multiscale method for fracture', *Computer Methods in Applied Mechanics and Engineering*, Vol. 319, pp.338–365.
- Budarapu, P., Reinoso, J. and Paggi, M. (2018a) 'Computational analysis of crack opening in photovoltaic solar cells in the presence of multiple interacting cracks', *Journal of Coupled Systems and Multiscale Dynamics*, December, Vol. 6, No. 4, pp.273–281.

- Budarapu, P., Javvaji, B., Reinoso, J., Paggi, M. and Rabczuk, T. (2018a) 'A three dimensional adaptive multiscale method for crack growth in silicon', *Theoretical and Applied Fracture Mechanics*, Vol. 96, pp.576–603.
- Budarapu, P., Kumar, S., Prusty, B.G. and Paggi, M. (2019a) 'Stress transfer through the interphase in curved-fiber pullout tests of nanocomposites', *Composites Part B: Engineering*, Vol. 165, pp.417–434.
- Budarapu, P.R., Zhuang, X., Rabczuk, T. and Bordas, S.P. (2019b) 'Multiscale modeling of material failure: theory and computational methods', in *Advances in Applied Mechanics*, Vol. 52, pp.1–103, Academic Press, Inc., New York; HK, London.
- Budarapu, P., Thakur, S., Kumar, S. and Paggi, M. (2021) 'Micromechanics of engineered interphases in nacre-like composite structures', *Mechanics of Advanced Materials and Structures*, Vol. 28, No. 22, pp.2327–2342.
- Bui, K., Grady, B.P. and Papavassiliou, D.V. (2011) 'Heat transfer in high volume fraction cnt nanocomposites: effects of inter-nanotube thermal resistance', *Chemical Physics Letters*, Vol. 508, Nos. 4–6, pp.248–251.
- Chen, T. and Guestrin, C. (2016) 'XGBoost: a scalable tree boosting system', in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.785–794.
- Dorogush, A.V., Ershov, V. and Gulin, A. (2018) *CatBoost: Gradient Boosting with Categorical Features Support*, arXiv preprint arXiv:1810.11363.
- Dusane, A., Budarapu, P., Pradhan, A., Natarajan, S., Reinoso, J. and Paggi, M. (2021) 'Simulation of bridging mechanisms in complex laminates using a hybrid PF-CZM method', *Mechanics of Advanced Materials and Structures*, pp.1–29.
- Fiamegkou, E., Athanasopoulos, N. and Kostopoulos, V. (2014) 'Prediction of the effective thermal conductivity of carbon nanotube-reinforced polymer systems', *Polymer Composites*, Vol. 35, No. 10, pp.1997–2009.
- Francis, A., Natarajan, S., Lee, C. and Budarapu, P.R. (2022) 'A cell-based smoothed finite element method for finite elasticity', *International Journal for Computational Methods in Engineering Science and Mechanics*, pp.1–15.
- Freitag, M., Steiner, M., Martin, Y., Perebeinos, V., Chen, Z., Tsang, J.C. and Avouris, P. (2009) 'Energy dissipation in graphene field-effect transistors', *Nano Letters*, Vol. 9, No. 5, pp.1883–1888.
- Friedman, J.H. (2001) 'Greedy function approximation: a gradient boosting machine', *Annals of Statistics*, Vol. 29, No. 5, pp.1189–1232.
- Ghosh, S., Bao, W., Nika, D.L., Subrina, S., Pokatilov, E.P., Lau, C.N. and Balandin, A.A. (2010) 'Dimensional crossover of thermal transport in few-layer graphene', *Nature Materials*, July, Vol. 9, p.555.
- Goswami, S., Anitescu, C., Chakraborty, S. and Rabczuk, T. (2020) 'Transfer learning enhanced physics informed neural network for phase-field modeling of fracture', *Theoretical and Applied Fracture Mechanics*, Vol. 106, p.102447.
- Guo, H., Zhuang, X., Chen, P., Alajlan, N. and Rabczuk, T. (2022) 'Stochastic deep collocation method based on neural architecture search and transfer learning for heterogeneous porous media', *Engineering with Computers*, pp.1–26.
- Harish, S., Ishikawa, K., Einarsson, E., Aikawa, S., Chiashi, S., Shiomi, J. and Maruyama, S. (2012) 'Enhanced thermal conductivity of ethylene glycol with single-walled carbon nanotube inclusions', *International Journal of Heat and Mass Transfer*, Vol. 55, Nos. 13–14, pp.3885–3890.
- He, B., Mortazavi, B., Zhuang, X. and Rabczuk, T. (2016) 'Modeling kapitza resistance of two-phase composite material', *Composite Structures*, Vol. 152, pp.939–946.

- Iman, R.L. and Conover, W. (1980) 'Small sample sensitivity analysis techniques for computer models. with an application to risk assessment', *Communications in Statistics – Theory and Methods*, Vol. 9, No. 17, pp.1749–1842.
- Javvaji, B., Budarapu, P.R., Paggi, M., Zhuang, X. and Rabczuk, T. (2018) 'Fracture properties of graphene-coated silicon for photovoltaics', *Advanced Theory and Simulations*, Vol. 1, No. 12, p.1800097.
- Jeong, S., Zhu, F., Lim, H., Kim, Y. and Yun, G.J. (2019) '3D stochastic computational homogenization model for carbon fiber reinforced CNT/epoxy composites with spatially random properties', *Composite Structures*, Vol. 207, pp.858–870.
- Kapadia, R.S., Louie, B.M. and Bandaru, P.R. (2014) 'The influence of carbon nanotube aspect ratio on thermal conductivity enhancement in nanotube – polymer composites', *Journal of Heat Transfer*, Vol. 136, No. 1, p.11303.
- Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. and Liu, T-Y. (2017) 'LightGBM: a highly efficient gradient boosting decision tree', *Advances in Neural Information Processing Systems*, Vol. 30.
- Keitel, H. (2012) *Bewertungsmethoden für die Prognosequalität von Kriechmodellen des Betons*, Doctoral thesis, Bauhaus-Universität Weimar.
- Kohavi, R. et al. (1995) 'A study of cross-validation and bootstrap for accuracy estimation and model selection', in *IJCAI*, Montreal, Canada, Vol. 14, pp.1137–1145.
- Korupolu, D., Budarapu, P., Vusa, V., Pandit, M. and Reddy, J. (2022) 'Impact analysis of hierarchical honeycomb core sandwich structures', *Composite Structures*, Vol. 280, p.114827.
- Kuhn, M., Johnson, K. et al. (2013) *Applied Predictive Modeling*, Vol. 26, Springer, New York.
- Kumar, Y., Rammohan, B., Budarapu, P., Harursamath, D. and Seetharamu, K. (2019) 'Dynamic instability analysis of multifunctional composite structures', *AIAA Journal*, Vol. 57, No. 10, pp.4241–4254.
- Liu, B., Vu-Bac, N. and Rabczuk, T. (2021) 'A stochastic multiscale method for the prediction of the thermal conductivity of polymer nanocomposites through hybrid machine learning algorithms', *Composite Structures*, Vol. 273, pp.114269.
- Liu, B., Vu-Bac, N., Zhuang, X. and Rabczuk, T. (2020) 'Stochastic multiscale modeling of heat conductivity of polymeric clay nanocomposites', *Mechanics of Materials*, Vol. 142, p.103280.
- Liu, B., Vu-Bac, N., Zhuang, X., Fu, X. and Rabczuk, T. (2022) 'Stochastic full-range multiscale modeling of thermal conductivity of polymeric carbon nanotubes composites: a machine learning approach', *Composite Structures*, Vol. 289, p.115393.
- Maruyama, S., Kojima, R., Miyauchi, Y., Chiashi, S. and Kohno, M. (2002) 'Low-temperature synthesis of high-purity single-walled carbon nanotubes from alcohol', *Chemical Physics Letters*, Vol. 360, Nos. 3–4, pp.229–234.
- Matos, M., Pinho, S. and Tagarielli, V. (2019) 'Application of machine learning to predict the multiaxial strain-sensing response of CNT-polymer composites', *Carbon*, Vol. 146.
- Moisala, A., Li, Q., Kinloch, I. and Windle, A. (2006) 'Thermal and electrical conductivity of single-and multi-walled carbon nanotube-epoxy composites', *Composites Science and Technology*, Vol. 66, No. 10, pp.1285–1288.
- Mortazavi, B., Baniassadi, M., Bardon, J., Ahzi, S. (2013a) 'Modeling of two-phase random composite materials by finite element, Mori-Tanaka and strong contrast methods', *Composites Part B: Engineering*, Vol. 45, No. 1, pp.1117–1125.
- Mortazavi, B., Bardon, J. and Ahzi, S. (2013b) 'Interphase effect on the elastic and thermal conductivity response of polymer nanocomposite materials: 3D finite element study', *Computational Materials Science*, Vol. 69, pp.100–106.
- Mortazavi, B., Benzerara, O., Meyer, H., Bardon, J. and Ahzi, S. (2013c) 'Combined molecular dynamics-finite element multiscale modeling of thermal conduction in graphene epoxy nanocomposites', *Carbon*, Vol. 60, pp.356–365.

- Mortazavi, B., Podryabinkin, E.V., Roche, S., Rabczuk, T., Zhuang, X. and Shapeev, A.V. (2020) 'Machine-learning interatomic potentials enable first-principles multiscale modeling of lattice thermal conductivity in graphene/borophene heterostructures', *Materials Horizons*, Vol. 7, No. 9, pp.2359–2367.
- Mortazavi, B., Silani, M., Podryabinkin, E.V., Rabczuk, T., Zhuang, X. and Shapeev, A.V. (2021) 'First-principles multiscale modeling of mechanical properties in graphene/borophene heterostructures empowered by machine-learning interatomic potentials', *Advanced Materials*, Vol. 33, No. 35, p.2102807.
- Novák, D., Teplý, B. and Keršner, Z. (1998) 'The role of Latin hypercube sampling method in reliability engineering', in *Proc. of ICOSSAR*, Vol. 97, pp.403–409.
- Ojo, S.O., Budarapu, P.R. and Paggi, M. (2017) 'A nonlocal adaptive discrete empirical interpolation method combined with modified hp-refinement for order reduction of molecular dynamics systems', *Computational Materials Science*, Vol. 140, pp.189–208.
- Pinsky, M. and Karlin, S. (2010) *An Introduction to Stochastic Modeling*, Elsevier, 30 Corporate Drive, Suite 400, Burlington, MA 01803, USA; The Boulevard, Langford Lane, Kidlington, Oxford, OX5 1GB, UK.
- Rafiee, R. and Zehtabzadeh, H. (2020) 'Predicting the strength of carbon nanotube reinforced polymers using stochastic bottomup modeling', *Applied Physics A*, Vol. 126, No. 8, pp.1–13.
- Rong, Q., Wei, H., Huang, X. and Bao, H. (2019) 'Predicting the effective thermal conductivity of composites from cross sections images using deep learning methods', *Composites Science and Technology*, Vol. 184, p.107861.
- Sharma, S., Awasthi, R., Sastry, Y.S. and Budarapu, P.R. (2021) 'Physics-informed neural networks for estimating stress transfer mechanics in single lap joints', *Journal of Zhejiang University-Science A*, Vol. 22, No. 8, pp.621–631.
- Shokrieh, M.M. and Rafiee, R. (2010) 'Stochastic multi-scale modeling of CNT/polymer composites', *Computational Materials Science*, Vol. 50, No. 2, pp.437–446.
- Siruvuri, S.V. and Budarapu, P. (2020) 'Studies on thermal management of lithium-ion battery pack using water as the cooling fluid', *Journal of Energy Storage*, Vol. 29, No. 2, p.101377.
- Siruvuri, S.V., Budarapu, P. and Paggi, M. (2021) 'Current-voltage characteristics of silicon based solar cells in the presence of cracks: MD simulations', *Semiconductor Science and Technology*, Vol. 37, No. 12, p.25011.
- Sutrakar, V., Javvaji, B. and Budarapu, P. (2021) 'Fracture strength and fracture toughness of graphene: MD simulations', *Applied Physics A*, Vol. 127, pp.1–11.
- Thostenson, E.T., Ren, Z. and Chou, T-W. (2001) 'Advances in the science and technology of carbon nanotubes and their composites: a review', *Composites Science and Technology*, Vol. 61, No. 13, pp.1899–1912.
- Vavouliotis, A., Fiamegou, E., Karapappas, P., Psarras, G. and Kostopoulos, V. (2010) 'DC and AC conductivity in epoxy resin/multiwall carbon nanotubes percolative system', *Polymer Composites*, Vol. 31, No. 11, pp.1874–1880.
- Vu-Bac, N., Lahmer, T., Keitel, H., Zhao, J., Zhuang, X. and Rabczuk, T. (2014a) 'Stochastic predictions of bulk properties of amorphous polyethylene based on molecular dynamics simulations', *Mechanics of Materials*, Vol. 68, pp.70–84.
- Vu-Bac, N., Lahmer, T., Zhang, Y., Zhuang, X. and Rabczuk, T. (2014b) 'Stochastic predictions of interfacial characteristic of polymeric nanocomposites (PNCS)', *Composites Part B: Engineering*, Vol. 59, pp.80–95.
- Vu-Bac, N., Silani, M., Lahmer, T., Zhuang, X. and Rabczuk, T. (2015a) 'A unified framework for stochastic predictions of mechanical properties of polymeric nanocomposites', *Computational Materials Science, Special Issue Polymeric Composites*, Vol. 96, No. Part B, pp.520–535.

- Vu-Bac, N., Rafiee, R., Zhuang, X., Lahmer, T. and Rabczuk, T. (2015b) ‘Uncertainty quantification for multiscale modeling of polymer nanocomposites with correlated parameters’, *Composites Part B: Engineering*, Vol. 68, pp.446–464.
- Yang, S-Y., Ma, C-C.M., Teng, C-C., Huang, Y-W., Liao, S-H., Huang, Y-L., Tien, H-W., Lee, T-M. and Chiou, K-C. (2010) ‘Effect of functionalized carbon nanotubes on the thermal conductivity of epoxy composites’, *Carbon*, Vol. 48, No. 3, pp.592–603.
- Yang, S-W., Budarapu, P.R., Mahapatra, D.R., Bordas, S.P., Zi, G. and Rabczuk, T. (2015) ‘A meshless adaptive multiscale method for fracture’, *Computational Materials Science, Special Issue Polymeric Composites*, Vol. 96, No. Part B, pp.382–395.
- Zhuang, X. and Zhou, S. (2019) ‘The prediction of self-healing capacity of bacteria-based concrete using machine learning approaches’, *Computers, Materials & Continua*, Vol. 59, No. 1, pp.57–77.