An efficient multiscale surrogate modelling framework for composite materials considering progressive damage based on artificial neural networks

Shibo Yan[a], Xi Zou[a,*], Mohammad Ilkhani[a], Arthur Jones[a]

[a]*Composites Research Group, Faculty of Engineering, University of Nottingham, Nottingham, NG7 2RD, United Kingdom*

Abstract

Modelling of the progressive damage behaviour of large-scale composite structures presents a significant challenge in terms of computational cost. This is due to the detailed description in finite element (FE) models for the materials, i.e., with each unidirectional layer defined as required by the applicability of laminate failure criteria, and numerous iterations required to capture the highly nonlinear behaviour after damage initiation. In this work, we propose a method to accelerate the nonlinear FE analysis by using a pre-computed surrogate model which acts as a general material database representing the nonlinear effective stress-strain relationship and the possible failure information. Developed using artificial neural network algorithms, the framework is separated into an offline training phase and an online application phase. The surrogate model is first trained with a vast number of sampling data obtained from mesoscale unit cell models offline, and then used for online predictions on a macroscale FE model. The prediction accuracy of the surrogate model was examined by comparing the results with conventional FE modelling and good agreement was observed. The presented method enables progressive damage analysis of composite structures with significant savings of the online computational cost. Lastly, the surrogate model is only based on material designs and is reusable for other structures with the same material.

Keywords: multiscale modelling, progressive damage, surrogate model, artificial neural network

---

*Corresponding author
Email address:* `xi.zou@nottingham.ac.uk` (Xi Zou)

1. Introduction

Composite materials have been increasingly used to meet the growing need for lightweight structures, due to their high specific stiffness and strength along with high energy absorption. Owing to their typically anisotropic designs, the failure mechanisms of these materials are especially complex when compared with homogeneous materials. Various failure modes can be observed in composite materials under different loading conditions and they often appear to be in a progressive damage manner, which makes the modelling of the failure behaviour of such materials more difficult.

Progressive damage modelling (PDM) methods, which in general employ failure criteria for predicting damage initiation, and evolution laws for damage propagation, have been developed and widely used by the research community to predict failure in composite materials [1, 2, 3]. However, due to the conventional PDM method's requirement for a mesoscopic representation of the material in the FE model, e.g., incorporating all the laminae for laminated composites or the whole fibre tow architectures for textile composites, and a large number of iterations required to capture the nonlinear behaviour after damage initiation, the resulting high computational cost prevents their application to large scale composite structures. As the need for building large-scale composite structures grows, much effort regarding more efficient modelling methods has been made to address this challenge. Recent attempts involve the use of a global-local coupling method [4, 5, 6], in which the large scale structure modelled by a homogenised material (at the macroscale) works as the global model, whilst the mesoscopic representations of selected sub-regions requiring detailed damage analysis are the local models. In order to account for the progressive damage behaviour, degraded materials properties obtained from the local analysis are fed into the global model. Although the computational cost can be reduced by using the global-local PDM method, there is a lack of rigorous criteria to be applied to the homogenised materials at the global level to determine the locations where local analysis should be performed.

On the other hand, material suppliers, end-users and regulatory agencies have been dedicated to creating a shared material property database for composites over the last few decades. This database, aiming to reduce repetitive testing efforts among multiple companies using the same materials, usually covers elastic properties and ultimate strengths for basic lamina and laminate materials under the standardised manufacturing processing. Meanwhile, it is well understood that the material properties of composite materials might be degraded after damage initiation because of their progressive damage behaviour, but these degraded properties have rarely been incorporated into the database. In this work, the idea of a more general material property database is proposed, referred to as a surrogate model, which can be used for fast progressive damage analyses of composite structures. Further to the previously discussed database of elastic properties and ultimate strengths, the surrogate model represents the nonlinear effective stress-strain relationship up to the ultimate failure of a specific material under any loading condition, as well as the

2

possible failure mode information. Surrogate models are normally built using a data-driven, bottom-up method. Therefore the construction of such a surrogate model requires a large amount of existing material behaviour data, which can be obtained from mesoscale unit cell modelling or even from experimental testing, although the latter is not practical at present due to the extensive work required.

A number of techniques for creating surrogate models have been employed in engineering such as Chaos polynomial expansion, response surface methodology, kriging (a.k.a. Gaussian process regression) and Artificial neural network (ANN) [7]. Artificial neural network, the basis of deep learning algorithms, creates a surrogate model for the relationship between input and output data from a given dataset [8]. Tye use of ANN assisted methods to model the mechanical behaviour of composites has attracted broad research interest, as reviewed in [9, 10]. More recently, Balokas et al. [11] employed ANN to generate a surrogate model for the elastic properties of 3D braided composites with uncertainty, with which they could reduce the computational cost in the sensitivity analysis to identify the most crucial uncertain parameters. Based on the properties of bulk matrix, and fibre and the fibre volume fraction, Oliveira and co-workers [12] used ANN to model the shear modulus (G_{12}) and longitudinal ultimate tensile strength (X_t) of unidirectional (UD) composites. Rong et al. [13] used cross-section images as the input of 2D convolutional neural network models (CNN) to predict the effective thermal conductivity of 3D particle filled composites, which was found to be two orders of magnitude faster than 3D CNNs due to the reduced number of parameters. A vibration based non-destructive testing method using an ANN model was developed by Farhana et al. [14] to predict glass fibre/matrix volume fractions in composites. The accuracy of the predictions was shown to be within the range of 90–98%, and they were in good agreement with those obtained through destructive tests. The authors also noted that the method is currently implemented only for glass fibre reinforced polyester and therefore the volume fraction prediction only works for glass fibre reinforced polyester of the same fibre fabric and orientation.

Apart from using neural networks for regression problems, a convolutional neural network based approach was adopted by Khan and colleagues [15] for the classification and prediction of delamination in smart composite laminates from the vibration-based spectrograms. A classification accuracy of 90.1% on one healthy and 12 delaminated cases was observed from their study. Ang et al. [16] developed an ANN model for predicting the damage initiation of glass/epoxy composite pipes under multi-axial loading. The model was trained based on a large amount of experimental data from the literature, covering a range of different material properties, geometry parameters, and applied experimental loads. In their validation, the classification accuracy was ranged from 85% to 95%, which showed that the ANN model can be used as an early procedure to predict the damage onset of glass/epoxy composite pipes before the standard qualification process. Poor accuracy was observed for pure hoop loading, as there were less available data covering this load case in their collected training sample. In other words, the trained model only worked well for pipes of specific configurations considered in the training data. This also highlights that neural networks are more

suitable for interpolation-oriented problems than extrapolation-oriented problems, as well as the importance of the selection of sampling data.

In this work, we introduce a surrogate modelling method based on artificial neural networks to accelerate the progressive damage analysis of composite materials. The surrogate model, defined at the macroscale, represents the nonlinear effective constitutive relationship of a homogenised composite material and also the possible damage information regarding a load condition. During the offline phase, mesoscale unit cell modelling is first performed under designed load cases to obtain the sampling material property data (Section 2). The detailed structure and generating method based on artificial neural networks of the surrogate model are described in Section 3. Section 4 illustrates the workflow of generating and using such a surrogate model for nonlinear FE analysis of composites. A demonstration on modelling of a open-hole composite plate under tension is performed and results are compared with the conventional FE modelling to examine the accuracy and computational acceleration of the proposed method.

2. Acquisition of material property data

The generation of the surrogate model requires a large amount of known material property data covering different load cases. Experimental testing under well-established standards is the most reliable method to obtain material property data, but it is not practical at present due to the extensive testing required for different load cases. In this study, numerical characterisation through mesoscale unit cell modelling is employed to obtain these data.

2.1. Unit cells

Unit cells (UCs) have been widely employed in modelling and characterisation of composite materials, due to the multiscale characteristic of these materials. Applications of unit cells for composites include the derivation of effective material properties, such as elastic, and thermal properties [17], from the properties of its constituents at the mesoscale, as well as progressive damage analysis of composites [18, 19].

In the presence of translational symmetry, or periodicity, the boundary conditions on the unit cell can be expressed in the following way [17]:

$$
\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} - \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \varepsilon_x^0 & \gamma_{xy}^0 & \gamma_{xz}^0 \\ 0 & \varepsilon_y^0 & \gamma_{yz}^0 \\ 0 & 0 & \varepsilon_z^0 \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}
\tag{1}
$$

where u, v, w and u', v', w' are the displacements at corresponding points on the unit cell; ε^0, γ^0 are macroscopic engineering (effective) strains of the unit cell with subscripts denoting directions. $[\Delta x, \Delta y, \Delta z]^{\mathrm{T}}$ is the relevant translation corresponding to each translational symmetry. After applying the above boundary conditions, the effective stress and strain components can be easily obtained through the key degrees of freedom as detailed in [17].

4

2.2. Damage model

Composite materials are often shown to be nonlinear due to the complex failure process involving both damage initiation and progression. The properties of these materials are degraded after damage initiation but they still remain capable of bearing further loads. Therefore, damage initiation and evolution are both considered in the unit cell model. Although some of the widely used failure criteria for unidirectional composites were proposed several decades ago, modelling their complex failure mechanisms is still the subject of significant researches [20, 21], as none of the 12 leading failure criteria have been demonstrated to be able to predict failure accurately within all the test cases examined by the so-called World-Wide Failure Exercise [22]. In this study, a damage model by Linde et al. [23] for fibre reinforced composites was used for demonstrating the proposed method, other than focusing on the accuracy and applicability of the damage model itself. In Linde's model, damage in the fibre and the matrix are considered via damage variables d_f and d_m, respectively. The interlaminar damage is not currently taken into account. The material behaviour of each lamina is assumed to be orthotropic, with stiffness much higher along the fibre direction and relatively lower in the transverse directions. The undamaged elastic matrix at a material point is denoted by $\mathbf{C}$ with its components C_{ij}, using conventional Voigt notation, $i, j = 1, 2, \ldots, 6$. For the damaged material, the elastic matrix becomes

$$\mathbf{C}_d = \begin{bmatrix} (1-d_\mathrm{f})C_{11} & (1-d_\mathrm{f})(1-d_\mathrm{m})C_{12} & (1-d_\mathrm{f})C_{13} & 0 & 0 & 0 \\ & (1-d_\mathrm{m})C_{22} & (1-d_\mathrm{m})C_{23} & 0 & 0 & 0 \\ & & C_{33} & 0 & 0 & 0 \\ & & & (1-d_\mathrm{f})(1-d_\mathrm{m})C_{44} & 0 & 0 \\ & \mathrm{sym} & & & C_{55} & 0 \\ & & & & & C_{66} \end{bmatrix}. \tag{2}$$

For orthogonal materials such as fibre reinforced polymer, the transverse properties are isotropic, and the stiffness matrix components read

$$\begin{aligned} C_{11} &= E_\mathrm{L}(1 - \nu_\mathrm{TT}^2)/\alpha, & C_{22} &= C_{33} = E_\mathrm{T}(1 - \nu_\mathrm{LT}\nu_\mathrm{TL})/\alpha, \\ C_{23} &= E_\mathrm{T}(\nu_\mathrm{TT} - \nu_\mathrm{LT}\nu_\mathrm{TL})/\alpha, & C_{12} &= C_{13} = E_\mathrm{T}\nu_\mathrm{LT}(1 - \nu_\mathrm{TT})/\alpha, \\ C_{33} &= G_\mathrm{TT}, & C_{44} &= C_{55} = G_\mathrm{LT}, \\ \alpha &= 1 - 2\nu_\mathrm{LT}\nu_\mathrm{TL} - \nu_\mathrm{TT}^2 - 2\nu_\mathrm{LT}\nu_\mathrm{TL}\nu_\mathrm{TT}, \end{aligned} \tag{3}$$

where the Young's module and Poisson's ratios are denoted by E and ν with subscripts L and T representing longitudinal and transverse directions respectively, and G denotes the shear modulus.

The in-plane strength of one lamina ply can be characterised by $\tilde{\varepsilon}^\mathrm{t}$, $\tilde{\varepsilon}^\mathrm{c}$ and $\tilde{\varepsilon}^\mathrm{s}$, which are the failure strain under tension, compression and shear, respectively. Conventionally the primary direction coincides with the fibre direction, and is designated with 1-direction, while its in-plane orthogonal is 2-direction. Taking

advantage of this notation, the damage initiation criteria for the fibre and the matrix read:

$$f_{\rm f}(\varepsilon_{11}) = \sqrt{\frac{\tilde{\varepsilon}_{11}^{\rm t}}{\tilde{\varepsilon}_{11}^{\rm c}}\varepsilon_{11}^2 + \left[\tilde{\varepsilon}_{11}^{\rm t} - \frac{(\tilde{\varepsilon}_{11}^{\rm t})^2}{\tilde{\varepsilon}_{11}^{\rm c}}\right]\varepsilon_{11}} - \tilde{\varepsilon}_{11}^{\rm t} > 0, \tag{4}$$

$$f_{\rm m}(\varepsilon_{22}, \varepsilon_{12}) = \sqrt{\frac{\tilde{\varepsilon}_{22}^{\rm t}}{\tilde{\varepsilon}_{22}^{\rm c}}\varepsilon_{22}^2 + \left[\tilde{\varepsilon}_{22}^{\rm t} - \frac{(\tilde{\varepsilon}_{22}^{\rm t})^2}{\tilde{\varepsilon}_{22}^{\rm c}}\right]\varepsilon_{22} + \left(\frac{\tilde{\varepsilon}_{22}^{\rm t}}{\tilde{\varepsilon}_{12}^{\rm s}}\right)^2 \varepsilon_{12}^2} - \tilde{\varepsilon}_{22}^{\rm t} > 0. \tag{5}$$

When any of the initiation criteria is satisfied, the damage variables follows the evolution equation

$$d_{\rm f} = 1 - \frac{\tilde{\varepsilon}_{11}^{\rm t}}{f_{\rm f} + \tilde{\varepsilon}_{11}^{\rm t}}\exp\left(-C_{11}\tilde{\varepsilon}_{11}^{\rm t}f_{\rm f}\frac{L_{\rm c}}{G_{\rm f}}\right), \tag{6}$$

$$d_{\rm m} = 1 - \frac{\tilde{\varepsilon}_{22}^{\rm t}}{f_{\rm m} + \tilde{\varepsilon}_{22}^{\rm t}}\exp\left(-C_{22}\tilde{\varepsilon}_{22}^{\rm t}f_{\rm m}\frac{L_{\rm c}}{G_{\rm m}}\right), \tag{7}$$

where $L_{\rm c}$ is the characteristic length, $G_{\rm f}$ and $G_{\rm m}$ are the fracture energies of the fibre and the matrix, respectively.

At a material integration point, the stress is updated with the constitutive equation

$$\boldsymbol{\sigma} = \mathbf{C}_d \boldsymbol{\varepsilon}. \tag{8}$$

For FE solvers which are using the incremental formulation, such as Abaqus, it is necessary to compute the Jacobian matrix at each increment in order to update the stiffness matrix. This is achieved by differentiating (8) using the chain rule,

$$\begin{aligned}
\frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\varepsilon}} &= \mathbf{C}_d + \frac{\partial \mathbf{C}_d}{\partial \boldsymbol{\varepsilon}}\boldsymbol{\varepsilon} \\
&= \mathbf{C}_d + \left(\frac{\partial \mathbf{C}_d}{\partial d_{\rm f}}\boldsymbol{\varepsilon}\right) \otimes \left(\frac{\partial d_{\rm f}}{\partial f_{\rm f}}\frac{\partial f_{\rm f}}{\partial \boldsymbol{\varepsilon}}\right) + \left(\frac{\partial \mathbf{C}_d}{\partial d_{\rm m}}\boldsymbol{\varepsilon}\right) \otimes \left(\frac{\partial d_{\rm m}}{\partial f_{\rm m}}\frac{\partial f_{\rm m}}{\partial \boldsymbol{\varepsilon}}\right).
\end{aligned} \tag{9}$$

Thus, the Cauchy stress at every integration point is updated in each increment by

$$\boldsymbol{\sigma} \leftarrow \boldsymbol{\sigma} + \frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\varepsilon}}\Delta\boldsymbol{\varepsilon}. \tag{10}$$

A typical implementation of the calculation of (9) resorts to a carefully written user material subroutine (UMAT) using the API provided by Abaqus. The requirement of recalculating the elastic tensor $\mathbf{C}_d$ at each increment makes it expensive for the whole analysis cost. It is also known that when the damage initiation happens, the nonlinearity causes the solution to become significantly slower than before, and it is tedious to search for the optimal tuning for the solver parameters due to their problem-dependency. In this study, a trained ANN is used as a surrogate model to approximate the true update of Cauchy stress at the macroscale, so as to achieve significant accelerations during this process. This surrogate model is implemented in another UMAT subroutine, in which the Cauchy stress is computed via the ANN and keeping the initial Jacobian simply unmodified to save the computational cost. The initial Jacobian is computed as (9) using homogenised material properties calculated following [17], but without any damages. Note such a simplification will turn the Newton-Raphson iteration into a quasi-Newton one in the nonlinear solution process. Other quasi-Newton schemes could also be implemented achieve even faster convergence.

6

2.3. Design of load cases for training data generation

As the ANN requires a large amount of sampling data to train, it is necessary to generate the dataset a priori with the established UC model. As the surrogate model represents the constitutive relationship and possible material damage states, the dataset contains the effective strain/stress with all components and corresponding damage state variables. In practice, the strategy is to collect the data from a batch run of the UC model under various prescribed load cases. The designed load cases applied to the UC are a series of strain states to which the loads are applied proportionally. The chosen strain states at the end of each analysis form an envelope of all possible combinations of the components, as illustrated in Figure 1 for plane stress problems, whilst the ranges for each strain components are $\epsilon_x, \epsilon_y \in [-0.02, 0.02]$; $\gamma_{xy} \in [0, 0.02]$. Regarding the surrogate model generated based on the load cases, it should be noted that the input strain should be enclosed in this envelope, otherwise the output stress will be unpredictable and losing physical meanings. To ensure this, a possible implementation in the code is to force the stress to be zero once an out-of-envelope strain is input to the surrogate model.
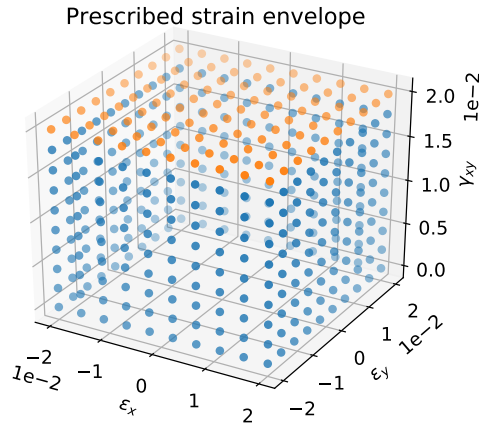

Figure 1: Scatter plot of sampling strain states at the end of each analysis. The states with maximum shear strain are coloured differently for clarity reasons.

3. Surrogate model

3.1. Methodology

A typical multiscale modelling method for composite materials involves at least two length scales, an upper one and a lower one. The objective of material characterisation using multiscale modelling is usually to evaluate the effective material properties in the upper scale based on the analyses conducted with the models at the lower scale. Not being constrained to only elastic properties, in this study, the scope of material characterisation is enriched to form a surrogate model representing a general material property

database. It covers nonlinear effective stress/strain relations (macroscale) accounting for damage initiation
and propagation with regard to any loading condition, along with material damage states. In addition, the
damage information also covers possible failure modes for composites.

To construct such a surrogate model, discrete effective stress/strain responses and damage parameters
data are first obtained through mesoscale UC damage modelling under a vast number of designed load cases.
Artificial neural networks are then employed to form the surrogate model, which is trained based on the
discrete data and used for predicting the behaviour under any loading condition including those not in the
training data. The trained ANN is a general material property database for a specific material, and is
designated to provide immediate information for future analysis of structures made of the same material
and layup.

In order to be used in FE analysis, the surrogate model is written into a material subroutine through
deploying the trained neural networks. Based on the application of Abaqus finite element software, the
structure and input/output of the material system is accordingly defined as shown in Figure 2. The inputs
of material system are the effective strain components, as required by the Abaqus user defined material
subroutine (UMAT), and the outputs are the corresponding effective stress components along with damage
information. Depending on the specific application, the inputs can be three dimensional for plane and shell
element usage or six dimensional for solid element usage. The workflow for using the proposed method is
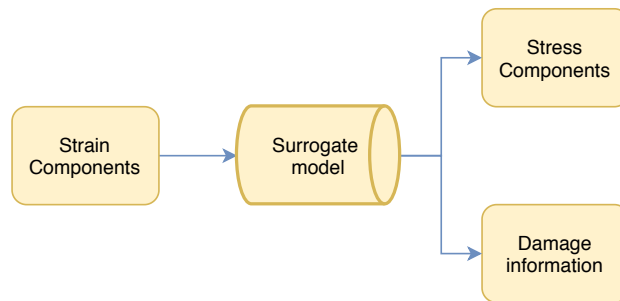summarised in Figure 3.

Figure 2: Input and output of the macroscale surrogate model.

3.2. Artificial neural networks

Artificial neural networks are inspired by the biological nervous systems, and is further evolved into many
varieties and being used in deep learning algorithms. Back to the basics, an ANN is built on a collection
of connected units (nodes) called artificial neurons, and the connections between neurons for transmitting
signals that are called synapses. A neuron first receives a signal then processes it, and finally signal neurons
connected to it. ANNs enable a computer to learn from observational data within the so-called training
process and then predict unknown results. ANNs have been applied to provide solutions to a variety of
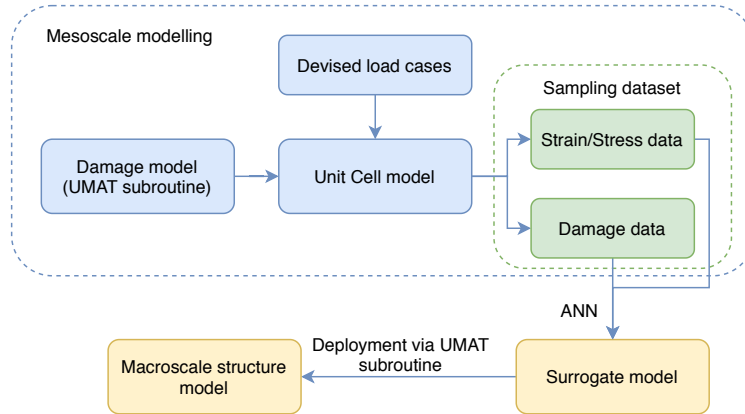
8

Figure 3: Proposed workflow for multiscale surrogate modelling of structure failure.

problems, including image recognition, speech recognition and machine translation. For applications in
composite materials, ANNs have been used to create surrogate models for the given input and output data,
which are usually related to material properties. After proper training, the predictions of new entries are far
more fast than running the simulation or experiment, due to the efficient information processing mechanism
of the ANNs. Consequently, using ANNs is a method to reduce the high computational costs of numerical
simulations, particularly for some large scale problems.

There are several types of artificial neural networks such as feed-forward, radial basis function and
recurrent neural networks. For regression analysis and classification problems in engineering, the most
commonly used one is the feed-forward neural network. This network contains one input layer, one output
layer and one or more hidden layers (Figure 4). The information propagates in one direction from the input
layer directly via any hidden layers to the output layer without loops.
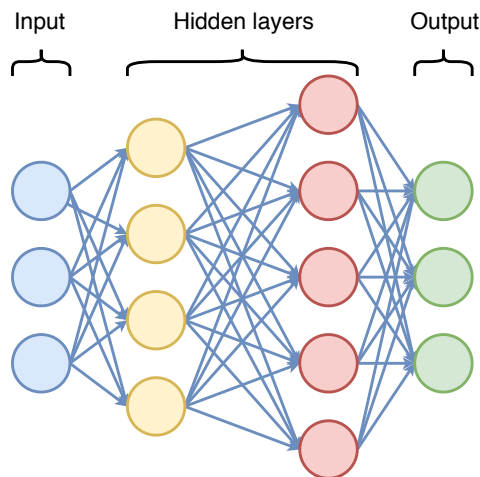

Figure 4: Illustration of a typical ANN with two hidden layers.

On each layer there are a number of neurons processing the information flow. The internal structure of a neuron is described in Figure 5. Let $\mathbf{w}^l$ be the weight matrix from the $(l-1)$-th layer to the l-th layer, $\boldsymbol{b}^l$ be the bias vector for inputs on the l-th layer. The I/O relationship of a neuron is expressed in a matrix form as

$$\boldsymbol{a}^l = \sigma\left(\mathbf{w}^l \boldsymbol{a}^{l-1} + \boldsymbol{b}^l\right).$$ (11)

where σ is the activation function. Commonly used activation functions are Tanh, ReLU, Sigmoid, Softplus and Linear etc., as illustrated in Figure 6. The definition of those functions are:

$$\text{Tanh:} \quad \sigma(x) = \tanh x$$

$$\text{ReLU:} \quad \sigma(x) = \max(0, x)$$

$$\text{Sigmoid:} \quad \sigma(x) = \frac{1}{1 + \exp(-x)} = \frac{\exp x}{\exp x + 1}$$

$$\text{Softplus:} \quad \sigma(x) = \ln(1 + \exp x)$$

$$\text{Softmax:} \quad \sigma(\boldsymbol{x})_i = \frac{\exp x_i}{\sum_j \exp x_j}.$$

For the case where the l-th layer has n neurons while the $(l-1)$-th layer has m neurons, it can be seen the weight matrix $\mathbf{w}^l \in \mathbb{R}^{n \times m}$, and the bias vector $\boldsymbol{b}^l \in \mathbb{R}^n$. Therefore (11) can be written in component form as

$$a_i^l = \sigma\left(\sum_{j=1}^m w_{ij}^l a_j^{l-1} + b_i^l\right),$$ (12)

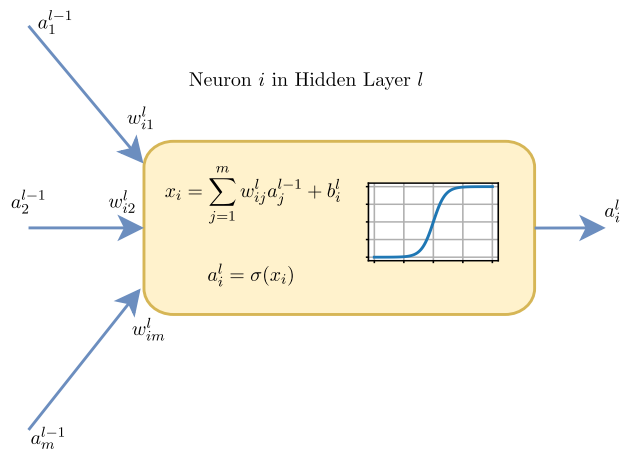where the indices $i = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, m$.

Figure 5: Illustration of a typical neuron.

After the network is established, it has to be trained to adapt to the given problem using a large dataset which was generated a priori. The training or learning process involves adjusting the weights and bias of the network to approximate closely the outputs of the training dataset through minimising the defined cost
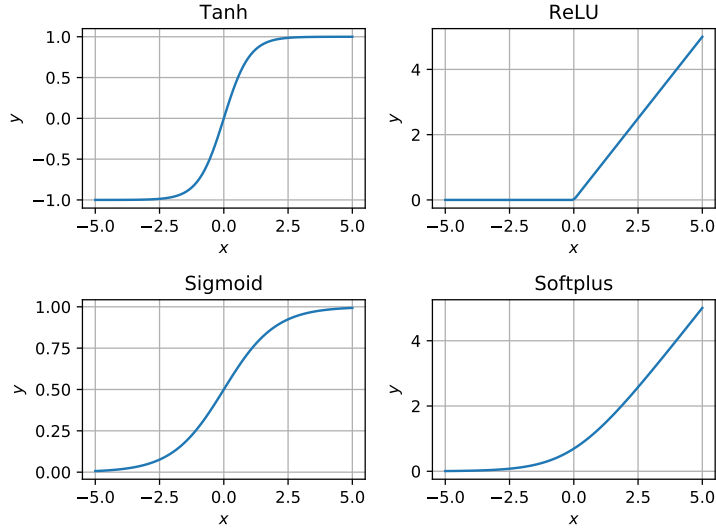
10

Figure 6: Plots of frequently used activation functions for a perceptron.

function (or loss function), e.g., the mean squared error (MSE) between the target output and the predicted output for regression problems, and the categorical cross-entropy for classification problems. In general, the training process can be considered as a typical application of a gradient-based optimisation algorithm and statistical estimation in a back-propagation manner. Further theoretical details about the training of ANN, the readers are referred to the literature such as [24]. Here we used Adamax [25], an algorithm for first-order gradient-based optimisation of stochastic objective functions, based on adaptive estimates of lower-order moments for the classification problems, and RMSprop [26], a gradient-based optimisation with adaptive learning rate adaption for the regression problem . In this optimisation process, the weights and bias of all the network are updated iteratively until the desired error tolerance is met or the maximum number of iterations (epoch) is exceeded. In this study, all problems are categorised as the supervised learning since the input and output data are paired.

4. Numerical example

In this Section, we demonstrate the steps to construct the proposed surrogate model and the application of the surrogate modelling method to predict the progressive damage behaviour of composites. A multi-directional laminate $[0, 90, 45, -45]_s$ under plane stress was used as a typical example to demonstrate the workflow for the proposed approach. The material used is aerospace-grade IM7/8551-7 carbon/epoxy [27], and the thickness for each ply is 0.125 mm.

11

4.1. Material property data acquisition

Mesoscale unit cell modelling was first conducted to acquire the material property data. An 8-layer mesoscale solid model was created as a unit cell for the example laminate $[0, 90, 45, -45]_{\mathrm{s}}$. Each layer is assumed to be a homogenised UD lamina with transversely isotropic properties given in Table 1. The the size of the unit cell is $1\,\mathrm{mm} \times 1\,\mathrm{mm} \times 1\,\mathrm{mm}$. Boundary conditions introduced in Section 2.1 were applied to the unit cell. Linde's damage model described in Section 2.2 was adopted here to demonstrate the proposed method through an Abaqus material subroutine (UMAT). The unit cell was loaded through applied strains for convenience as they would be required by the material subroutine as inputs for the surrogate model. The ranges for each strain components are $\epsilon_x, \epsilon_y \in [-0.02, 0.02]$; $\gamma_{xy} \in [0, 0.02]$. Each of the 8 layers might have different damage parameters and for convenience, for each load case only one maximum fibre damage and one maximum matrix damage parameters used selected to represent the whole laminate. A Python script was used to automate the collection of the material property data through a batch run of unit cell model under the designed load cases with Abaqus/Standard 2016. In this example, 12 754 sets of data including strains, stresses and damage parameters respectively have been collected for the subsequent training of the ANNs.

Table 1: Material properties of IM7/8551-7 lamina [27], fibre volume fraction $V_{\mathrm{f}} = 60\%$.

E_{11}	$165\,\mathrm{GPa}$	Longitudinal tensile strength X_{t}	$2560\,\mathrm{MPa}$
$E_{22} = E_{33}$	$8.4\,\mathrm{GPa}$	Longitudinal compressive strength X_{c}	$1590\,\mathrm{MPa}$
$\nu_{12} = \nu_{13}$	0.34	Transverse tensile strength Y_{t}	$73\,\mathrm{MPa}$
ν_{23}	0.5	Transverse compressive strength Y_{c}	$185\,\mathrm{MPa}$
$G_{12} = G_{13}$	$5.6\,\mathrm{GPa}$	In-plane/Transverse shear strength $S_{12} = S_{13}$	$90\,\mathrm{MPa}$
G_{23}	$2.8\,\mathrm{GPa}$	Through-thickness shear strength S_{23}	$57\,\mathrm{MPa}$

4.2. Construction of surrogate model

As the proposed surrogate model covers both constitutive relationship and damage state information, multiple ANNs are used to represent them respectively as they are for different problems. The regression analysis is carried out for nonlinear constitutive relationship between the obtained strain and stress data, and classification is performed for damage identification between the strain and damage state variable data. In this example, both regression and classification ANN were trained using Keras [28], which is a high-level open source neural network library written in Python and capable of running on top of TensorFlow (by Google), CNTK (by Microsoft), or Theano [29].

4.2.1. Regression ANN for constitutive relationship

Progressive damage behaviour is considered in the proposed method so that the regression for the constitutive law is nonlinear. A deep neural network with two hidden layers (60 and 50 neurons each) were employed for the plane stress case. The numbers of layers and neurons are hyperparameters in neural networks, which are problem-dependent and can not be learned during the training. Manual search is a simple way to determine the hyperparameters and also there are some complex hyperparameter optimisation methods as introduced in [30]. In this case, a manual search has been performed to identity these numbers through comparing accuracy/loss during validation. The input/output and hyperparameters for this classification ANN are presented in Table 2. It is noted that for 3D stress application the required hidden layers or number of neurons per layer could be more in the regression analysis. The MSE was used as the loss function and RMSprop was adopted as the optimiser. The training and validation process is illustrated in terms of the evolution of coefficient of determination R^2 (The metric "Accuracy" provided by Keras is the correctness rate of predictions in classification problems but is not meaningful for regression problems) and loss function during the epochs, which are shown in Figure 7. To avoid overfitting in the neural networks, one should monitor if there is any significant increase in the validation loss globally, and can use various methods provided by Keras such as adding dropout layers, adding early-stop callbacks in the training.

Table 2: Structure of the ANN for regression of stress-strain constitutive law.

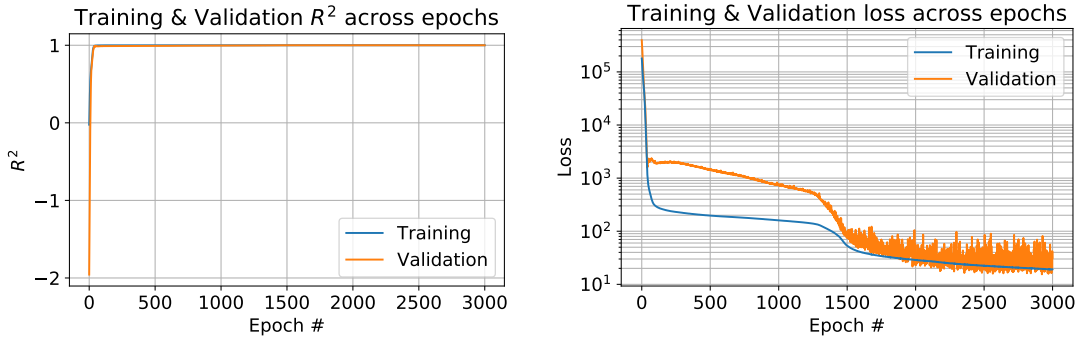| Input layer	Hidden layer 1	Hidden layer 2	Output layer	Loss function	Optimiser
$[\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{12}]^{\mathrm{T}}$	$60 \times \tanh$	$50 \times \tanh$	$[\sigma_{11}, \sigma_{22}, \sigma_{12}]^{\mathrm{T}}$	MSE	RMSprop

Figure 7: R^2 (left) and loss (right) function evolution during the training for the regression problem.

4.2.2. Classification ANN for damage identification

To perform the identification of damage initiation or material failure, a proper definition of the damage/failure state is required. In most applications, it is also required to provide a credibility measure as the

13

confidence of the simulation. An advantage of using the ANN is that the output of a classification problem is actually a probability of the categorisation based on pre-defined damage/failure criteria. In this section, we introduce two classification problem based on a previously introduced dataset for the damage initiation and failure identification problems of the 8-layer laminate. Note that for the macroscale model, there is only one layer of plane stress elements, it is more interesting to tell which element is damaged or failed regardless of the layerwised answer. Therefore, a simplification is made in this example to reduce the number of possible damage/failure states.

We define two simple damage states for each element in the plane stress macroscale model: the intact state in which both fibre and matrix have their damage variable equal to zero for all plies; and the damaged state in which either damage variable is above zero for any ply. This induces two corresponding probabilities as the ANN output: $\Pr(\text{Intact}) := \Pr(d_\text{f} = 0 \text{ and } d_\text{m} = 0)$ and $\Pr(\text{Damaged}) := \Pr(d_\text{f} > 0 \text{ or } d_\text{m} > 0)$. In addition, the training dataset is also labelled in this manner during preprocessing before training the ANN. The hyperparameters for this classification ANN are presented in Table 3. Note that for all classification problems, the categorical cross-entropy function is typically used as the loss function while the softmax activation function is used for each neuron in the output layer, so that the output will be the probability for each category. The training and validation process is illustrated in terms of the evolution of accuracy and loss function during the epochs, which are shown in Figure 8.

Table 3: Structure of the ANN for classification of damage identification.

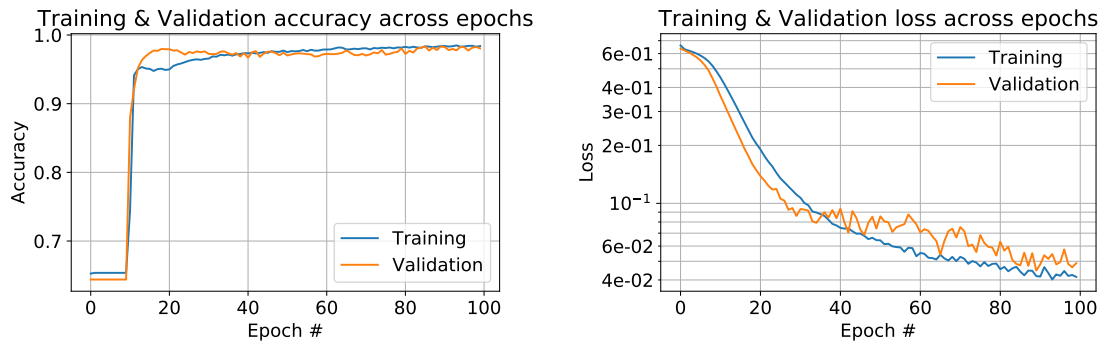| Input layer	Hidden layer 1	Hidden layer 2	Output layer	Loss function	Optimiser
$[\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{12}]^\text{T}$	$8 \times \text{ReLU}$	$10 \times \text{ReLU}$	$\begin{bmatrix} \Pr(\text{Intact}) \\ \Pr(\text{Damaged}) \end{bmatrix}$	Categorical cross-entropy	Adamax

Figure 8: Accuracy (left) and loss (right) function evolution during the training for the classification problem to determine damage initiation.

As a slightly more complex classification example problem, now we introduce 4 states for the failure

14

identification of the same 8-layered laminate regarding the damage variables of fibre and matrix of all laminae. The failure states are:

- No failure for all plies: $\Pr(\text{No failure}) := \Pr(d_{\text{f}} \leq 0.9 \text{ and } d_{\text{m}} \leq 0.9)$;

- Only fibre failure for any ply: $\Pr(\text{Fibre failure}) := \Pr(d_{\text{f}} > 0.9 \text{ and } d_{\text{m}} \leq 0.9)$;

- Only matrix failure for any ply: $\Pr(\text{Matrix failure}) := \Pr(d_{\text{f}} \leq 0.9 \text{ and } d_{\text{m}} > 0.9)$;

- Both fibre and matrix failure for any ply: $\Pr(\text{Both failure}) := \Pr(d_{\text{f}} > 0.9 \text{ and } d_{\text{m}} > 0.9)$.

Note that the failure threshold of damage variable is taken to be 0.9, so that the results would tend to be conservative and avoid the sharp increase of damage variable when approaching 1. All the failure states are mutually exclusive so that the conservation of probability is guaranteed. The hyperparameters for this failure identification ANN are listed in Table 4, and the evolution of accuracy and loss function plot are shown in Figure 9.

Table 4: Structure of the ANN for classification of failure identification.

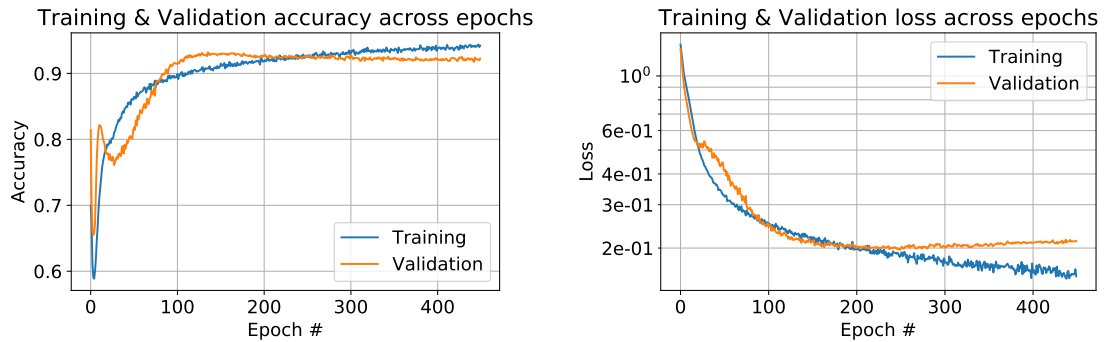| Input layer	Hidden layer 1	Hidden layer 2	Output layer	Loss function	Optimiser
$[\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{12}]^{\text{T}}$	$12 \times \text{ReLU}$	$20 \times \text{ReLU}$	$\begin{bmatrix} \Pr(\text{No failure}) \\ \Pr(\text{Fibre failure}) \\ \Pr(\text{Matrix failure}) \\ \Pr(\text{Both failure}) \end{bmatrix}$	Categorical cross-entropy	Adamax

Figure 9: Accuracy (left) and loss (right) function evolution during the training for the classification problem to determine material failure.

4.3. Validation of the trained ANN

During the training process, the dataset was split into two subsets: the majority was used for training the ANN, while a small portion, in this example 200 samples, was used for on-the-fly validation during the

15

training process. The orange curves in Figure 7, 8 and 9 show the evolution of accuracy and loss functions with the validation subset, which provides an indication of the immediate performance of the ANN being trained.

To perform an offline validation of the trained regression ANN, here we use a manipulated load case by controlling the strain with $\varepsilon_x \in [0, 0.02]$, while other strain components are constrained to zero. This load case is out of the training dataset, thus the ANN has never "seen" it during the training. The mesoscale UC modelling is used to provide a reference, with the same load case applied. Such a load case produces a uniaxial strain state, with the stress component σ_x being significantly larger than the other components. For an illustrative comparison, the σ_x-ε_x curves from the UC-based FE model and the ANN prediction are plotted in Figure 10 which agree well.
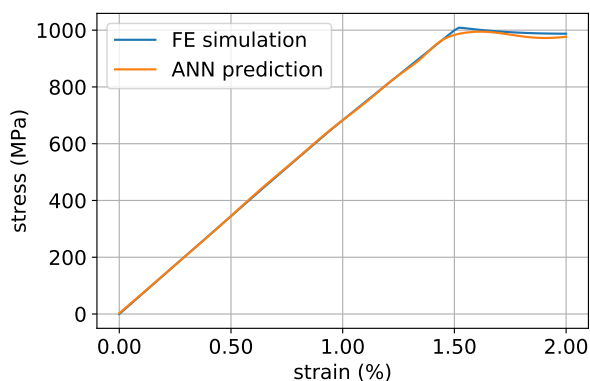
Figure 10: Comparison of surrogate model predicted and FE simulated nonlinear material stress-strain curve for a manipulated load case.

4.4. Example of application: plate with a hole under tension

To demonstrate the usage of the proposed surrogate model for structural damage analysis, the analysis of an open-hole laminated plate under in-plane tension was performed using the surrogate model, and compared with the results of conventional 3D finite element analysis. For structural analysis application, the above trained ANNs were deployed through a UMAT subroutine, in which the stresses are updated by the prediction from the trained ANNs. By using the surrogate model, the open-hole plate can be modelled as a single layer 2D model with homogenised material properties defined by the ANN subroutine, whilst the reference model is a multi-layer 3D solid model running with Linde's damage model subroutine and mesoscale lamina property. An illustration of the two finite element models is given in Figure 11, both of which were solved using Abaqus/Standard 2016 on the same workstation equipped with Intel Xeon E5-1620 v4 CPU and 32 GB DDR4 RAM running Windows 10. The UMAT subroutine was compiled with Intel Fortran Compiler 2016.
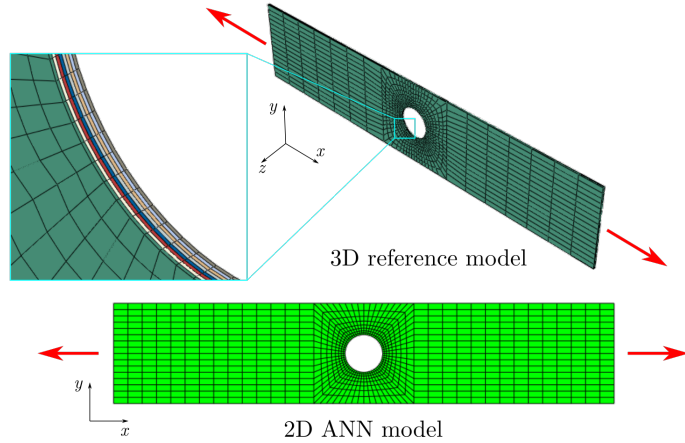
16

Figure 11: Illustration of the FE model of open-hole plate. Top: 3D reference model, layers are zoomed in and shown on the left. Bottom: 2D single layer model using ANN.

Figure 12 shows the comparison of load-displacement responses from the surrogate model and reference model for the open-hole plate under tension. In general, good agreement is observed with the exception of minor discrepancy near the ultimate strength. As the surrogate model is defined in terms of the homogenised materials at the macroscale, it may give sharp response near displacement at material failure because it loses the details caused by lamina failure as seen in the mesoscale model.
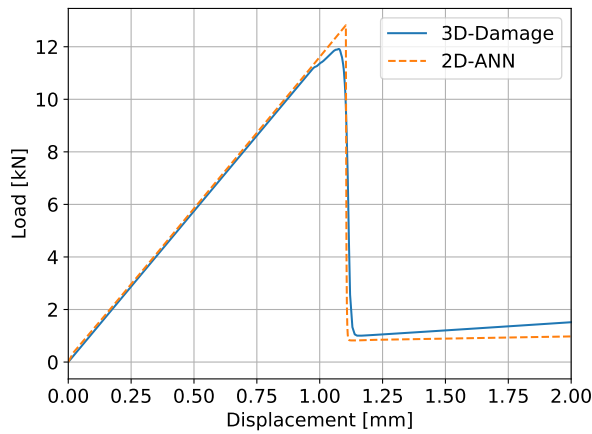
Figure 12: Load-displacement curves for the example problem simulated using 2D ANN surrogate model and the reference 3D damage model.

When comparing the damage events and locations, in the reference model the fibre damage initiates first near the hole (Figure 13) immediately followed by matrix damage (Figure 14) at the same location. The cracks then propagated rapidly to the whole cross section, as a result, the whole structure reached its maximum load carrying capacity. For the surrogate model, only the probability contour for damage

17

within both fibre and matrix damage is plotted in Figure 15 as the event for fibre damage only existed briefly. Besides, a video showing the evolution of the probability contours for all designed failure states with changes of loading displacement is provided in the Supplementary Data. Cracks (in terms of damage in both constituents) initiated at the same location and propagated towards the cross section similar to what was observed in the reference model. The displacement for crack initiation is around 1.10 mm close to 1.06 mm for the reference model when the fibre and matrix both have first reached a full damage state. It should also been noted that the surrogate model would loss the detail of ply damage prediction in the conventional FE models.
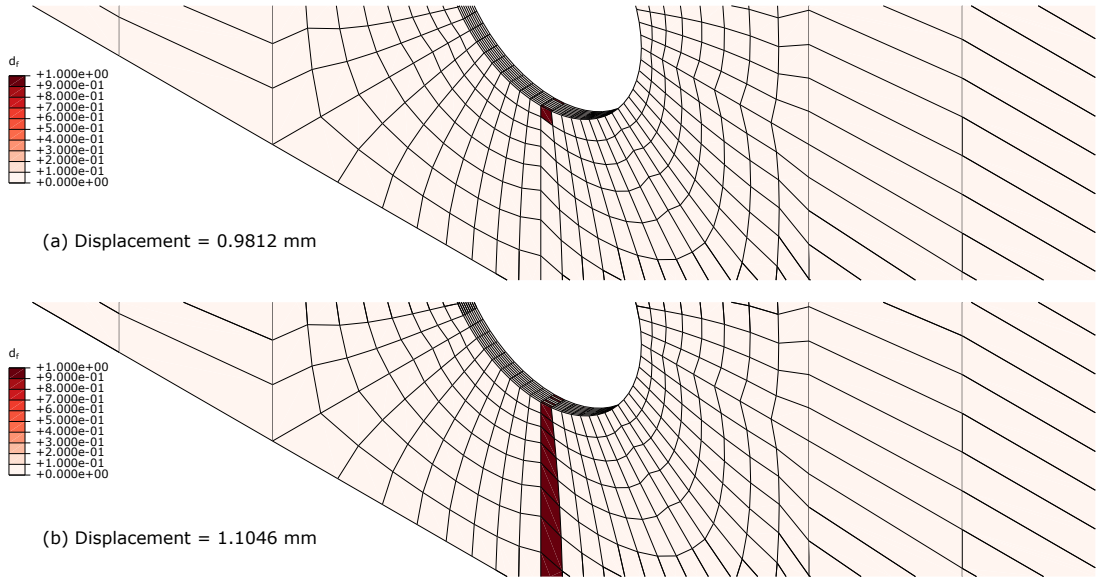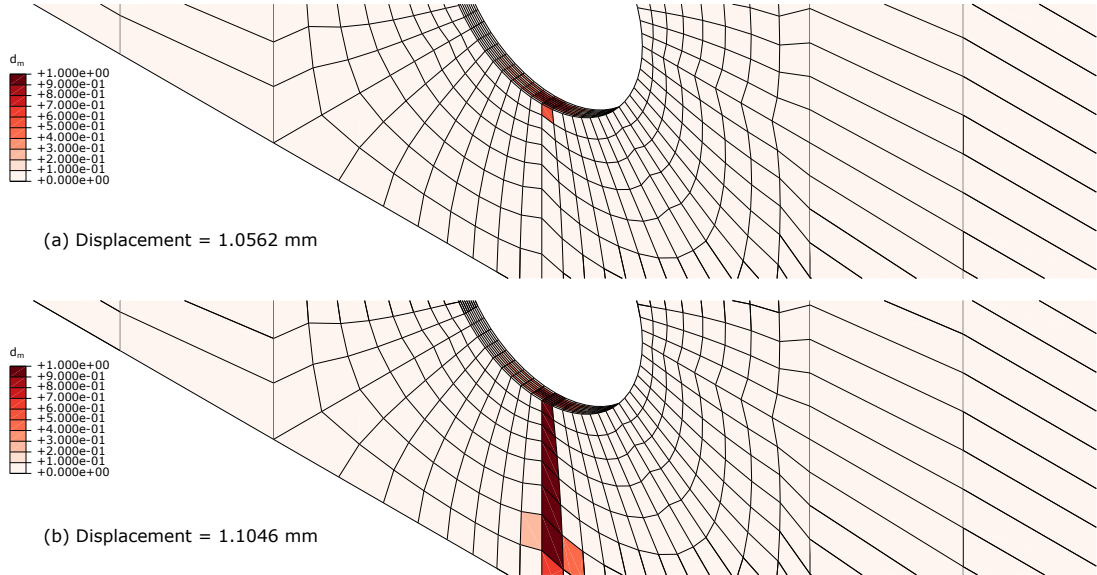

Figure 13: Contours for fibre damage variable d_f. (a): Damage initiated at displacement around 0.9812 mm. (b): Whole cross section are damaged at displacement around 1.1046 mm.

The computational costs for the two models are presented in Table 5. Using the surrogate model can offer significant computational benefits as the CPU time has been reduced to approximately 1/12 of the conventional FE solution. This great saving resulted not only from the decreased number of degrees of freedom (DOF) in the surrogate model, but also the reduced number of iterations and calculations in the material subroutine as the variables to be updated can be directly obtained from the trained ANNs.

Table 5: Comparison of computational costs.

Model	DOF	CPU time [s]	CPU	Wallclock time [s]	Acceleration
3D Ref.	22002	712.5	4	229	N/A
2D ANN	2116	59.7	1	60	×11.935

18

Figure 14: Contours for matrix damage variable d_{m}. (a): Damage initiated at displacement around 1.0562 mm. (b): Whole cross section are damaged at displacement around 1.1046 mm.
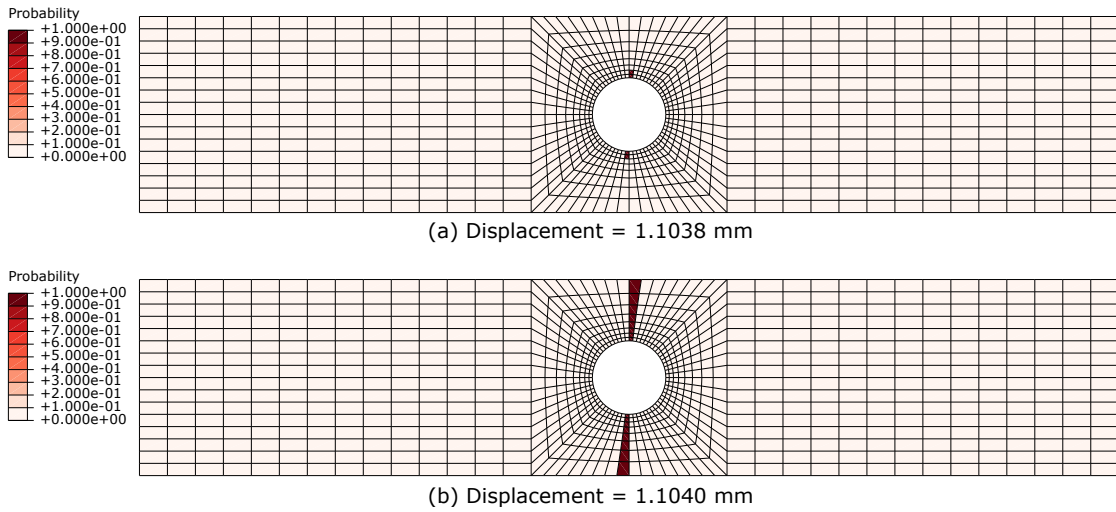
Figure 15: Contours for probability (Pr(Both failure)) of element failure. (a): Crack initiated at displacement around 1.1038 mm. (b): Whole cross section failure at displacement around 1.1040 mm.

5. Conclusions

A significant challenge for modelling the progressive damage behaviour of composite structures is the high computational cost. In this paper, an efficient multiscale framework based on surrogate modelling for composite materials considering progressive damage behaviour was proposed. This approach uses a surrogate model to represent the nonlinear effective stress-strain relationship and the possible failure information. Firstly, mesoscale unit cell modelling under designed load cases was performed to obtain the sampling data including effective stress/strain and damage parameters for the surrogate model. Artificial neural network algorithms were employed to construct the surrogate model by conducting regression for the constitutive law and classification for the damage information. The trained ANNs were deployed via an Abaqus user subroutine UMAT for use within finite element analysis. To demonstrate the proposed method, the analysis of an open-hole laminated plate under in-plane tension was performed using the surrogate model, and compared with the results of conventional 3D finite element analysis. The proposed model was found to offer huge computational benefits over conventional FE models, while maintaining sufficient levels of accuracy. Acting as a general material property database, the presented surrogate model is reusable for other structures made of the same material. However, the adopted neural network for regression requires a bijective function (one-to-one correspondence) between the input and output sampling dataset, and therefore it is only applicable for monotonic loading scenarios. Further investigation is needed for non-monotonic problems like cyclic loading.

20

References

[1] F. A. Leone, C. G. Dávila, D. Girolamo, Progressive damage analysis as a design tool for composite bonded joints, Composites Part B: Engineering 77 (0) (2015) 474–483. doi:http://dx.doi.org/10.1016/j.compositesb.2015.03.046.

[2] S. Yan, X. Zeng, A. Long, Meso-scale modelling of 3d woven composite t-joints with weave variations, Composites Science and Technology 171 (2019) 171–179. doi:https://doi.org/10.1016/j.compscitech.2018.12.024.

[3] J. Zhou, M. Z. Hassan, Z. Guan, W. J. Cantwell, The low velocity impact response of foam-based sandwich panels, Composites Science and Technology 72 (14) (2012) 1781 – 1790. doi:https://doi.org/10.1016/j.compscitech.2012.07.006.

[4] S. Hühne, J. Reinoso, E. Jansen, R. Rolfes, A two-way loose coupling procedure for investigating the buckling and damage behaviour of stiffened composite panels, Composite Structures 136 (Supplement C) (2016) 513–525. doi:https://doi.org/10.1016/j.compstruct.2015.09.056.

[5] M. Akterskaia, E. Jansen, S. Hallett, P. Weaver, R. Rolfes, Analysis of skin-stringer debonding in composite panels through a two-way global-local method, Composite Structuresdoi:10.1016/j.compstruct.2018.06.064.

[6] M. Akterskaia, E. Jansen, S. Hühne, R. Rolfes, Efficient progressive failure analysis of multi-stringer stiffened composite panels through a two-way loose coupling global-local approach, Composite Structures 183 (2018) 137–145. doi:https://doi.org/10.1016/j.compstruct.2017.02.011.

[7] B. E. Said, S. R. Hallett, Multiscale surrogate modelling of the elastic response of thick composite structures with embedded defects and features, Composite Structures 200 (2018) 781 – 798. doi:https://doi.org/10.1016/j.compstruct.2018.05.078.

[8] K. Lachhwani, Application of neural network models for mathematical programming problems: A state of art review, Archives of Computational Methods in Engineeringdoi:10.1007/s11831-018-09309-5.

[9] H. El Kadi, Modeling the mechanical behavior of fiber-reinforced polymeric composite materials using artificial neural networks—A review, Composite Structures 73 (1) (2006) 1–23. doi:https://doi.org/10.1016/j.compstruct.2005.01.020.

[10] Z. Zhang, K. Friedrich, Artificial neural networks applied to polymer composites: a review, Composites Science and Technology 63 (14) (2003) 2029 – 2044, polymer Composites: Design, Materials, Manufacturing, Dedicated to Professor M. Neitzel. doi:https://doi.org/10.1016/S0266-3538(03)00106-4.

[11] G. Balokas, S. Czichon, R. Rolfes, Neural network assisted multiscale analysis for the elastic properties prediction of 3D braided composites under uncertainty, Composite Structures 183 (2018) 550–562. doi:https://doi.org/10.1016/j.compstruct.2017.06.037.

[12] G. A. B. Oliveira, E. C. B. Câmara, R. C. S. Freire Júnior, Obtaining g12 and xt using mixed anns based on matrix and fiber properties, Composites Part B: Engineering 175 (2019) 107171.

[13] Q. Rong, H. Wei, X. Huang, H. Bao, Predicting the effective thermal conductivity of composites from cross sections images using deep learning methods, Composites Science and Technology (2019) 107861doi:https://doi.org/10.1016/j.compscitech.2019.107861.

[14] N. Farhana, M. A. Majid, M. Paulraj, E. Ahmadhilmi, M. Fakhzan, A. Gibson, A novel vibration based non-destructive testing for predicting glass fibre/matrix volume fraction in composites using a neural network model, Composite Structures 144 (2016) 96 – 107. doi:https://doi.org/10.1016/j.compstruct.2016.02.066.

[15] A. Khan, D.-K. Ko, S. C. Lim, H. S. Kim, Structural vibration-based classification and prediction of delamination in smart composite laminates using deep learning neural network, Composites Part B: Engineering 161 (2019) 586–594. doi:https://doi.org/10.1016/j.compositesb.2018.12.118.

[16] J. Y. Ang, M. S. Abdul Majid, A. Mohd Nor, S. Yaacob, M. J. M. Ridzuan, First-ply failure prediction of glass/epoxy

composite pipes using an artificial neural network model, Composite Structures 200 (2018) 579–588. `doi:https://doi.org/10.1016/j.compstruct.2018.05.139`.

[17] S. Li, A. Wongsto, Unit cells for micromechanical analyses of particle-reinforced composites, Mechanics of Materials 36 (7) (2004) 543–572. `doi:http://dx.doi.org/10.1016/S0167-6636(03)00062-0`.

[18] D. S. Ivanov, F. Baudry, B. Van Den Broucke, S. V. Lomov, H. Xie, I. Verpoest, Failure analysis of triaxial braided composite, Composites Science and Technology 69 (9) (2009) 1372–1380. `doi:http://dx.doi.org/10.1016/j.compscitech.2008.09.013`.

[19] T. Wehrkamp-Richter, N. V. D. Carvalho, S. T. Pinho, Predicting the non-linear mechanical response of triaxial braided composites, Composites Part A: Applied Science and Manufacturing 114 (2018) 117 – 135. `doi:https://doi.org/10.1016/j.compositesa.2018.08.011`.

[20] R. M. Christensen, Failure criteria for fiber composite materials, the astonishing sixty year search, definitive usable results, Composites Science and Technology 182 (2019) 107718. `doi:https://doi.org/10.1016/j.compscitech.2019.107718`.

[21] Q. Sun, G. Zhou, Z. Meng, H. Guo, Z. Chen, H. Liu, H. Kang, S. Keten, X. Su, Failure criteria of unidirectional carbon fiber reinforced polymer composites informed by a computational micromechanics model, Composites Science and Technology 172 (2019) 81–95. `doi:https://doi.org/10.1016/j.compscitech.2019.01.012`.

[22] A. S. Kaddour, M. J. Hinton, Maturity of 3d failure criteria for fibre-reinforced composites: Comparison between theories and experiments: Part b of wwfe-ii, Journal of Composite Materials 47 (6-7) (2013) 925–966. `doi:10.1177/0021998313478710`.

[23] P. Linde, J. Pleitner, H. de Boer, C. Carmone, Modelling and simulation of fibre metal laminates, in: ABAQUS Users' conference, 2004, pp. 421–439.

[24] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016, `http://www.deeplearningbook.org`.

[25] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization (2014). `arXiv:1412.6980`.

[26] T. Tieleman, G. Hinton, Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, COURSERA: Neural networks for machine learning 4 (2) (2012) 26–31.

[27] A. Kaddour, M. Hinton, Input data for test cases used in benchmarking triaxial failure theories of composites, Journal of Composite Materials 46 (19-20) (2012) 2295–2312. `doi:10.1177/0021998312449886`.

[28] Keras documentation, `https://keras.io/`, accessed: 2019-09-30.

[29] Theano Development Team, Theano: A Python framework for fast computation of mathematical expressions, arXiv e-prints abs/1605.02688.
URL `http://arxiv.org/abs/1605.02688`

[30] F. Hutter, J. Lücke, L. Schmidt-Thieme, Beyond manual tuning of hyperparameters, KI-Künstliche Intelligenz 29 (4) (2015) 329–337.