

# The Virtual Fields Method to Indirectly Train Artificial Neural Networks for Implicit Constitutive Modelling

Rúben Lourenço<sup>1,a\*</sup>, António Andrade-Campos<sup>1,b</sup>, Pétia Georgieva<sup>2,c</sup>

<sup>1</sup>Dept. of Mechanical Engineering, Centre for Mechanical Technology and Automation, University of Aveiro, Portugal

<sup>2</sup>Dept. of Electronics Telecommunications and Informatics, Institute of Electronics and Informatics Engineering, University of Aveiro, Portugal

<sup>a</sup>rubenl@ua.pt, <sup>b</sup>gilac@ua.pt, <sup>c</sup>petia@ua.pt

**Keywords:** Constitutive Model, Elastoplasticity, Neural Networks, Indirect Training, Equilibrium Conditions

**Abstract.** In the current work, displacement and global force data are used to feed and indirectly train an ANN to predict the stress state of a material. An experimental test is recreated numerically in order to obtain displacement and global force data for different load distributions, i.e. obtaining synthetic data using a virtual experiment. The strain from the current and previous time increments are indirectly obtained from the corresponding displacements and used as inputs for the ANN to predict the current state of stress. Training is carried out without stress labels to compute the loss. Instead, the local and global equilibrium conditions corresponding to the application of the Virtual Fields Method (VFM) to the physical model are employed to compute the loss and update the network parameters, until the predicted stress state is accurate.

## Introduction

Constitutive laws are established according to first-principle assumptions compiled in a set of mathematical expressions, supported by a number of empirical parameters that need to be calibrated via experimentation. Materials with complex behaviors require complex models with a higher number of parameters, resulting in expensive and time-consuming experimental campaigns [1].

Artificial Neural Networks (ANNs) have the potential to provide a radically different approach to the field, as they are able to implicitly learn patterns directly from data, without having to postulate a mathematical formulation or identify parameters [2, 3]. Several successful applications of ANNs targeted at the implicit modelling of material behavior have been reported in the literature (e.g. [4–6] among others). However, the vast majority of the approaches rely on training the ANNs with paired data, usually stress-strain, from numerically generated datasets. However, with numerical generated data all variables are easily accessible, while in a real experimental setting certain variables (e.g. stress) are not directly measured, only the displacements and global force are measurable quantities and should be used to indirectly train the ANN model.

In this work, displacement and global force data are used to indirectly train an ANN model to predict the stress state of a material. A virtual experiment was recreated in order to obtain the necessary variables considering different load distributions. The strains corresponding to two subsequent time increments and used as inputs for an ANN to predict the current state of stress. Training is carried out without stress labels. Instead, the local and global equilibrium conditions corresponding to the application of the Virtual Fields Method (VFM) [1] to the physical model are employed to compute the loss and update the network parameters. Hence, the main objective of this paper is to show the VFM as capable and viable method to indirectly train ANNs for implicit constitutive modelling.

## Fundamentals on ANNs

**Direct training.** Feedforward neural networks (FFNNs) are the most commonly used topology, consisting of: an input layer, an output layer and one or more hidden layers, which provide complexity for non-linear problems [7]. Considering the generic FFNN, shown in Fig. 1, the dimensions of the vector  $\mathbf{x}$  dictate the number of neurons in the input layer [7]. These inputs are subsequently mapped to the next layers, triggering the respective activations  $\{a_1^{(1)}, \dots, a_m^{(k)}\}$  that form a vector  $\mathbf{a}^{(k)} \in \mathbb{R}^m$ . The activation potential, from layer  $(k-1)$  to layer  $k$ , is controlled by a matrix of parameters  $\mathbf{W}^{(k)} \in \mathbb{R}^{m \times n}$  and computed as the sum of the weighted output values  $\{x_1, \dots, x_n\}$  of all incoming connections. A function  $g(\cdot)$  is then applied to this weighted sum, leading to the activations of the  $k^{\text{th}}$  layer being computed as [7]:

$$\mathbf{a}^{(k)} = g\left(\mathbf{W}^{(k)}\mathbf{x} + \mathbf{b}^{(k)}\right). \quad (1)$$

A set of biases  $\mathbf{b} \in \mathbb{R}^m$  is added as the weight of a link that always transmits a value of 1. During training, the network learns the parameters  $(\mathbf{W}, \mathbf{b})$  that minimize a given loss function. A standard supervised learning procedure uses labeled datasets, where each training sample is associated with an observed value [8]. In the case of implicit constitutive modelling, usually one seeks to predict the stress components, therefore, stress labels are fed to the network in order to compute the cost, such that [4]:

$$L(\boldsymbol{\sigma}, \hat{\boldsymbol{\sigma}}) = \frac{1}{n} \sum_{i=0}^n (\sigma_i - \hat{\sigma}_i)^2, \quad (2)$$

where  $n$  is the total number of training instances,  $\boldsymbol{\sigma}$  is the vector of observed values and  $\hat{\boldsymbol{\sigma}}$  the vector of predictions. There is a wide variety of optimization algorithms available to minimize the cost function. A gradient-based algorithm is normally used to minimize the cost and drive the parameters update, such that [7, 9]:

$$\mathbf{W}^{(k)} := \mathbf{W}^{(k)} - \alpha \frac{\partial L}{\partial \mathbf{W}^{(k)}}, \quad (3)$$

$$\mathbf{b}^{(k)} := \mathbf{b}^{(k)} - \alpha \frac{\partial L}{\partial \mathbf{b}^{(k)}}, \quad (4)$$

with  $\alpha$  being the learning rate, controlling how quickly the model adapts to the problem [9]. The partial derivatives of the loss with respect to the parameters can be efficiently computed by an error backpropagation.

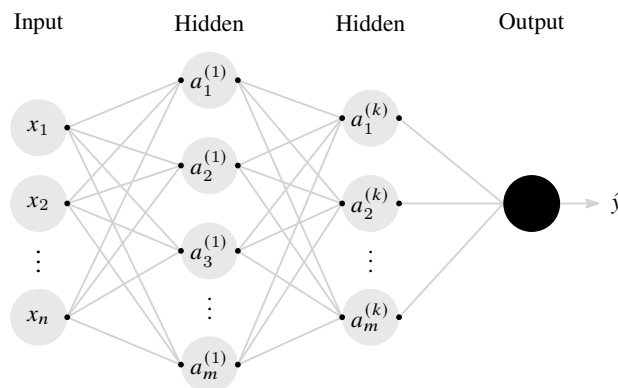


Fig. 1: Multi-layer feedforward neural network with one output and  $n$  inputs.

**Indirect training.** Implicit constitutive modelling based on ANNs mostly relies on paired data, with the strain and stress tensors usually being fed to the system in order to learn the material behavior. However, variables such as stress cannot be directly measured from experiments [10]. Therefore, the training process must be carried out indirectly, using only measurable data. Some authors recently reported different approaches to tackle this issue. For example, Xu *et al* [11] presented a method for an ANN to model viscoelasticity, based on displacement data, using partial differential equations to introduce the physical constraints during training, and Liu *et al* [10] addressed the issue via coupling the ANN model with the Finite Element Method (FEM) to learn constitutive laws based on force and displacement data (Fig. 2). In the latter approach, the physical constraints are implicitly imposed as the network's outputs must go through the FEM to generate correct input data.

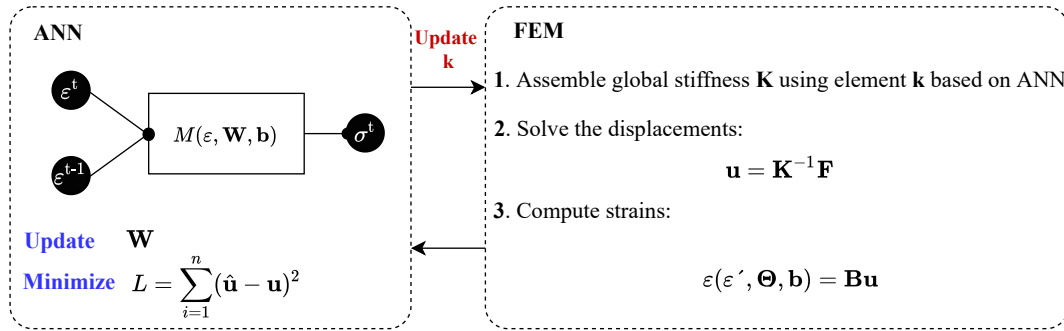


Fig. 2: Coupled ANN-FEM model approach presented by Liu *et al*. Adapted from [10].

In the current work we propose a disruptive approach in comparison to Liu *et al* [10], making use of the VFM to guarantee the global equilibrium (Fig. 3). The VFM, first introduced by Grédiac [12], is known by its computational efficiency and does not need to resort to FEM in order to conduct any forward calculations [13]. The key elements behind the VFM are the Principle of Virtual Work (PVW) and the choice of virtual fields. According to the PVW, the internal virtual work must be equal to the external virtual work performed by the external forces, and is written by [14]:

$$-\int_V \boldsymbol{\sigma} : \boldsymbol{\varepsilon}^* dV + \int_{\partial V} \mathbf{T} \cdot \mathbf{u}^* dS = 0, \quad (5)$$

where  $\boldsymbol{\varepsilon}^*$  is the virtual strain,  $\mathbf{u}^*$  is the virtual displacement,  $V$  is the volume of the solid and  $\mathbf{T}$  is the traction vector. These virtual entities are mathematical test functions, work as weights and can be defined independently of the measured displacements/strains. An infinite number of virtual fields can be used, nonetheless the following two conditions should be met [1, 14]: the chosen virtual fields should be kinematically admissible, meaning that the displacement boundary conditions must be satisfied, and the virtual fields should be constant along the boundary where the force is applied.

By coupling the VFM with the ANN model, one can use force and displacement data to indirectly train the ANN. The strains are obtained from the displacements and fed as inputs to the neural network, which will provide the stress tensor components. Then the stress equilibrium is evaluated globally by means of the PVW and the parameters  $(\mathbf{W}, \mathbf{b})$  are optimized until the equilibrium is respected, that is, by minimizing the loss:

$$L = \frac{1}{n_V} \sum_{i=1}^{n_V} \left( -\int_V \boldsymbol{\sigma} : \boldsymbol{\varepsilon}^* dV + \int_{\partial V} \mathbf{T} \cdot \mathbf{u}^* dS \right)^2, \quad (6)$$

where  $n_V$  represents the number of virtual fields.

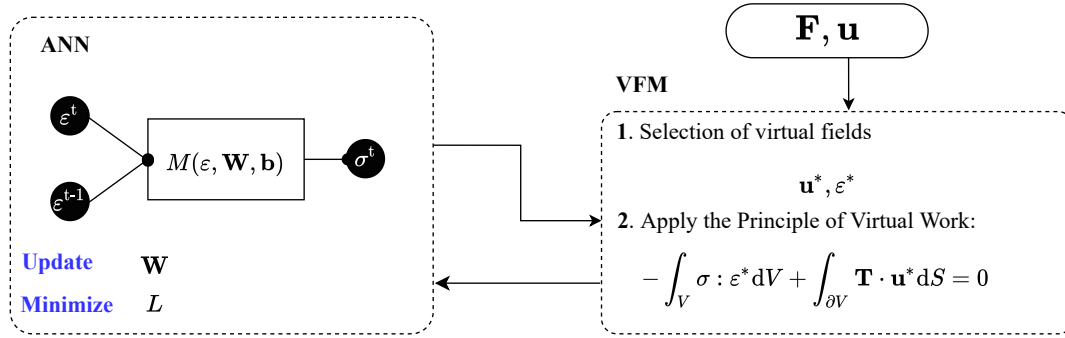


Fig. 3: Coupled ANN-VFM method for implicit constitutive modelling. The variables  $\varepsilon^t$  and  $\varepsilon^{t-1}$  are the deformation tensors fed as input to the ANN in order to obtain the output  $\sigma^t$ , the stress tensor.

### Application Example: Implicit Elastoplastic Modelling Using the VFM

**Numerical model.** The heterogeneous test developed by Martins *et al* [15] was chosen in order to generate synthetic experimental data to train the ANN model. The configuration consists of a solid  $3 \times 3 \text{ mm}^2$  plate with a thickness  $t = 0.1 \text{ mm}$ . The physical domain is discretized by a mesh with a total of 9 four-node bilinear plane stress elements. The initial mesh, geometry and boundary conditions are depicted in Fig. 4. Symmetry boundary conditions are applied to the boundaries at  $x = 0$  and  $y = 0$ , and a surface traction is applied to the boundary at  $x = 3 \text{ mm}$ . The traction follows a non-uniform distribution and is composed by a single component along the  $x$ -direction, which varies linearly in the  $y$ -direction according to:  $f_x(y) = my + b$ , where  $m$  and  $b$  respectively control the slope and intercept of the distribution.

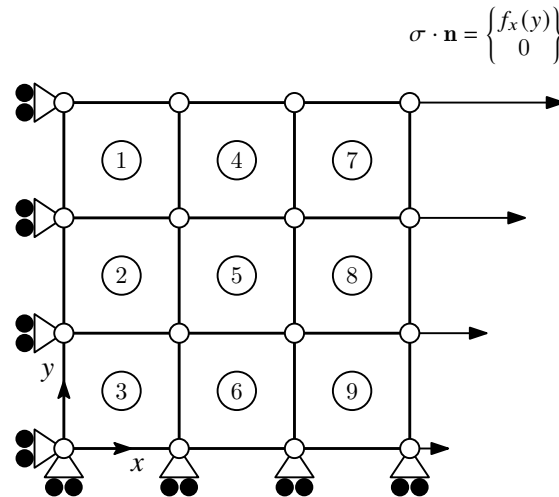


Fig. 4: Heterogeneous test: initial geometry, mesh and boundary conditions.

The numerical simulations were conducted using the commercial finite element code Abaqus. The model was built with CPS4R elements (bilinear reduced integration plane stress). The material was simulated employing a non-linear isotropic elasto-plastic model, with the isotropic hardening response obeying to the Swift's law, given by:

$$\sigma_y = K(\varepsilon_0 + \bar{\varepsilon}^p)^n, \quad (7)$$

where  $\sigma_y$  is the flow stress,  $K$  is a hardening coefficient,  $n$  is the hardening exponent,  $\sigma_0$  is the yield stress and  $\varepsilon_0$  the deformation at the yielding point, computed as:



$$\varepsilon_0 = \left( \frac{\sigma_0}{K} \right)^{1/n}. \quad (8)$$

The elastic parameters were defined as  $E = 210$  GPa and  $\nu = 0.3$ , while  $\sigma_0 = 160$  MPa,  $K = 565$  MPa and  $n = 0.26$  were adopted for the Swift's hardening law, corresponding to a soft steel.

**Data generation and ANN training.** To generate the training data, all simulations were performed using a small displacement formulation, with the time period set to 1 and using a fixed time increment  $\Delta t = 0.001$ . For each time increment, the deformation components at the centroid were extracted for all the elements and the global force was determined by means of computing the equilibrium of the internal forces, such that:

$$Fl = \sum_{i=1}^n \sigma_i A_i t, \quad (9)$$

where  $F$  is the global force,  $l$  is the length of the solid,  $A$  is the element area and  $t$  is the thickness. The training data was generated for different load distributions, keeping the slope fixed at  $m = 10$  N/mm and varying the intercept parameter, such that:  $b = \{50, 170, 270\}$  N. Prior to training and for each mechanical trial, the dataset was organized into batches of 9 elements per time increment and shuffled before being split into training (67%) and test data (33%). The input features were normalized to the interval  $[0, 1]$ . Two models were trained and aimed at predicting the linear elastic and elasto-plastic responses of the material. The latter included training samples with both elastic and plastic data. Once trained, the models were validated with different mechanical trials using the following load distributions:  $m = 12$ ,  $b = 100$  for the elastic model and  $m = 10$ ,  $b = 200$  for the elasto-plastic model.

The neural network model used for both cases was a FFNN with one hidden layer with of 8 neurons. The PReLU was chosen for activation function over a standard ReLU, due to the fact that the latter makes neurons with negative inputs to always output zero. Thus, the gradient flowing through them will forever be zero, irrespective of the input. The issue is commonly known as “dead” neurons and the PReLU circumvents it by having a slope for negative input values, thus making the gradients non-zero. The slope itself is a learnable parameter that the neural network automatically adjusts during training. The ANN architecture is summarized in Table 1.

Table 1: ANN model architecture.

Layers	Neurons	Activation	Weight initializer
Input	6	-	-
Hidden 1	8	PReLU	Kaiming He
Output	3	-	-

The inputs given to the model were the deformation components in the current and previous time increments,  $\varepsilon^t$  and  $\varepsilon^{t-1}$ , respectively, and the outputs were the stress tensor components at the current time increment,  $\sigma^t$ . The Adam algorithm was used to optimize the network weights, with the initial learning rate set to 0.1, scheduled to be reduced using a multiplier of 0.2 if no improvement in the training loss was registered after 3 epochs. For the elastic response model, the network was trained during 20 epochs and for the elasto-plastic response model the training was set to occur during a maximum of 150 epochs. However an early-stopping criteria was defined such that it would be interrupted if no further improvement was registered in the test loss. The complete set of virtual fields used to train both models is shown in Fig. 5.

**Results and discussion.** The learning curves for both models are depicted in Fig. 6. The plots show that loss decreases sharply during training after some point for both models, but convergence is achieved earlier for the elastic model which was easier to train using only 2 virtual fields. For the elasto-plastic response model 15 virtual fields were used during training, substantially lowering the initial loss (Fig. 6(a)).

The validation results (Fig. 7) show that the ANN was able to learn both elastic and plastic behaviors, providing reasonable predictions for the stress response along the  $x$ -direction, indicating that the VFM approach is at least working. Although using additional virtual fields improves the shear stress response just enough for the plastic model (Fig. 7(b)), in general the ANN has a noticeable lower sensitivity to the stress responses along  $y$  and  $xy$ . The issue may be due to fact that either the number of chosen virtual fields is not enough to capture the material behavior or the chosen set of virtual fields provides more weight for the stresses along  $x$  in detriment of the remaining components. Another factor at play here is that the virtual fields were chosen manually. This strategy is often used for non-linear models and is the easiest to implement, nonetheless it does not guarantee the chosen virtual fields produce the best results and is tied to the expertise of the user [15].

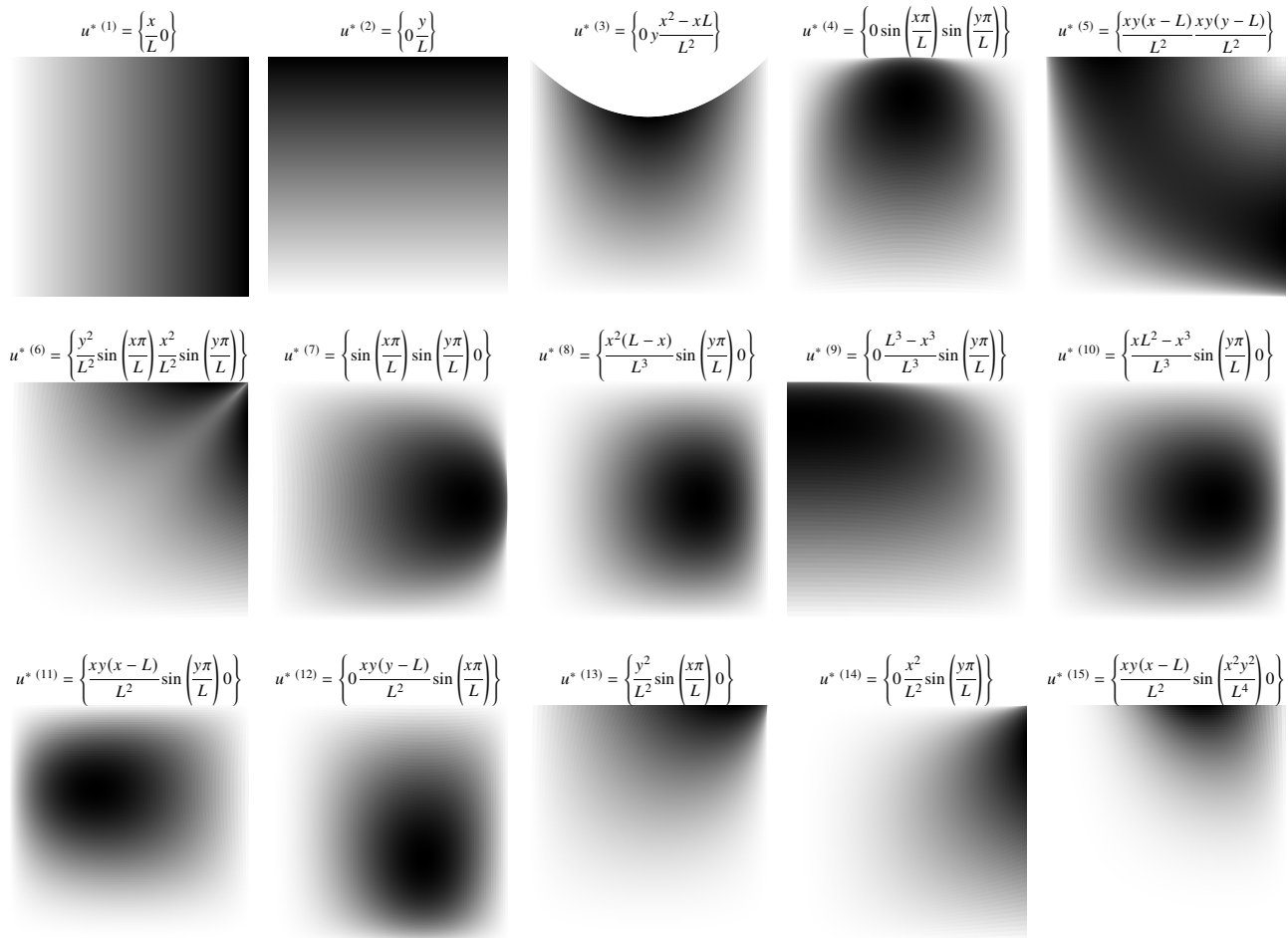


Fig. 5: Complete set of virtual fields used to train the both models. Only  $u^{*(1)}$  and  $u^{*(2)}$  were used to train the elastic model.

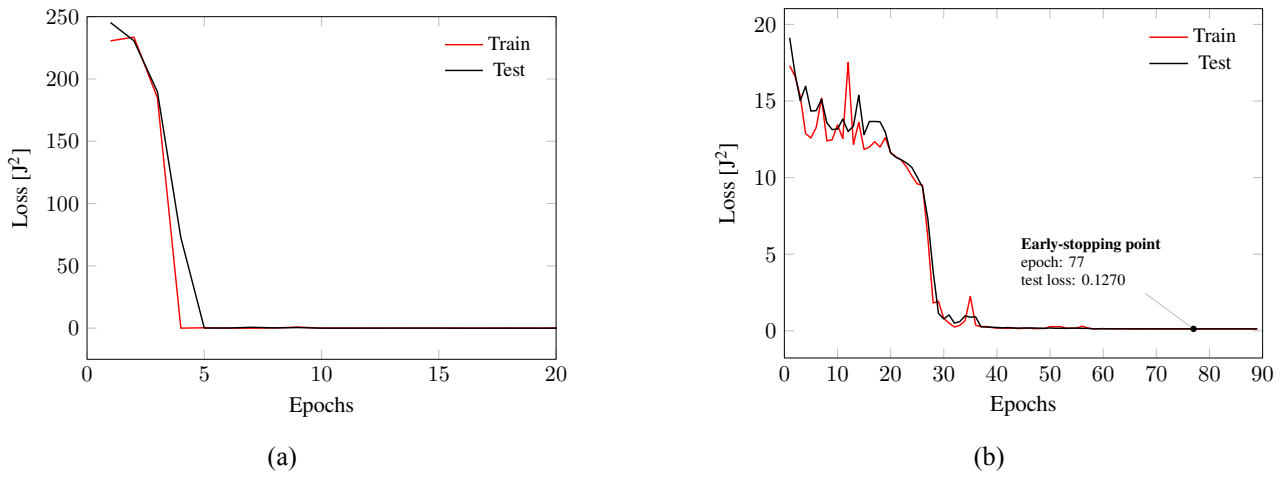
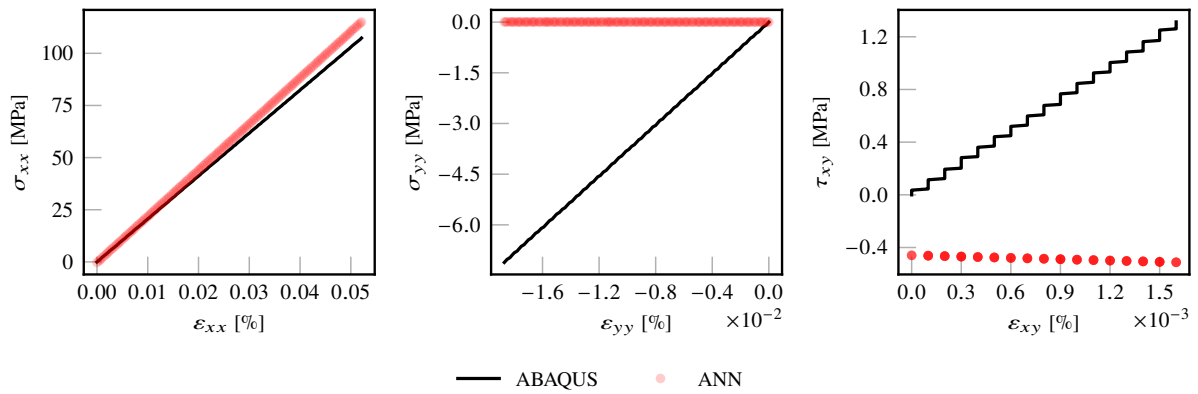
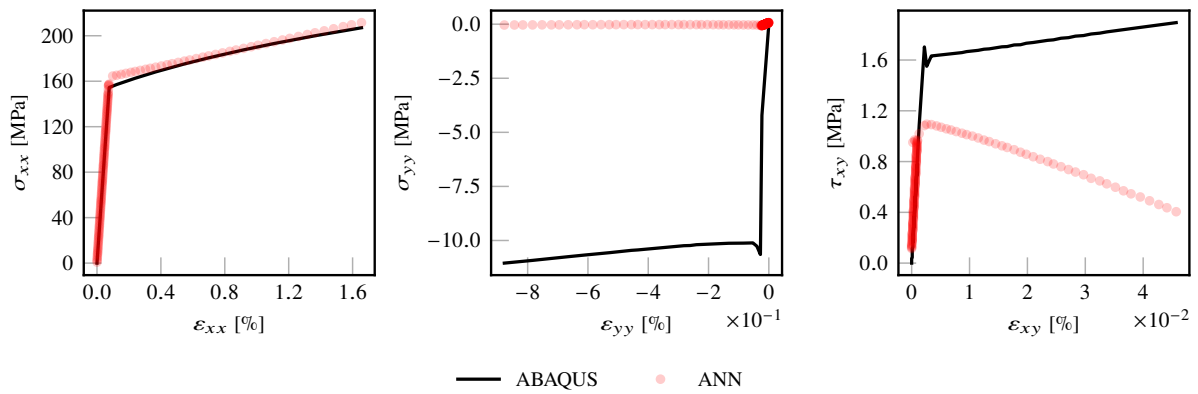


Fig. 6: Learning curves for (a) the elastic and (b) the elasto-plastic response models.



(a) Element #9:  $m = 12$ ,  $b = 100$



(b) Element #9:  $m = 10$ ,  $b = 200$

Fig. 7: Learning curves for trained (a) elastic and (b) elasto-plastic response models.

## Conclusions

Implicit constitutive modelling using ANNs does not require a mathematical formulation or parameter identification, providing a radically different approach to the field. A new methodology to indirectly train an ANN-based constitutive model was proposed using the VFM. The new approach allows to train the model using directly measurable data, e.g. displacements and global force, and does not require the FEM for further calculations. According to the methodology two models were trained to capture the elastic and elasto-plastic responses of a fictional material, using a set of hand-picked virtual fields. The results show that both models were capable of learning the constitutive behavior, with convergence being achieved very early during training. The models were able to provide good predictions for the stress response along the main direction of applied force. However, in general, predictions were poor for the remaining stress components. It is evident that the addition of virtual fields helps to reduce the training loss, nonetheless it is not enough to guarantee a good accuracy, which may indicate that either the number of virtual fields was not enough or the chosen set privileges more the stress response in one direction in detriment of the others. In that case, a more careful selection of virtual fields may improve the results. Another factor at play here that may explain the lack of accuracy for  $\sigma_{yy}$  and  $\tau_{xy}$  is the optimization process being unconstrained. As such, there may be multiple possible solutions that provide the equilibrium between the internal and external virtual works, thus minimizing the loss, however, many of those solutions may not have physical meaning. Albeit with margin for further improvements in the future, we consider the application example has effectively proven that this new methodology works.

## Acknowledgements

Rúben Lourenço acknowledges the Portuguese Foundation for Science and Technology (FCT) for the financial support provided through the grant 2020.05279.BD, co-financed by the European Social Fund, through the Regional Operational Programme CENTRO 2020. The authors also gratefully acknowledge the financial support of the Portuguese Foundation for Science and Technology (FCT) under the projects CENTRO-01-0145-FEDER-029713, POCI-01-0145-FEDER-031243 and POCI-01-0145-FEDER-030592 by UE/FEDER through the programs CENTRO 2020 and COMPETE 2020, and UIDB/00481/2020 and UIDP/00481/2020-FCT under CENTRO-01-0145-FEDER-022083. This project also supported by the Research Fund for Coal and Steel under grant agreement No 888153.

## References

- [1] M. Grédiac, F. Pierron, S. Avril, and E. Toussaint. The virtual fields method for extracting constitutive parameters from full-field measurements: a review. *Strain*, 42(4):233–253, 2006.
- [2] Y. C. Lin, Jun Zhang, and Jue Zhong. Application of neural networks to predict the elevated temperature flow behavior of a low alloy steel. *Computational Materials Science*, 43(4):752–758, oct 2008.
- [3] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.
- [4] Jamshid Ghaboussi, David A. Pecknold, Mingfu Zhang, and Rami M. Haj-Ali. Autoprogressive training of neural network constitutive models. *International Journal for Numerical Methods in Engineering*, 42(1):105–126, may 1998.
- [5] Annan Zhang and Dirk Mohr. Using neural networks to represent von Mises plasticity with isotropic hardening. *International Journal of Plasticity*, 132:102732, sep 2020.

- 
- [6] Filippo Masi, Ioannis Stefanou, Paolo Vannucci, and Victor Maffi-Berthier. Thermodynamics-based Artificial Neural Networks for constitutive modeling. *Journal of the Mechanics and Physics of Solids*, 147:104277, feb 2021.
- [7] Charu C. Aggarwal. *Neural Networks and Deep Learning*. Springer, 2018.
- [8] Wesam Salah Alaloul and Abdul Hannan Qureshi. Data processing using artificial neural networks. In Dinesh G. Harkut, editor, *Dynamic Data Assimilation - Beating the Uncertainties*. IntechOpen, 2020.
- [9] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.
- [10] Xin Liu, Fei Tao, Haodong Du, Wenbin Yu, and Kailai Xu. Learning nonlinear constitutive laws using neural network models based on indirectly measurable data. *Journal of Applied Mechanics, Transactions ASME*, 87(8):1–8, 2020.
- [11] Kailai Xu, Alexandre M. Tartakovsky, Jeff Burghardt, and Eric Darve. Learning viscoelasticity models from indirect data using deep neural networks. *Computer Methods in Applied Mechanics and Engineering*, 387:114124, dec 2021.
- [12] M. Grédiac. Principe des travaux virtuels et identification. *Comptes Rendus de l'Académie des Sciences*, 309(309):1–5, 1989.
- [13] Aleksander Marek, Frances M Davis, Fabrice Pierron, and Fabrice Pierron F Pierron. Sensitivity-based virtual fields for the non-linear virtual fields method. *Computational Mechanics*, 60:409–431, 2017.
- [14] F. Pierron and M. Grédiac. *The virtual fields method: extracting constitutive mechanical parameters from full-field deformation measurements*. Springer Science & Business Media, 2012.
- [15] J.M.P. Martins, A. Andrade-Campos, and S. Thuillier. Comparison of inverse identification strategies for constitutive mechanical models using full-field measurements. *International Journal of Mechanical Sciences*, 145:330 – 345, 2018.