Numerical simulation of heterogeneous materials combining Artificial Intelligence and physics-based modeling

Aymen Danoun

université de BORDEAUX

THÈSE PRÉSENTÉE

POUR OBTENIR LE GRADE DE

DOCTEUR

DE L'UNIVERSITÉ DE BORDEAUX

ECOLE DOCTORALE SCIENCES PHYSIQUES ET DE L'INGÉNIEUR

Spécialité **Mécanique**

Par **Aymen Danoun**

Simulation numérique de matériaux hétérogènes non-linéaires : Intelligence artificielle et méta-modèles de comportement

Sous la direction de : **Yves Chemisky**
Co-encadrement : **Etienne Prulière**

Soutenue le 8 Novembre 2022

Membres du jury :

M. David Ryckelynck	Professeur des universités	MINES ParisTech	Président
M. Julien Yvonnet	Professeur des universités	Université Gustave Eiffel	Rapporteur
M. Björn Kiefer	Professeur des universités	École des mines de Freiberg	Rapporteur
M. Justin Dirrenberger	Maître de conférences, HDR	CNAM Paris	Examinateur
M. Etienne Prulière	Maître de conférences	Arts et Métiers Bordeaux	Examinateur
M. Yves Chemisky	Professeur des universités	Université de Bordeaux	Directeur de thèse

UNIVERSITY OF BORDEAUX
DOCTORAL SCHOOL EDSPI
SCIENCES PHYSIQUES ET DE L'INGÉNIEUR

P H D T H E S I S

to obtain the title of

PhD in Engineering Sciences

of University of Bordeaux

Defended by

Aymen DANOUN

Numerical simulation of heterogeneous materials combining Artificial Intelligence and physics-based modeling

Thesis Advisor: Yves CHEMISKY

Thesis Co-supervisor: Etienne PRULIERE

Defended on November 8, 2022

Jury :

Reviewers :	Prof. Julien YVONNET	-	Université Gustave Eiffel
	Prof. Björn KIEFER	-	TU Bergakademie Freiberg
Examinators :	Prof. Justin DIRRENBERGER	-	CNAM Paris
	Prof. David RYCKELYNCK	-	MINES ParisTech
Thesis Advisor :	Prof. Yves CHEMISKY	-	Université de Bordeaux
Thesis Co-supervisor :	Prof. Etienne PRULIÈRE	-	Arts et Métiers Bordeaux

Simulation numérique de matériaux hétérogènes non-linéaires: Intelligence artificielle et méta-modèles de comportement

Résumé : L'innovation technologique s'appuie de plus en plus sur l'utilisation des matériaux hétérogènes pour des domaines de pointe (Énergies renouvelables, aéronautique, biomécanique). L'utilisation de ces matériaux est devenue incontournable en raison de leur excellentes propriétés mécaniques spécifiques. Cependant, le comportement thermomécanique résultant de ces microstructures est très complexe. Il présente des mécanismes de déformation non-linéaires dont l'identification demeure un véritable challenge. D'autre part, le comportement de ces matériaux est fortement dépendant du couple matériau-microstructure, il s'agit donc d'une problématique de modélisation multi-échelles. Des approches standard comme la méthode FE^2 ont été largement développées dans la littérature pour la simulation de la réponse mécanique de structures hétérogènes. Néanmoins, l'utilisation de cette méthode reste dans la plupart des cas très coûteuse en terme de temps de calcul et nécessite parfois des ressources informatiques assez robustes. Ce projet de thèse a donc pour objectif de repenser en profondeur le paradigme de la simulation multi-échelles du comportement mécanique des structures architecturés. En combinant Intelligence Artificielle (IA), thermodynamique des matériaux et simulation numérique du comportement de structures, nous avons l'ambition de développer et valider des modèles hybrides permettant de simuler, avec un gain de calcul très important, le comportement multi-échelle de structures hétérogènes. L'objectif est donc d'effectuer en quelques minutes, sur un ordinateur de bureau, des simulations prenant plusieurs jours sur des centres de calcul intensif.
Mots-clés : Intelligence Artificielle, Modélisation multi-échelle, Matériaux hétérogènes, Homogénéisation, Réseaux de neurones artificiels

---

Numerical simulation of heterogeneous materials combining Artificial Intelligence and physics-based modeling

Abstract: The new industrial constraints are nowadays pushing the limits of materials in terms of mechanical properties for advanced application fields (renewable energy, healthcare, transport). One solution to adapt the material properties to target applications is to rely on heterogeneous materials due to their excellent specific characteristics (high strength-to-weight ratio, energy absorption, mechanical resistance). However, the design of such structures can be challenging given the complexity of material mechanical responses when multiple physical mechanisms are involved at different length scales. Consequently, the evaluation of the resulting mechanical behaviors requires powerful numerical tools capable of solving highly non-linear multiscale problems. Standard methodologies including multi level finite element approaches FE^2 have been extensively developed in literature to simulate the macroscopic responses of heterogeneous structures. Nevertheless, the use of such methods still suffers from the high computational costs, therefore preventing it from being efficiently applied to multiscale structural analysis. The main objective of the present thesis is to propose a complete change of paradigm in the design and simulation of complex heterogeneous structures using hybrid models that combine physical knowledge and Artificial Neural Networks (ANN) based approaches. By adopting this strategy, we have observed a drastic downscale of the expected computational cost as well as the possibility of conducting expensive simulations on desktop computers instead of HPC clusters.
Keywords: Artificial Intelligence, Multiscale modeling, Heterogeneous materials, Homogenization, Artificial Neural Networks

---

Unité de recherche
Université de Bordeaux, I2M, UMR 5295 351 Cours de la Libération, 33400 Talence, France.

Remerciements - Acknowledgments

Suite à l'achèvement d'une étape importante de mon parcours scientifique, j'aimerais témoigner en quelques lignes ma profonde gratitude envers toutes les personnes qui ont contribué à l'élaboration de ce projet de thèse.

Je souhaite tout d'abord remercier mes parents pour leur amour inestimable, leurs sacrifices et leur soutien que ce soit moral ou financier. C'est grâce à vous que j'ai pu grandir dans les meilleures conditions et bénéficier d'une formation de qualité. Mes remerciements vont également à mes deux frères Anouar et Adham ainsi que ma petite soeur Alae pour leur encouragement durant cette période.

Je souhaite exprimer ma gratitude envers les membres du jury pour le temps qu'ils m'ont accordé et particulièrement les rapporteurs pour avoir accepté cette lourde tâche. C'est un très grand privilège pour moi d'avoir un jury de thèse composé de chercheurs reconnus au niveau international.

Je tiens à remercier mes encadrants de thèse pour leur engagement et leur accompagnement tout au long de ces trois années, j'espère avoir été à la hauteur de vos attentes. M.Yves Chemisky, je vous remercie profondément pour la confiance que vous m'avez accordée. Grâce à vous, j'ai pu bénéficier d'un encadrement de qualité, je vous remercie pour votre implication et vos nombreux conseils avisés et remarques constructifs qui m'ont permis de progresser en mécanique des matériaux. Merci aussi d'avoir rendu possible mes déplacements en conférences qui ont rythmé cette période de thèse. Mes remerciements vont également à M.Étienne Prulière pour son soutien considérable durant ces trois années, vous étiez toujours disponible pour m'assister lors des développements numériques et pour répondre à mes questions lors de mes passages réguliers à votre bureau. J'ai également énormément apprécié nos interactions et la qualité des échanges lors de nos réunions d'équipes.

Je remercie l'ensemble des membres du laboratoire I2M,DuMAS. Merci aux anciens et présents doctorants et post doctorants avec qui j'ai passé d'agréables moments: Kévin Marchais, Louise Le Barbenchon, Marie Pirotais, Rahul Subramanian, Hugo Roirand, Simon Lottier, Sarah Milhomme, Luc Brémaud, Vincent Longchamp, Julie Liegey, Quentin Henry, Vincent Fournier, Morgane Suhas, Dhawale Trunal Bhujangrao, Driss El khoukhi, Sudeep Sahoo.

Enfin, je remercie mes amis Oussama, Achraf, Mouaad, Walid, Ali et Ilyass pour leur encouragement durant ces trois années de thèse.

Que les oubliés me pardonnent ...

Contents

List of Figures

List of Tables

General Introduction

The use of heterogeneous materials has increasingly became more indispensable for advanced engineering applications, including renewable energy, healthcare and aerospace. For instance, composite materials have gained major prominence in the aeronautical industry due to their excellent and distinctive mechanical properties, which allow to design lightweight structures while maintaining high performance and meeting safety requirements. In addition, the use of composites presents a significant environmental impact in terms of energetic efficiency and CO_2 emissions reduction. The emergence of cellular architectured materials has also revolutionized a wide spectrum of engineering fields. Due to their high strength-to-weight ratio and significant energy absorption capabilities, lattice-type architectured structures can be employed for many applications, and their tunable mechanical properties is particularly suitable in biomechanics to conceive orthopedic implants. Through the optimization of microstructures topology, architectured prostheses can be designed in an optimal fashion to ensure a compatibility with human bones. Moreover, they provide both the appropriate porosity and mechanical strength to the bone tissue as well as the necessary lightness that allows the prosthesis to withstand significant loading conditions. The major growth of this type of materials is mainly due to the significant development of topological optimization and additive manufacturing methods which enable the production of quite complex shapes that were not accessible with traditional methods such as machining and casting processes.

However, modeling the thermomechanical response of heterogeneous structures can be very challenging due to many factors: First, the mechanical behavior resulting from the microstructures is very complex to identify as multiple physical mechanisms are usually involved at different length scales. Local material behaviors are particularly governed by strongly non-linear dissipative phenomena that requires a proper identification of the constitutive laws. In addition, multiscale modeling strategies are necessarily required to have a better understanding of the effect of the microstructure on the macroscopic response. Consequently, the evaluation of heterogeneous structures mechanical behaviors requires the development of advanced and powerful numerical tools, capable of taking into consideration the micro-macro interaction in real time and predicting accurately the resulting global response. Among these computational approaches, the multi level finite element method FE^2 [Feyel 1999] have been extensively developed in the literature for multiscale structural analysis. This full field approach based on periodic homogenization theory, consist on attributing a Representative Volume Element (RVE) to each Gauss integration point. The estimation of the overall response is then obtained by solving micro and macro problems simultaneously through localization and homogenization principles. Although FE^2 remains more advantageous in terms of execution time compared to direct FE simulations on fully meshed heterogeneous structures, it is clear that such modeling strategy also suffers from numerous limitations given the following reasons: as described in the above mentioned FE^2 procedure, nonlinear problems are solved simultaneously on both micro and macro scales using Newton Raphson iterative schemes. This process is commonly time consuming due to the treatment of material non-linearities in each microscopic computation and at each iteration. In addition, other considerations may increase the computational time including the number of Gauss integration points, the complexity of loading conditions, the number of increments and the meshing resolution of the RVE and the macrostructure. Therefore, all these considerations may sometimes hinder FE^2 approach from being efficiently applicable to structural analysis.

In the wake of the Fourth Industrial Revolution (4IR or Industry 4.0), numerous technological innovations are currently impacting a large span of our society from autonomous vehicles, robotics, the Internet of Things (IoT) and 3D printing. Artificial Intelligence (AI) is one of the most prominent technologies driving the Industry 4.0 through Machine Learning (ML) and Deep Learning (DL). For instance, machine learning had profoundly influenced the healthcare system in recent years, supporting the industry's transition from traditional methods. Through the development of comprehensive algorithms and softwares, doctors are currently assisted by ML in the diagnosing of patients and diseases. The impressive progress in the field of AI are only a reflection of the exponential growth of computing resources and the availability of massive amount of data. In material science, the recent developments in experimental measurement techniques have enabled the generation of large databases of material behaviors that can be used for constitutive models identification. On the other hand, the significant advances in computational capabilities have facilitated the production of extensive quantities of numerical simulations data. Therefore, the data availability, considered as the main source of the enrichment for AI models, is no longer a major obstacle. From this perspective, the employment of machine learning and deep learning has considerably attracted the interest of the material science community. The essence of ML and DL techniques lies in their ability to quickly predict target mechanical responses or material properties after conducting a training phase. Such methods also find their interest in computational mechanics to accelerate multiscale simulations of heterogeneous materials. As discussed previously, the evaluation of the mechanical response of complex architectured structures is very challenging, the traditional methodologies to perform these simulations such as FE^2 require costly computing resources. The integration of deep learning approaches in solid mechanics numerical solvers can be considered as a promising alternative way to overcome these issues.

In that sense, the main purpose of this thesis is **to propose a change of paradigm in the design and simulation of complex heterogeneous structures using hybrid models that combine physical knowledge and Artificial Neural Networks (ANN) based approaches.** To address this objective, the present manuscript has been structured into two parts and four chapters.

Part.I: Hybrid approaches to predict nonlinear mechanical behaviour of materials using Artificial Neural Network and physics based modeling.

- **Chapter 1:** The first chapter of this manuscript is dedicated to a state or art review regarding the use of machine learning techniques and data driven approaches in material science and for engineering applications. First, the main ML training strategies are presented including Supervised Learning (SL), Unsupervised Learning (UL), Semi Supervised Learning (SSL) and Reinforcement Learning (RL). The use of a specific learning algorithm depends on the the problem to be addressed and the nature of available data. In the second section of this chapter, a general overview of the application of ML methods in mechanics of materials is highlighted. Based on the recent contributions from the material science community, numerous practical examples including constitutive modeling and multiscale modeling are discussed.

- **Chapter 2:** In this chapter, we aimed to develop a hybrid physics-AI based model to predict nonlinear mechanical behaviors of dissipative materials. By introducing a specific Neural Network architecture called Thermodynamically Consistent Recurrent Neural Networks (ThC-RNN), this work proposes a new

paradigm for the simulation of dissipative materials under complex loading conditions. The design of such architecture allows to take into account the material loading history subjected to multi-axial and non-proportional loading paths, similarly to internal variables for homogeneous materials. In addition, a special focus is given to the respect of thermodynamics principles in the ThC-RNN model by introducing specific thermodynamical constraints during the training phase. Finally, the model's reliability is tested on different plasticity models once the training is completed.

Part.II: Multiscale modeling of heterogeneous materials using Artificial Neural Networks based approaches

- **Chapter 3:** In the third chapter of this manuscript, an investigation was carried out to verify hybrid models capabilities to predict the effective properties of heterogeneous materials. A hybrid model ANN $-\varphi$ is developed by combining Artificial Neural Networks and micromechanical modeling. The homogenization approach used in this study is mainly based on Eshelby's inclusion problem. ANN $-\varphi$ predictive capabilities of composite materials homogenized properties and local behaviors is investigated. The accuracy of the obtained results are compared with concurrent numerical estimations.

- **Chapter 4:** The last chapter of this manuscript is devoted to the development of new numerical modeling strategies using deep neural networks to accelerate multiscale simulation of architectured materials. A so called FE-LSTM approach is designed for multiscale modeling of heterogeneous structures by combining Finite Element method (FE) and Long Short Term Memory (LSTM) neural networks. The developed approach is tested on several examples of 3D architectured structures under proportional and non proportional loading paths. The obtained results with FE-LSTM are compared with FE^2 method in terms of accuracy and computational costs savings (execution time, required memory, CPU usage). The last section of this chapter presents a a first attempt of using multi-inputs neural networks architectures combining convolutional neural networks and multilayer perceptrons to predict the effective properties of architectured materials. Through the development of a CNN-MLP model, the achievable capabilities of such neural networks architectures that takes into account data of different nature (images, numerical data, sequences) are investigated.

Part I

Hybrid approaches to predict nonlinear mechanical behaviour of materials using Artificial Neural Network and physics based modeling

Chapter 1

State of the art

Contents

1.1 Introduction

Historically, the development of material science has undergone regular advances over the centuries regarding the challenges and the adopted strategies to deal with engineering problems. Beginning with empirical modeling where scientists relied solely on observation and experimentation to describe physical phenomena, theoretical modeling allowed to formalize properly these physical laws using adequate mathematical expressions. Since the 19^{th} century, significant progress has been made in material science thanks to Augustin-Louis Cauchy, the pioneer of continuum mechanics who established a complete framework to characterize the mechanical behaviour of materials and structures. The governing equations of this framework includes the kinematic relations, conservation laws (for mass, momentum and energy) and lastly materials constitutive models which are restricted by physical laws and thermodynamics principles. Another major breakthrough in material science is the development of micromechanics to deal with the mechanical behavior of heterogeneous materials commonly employed for various engineering applications. While the theoretical approaches provided a solid foundation in structural mechanics, the resolution of engineering problems was not always achievable using analytical methods especially when dealing with complex non-linear behaviors. The emergence of the computational era has revolutionized the field of mechanical engineering, allowing to overcome many restrictive assumptions regarding microstructures geometrical properties, material behaviors and problems dimension. For instance, the advent of computational methods including Finite Element Analysis (FEA) has enabled engineers to address heterogeneous material problems with more realistic microstructures including architectured materials with non-linear mechanical behaviors, therefore keeping a full and detailed description of the microstructure contrary to micromechanical models. However, the high computational costs constitute a major issue often encountered when using this type of approach particularly for multiscale simulations.

More recently, a new paradigm is attracting the attention of the research community, Data science and its incorporation in the field of mechanical engineering. The impressive growth of big data and Artificial Intelligence (AI) algorithms including Machine learning and Deep learning have considerably impacted a wide spectrum of material science, offering new insights and perspectives for engineers and scientist and giving more flexibility and freedom in material design. The main concept behind AI based approaches lies in their fast and powerful predictive capabilities of target mechanical behaviors or properties after a training phase on materials data sets. Nowadays, Data availability, considered as the primary source of AI models enrichment, is no longer an obstacle thanks to the recent development of experimental measurement techniques and current computing capabilities, thus allowing to generate a large quantity of experimental data or numerical simulations. With the aim of optimizing the mechanical behavior of complex multi functional structures, a promising alternative to the conventional methods is to fit a machine learning model using these generated data to explore massive design spaces in a short amount of time. An additional application scope of such Data driven approaches is the acceleration of computational methods in solid mechanics. As mentioned previously, multiscale simulations of complex architectured materials using Finite Element Analysis require expensive computing resources that have a significant energetic impact. An alternative way to approach this problem is the incorporation of machine learning algorithms in the numerical solvers. More specifically, the general idea would be to substitute the rather costly numerical estimations by a surrogate model that provides rich interpolation spaces. By adopting this strategy, we could envision the possibility of downscaling the computational costs of complex, non-linear and multiscale simulations during the online phase, once the machine learning model is trained

with an offline database.

Despite the great success of artificial intelligence-based approaches, it is worth mentioning that these methods are mostly considered as "black boxes", sometimes lacking any explanation or interpretation of why a machine made a specific decision or predicted a specific value for a certain quantity of interest. The reason lies in the fact that a machine learning model is simply a mathematical tool that relies on a set of data to map any non-linear function after a learning phase. Even if the data comes from physical models, the algorithm itself does not have any scientific knowledge in its architecture or during the training process. As the complexity of physical systems increases, the generation of large data sets is required to calibrate the model, a task that is not always achievable given the high computational costs. Moreover, even if the issue of generating large databases do not arise, the compatibility of model's predictions with physical principles is not always guaranteed since they are purely data-driven. A new promising way to enhance the robustness of AI based models is to rely on Hybrid approaches by incorporating scientific knowledge and physical principles in the design of ML algorithms. The main motivation of this hybridization in mechanics is to benefit from the valuable knowledge in materials science that has been developed over centuries and which provides a fine description of physical mechanisms in complex materials that can not easily be captured with regular ML techniques. As shown in Figure 1.1, a hybrid model can be defined as the combination of a ML technique trained on a set of data resulting from numerical simulations (or experimental tests) in which physical laws are integrated simultaneously in the core of the AI algorithm either in the architecture design or during model training process. The development of such hybrid models can potentially be beneficial for many engineering applications given the reliability of the predictions that are physically consistent, while at the same time compensating for the lack of model training data.

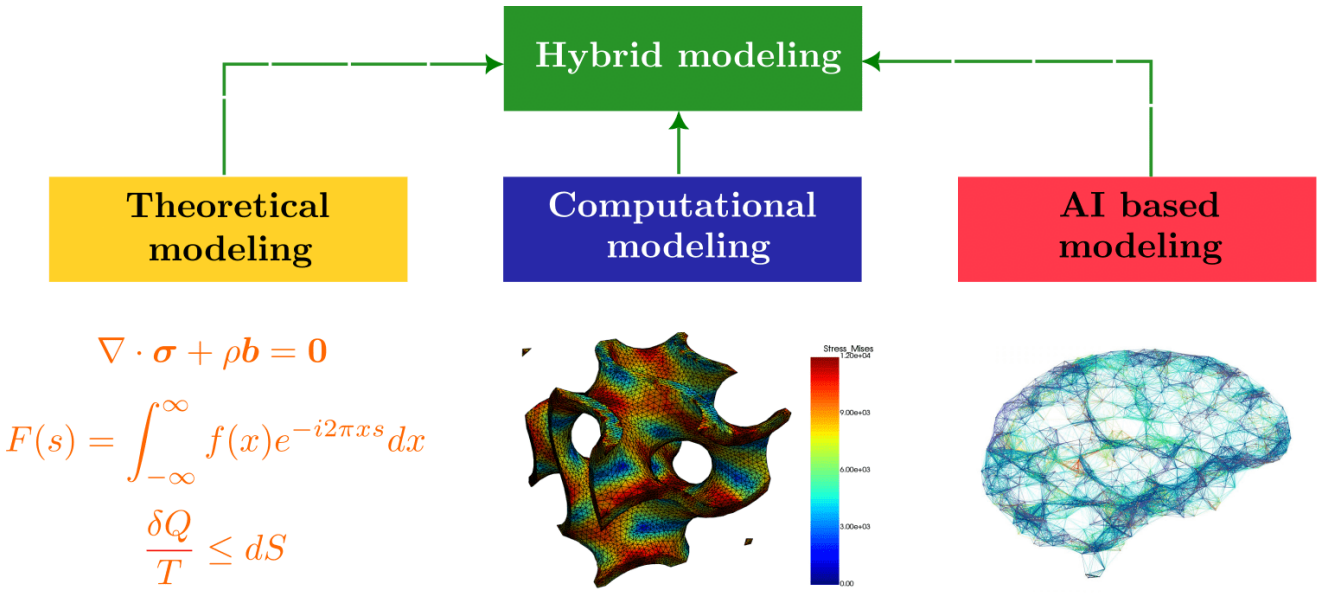


Figure 1.1: An illustrative diagram of a hybrid model combining an AI based method, a set of computational training data and physical laws resulting from theoretical frameworks

The first chapter of this manuscript is devoted to a state of art regarding the use of ML techniques and Data driven approaches in material science. A general overview of the application of such methods in mechanics is presented. Several examples and cases of study including constitutive modeling, multiscale modeling are discussed based on the recent contributions from the material science community in this research area.

1.2 Machine learning algorithms for engineering science applications

In this section, we review recent developments and scientific achievements in the use of data driven techniques, including machine learning, in material science. Several topics will be explored in this review and various practical examples in mechanics will be discussed such as Data-driven constitutive modeling, Data-driven computational mechanics, Data-driven multiscale homogenization and Hybrid modeling. Before addressing the aforementioned applications, it is first necessary to properly define a machine learning algorithm and its associated workflow while solving a given problem. Machine learning is a set of algorithms and statistical methods implemented in computer systems whose objective is to perform specific tasks without requiring any prior programming. This implies that ML models are only enhanced by past experiences where the machine draw inferences from patterns in relevant data. This process involves a learning phase using mathematical optimization methods that adjusts systems parameters to establish reliable predictive models. Machine learning algorithms training techniques can be divided into four main categories: *Supervised learning, Unsupervised learning, Semi-supervised learning* and *Reinforcement learning.*

1.2.1 Supervised Learning (SL)

Supervised learning consists on using labeled data to train machine learning models. Each input of the training database is associated with a label or a ground truth, the objective of the algorithm is to learn the correspondence between inputs and outputs by solving an optimization problem i.e. minimizing the error between the predicted values and the target values. The principle of supervised learning is illustrated in figure 1.2. This learning technique is widely used for solving various ML tasks including regression and classification problems. For classification, this could be for instance in medical diagnostics, fraud detection or image recognition. For regression, the are also many applications such as population growth prediction, market forecasting or materials durability prediction in mechanics.

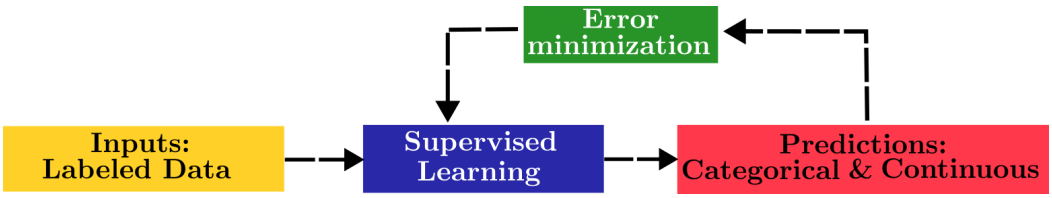


Figure 1.2: An illustrative diagram of supervised learning

1.2.2 Unsupervised Learning (UL)

Unsupervised learning handles unlabeled data sets where the main goal is to identify undetected patterns and trends in the data. Unsupervised learning can be employed for various applications such as clustering or dimensionality reduction. Cluster analysis is a statistical method that aims on grouping the data into clusters based on their degree of similarity or shared characteristics. Dimensionality reduction consists on finding a reduced representation of high dimensional data, it can be applied in the context of model order reduction using various algorithms such as Proper Orthogonal Decomposition (POD) [Kerschen *et al.* 2005] or Proper Generalized Decomposition (PGD) [Chinesta *et al.* 2014]. Unsupervised learning can be applied to a variety of tasks such as medical imaging, big data visualization and anomaly detection using Generative Adversarial Networks (GANs).

Figure 1.3: An illustrative diagram of unsupervised learning

1.2.3 Semi-Supervised Learning (SSL)

Semi-supervised learning is a technique that bridges supervised and unsupervised learning by using a small portion of labeled data and a high portion of unlabeled data to train predictive machine learning models. The main concept of this method can be illustrated in figure 1.4, the first step of the semi supervised learning workflow is to train an initial model using a small subset of labeled data via supervised methods. The second step is to use the partially trained model to make predictions for the rest of unlabeled data which constitute the big portion of the database. The third step is where pseudo labels are generated as predictions of the initial classifier. Afterwards, the small subset of labeled data is merged with the most confident pseudo labels to create a new labeled database, then a new improved classifier is trained based on this data with the help of a supervised technique. This process is performed for several iterations with more confident pseudo labels being added each time until achieving an optimal performance. The main advantage of semi supervised learning is to overcome the limitations related to the labeling of large databases which is a resource-intensive and time consuming manual operation.

1.2.4 Reinforcement Learning (RL)

Reinforcement learning is a new paradigm of training ML models where the objective is to learn from successive experiments through a trial-error process. Typically in reinforcement learning, an "agent" is interacting with an "environment" to solve a given problem. Contrary to the supervised and unsupervised learning, RL is an interactive approach where the "agent" explores several solutions while observing the reaction of its "environment" and trying to adapt its behavior to find the optimal strategy. As shown in figure 1.5, for each action A_t in the environment, a reward R_t is attributed to the agent and its state S_t is updated based on its previous interaction and exploration. This type of learning is often used for autonomous cars, and for video game design. A concrete example is AlphaGo, a deep AI from Google that has defeated Lee Sedol the world champions at the game of Go using reinforcement learning.

The choice of a machine learning model depends mainly on the problem to be solved. For regression and classification problems, supervised learning is the most appropriate learning technique. As shown in

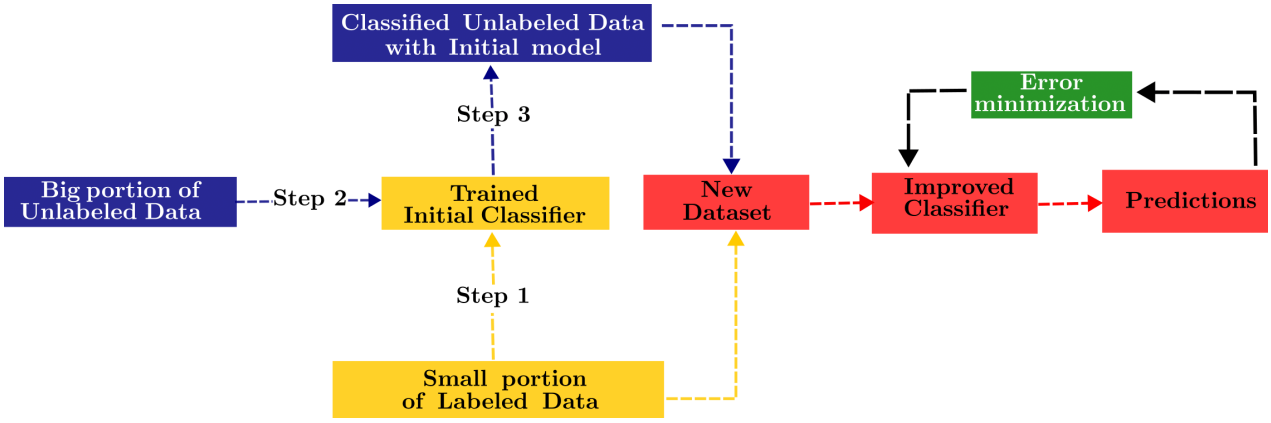


Figure 1.4: An illustrative diagram of semi-supervised learning

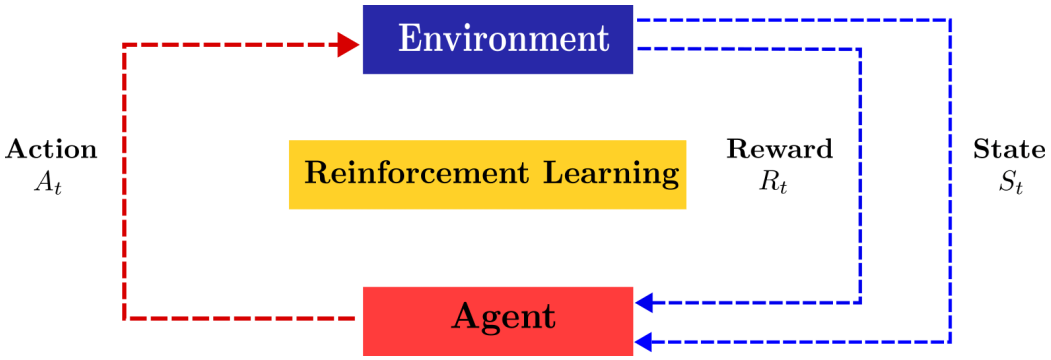


Figure 1.5: An illustrative diagram of reinforcement learning

the figures 1.6 and 1.7, several SL-based algorithms can be applied for regression, such as Feed Forward Neural Networks (FFNN), Recurrent Neural Networks (RNN), decision trees, random forest, and Gaussian process regression (GPR). For classification applications, Convolutional Neural Networks (CNN) are currently the most robust technique for computer vision. Clustering and dimensionality reduction problems require unsupervised learning algorithms such as K-means, Support Vector Machine (SVM), and model order reduction methods.

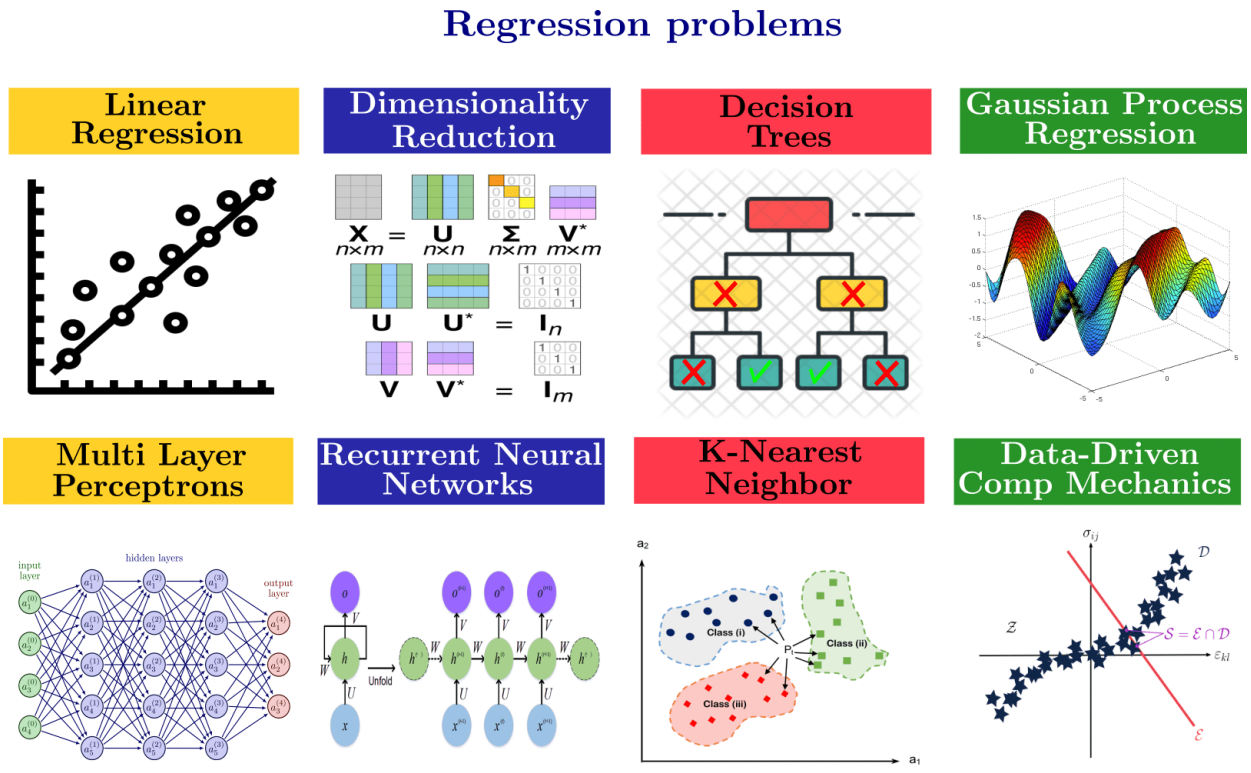


Figure 1.6: An overview of some machine learning techniques and data driven approaches for solving regression problems

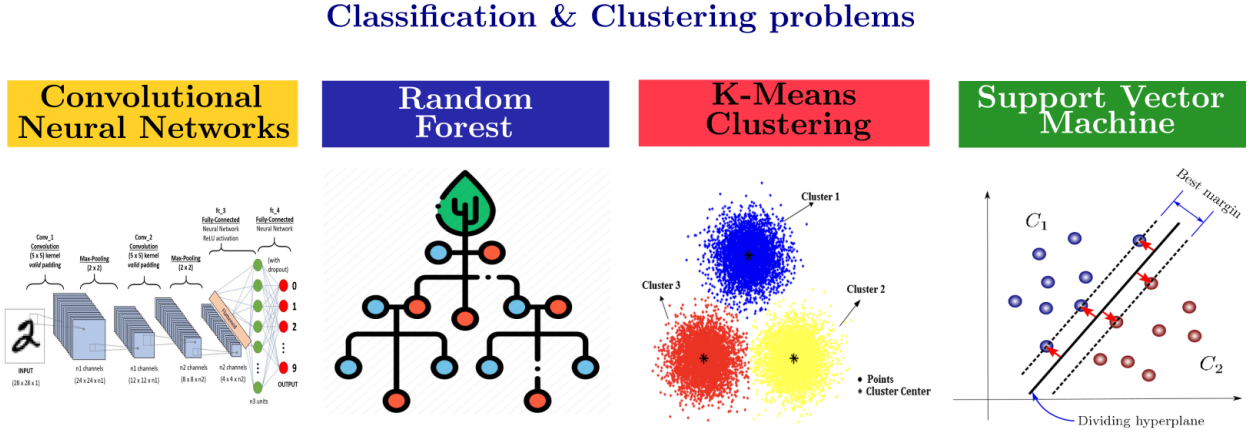


Figure 1.7: An overview of some machine learning techniques and data driven approaches for solving classification and clustering problems

1.3 An overview of using Data driven and Machine learning techniques in material science

1.3.1 Linear regression

Linear regression models are considered the most basic and elementary predictive machine learning algorithms. Despite their simplicity in terms of implementation, they can sometimes be effective for solving specific problems such as identifying relationships between different variables. For instance, medical scientists often use linear regression models to understand the correlation between drug dosage and blood pressure of patients. A linear model consists in identifying a linear relationship between a set of data points $X = (X^i)_{1 \leqslant i \leq n}$ and an output $Y = (Y^i)_{1 \leqslant i \leqslant n}$. For a given data point X^i with d features, the predicted output Y_p^i is expressed as:

$$Y_p^i = \sum_{j=0}^{d} \omega_j x_j^i = W^T X^i,$$

where W is the weight column vector.

By defining a loss function $\mathcal{L}$ that measures the error between the predicted values Y^i and target values Y_p^i, the vector W can be obtained by solving a mathematical optimization problem i,e minimizing $\mathcal{L}$. The most common loss is the Mean Square Error (MSE):

$$\mathcal{L} = \frac{1}{n} \sum_{i=1}^{n} \left(Y_p^i - Y^i\right)^2 = \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=0}^{d} \omega_j x_j^i - Y^i \right)^2$$

The most encountered issue of regression algorithms is overfitting. This problem occurs when the model is only performing well on training data but is unable to generalize on unseen data. To overcome this issue, regularisation can be used to reduce the effect of overfitting by adding a penalty term to the loss. Two regularized models are commonly used in linear regression, Least Absolute Shrinkage and Selection Operator (LASSO) [Tibshirani 1996] and Ridge regression [Cortes *et al.* 2012]:

Least Absolute Shrinkage and Selection Operator (LASSO) regression or L^1 regularisation:

$$\mathcal{L}^{Lasso} = \frac{1}{n} \sum_{i=1}^{n} \left(Y_p^i - Y^i\right)^2 + \lambda \|W\|_{L^1} = \frac{1}{n} \sum_{i=1}^{n} \left(Y_p^i - Y^i\right)^2 + \lambda \sum_{j=1}^{d} |w_j| \tag{1.1}$$

Ridge regression or L^2 regularisation:

$$\mathcal{L}^{Ridge} = \frac{1}{n} \sum_{i=1}^{n} \left(Y_p^i - Y^i\right)^2 + \lambda \|W\|_{L^2} = \frac{1}{n} \sum_{i=1}^{n} \left(Y_p^i - Y^i\right)^2 + \lambda \sum_{j=1}^{d} w_j^2 \tag{1.2}$$

As shown in equations 1.1 and 1.2, Ridge and Lasso regression models constrain the norm of the vector W by adding a penalty term λ when the coefficients take larger values. By shrinking the weights

coefficients, this regularisation techniques can significantly reduce models complexity, multicollinearity and prevent overfitting.

Linear regression methods are among the first machine learning models used for engineering material design due to their capabilities of identifying some relationships between different mechanical properties. For instance, [Doreswamy 2011] applied a regression analysis to predict linear relationships between the yield strength and tensile strength for a variety of metals and polymers. This study suggests the possibility of using material databases to fit a simple linear models to determine certain mechanical properties that are difficult to obtain experimentally. Another study by [Pavlina & Van Tyne 2008] revealed that using regression analysis, there exists a correlation of yield strength and tensile strength with hardness for a certain group of steel materials. However, this correlation is linear only for a certain range of yield and tensile strength values and above a certain threshold these mechanical properties are totally uncorrelated. The limitations of linear regression models were recently highlighted by [Zhao *et al.* 2021] for a study that aims to establish a relationship between chemical composition, aging time, and target mechanical properties such as hardness of copper alloys. Four machine learning algorithms were tested including linear regression, Support Vector Regression (SVR), regression trees and Gaussian Process Regression (GPR). The linear regression was found to be the least accurate compared to other models. This is an expected result given the complexity of material behavior. Hence, this reinforce the need of robust algorithms when dealing with complex non-linear material problems. In [Yang *et al.* 2019], LASSO algorithm has been compared with several machine learning models such as polynomial regression, random forest and feed forward neural networks to predict the elastic properties of silicate glasses. This application which involved dynamics simulation databases to train the aforementioned ML algorithms, showed that neural networks were the highest in terms of accuracy. LASSO was found to be slightly less precise than the other models but offered better interpretability and simplicity.

1.3.2 Feed Forward Neural Networks (FFNN)

Feed Forward Neural Networks (FFNN) also called Multi Layer Perceptrons (MLP) are known to be one of the fundamental architectures of Artificial Neural Networks (ANN) [Bebis & Georgiopoulos 1994]. Artificial Neural Networks are currently at the center of Deep Learning (DL) revolution due to their powerful capabilities to solve complex and highly non-linear problems. Deep learning represents a branch of machine learning methods capable of mimicking the actions of the human brain through artificial neural networks.

Inspired by the structure of biological neurons, ANN are capable of learning and executing tasks with high levels of abstraction. The major growth of DL methods has led to rapid and significant advances in many fields including speech recognition, computer vision and automated language processing. Figure 1.8 illustrates the relationship between different branches and disciplines of AI, including machine learning and deep learning. The first neural network architecture was the perceptron created by Rosenblatt in 1958 [Rosenblatt 1958] for machines supervised tasks. Then, hidden layers have been progressively added to the architecture to learn complex non-linear relationships between inputs and outputs. The effect of stacking multiple layers in the ANN architecture led to the development of Deep Neural Networks (DNN). The most common DNN architectures are FFNN which are introduced in this subsection, Recurrent neural networks (RNN) for processing sequential data, and convolutional neural networks (CNN) commonly used for images processing which will be detailed in the next subsection.

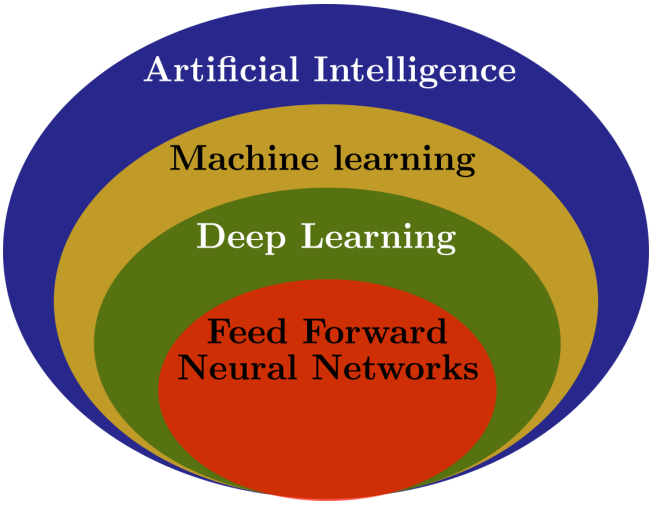


Figure 1.8: An overview of the different branches of AI and its relationship to Machine learning and Deep learning

Feed forward neural networks are a multiple fully connected layers Neural Networks whose information flow is propagated only in the forward direction from the inputs to the outputs. Multi Layer Perceptrons are able to learn the representation in the training data and how to best relate it to the outputs that needs to be predicted. Mathematically, they are capable of learning any mapping function and identifying complex non-linear relationships between input and output data sets. MLP have indeed been proven to be a universal approximation algorithm [Hornik *et al.* 1989]. Figure 1.9 shows an example of a FFNN with an input layer of four input parameters, three fully connected hidden layers and an output layer with three output parameters.

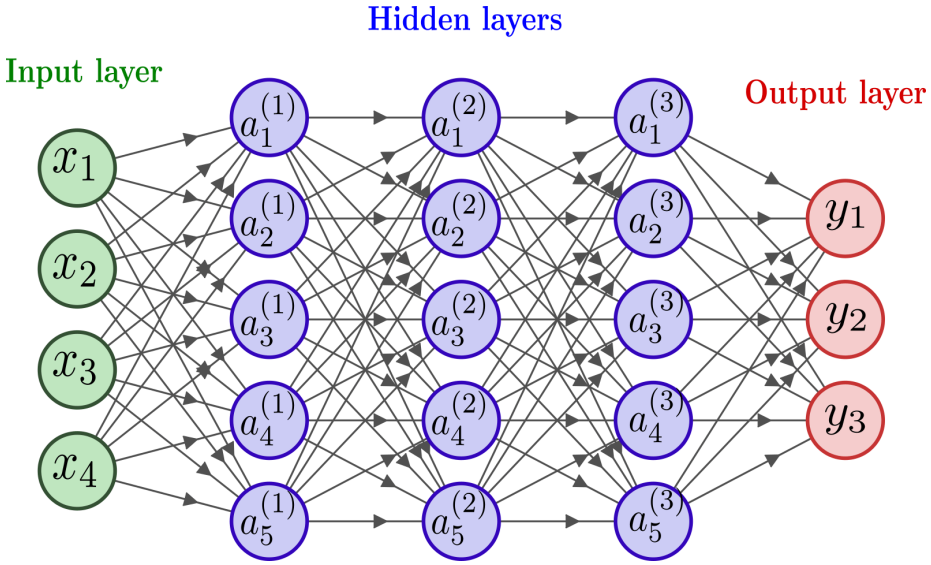


Figure 1.9: An example of a feed forward neural network architecture

Multi layer perceptrons training procedure is described as follows. MLP maps inputs into the corresponding outputs by tuning model's parameters. MLP consists of L layers of neurons and for each layer $l = 1, ..., L$, a neuron i is linked to every neuron j from the previous layer $l-1$ by the weight ω_{ij}^l. As shown in figure 1.10, the input x_i^l of the neuron i can be expressed using the weighted sum of all outputs of previous layer a_i^{l-1} and the bias b_i^l.

$$x_i^l = \sum_{j=1}^{n} \omega_{ij}^l a(x_j^{l-1}) + b_i^l$$

The output of the neuron i is defined by a non linear activation function a that governs the threshold at which the neuron is activated and its contribution for the next layer $l+1$. Several non linear functions can be used as activations to solve deep learning problems. Sigmoid, Rectified Linear Unit (ReLU), SoftPlus, Hyperbolic Tangent (TanH) and Exponential Linear Unit (ELU) are examples of the most commonly used activation functions:

- **Exponential Linear Unit (ELU):**

$$a(x) = \begin{cases} \alpha(e^x - 1) \text{ for } x < 0 \\ x \text{ for } x \geqslant 0 \end{cases}$$

- **Rectified Linear Unit (ReLU):**

$$a(x) = \begin{cases} 0 \text{ for } x < 0 \\ x \text{ for } x \geqslant 0 \end{cases}$$

- **SoftPlus:**

$$a(x) = \ln(1 + e^x)$$

- **Hyperbolic Tangent (TanH):**

$$a(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

- **Sigmoid:**

$$a(x) = \frac{1}{1 + e^{-x}}$$

Using the activation function a, the output of the neuron i is given by:

$$a_i^l = a(x_i^l) = a(\sum_{j=1}^{n} \omega_{ij}^l a(x_j^{l-1}) + b_i^l)$$

The same procedure is applied to calculate the outputs of each neurons in the layer $l+1$ until reaching the last layer L.

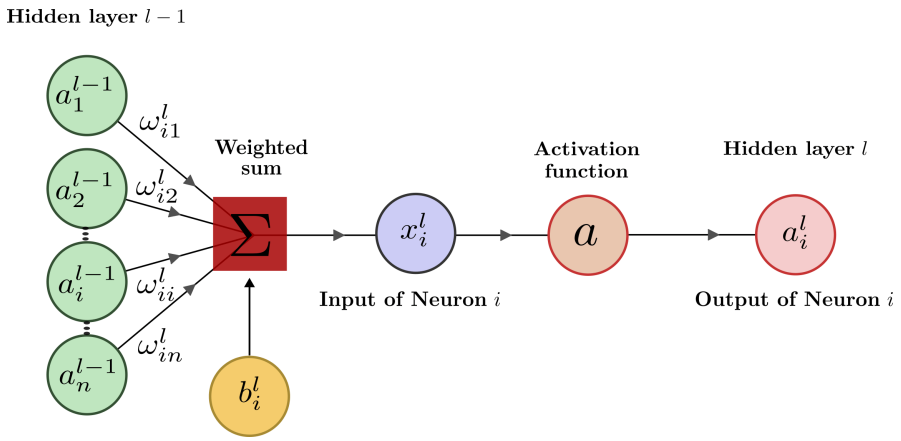


Figure 1.10: A representation of the forward propagation in a neuron i of a multilayer perceptron

Building a Feed Forward Neural Network requires the definition of model's hyperparameters which are among activation functions, optimizers or the number of hidden neurons. Model's hyperparameters are the set of parameters that needs to be chosen before the training phase. Thus, the selection of the right combination is a crucial step since this choice has a direct impact on model's accuracy. Hyperparameter optimization [Bergstra *et al.* 2011] often involves experimentation using methods such as Grid Search or Random Search.

Grid Search [Bergstra *et al.*] consists on evaluating the model for each possible combination of tested hyperparameters and selecting the set of parameters which provides the most accurate predictions. In order to avoid overfitting [Caruana *et al.* 2001], the best parameters are selected based on cross-validation score [Browne 2000]. By contrast, Random Search [Bergstra & Bengio] sets up a grid of hyperparameter values and selects random combinations to train the model. This approach allows to explicitly control the attempted number of parameter combinations which are set based on time or computational resources.

To measure the neural network's accuracy, a loss function $\mathcal{L}$ needs to be defined. The loss function estimates the error between predicted values by ANN y_p and given output data y_{true}. For regression problems, Mean Squared Error (MSE) or Mean Absolute Error loss functions can be used:

$$\mathcal{L}^{MSE} = \frac{1}{N}\sum_{k=1}^{N}(y_p^{(k)} - y_{true}^{(k)})^2$$

$$\mathcal{L}^{MAE} = \frac{1}{N}\sum_{k=1}^{N}|y_p^{(k)} - y_{true}^{(k)}|$$

N is the total number of training data.

For classification problems, cross entropy loss functions are employed. The word entropy has a statistical interpretation as it measures the randomness in processed data, and the cross entropy is a measure of the difference of randomness between the predictions and the targets. For two-classes classification prob-

lems, Binary Cross Entropy (BCE) loss function $\mathcal{L}^{BCE}$ is commonly used. For multi classes classifications problems, Multi-Class Cross Entropy (MCCE) loss functions $\mathcal{L}^{MCCE}$ are employed.

$$\mathcal{L}^{BCE} = -\sum_{k=1}^{N} y_{true}^{(k)} \log\left(y_p^{(k)}\right) + \left(1 - y_{true}^{(k)}\right) \log\left(1 - y_p^{(k)}\right)$$

$$\mathcal{L}^{MCCE} = -\sum^{C}\sum_{k=1}^{N} y_{true}^{(k)} \log(y_p^{(k)})$$

N is the total number of training data, C is the number of predicted classes.

If the divergence of the predictions from the actual targets increases, the cross-entropy loss increases. Minimizing the cross-entropy loss functions consists in finding the best model parameters for which the difference in randomness between the predictions and the targets tends towards zero.

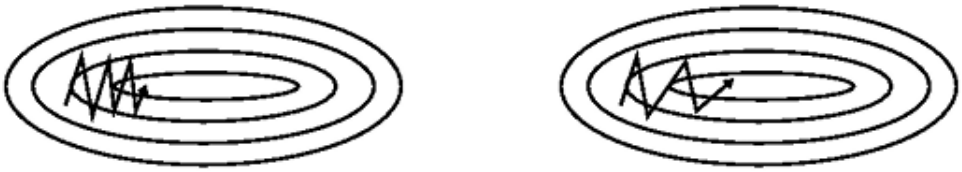
The main objective of Neural Network's training phase is to calibrate the weights ω and biases b by solving an optimization problem *i.e*, minimize the loss function. For this purpose, back-propagation algorithm [Lecun 2001a] is used. Back-propagation is the essence of Neural Network's training, this algorithm is used to train the ANN through a chain rule method by computing the partial derivative of the loss function with respect to the weights ω_{ij} and biases b_i. After each forward pass through the network, backpropagation performs a backward pass while adjusting the model's parameters by a gradient descent technique [Ruder 2017]. Gradient descent optimization algorithms compute the gradient of loss function with respect to the weights and biases and updates them after each Epoch[1] k with a learning rate η,

$$\omega_{ij}^{(k+1)} = \omega_{ij}^{(k)} - \eta \frac{\partial \mathcal{L}}{\partial \omega_{ij}^{l}}, b_{ij}^{(k+1)} = b_{ij}^{(k)} - \eta \frac{\partial \mathcal{L}}{\partial b_{ij}^{l}}$$

This learning process is iterated until the minimum of loss function is reached. However, it is very common that neural networks encounter difficulties during training with traditional descent gradient techniques. A common problem with these optimization algorithms is that they can bounce around in the search space of optimization problems that have curvatures and noisy gradients, and they can get stuck in flat areas of the search space that have no gradient. To overcome this issue, gradient descent techniques with Adaptative Learning Rates or Momentum are introduced. Momentum is an extension to the gradient descent optimization algorithm that allows to build inertia in relevant directions of search space and overcome the oscillations of noisy gradients and coast across flat spots of the search space (figure 1.11). The most common adaptative learning rates optimization algorithms are RMSProp [Ruder 2017], Adam [Kingma & Ba 2014], AdaGrad [Duchi *et al.* 2011] and Nesterov accelerated gradient [Botev *et al.* 2016].

The use of ANN as surrogates has been proven to be a reliable tool for several applications including structural analysis, constitutive modeling and multi-scale modeling. The first application of neural networks in material science was in 1991 where [Ghaboussi *et al.* 1991] used a multilayer perceptron to model the mechanical behavior of concrete under monotonic biaxial loading and compressive uniaxial loading. [Furukawa & Yagawa 1998] proposed an implicit viscoplastic constitutive model using a neural network

[1] One Epoch is when the entire training samples are passed forward and backward through the ANN only one time



(a) SGD without momentum (b) SGD with momentum

Figure 1.11: The effect of adding momentum to gradient descent techniques when trying to reach global minima [Ruder 2017]

that takes as inputs the current viscoplastic strain, internal variables and current stress, then predict the current rates of change of the viscoplastic strain and material internal variables. Another extension of ANN to rate dependant materials modeling has been proposed by [Jung & Ghaboussi 2006]. This study consists of the implementation of a neural network, trained to capture viscoelasticity, in a finite element code and then applied to structural analysis of concrete. Using Neural Networks backpropagation algorithm as a replacement for numerical procedures, [Waszczyszyn & Ziemiański 2001] treated several problems regarding elastoplasticity such as the bending analysis of elastoplastic beams and detection of damage in steel beams. [Zhang & Mohr 2020] reformulated the standard return-mapping algorithm by substituting the non linear procedures (used to compute stress-strain relationships and elastoplastic tangent matrix) with a feed forward neural network. This method has been employed for the prediction of Von Mises plasticity with isotropic hardening. In a recent study, [Ali *et al.* 2019] applied an MLP architecture to predict stress-strain responses and texture evolution of polycrystalline metals under shear and tension loading. As shown in figure 1.12, the computation time saving was estimated at 10000 by adopting this approach. Similarly, [Shen *et al.* 2020] used an ANN model to predict plastic yield surfaces for porous materials with different loading states and different porosity coefficients. [Settgast *et al.* 2019a] also applied a feed forward neural networks as a constitutive model to describe deformation behavior of open-cell foams under proportional loading paths, the computational time saving factor was also estimated at 10000 during this study.

According to recent studies, artificial neural networks have proven to be reliable and effective methods for engineering material design due to their capabilities to accurately predict target material properties. For instance, a feed forward neural network has been employed by [Ravinder *et al.* 2020] to capture the associated non-linear composition–property relationships of oxide glasses. By training this model on a large database containing glass compositions, the neural network is able to predict accurately a variety of mechanical properties including Young's modulus, hardness and shear modulus. In another study conducted by [Liu *et al.* 2020], ANN have been used in a fracture mechanics framework to predict the toughness of polysilicon materials. The high predictive capability of deep neural networks, even when trained on small datasets, has been highlighted by [Ouyang *et al.* 2021] to predict concrete's strength as a function of the material mixture proportions. This study showed that ANN are able to quickly and successfully learn input-output relationships compared to other machine learning algorithms.

A few recent contributions have also demonstrated that deep learning methods can be integrated into a computational homogenization framework to accelerate multiscale simulations. For example,

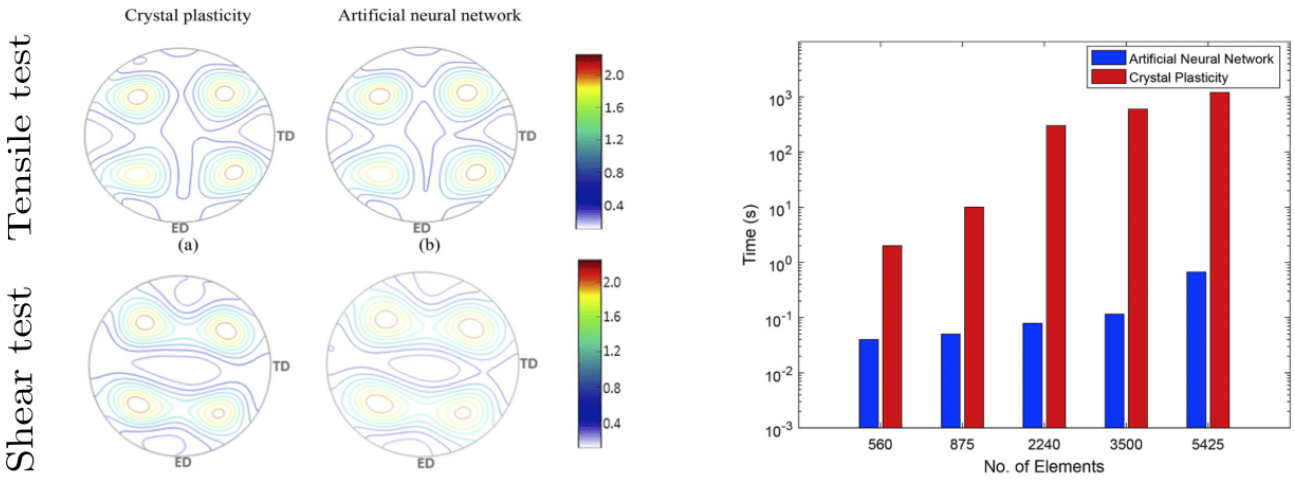


Figure 1.12: Tensile and shear <1 1 1> pole figures comparison between crystal plasticity simulations and ANN under 10% strain for AA6063 – T6 polycrystal (left figure). Runtime comparison between crystal plasticity and ANN (right figure) [Ali *et al.* 2019]

[Le *et al.* 2015] used a neural network to compute the mechanical behavior of elastic non-linear heterogeneous structures under proportional loading paths. The approach involved replacing the strain energy density of the homogenized material with a response surface predicted by an ANN. The derivation of this effective potential allowed to obtain the macroscopic constitutive law. As shown in figure 1.13, this method was applied to compute the mechanical response of a multi-scale composite structure. The ANN-based approach was compared with a FE^2 reference solution. The obtained results showed a very good correspondence between the two solutions, and the online computational time saving factor was estimated to be 120.

Another example of substituting the energy density of heterogeneous materials by an ANN-based approach was studied by [Minh Nguyen-Thanh *et al.* 2020]. This method was applied to the computational homogenization of non-linear elastic structures with finite strain. This study demonstrated that the multilayer perceptron, trained on a database of FFT simulations, showed excellent predictive capabilities of the macroscopic potentials. This approach was then applied to compute the mechanical behavior of heterogeneous structures such as multiscale membranes. The comparison between the obtained results with full field solutions and by ANN based model have revealed good agreements, thus allowing to be potentially incorporated in a computational homogenization framework. The robustness and reliability of deep learning methods in computational homogenization was also highlighted in another study conducted by [Lu *et al.* 2018] to characterize the non-linear electric conduction in random graphene-polymer nanocomposites. This data driven approach involved using an ANN-based surrogate to construct the effective electrical constitutive law after a training phase on RVE non-linear electrical conduction simulations. By adopting this approach, the online speed up factor was estimated to be 104, thus significantly reducing the computational time compared to regular FE^2 method. However, it is worth mentioning that the conducted studies by [Le *et al.* 2015, Lu *et al.* 2018, Minh Nguyen-Thanh *et al.* 2020] were limited to non-dissipative materials under proportional loading paths, thus leading to a significant simplification of the problem.

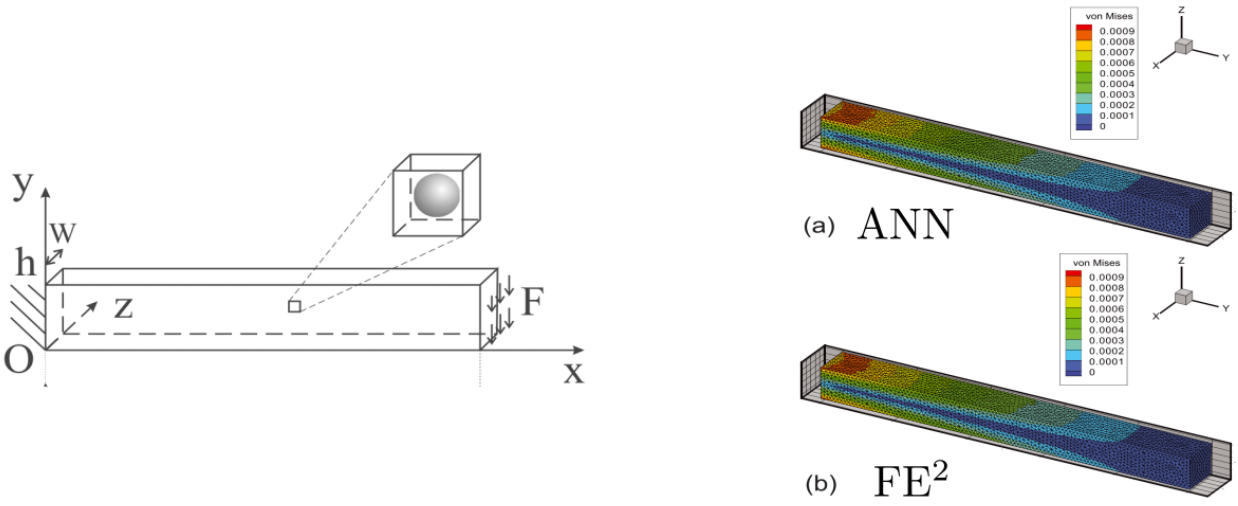


Figure 1.13: Geometry and boundary conditions of a two-scale heterogeneous composite structure (left figure). Comparison between the von mises stress obtained by an FE^2 reference solution and by an ANN based approach (right figure) [Le *et al.* 2015]

1.3.3 Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNN or ConvNets) are a type of deep neural networks intended for processing data that has a known grid-like topology such as images, audio files, time series and signals. For example, an image can be considered as a 2D grid of pixels and a time-series data can be assimilated as a 1D grid of samples at regular time intervals. The term "convolutional neural network" refers to the application of a linear mathematical operation called "convolution". By performing this operation, CNN are able to extract features and find patterns in many types of data and can be applied for instance in image classification or object detection. In recent years, CNN have been widely used in various fields, especially in computer vision. The major advances in self-driving vehicles, facial recognition and robotics are only the consequence of significant progress of artificial intelligence based approaches including convolutional neural networks.

In material design, CNN ability to capture and extract features from data is an attractive property that material science can take advantage of. ConvNets are therefore well suited for applications that have traditionally required human intervention, such as image segmentation. In a study conducted by [Strohmann *et al.* 2019], CNN have been used for X-ray tomographic images segmentation of complex 3D Al-Si microstructures. The results of this study demonstrated that the entire operation time for segmentation using the trained ConvNet was reduced to less than 1% of the time required by human segmentation without a loss of accuracy. [Bertoldo *et al.* 2021] have also developed a CNN based model called modular U-Net for automated segmentation of composites X-ray tomography images. As shown in figure 1.14, the CNN architecture (modular U-net) is trained to segment 3D X-ray tomography images of a three-phase glass fiber reinforced Polyamide 66 resulting from synchrotron tests. The achieved performances by this type of architecture have shown promising results in the adoption of CNN-based models for automated X-ray tomography segmentation tasks. Furthermore, it has been shown that this

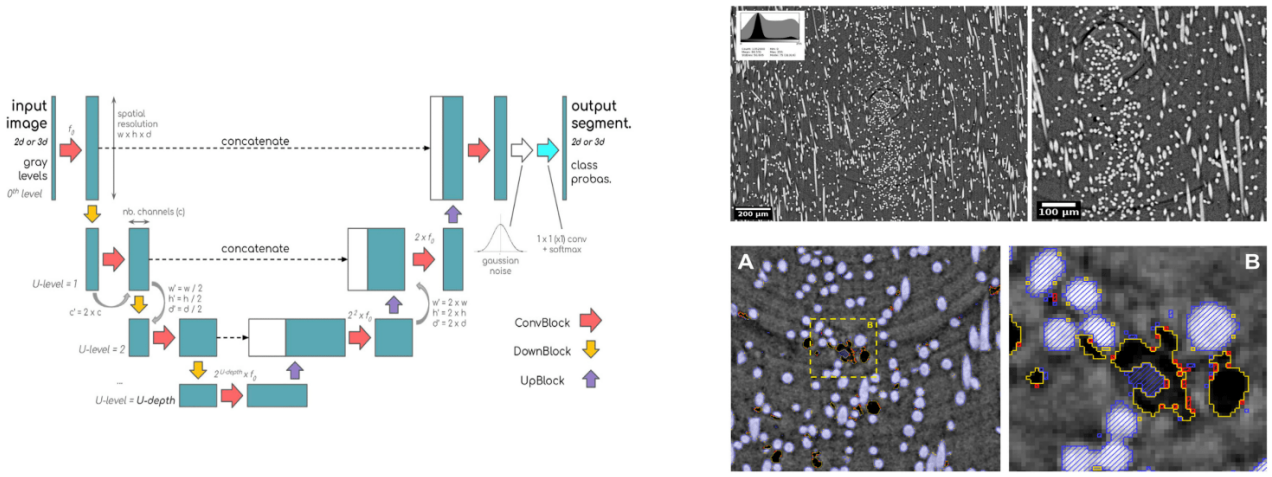


Figure 1.14: The CNN architecture (Modular U-Net) applied for X-Ray tomography images segmentation (left figure). Glass fiber reinforced Polyamide 66 tomography and segmentation results by CNN on test data set, blue represents voxels correctly classified as fiber (hatched), yellow as porosity (contours), and red represents misclassifications (right figure) [Bertoldo *et al.* 2021]

approach can be potentially integrated to automate synchrotron X-ray tomography processing workflows.

Although CNN are universally recognized for solving classification problems, they can be also be employed to solve non linear regression problems where the predicted features are continuous and not categorical. A concrete example to illustrate this application is a study carried out by [Yang *et al.* 2018] to establish the structure-properties relationship of high contrast composite materials using 3D-CNN. As shown in figure 1.15, a 3D ConvNet, trained on a database of composites RVEs voxels, is applied to capture non-linear relationship between the microstructure and its homogenized properties. Results from this study showed that the 3D CNN can outperform a traditional approach by nearly 54% in terms of validation errors. Another similar study by [Liu *et al.* 2015] showed the possibility of extracting features from microstructure by 3D CNN to capture elastic localization relationships in high contrast composite materials. Other examples of application of ConvNets for the prediction of the effective behavior of heterogeneous materials are investigated in [Liu *et al.* 2017, Cecen *et al.* 2018, Wang *et al.* 2020, Chen *et al.* 2020]. 3D ConvNets have also been used to solve more complex problems in fracture mechanics. In a study conducted by [Pierson *et al.* 2019], 3D ConvNets were used to predict the microstructure-sensitive evolution of 3D fatigue crack surface of Al-Mg-Si alloys. By training a 3D CNN on microstructural and micromechanical data from uncracked polycrystals and by using some prior knowledge of crack initiation, the model was able to accurately predict fatigue crack paths.

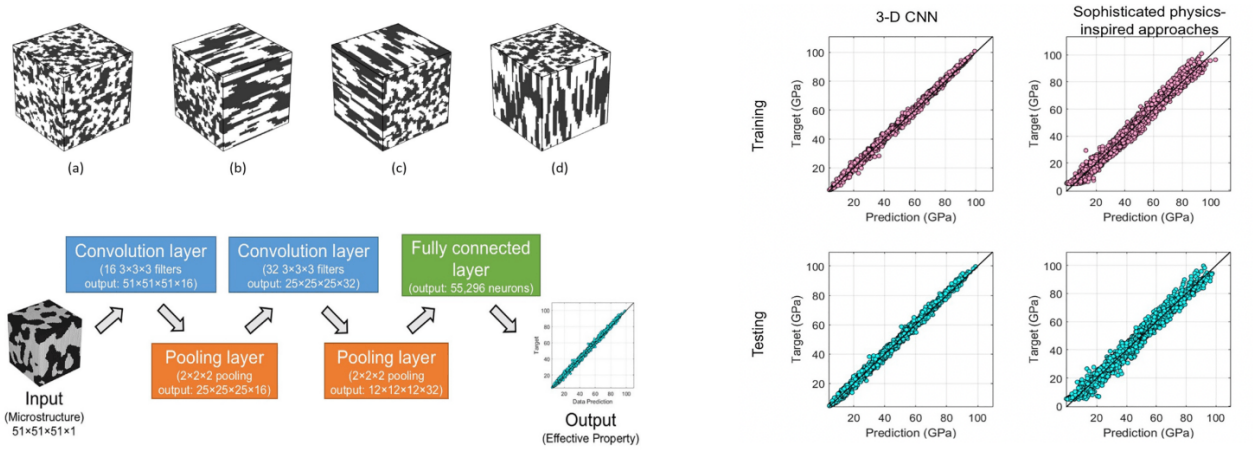


Figure 1.15: Database of composites RVEs and 3D CNN architecture to predict the macroscopic effective stiffness (left figure). Correlation plots between predicted values and target values using 3D CNN and physics inspired approach (right figure). [Yang *et al.* 2018]

1.3.4 Decision Trees & Random Forest

A decision tree [Quinlan 1986] is a non-parametric supervised learning technique that consists of establishing feature-targets relationships using simple decision rules. It has a hierarchical tree structure, composed of a root node, decision nodes and leaf nodes. The root node is the node at the top of the tree that start the graph, it contains a feature that best splits the dataset. Decision nodes is where the variables are evaluated and leaf nodes are the final nodes of the tree that do not have any children and where the predictions are made. Decision tree learning uses a divide-and-conquer strategy by performing a greedy search to identify optimal division points in a tree. This splitting process is then repeated from root to leafs, recursively until each leaf is associated with a class, which is the output of the predictor. Being one of the most intuitive machine learning algorithms, decision trees can be applied for both regression and classification tasks due to their capabilities to handle categorical and numerical data. The main advantage that decision trees offer over black box machine learning algorithms is interpretability, it creates an easy to digest representation of decision making, allowing to better understand why a decision was made. Figure 1.16 shows an example of a binary classification problem of red and green dots based on their coordinates with decision trees. The model starts with a root node and then splits the dataset recursively using the decision nodes (condition $X \leq X_i$ or $Y \leq Y_i$) until reaching pure leaf nodes (classify red or green), if a data sample satisfies the condition at a decision node then it moves to the left child, else it moves to the right until it reaches a leaf node where the predicted class is assigned.

The training process of decision trees for classification problems begins with the determination of the root node that contains the feature that best splits the dataset. This operation is conducted by computing the associated entropy of each feature using equation 1.3 and then keeping the feature with the lowest value. Entropy is a measure of impurity and heterogeneity of a node, its value ranges from 0 (pure) and 1 (impure), a leaf node is a pure node with zero entropy.

$$E(S) = \sum_{i=1}^{C} -p_i \log_2 p_i, \tag{1.3}$$

where C is the total number of classes and p_i is the probability of class i.

Afterward, an information Gain (IG) is calculated for every possible split, this metric given by equation 1.4 represents an average of all entropy values based based on a specific split.

$$\text{IG} = E(\text{ parent }) - \sum w_i E(\text{ child }_i) \tag{1.4}$$

where w_i is the relative size of a child i with respect to the parent.

The main objective is to determine the best split feature and threshold that maximises the information gain. The model performs a greedy search, goes through all the input features and their unique values, computes the information gain for each combination, and records the best feature and the best threshold for each node. In this way, the tree is built recursively. The recursion process could continue indefinitely, in that case some exit conditions can be specified such as the maximum depth and minimum samples at the node.

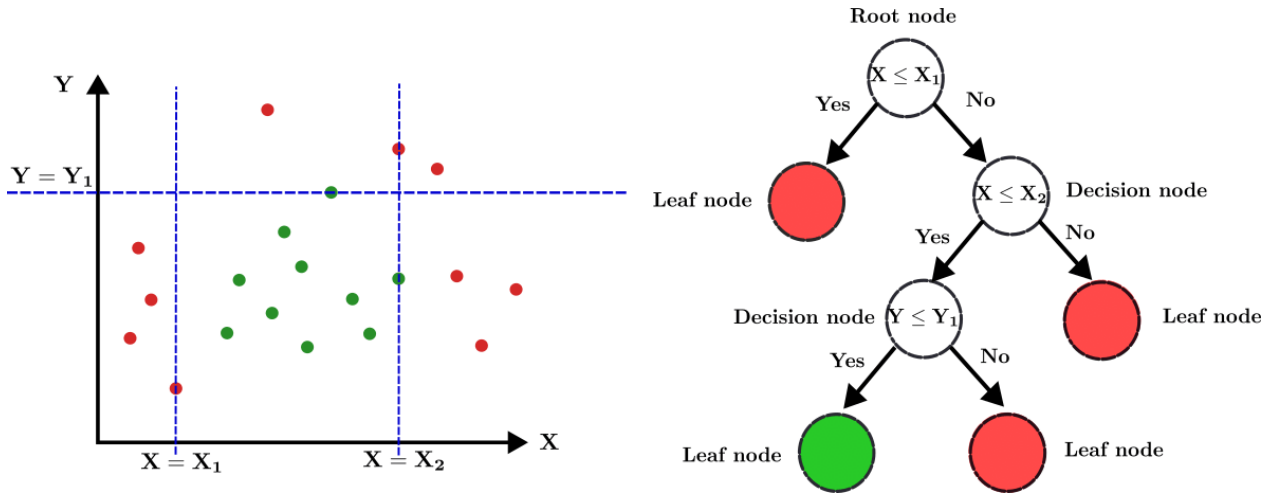


Figure 1.16: An example of a binary decision tree classifier

Despite the concept simplicity behind decision trees, their use remains rather limited due to some drawbacks. Decision trees are generally prone to overfitting and highly sensitive to the training data, which means that small variations within the data can result in a different decision tree, thus they are a high variance estimators. The idea behind the creation of Random Forest (RF) [Breiman 2001] was to address the aforementioned challenges. Random forest is a collection of multiple random decision trees (hence the nomination of random forest) that gathers the predictions of several decision trees and create a final result using an averaging mechanism (mean values for regression problems and majority vote for classification problems). The process of combining results from multiple models is called aggregation. The way random forest handles overfitting and high variance problems better than decision trees is due

to bootstrapping and random feature selection. These two random processes with aggregating are the main cause of the name behind "Random forest". Bootstrapping is the operation of taking a random sample of that data that will be assigned and used for each decision tree. Bootstrapping ensures that same data are note used for every tree, it helps the model to be less sensitive to the original training data. Random feature selection consists of choosing a random set of features instead of all input features. The random feature selection helps to reduce the correlation between the trees, if all the features are used every time for each tree, this will lead to the same decision nodes and the trees will act very similarly. Figure 1.17 shows an example of a random forest with n decision tree to solve a binary classification problem. Each test data point is passed through each tree while saving the associated predictions, then the predicted class is obtained through majority voting.

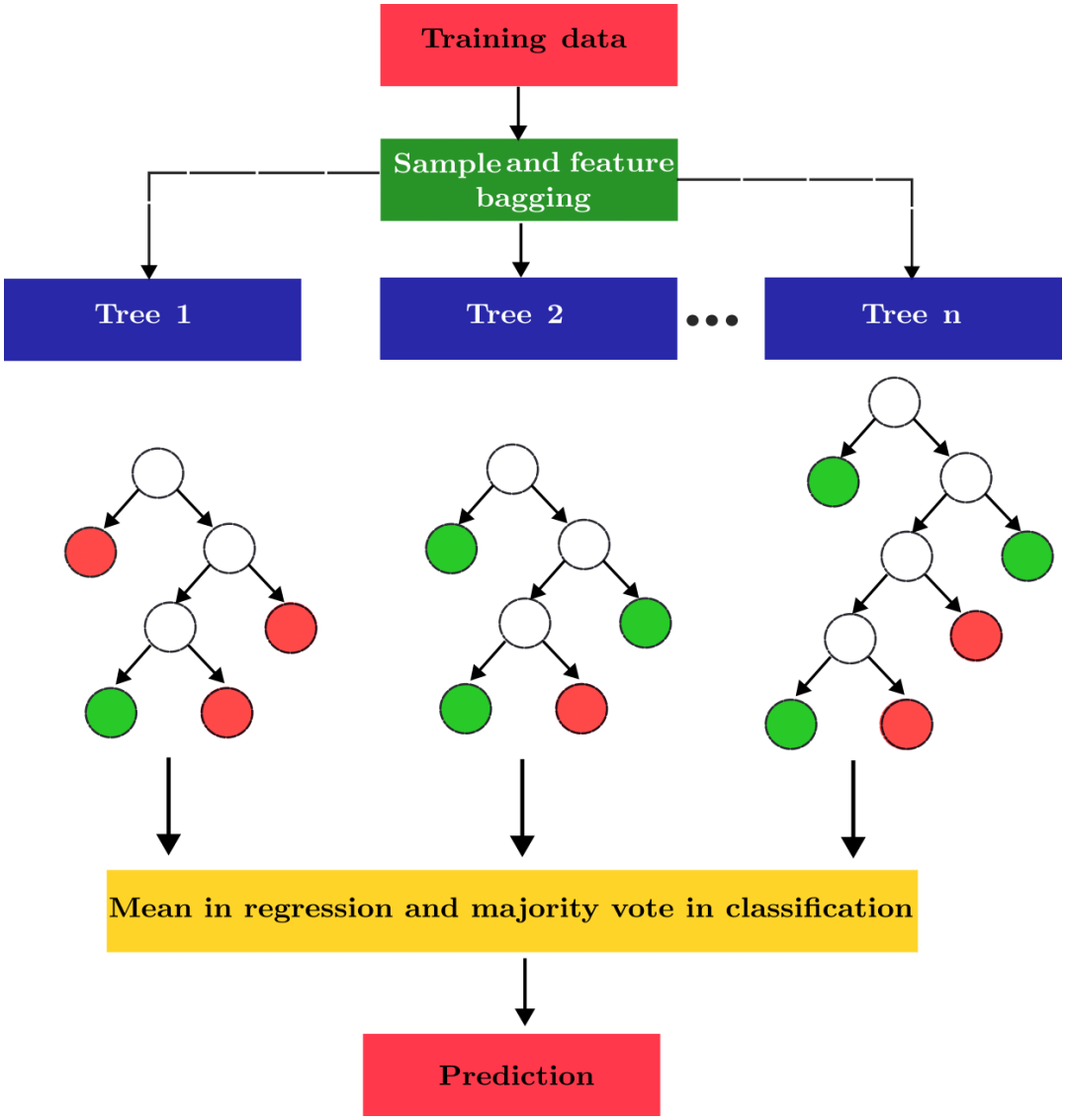


Figure 1.17: An illustration of a random forest with n decision trees to solve a binary classification problem

Despite the various advantages of using tree-based regressors and classifiers in terms of interpretability and performance, these methods are still not widely used in the materials science community compared to black-box machine learning algorithms. Nevertheless, it is noteworthy that decision trees and random forest have achieved high performances in the applications in which they have been used. For instance, RF algorithm have been used by [Yang *et al.* 2019] to predict the elastic properties of silicate glasses. As shown in figure 1.18, a RF algorithm with 200 decision trees, trained on a silicate glass molecular composition database, offers a robust interpolation capability for both the training set and the test set given the obtained values of R^2 parameter. Furthermore, the model does not yield to any noticeable overfitting while being at the same time easy to interpret. Another study conducted by Bulgarevich et Al in [Bulgarevich *et al.* 2018, Bulgarevich *et al.* 2019] showed the great potential of tree-based algorithms to solve classification problems. In this study, an RF algorithm is trained to automatically label various steel materials microstructures (Ferrite/Pearlite, Ferrite/Pearlite/Bainite, and Bainite/Martensite type microstructures) obtained by optical microscopy images. The obtained accuracy by RF was around 99% which exceed the performances achieved by other ML algorithms including Native Bayes, K-Means, K-Nearest Neighbors and Logistic Regression. Regression-tree based algorithms have also been applied by [Liu *et al.* 2020] to solve fracture mechanics problems. In this study, a random forest based metamodel was established to measure fracture toughness of pre-notched pentagonal cross-section cantilevers as a function of specimen geometric dimensions. This study showed that a deep random forest with 512 decision trees of depth equal to 8 gave an average absolute percentage error of nearly 2%.

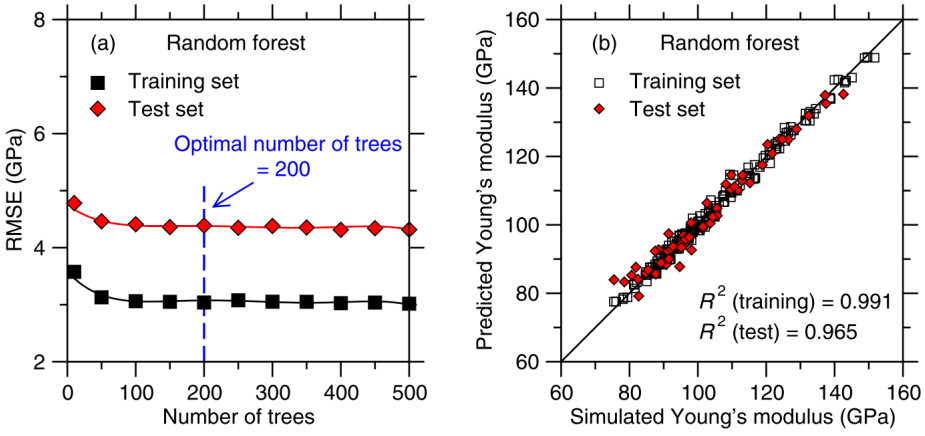


Figure 1.18: RF training and test accuracy as a function of the number of trees considered in each model (left figure). Comparison between the Young's modulus values predicted by RF algorithm and computed by molecular dynamics simulations (right figure) [Yang *et al.* 2019]

1.3.5 Hybrid models

In this chapter, we have highlighted the recent massive use of machine learning methods in various areas of material science and mechanical engineering to investigate complex phenomena. Whether it is for solving regression problems such as constitutive modeling and multiscale modeling or for solving classification problems such as image segmentation, these methods have shown great potential and have considerable advantages over traditional methods. However, all regression-based machine learning algo-

rithms discussed so far are purely data-driven, meaning that they rely solely on data to approximate non-linear behaviors after a training phase. As discussed in 1.1, these approaches, considered as black boxes, do not always guarantee reliable predictions in terms of physical consistency, and their application in engineering sciences can therefore be questionable. To overcome these limitations, material science community has recently become increasingly attractive to the development of hybrid models that integrate scientific knowledge and physical principles into the design of machine learning algorithms. Therefore, a hybrid model can be defined as the combination of a ML technique trained on a set of data resulting from numerical simulations (or experimental tests) in which physical laws are integrated simultaneously in the core of the AI algorithm either in the architecture design or during model training process.

One of the first developed hybrid approaches is the Data Driven Computational Mechanics (DDCM) framework proposed by Kirchdoerfer and Ortiz [Kirchdoerfer & Ortiz 2016]. The general concept behind DDCM is to solve Boundary Value Problems (BVP) by substituting empirical materials constitutive equations with a database of material data points, derived from experimental tests or numerical simulations, while ensuring compatibility and equilibrium conditions. This new paradigm allows to directly integrate material data into numerical computations while bypassing completely material constitutive laws that require calibration step of material parameters which are often not easily identifiable. The DDCM solver consists in minimizing the distance function $\overline{\mathscr{F}}$ (see equation 1.5) between two pairs of strain-stress fields $(\bar{\varepsilon}, \bar{\sigma})$ and $(\bar{\varepsilon}', \bar{\sigma}')$.

$$\overline{\mathscr{F}}(\bar{\varepsilon}, \overline{\boldsymbol{\sigma}}) = \min_{(\bar{\boldsymbol{\varepsilon}}', \bar{\boldsymbol{\sigma}}') \in \bar{D}} \frac{1}{2} \int_{\Omega} \left(\left(\bar{\varepsilon} - \bar{\varepsilon}'\right) : \overline{\mathbb{C}} : \left(\bar{\varepsilon} - \bar{\varepsilon}'\right) + \left(\overline{\boldsymbol{\sigma}} - \overline{\boldsymbol{\sigma}}'\right) : \overline{\mathbb{C}}^{-1} : \left(\overline{\boldsymbol{\sigma}} - \overline{\boldsymbol{\sigma}}'\right) \right) d\Omega \tag{1.5}$$

$(\bar{\varepsilon}', \bar{\sigma}')$ is the pair corresponding to the material response derived from material database $\bar{D}$ and $(\bar{\varepsilon}, \bar{\sigma})$ is the pair that represents the body mechanical state that is constrained by equilibrium and compatibility conditions. As shown in figure 1.19, the DDCM solver aims to find for each material point of the body a mechanical state closest to the material dataset that satisfies the above mentioned conservation laws, this closest point is thus retained as a solution.

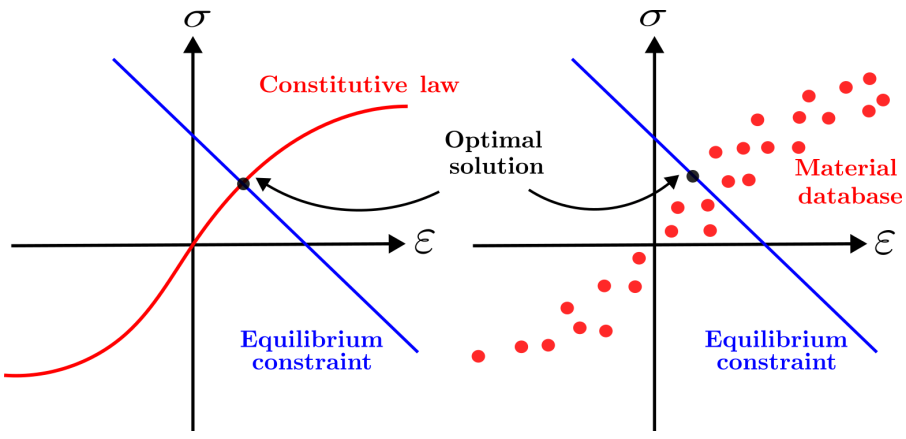


Figure 1.19: The difference between traditional computational mechanics (left figure) and Data driven computational mechanics (right figure)

In recent years, DDCM-based approaches have begun to attract community interest, providing a new avenue for solving engineering problems for which large material databases are available. The DDCM framework was first applied to solve linear elasticity problems [Kirchdoerfer & Ortiz 2016] even in the case of noisy material datasets [Kirchdoerfer & Ortiz 2017]. This data-driven approach was then extended to solve more complex problems, including non-linear elasticity [Nguyen & Keip 2018], computational plasticity [Ibañez *et al.* 2018], finite strain elasticity [Platzer *et al.* 2021], dynamics with noisy data [Kirchdoerfer & Ortiz 2018] and multiscale analysis [Xu *et al.* 2020].

Recent studies have also attempted to hybridize deep learning methods by incorporating physical knowledge into the core of neural network architectures. We recall that the success of deep learning methods lies in the fact that they are universal approximators capable of solving complex and highly non-linear problems. However, the "universal approximator" property comes at a cost, i.e., neural networks models must be deep enough and trained with sufficient data to capture complex behavior. As the complexity of the physical phenomena increases, the generation of large data sets is required to calibrate model parameters. This task is not always feasible due to the high computational costs of this process. Furthermore, even if the issue of generating large databases does not arise, the model's prediction compatibility with physical principles is not always guaranteed since they are purely data-driven. Therefore, the main motivation behind hybrid models in mechanics is to take advantage of the valuable material science knowledge that has been developed over the centuries. This background provides a fine description of physical mechanisms in complex materials that cannot be easily captured by standard ML techniques. In this regard, a new kind of DNN based models called Physics Informed Neural Networks (PINNs) [Raissi *et al.* 2019] has emerged in recent years for the resolution of non-linear Partial Differential Equations (PDEs). This new paradigm integrates the underlying physical laws in the form of PDEs with artificial neural networks to approximate the corresponding solutions. The main difference between PINNs and standard feed forward neural networks lies in the design of the network architecture and the definition of loss functions. Figure 1.20 shows a typical PINN framework to solve non-linear PDEs with Dirichlet boundary conditions.

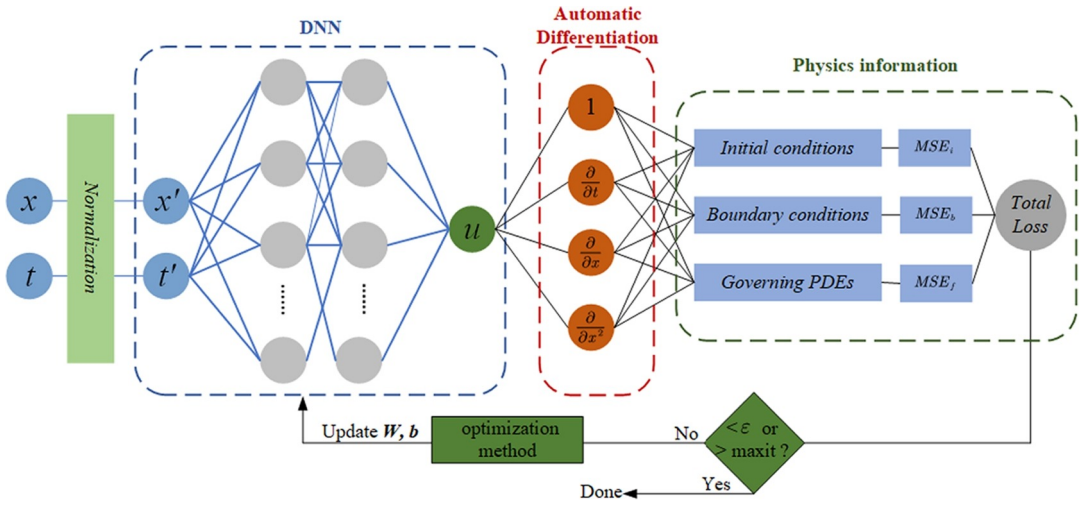


Figure 1.20: PINNs architecture to solve non-linear PDEs with Dirichlet boundary conditions [Yuan *et al.* 2022]

Considering an n-order partial differential equation given by the following system:

$$\begin{aligned} &\lambda \frac{\partial^n u(x,t)}{\partial x^n} + f(x,t) = 0 \\ &u(0,t) = \varphi(0,t) \\ &u(x,0) = g(x,0) \end{aligned} \tag{1.6}$$

where $u(x,t)$ is the unknown solution, $\varphi(0,t)$ is the Dirichlet boundary condition at $x = 0$ and $g(x,0)$ is the initial condition at $t = 0$.

The DNN part of the PINN architecture takes the position x_i and the time t_i as inputs and returns the predicted solution of the PDE $u_{\text{pred}}\ (x_i, t_i)$. Then by using the automatic differentiation algorithm of FFNN, the partial derivatives of $u_{\text{pred}}\ (x_i, t_i)$ with respect to x_i and t_i are computed. Theses quantities will serve to calculate the residuals mean squared errors of PDE MSE_u, boundary conditions MSE_φ and initial conditions MSE_g. Furthermore, a penalty term MSE_r is added to the total loss MSE_Σ. This regularization term is responsible for forcing the predictions to satisfy the governing equation of the BVP. The expression of the above mentioned loss function is given by:

$$\begin{aligned} MSE_u &= \frac{1}{N_u} \sum_{i=1}^{N_u} \left(u_{\text{pred}}\ (x_i, t_i) - u_{\text{exact}}\ (x_i, t_i)\right)^2 \\ MSE_\varphi &= \frac{1}{N_\varphi} \sum_{i=1}^{N_\varphi} \left(u_{\text{pred}}\ (0, t_i) - \varphi\,(0, t_i)\right)^2 \\ MSE_g &= \frac{1}{N_g} \sum_{i=1}^{N_g} \left(u_{\text{pred}}\ (x_i, 0;) - g\,(x_i, 0)\right)^2 \\ MSE_r &= \frac{1}{N_u} \sum_{i=1}^{N_u} \left(\lambda \frac{\partial^n u_{\text{pred}}\ (x_i, t_i)}{\partial x^n} + f\,(x_i, t_i)\right)^2 \\ MSE_\Sigma &= MSE_u + MSE_\varphi + MSE_g + MSE_r \end{aligned}$$

where N_u,N_φ,N_g are the number of training sampling points of the equation domain, boundary conditions and initial conditions respectively. $u_{\text{exact}}\ (x_i, t_i)$, $\varphi\,(0, t_i)$ and $g\,(x_i, 0)$ are the exact values at the sampling points for the solution u, Dirichlet boundary conditions and initial condition respectively.

The training process of the PINN model consists in minimizing the loss function MSE_Σ which is penalized by a regularization term that constraint the predictions to satisfy the problem governing equations.

Successful applications of PINNs models have been highlighted in [Cai *et al.* 2021] to solve heat transfer equations including convection and multi-physics problems, and in [Cai *et al.* 2022] to solve fluid mechanics problems with a particular focus on Navier Stokes equations. In material science, a new class of artificial neural networks called Thermodynamic-based Neural Network (TANN) [Masi *et al.* 2021] has been recently developed for material points constitutive modeling. This approach, inspired by PINNs

models that benefit from the automatic differentiation property, allows to encode the thermodynamics principles in the ANN architecture to obtain consistent predictions with the laws of physics. The general concept of TANN consists on training a feed forward neural network on two scalar quantities: free energy potential and the dissipation rate. Then using the automatic differentiation properties in a similar way to PINNs, the relationship between the free energy and the stress on one hand and the dissipation rate and the internal variables on the other hand can be computed using the derivation operators. By adopting such methodology, the model does not require an identification of the underlying thermodynamic laws during the training process as they are already incorporated in the TANN architecture. TANNs have been applied to predict the mechanical responses of elastoplastic materials with strain hardening and strain softening, the preliminary results shows that TANNs can achieve higher accuracy than standard ANN approaches. The integration of the laws of thermodynamics in the ANN core have significantly improved the model generalization capabilities, especially for unseen test datasets.

According to recent studies conducted by [Settgast *et al.* 2019b, Malik *et al.* 2021], hybrid approaches have also been extended for multiscale analysis to describe the macroscopic behavior of 3D foam structures under proportional loading paths. This hybrid scale bridging approach has proven to be more efficient and reliable for capturing complex inelastic responses of cellular structures compared to the rather costly traditional methods including FE^2. The originality of this approach lies in the fact that the formulation of the constitutive laws using plasticity frameworks is employed at both the micro and macro scales. In addition, neural networks are used to approximate the macroscale yield function and flow directions to obtain a complete formulation of the macroscopic problem. To assess the robustness of this hybrid approach, several numerical examples on Wheire-Phelan foam structures have been performed and the obtained results have shown model's reliability for capturing stress-strain relationships for different loading cases and the evolution of internal state variables.

1.4 Conclusion

The first chapter of this manuscript was dedicated to a state of art review regarding the use of Data-Driven and machine learning approaches for material science and engineering applications. Numerous practical examples of such applications have been highlighted in this chapter, including multiscale analysis, constitutive modeling, material design and durability. According to the contributions of the mechanics of materials community in this research area, it has been proven that this type of approach has a great potential to solve complex engineering problems in a fast and efficient way compared to traditional methods. The adopted strategy for treating mechanics of materials problems with a machine learning based technique relies mainly on three necessary steps. First, a well defined research problem (regression, classification or clustering ?) that cannot be easily addressed with the conventional methods. Then, a generated database with sufficient samples obtained from computational simulations or experimental tests. The data can be labeled or unlabeled depending on the adopted training technique. The last step involves the choice of the most suitable ML algorithm since different training strategies can be considered depending on the problem to be addressed such as supervised learning technique for regression and classification problems or unsupervised learning for clustering applications. Special attention has also been given in this chapter to the use of hybrid approaches combining ML techniques and physics-based knowledge to enhance the robustness of AI-based methods. This new paradigm allows to take into

account fundamental physical laws when predicting complex mechanical behaviors. In that sense, the next chapter of this manuscript aims at proposing a new hybrid physics-AI based model to predict non-linear mechanical behaviors of dissipative materials.

Chapter 2

Thermodynamically consistent Recurrent Neural Networks for dissipative materials constitutive modeling

Contents

2.1 Introduction

The recent development of data-driven approaches to overcome the computational costs limitations is attracting the interest of the material science community. The use of Artificial Neural Networks (ANN) as surrogate models has been proven to be a reliable tool for several applications including structural analysis, constitutive modeling and multi-scale modeling. This is due to fact that ANN are considered to be a universal approximation algorithms capable of learning any mapping function and identifying complex non-linear relationships. For a specific class of problems where data is available, the development of such models can be a promising substitute for traditional numerical methods for multiscale structural analysis (Finite Element Analysis, FFT methods) which often require important computational costs. As presented in chapter 1, different ANN models and architectures have been developed to deal with the challenges of mechanics of materials. The most common one being the Multi Layer Perceptron (MLP) presented in 1.3.2. The MLP architecture has been used by several authors as a surrogate for constitutive modeling applications [Furukawa & Yagawa 1998, Waszczyszyn & Ziemiański 2001, Settgast *et al.* 2019a, Ali *et al.* 2019]. However, we would like to point out that these studies have been conducted on a particular case which is the proportionality of the loading paths, thus leading to a significant simplification of the problem. According to [Gorji *et al.* 2020], it has been shown that feed forward neural networks are not adapted for history-dependant constitutive modeling. Using an MLP as a non linear constitutive law surrogate for dissipative materials under non-proportional loading paths is very challenging given that no direct relationship exists between the mechanical state at the current step and at the previous time step. Thus, the loading history is not taken into account in that case as we recall that the information flow is propagated only in the forward direction for feed forward neural networks. The above stated issue can be solved by introducing a specific ANN architecture known as Recurrent Neural Network (RNN) where the loading history dependency can be taken into consideration. RNN [Jain & Medsker 1999] are designed to handle temporal sequences using its "memory". Unlike MLP, RNN inputs are not only related to the inputs at the current time step, but also to the outputs from all the previous steps. This type of architecture have been extensively applied in tasks such as speech recognition, text generation or machine translation. For material science applications, RNN based architectures have been recently employed for constitutive modeling of plasticity and thermo-viscoplasticity, respectively [Gorji *et al.* 2020, Abueidda *et al.* 2021].

While RNN based methods have shown high potential as reliable tools for modeling path dependant behaviors, we recall that data driven techniques in general including RNN are approximation algorithms considered as "Black box" that maps inputs data into outputs after a training process. This lack of physical consistency can present major limitations on model's reliability when the explicit forms of the governing physical laws are not taken into consideration while making predictions. Therefore, the use of "Black box" models for engineering applications can sometimes be questionable when the respect of physical laws is not guaranteed. To overcome this limitation, the development of hybrid models with a full use of data and physics can potentially provide reliable results in terms on ANN generalization ability as well as a physical consistency. For this purpose, a hybrid Thermodynamically Consistent Recurrent Neural Network (ThC-RNN) is developed by incorporating physics-based loss functions to train a Stacked Long Short Term Memory (LSTM) network. To the best of our knowledge, only one attempt have been made by [Daw *et al.* 2017] for lake temperature modeling with physics guided MLP Neural Networks using a similar approach to the one presented in this manuscript. Recently, F.Masi et Al [Masi *et al.* 2021] have

also developed a Thermodynamic-based ANN (TANN) for constitutive modeling. The main differences between the two approaches will be discussed in 2.3.3. The main objective of this chapter is to investigate the effect of combining physical laws and particularly thermodynamics laws with RNN based models for dissipative materials constitutive modeling.

For that propose, the present chapter is organized as follows: First, the thermodynamic framework of dissipative materials constitutive modeling is presented in Section 2.2. This framework allows to express properly the governing physical laws that need to be incorporated with AI based models. Next, a brief recall of the thermodynamic foundations of two constitutive laws is given for the *Elastoplasticity with isotropic hardening* and *Chaboche plasticity model* with an isotropic hardening and two non-linear kinematic hardening [Lemaitre & Chaboche 1990]. In Section 2.3, the basic principle of RNN is briefly recalled and selection procedure of the suitable RNN architecture for each constitutive law is presented. The essence of RNN model's training procedure which is the Back Propagation Through Time (BPTT) is explained along with the methodology of introducing physics based laws during model's training phase. Section 2.4 shows a comparison between the predicted mechanical behavior with a standard RNN model and ThC-RNN to highlight the effect of the thermodynamic consistency on models accuracy. The application of ThC-RNN model in a Finite Element analysis framework to simulate structures in real-life conditions is also presented. Finally, some concluding remarks are given in Section 2.5.

2.2 Thermodynamic formulation of constitutive laws for dissipative materials

The present section is dedicated to briefly recall the thermodynamic foundations of constitutive modeling. All the following constitutive models are based on the thermodynamics of irreversible processes with the formalism described in [Maugin 1999] by Maugin. The necessary equations resulting from this formalism will be presented with the notations used in chapter 2 monograph of Chatzigeorgiou et al [Chatzigeorgiou *et al.* 2018]. The thermodynamic quantities used in this chapter are:

- $\boldsymbol{\gamma_{loc}}$: the local generated entropy or the intrinsic dissipation in the material.

$\boldsymbol{\gamma_{loc}}$ is given by equation 2.1 (first principle of thermodynamics) and must satisfy the inequality 2.2 as a result of the application of the second law of thermodynamics:

$$\gamma_{\text{loc}} = \boldsymbol{\sigma} : \dot{\boldsymbol{\varepsilon}} + \theta\dot{\eta} - \dot{E}, \tag{2.1}$$

$$\gamma_{\text{loc}} \geq 0. \tag{2.2}$$

- $\boldsymbol{\sigma}$: the second order Cauchy stress tensor.
- $\dot{\boldsymbol{\varepsilon}}$: the second order strain tensor rate.
- $\boldsymbol{E}$: the internal energy per unit of volume.
- θ : the absolute temperature.
- η : the entropy per unit of volume.

Using a free energy potential, $\boldsymbol{\gamma_{loc}}$ can be expressed by equation 2.3:

$$\gamma_{loc} = -\frac{\partial \Psi}{\partial \zeta} : \dot{\zeta}, \quad \gamma_{loc} \geq 0. \tag{2.3}$$

- $\boldsymbol{\Psi}$: Helmholtz free energy potential.
- $\boldsymbol{\zeta}$: the set internal state variables.

Helmholtz free energy potential can be split into a recoverable and a irrecoverable part as expressed in equation 2.4, as proposed in [Chatzigeorgiou *et al.* 2018].

$$\Psi(\varepsilon, \theta, \boldsymbol{\zeta}) = \Psi^{\mathrm{r}}(\varepsilon, \theta, \boldsymbol{\zeta}) + \Psi^{\mathrm{ir}}(\theta, \boldsymbol{\zeta}). \tag{2.4}$$

- $\boldsymbol{\Psi^{\mathrm{r}}}$: the part of the internal energy that can be restored during a thermomechanical loading path even if it is followed by a change in the state variables.
- $\boldsymbol{\Psi^{\mathrm{ir}}}$: the part of the internal energy which permanently changes with the internal variables ζ.

According to the decomposition given by equation 2.4, the stress and strain tensors are only defined from the recoverable part of Ψ.

$$\sigma = \frac{\partial \Psi^{\mathrm{r}}}{\partial \varepsilon}. \tag{2.5}$$

The intrinsic dissipation γ_{loc} is defined from both parts using equations 2.3 and 2.4.

$$\gamma_{\mathrm{loc}} = -\frac{\partial \Psi^{\mathrm{r}}}{\partial \boldsymbol{\zeta}} : \dot{\zeta} - \frac{\partial \Psi^{\mathrm{ir}}}{\partial \boldsymbol{\zeta}} : \dot{\zeta}. \tag{2.6}$$

The total mechanical power $\dot{\mathrm{W}}_{\mathrm{m}}$ can be decomposed into 3 terms as expressed in equation 2.7:

$$\dot{\mathrm{W}}_{\mathrm{m}} = \dot{\mathrm{W}}^{\mathrm{r}}_{\mathrm{m}} + \dot{\mathrm{W}}^{\mathrm{ir}}_{\mathrm{m}} + \dot{\mathrm{W}}^{\mathrm{d}}_{\mathrm{m}}. \tag{2.7}$$

- $\boldsymbol{\dot{\mathrm{W}}^{\mathrm{r}}_{\mathrm{m}}}$: the recoverable part of the total mechanical power.

$$\dot{\mathrm{W}}^{\mathrm{r}}_{\mathrm{m}} = \boldsymbol{\sigma} : \dot{\varepsilon} + \frac{\partial \Psi^{\mathrm{r}}}{\partial \zeta} : \dot{\zeta}. \tag{2.8}$$

- $\boldsymbol{\dot{\mathrm{W}}^{\mathrm{ir}}_{\mathrm{m}}}$: the irrecoverable part of the total mechanical power.

$$\dot{\mathrm{W}}^{\mathrm{ir}}_{\mathrm{m}} = \frac{\partial \Psi^{\mathrm{ir}}}{\partial \boldsymbol{\zeta}} : \dot{\zeta}. \tag{2.9}$$

- $\boldsymbol{\dot{\mathrm{W}}^{\mathrm{d}}_{\mathrm{m}}}$: the dissipative part of the total mechanical power.

$$\dot{\mathrm{W}}^{\mathrm{d}}_{\mathrm{m}} = \gamma_{\mathrm{loc}} = -\frac{\partial \Psi^{\mathrm{r}}}{\partial \boldsymbol{\zeta}} : \dot{\zeta} - \frac{\partial \Psi^{\mathrm{ir}}}{\partial \boldsymbol{\zeta}} : \dot{\zeta}. \tag{2.10}$$

- $\boldsymbol{\dot{\mathrm{W}}_{\mathrm{m}}}$: the total mechanical power.

$$\dot{\mathrm{W}}_{\mathrm{m}} = \boldsymbol{\sigma} : \dot{\varepsilon}. \tag{2.11}$$

Once the thermodynamic framework of material constitutive modeling is established and the mechanical and energetic quantities are expressed as a function of free energy potentials, the methodology is applied to formulate the associated constitutive law of dissipative materials. Two examples are presented in the following subsections, the *Elastoplasticity with isotropic hardening* and *Chaboche plasticity model with combined isotropic hardening and two non-linear kinematic hardening.*

2.2.1 Elastoplasticity with isotropic hardening

Under small deformations assumptions, the total strain ε for elastoplastic material under pure mechanical loading is additively decomposed into an elastic term ε^e and a plastic term ε^p.

$$\varepsilon = \varepsilon^e + \varepsilon^p. \tag{2.12}$$

The associated Helmholtz free energy potential of an elastoplastic material with isotropic hardening can be split into a recoverable part Ψ^r and an irrecoverable part Ψ^{ir}. Ψ^r depends on the total strain ε and the plastic strain ε^p. Ψ^{ir} is expressed in terms of an internal variable p corresponding to the accumulated plastic strain.

$$\Psi\left(\varepsilon, \varepsilon^{\mathrm{p}}, p\right) = \Psi^{\mathrm{r}}\left(\varepsilon, \varepsilon^{\mathrm{p}}\right) + \Psi^{\mathrm{ir}}(p), \tag{2.13}$$

$$\Psi^{\mathrm{r}}\left(\varepsilon, \varepsilon^{\mathrm{p}}\right) = \frac{1}{2}\left[\varepsilon - \varepsilon^{\mathrm{p}}\right] : \boldsymbol{L} : \left[\varepsilon - \varepsilon^{\mathrm{p}}\right], \tag{2.14}$$

$$\Psi^{\mathrm{ir}}(p) = f(p). \tag{2.15}$$

$\boldsymbol{L}$ is the elastic stifness tensor and $f(p)$ is the elastoplastic hardening function.

Using equations 2.5 and 2.6, the associated constitutive equation and the intrinsic dissipation are given by the following expressions:

$$\boldsymbol{\sigma} = L : \left[\varepsilon - \varepsilon^{\mathrm{p}}\right], \tag{2.16}$$

$$\gamma_{\mathrm{loc}} = -\frac{\partial \Psi^{\mathrm{r}}}{\partial \varepsilon^{\mathrm{p}}} : \dot{\varepsilon}^{\mathrm{p}} - \frac{\partial \Psi^{\mathrm{ir}}}{\partial p}\dot{p} = \boldsymbol{\sigma} : \dot{\varepsilon}^{\mathrm{p}} - f_p \dot{p}, \tag{2.17}$$

where $f_p = \partial f / \partial p$.

For Von Mises type material with exponential isotropic hardening, the yield surface is given by:

$$\Phi = \sigma^{\mathrm{vM}} - f_p - \sigma_Y = \sigma^{\mathrm{vM}} - H p^n - \sigma_Y, \tag{2.18}$$

H and n are the hardening parameter and exponent respectively. σ_Y is the yield stress limit and σ^{vM} is the Von Mises equivalent stress.

$$\sigma^{\mathrm{vM}} = \sqrt{\frac{3}{2}\boldsymbol{\sigma}^{\mathrm{dev}} : \boldsymbol{\sigma}^{\mathrm{dev}}}, \quad \boldsymbol{\sigma}^{\mathrm{dev}} = \boldsymbol{\sigma} - \frac{1}{3}\operatorname{tr}\boldsymbol{\sigma}\boldsymbol{I}. \tag{2.19}$$

Using the convex analysis for generalized standard materials [Halphen & Nguyen 1975], the evolution of the plastic strain using the flow rule is given by the following set of equations:

$$\dot{\varepsilon}^{\mathrm{p}} = \dot{p}\Lambda, \quad \Lambda = \frac{\partial \Phi}{\partial \boldsymbol{\sigma}} = \frac{3}{2}\frac{\sigma^{\mathrm{dev}}}{\sigma^{\mathrm{vM}}}. \tag{2.20}$$

The Kuhn-Tucker conditions define the non-linear problem to be solved:

$$\Phi \leq 0, \quad \dot{p} \geq 0, \quad \Phi \dot{p} = 0. \tag{2.21}$$

The corresponding mechanical powers for elastoplastic materials with an exponential isotropic hardening can be expressed using the definitions given in equations 2.8,2.9 and 2.10:

$$\dot{\mathrm{W}}^{\mathrm{r}}_{\mathrm{m}} = \boldsymbol{\sigma} : \dot{\varepsilon} + \frac{\partial \Psi^{\mathrm{r}}}{\partial \varepsilon^{\mathrm{p}}} : \dot{\varepsilon}^{\mathrm{p}} = \sigma : \left[\dot{\varepsilon} - \dot{\varepsilon}^{\mathrm{p}}\right], \tag{2.22}$$

$$\dot{\mathrm{W}}^{\mathrm{ir}}_{\mathrm{m}} = \frac{\partial \Psi^{\mathrm{ir}}}{\partial p}\dot{p} = H p^{n} \dot{p}, \tag{2.23}$$

$$\dot{\mathrm{W}}^{\mathrm{d}}_{\mathrm{m}} = -\frac{\partial \Psi^{\mathrm{r}}}{\partial \varepsilon^{\mathrm{p}}} : \dot{\varepsilon}^{\mathrm{p}} - \frac{\partial \Psi^{\mathrm{ir}}}{\partial p}\dot{p} = \boldsymbol{\sigma} : \dot{\varepsilon}^{\mathrm{p}} - H p^{n} \dot{p}, \tag{2.24}$$

2.2.2 Chaboche plasticity model with a combined isotropic hardening and two non-linear kinematic hardening

The present Chaboche plasticity model is a constitutive law combining an isotropic and two non-linear kinematic hardening. This superposition results in a translation and an expansion of material elastic domain. Ψ^{r} depends on the total strain ε, the plastic strain ε^{p} and the second order tensors $\boldsymbol{a_1}$ and $\boldsymbol{a_2}$ related to the kinematic hardening. Ψ^{ir} is expressed in terms of the internal variable p corresponding to the accumulated plastic strain.

$$\Psi\left(\varepsilon, \varepsilon^{\mathrm{p}}, \boldsymbol{a_1}, \boldsymbol{a_2}, p\right) = \Psi^{\mathrm{r}}\left(\varepsilon, \varepsilon^{\mathrm{p}}, \boldsymbol{a_1}, \boldsymbol{a_2}\right) + \Psi^{\mathrm{ir}}(p), \tag{2.25}$$

$$\begin{aligned} \Psi^{\mathrm{r}}\left(\varepsilon, \varepsilon^{\mathrm{p}}, \boldsymbol{a_1}, \boldsymbol{a_2}\right) = & \frac{1}{2}\left[\varepsilon - \varepsilon^{\mathrm{p}}\right] : \boldsymbol{L} : \left[\varepsilon - \varepsilon^{\mathrm{p}}\right] \\ & + \frac{1}{3} C_1 \boldsymbol{a_1} : \boldsymbol{a_1} + \frac{1}{3} C_2 \boldsymbol{a_2} : \boldsymbol{a_2}, \end{aligned} \tag{2.26}$$

$$\Psi^{\mathrm{ir}}(p) = H(p). \tag{2.27}$$

The back stress tensors $\boldsymbol{X_1}$ and $\boldsymbol{X_2}$ are related to $\boldsymbol{a_1}$ and $\boldsymbol{a_2}$ by the following expression:

$$\boldsymbol{X_1} = \frac{2}{3} C_1 \boldsymbol{a_1} \quad , \quad \boldsymbol{X_2} = \frac{2}{3} C_2 \boldsymbol{a_2}, \tag{2.28}$$

$$\boldsymbol{X} = \boldsymbol{X_1} + \boldsymbol{X_2}. \tag{2.29}$$

The evolution equations of the kinematic variables $\boldsymbol{X_1}$ and $\boldsymbol{X_2}$ depends on the accumulated plastic strain p:

$$\dot{\boldsymbol{X}}_{\mathbf{1}} = \frac{2}{3} C_1 \dot{\boldsymbol{\varepsilon}}^p - D_1 \boldsymbol{X}_{\mathbf{1}} \dot{p}, \tag{2.30}$$

$$\dot{\boldsymbol{X}}_{\mathbf{2}} = \frac{2}{3} C_2 \dot{\boldsymbol{\varepsilon}}^p - D_2 \boldsymbol{X}_{\mathbf{1}} \dot{p}. \tag{2.31}$$

C_1,D_1,C_2,D_2 are material parameters related with kinematic hardening, the terms $D_1 \boldsymbol{X}_{\mathbf{1}} \dot{p}$ and $D_2 \boldsymbol{X}_{\mathbf{1}} \dot{p}$ correspond to the material softening effect.

The evolution of the isotropic hardening function $H(p)$ as a function of the accumulated plastic strain p is given by the following equation:

$$\dot{H} = b(Q - H)\dot{p}. \tag{2.32}$$

The parameter Q is defined as the asymptotic value of the elastic limit and b is a parameter related to the rate of evolution of the elastic limit.

Using equations 2.5 and 2.6, the associated constitutive equation and the intrinsic dissipation are given by the following expressions:

$$\boldsymbol{\sigma} = L : \left[\varepsilon - \varepsilon^{\mathrm{p}}\right], \tag{2.33}$$

$$\begin{aligned} \gamma_{\mathrm{loc}} &= -\frac{\partial \Psi^{\mathrm{r}}}{\partial \varepsilon^{\mathrm{p}}} : \dot{\varepsilon}^{\mathrm{p}} - \frac{\partial \Psi^{\mathrm{r}}}{\partial a_1} : \dot{a_1} - \frac{\partial \Psi^{\mathrm{r}}}{\partial a_2} : \dot{a_2} - \frac{\partial \Psi^{\mathrm{ir}}}{\partial p} \dot{p} \\ \gamma_{\mathrm{loc}} &= \boldsymbol{\sigma} : \dot{\varepsilon}^{\mathrm{p}} - \boldsymbol{X_1} : \dot{\boldsymbol{a_1}} - \boldsymbol{X_2} : \dot{\boldsymbol{a_2}} - \frac{\partial H}{\partial p} \dot{p}. \end{aligned} \tag{2.34}$$

For combined isotropic and kinematic hardening, the yield surface is given by the form:

$$\Phi = J(\boldsymbol{\sigma} - \boldsymbol{X}) - H_p - \sigma_Y, \tag{2.35}$$

where

$$\begin{aligned} J(\boldsymbol{\sigma} - \boldsymbol{X}) &= \sqrt{\frac{3}{2}[\boldsymbol{\sigma} - \boldsymbol{X}]' : [\boldsymbol{\sigma} - \boldsymbol{X}]'} \\ [\boldsymbol{\sigma} - \boldsymbol{X}]' &= \boldsymbol{\sigma} - \boldsymbol{X} - \frac{1}{3} \operatorname{tr}(\boldsymbol{\sigma} - \boldsymbol{X}) \boldsymbol{I} \\ H_p &= \partial H / \partial p = b(Q - H). \end{aligned}$$

Using the convex analysis for generalized standard materials, the evolution of the plastic strain for a material obeying Von Mises criterion is given by the following set of equations:

$$\dot{\varepsilon}^{\mathrm{p}} = \dot{p}\Lambda, \quad \boldsymbol{\Lambda} = \frac{3[\boldsymbol{\sigma} - \boldsymbol{X}]'}{2J(\boldsymbol{\sigma} - \boldsymbol{X})}. \tag{2.36}$$

The Kuhn-Tucker conditions define the non-linear problem to be solved:

$$\Phi \leq 0, \quad \dot{p} \geq 0, \quad \Phi \dot{p} = 0. \tag{2.37}$$

The corresponding work rates for Chaboche plasticity model can be expressed using the definitions given in equations 2.8,2.9 and 2.10:

$$\dot{\mathrm{W}}_{\mathrm{m}}^{\mathrm{r}} = \sigma : \left[\dot{\varepsilon} - \dot{\varepsilon}^{\mathrm{P}}\right] + \boldsymbol{X_1} : \dot{a}_1 + \boldsymbol{X_2} : \dot{a}_2, \tag{2.38}$$

$$\dot{\mathrm{W}}_{\mathrm{m}}^{\mathrm{ir}} = \frac{\partial H}{\partial p}\dot{p}, \tag{2.39}$$

$$\dot{\mathrm{W}}_{\mathrm{m}}^{\mathrm{d}} = \boldsymbol{\sigma} : \dot{\varepsilon}^{\mathrm{p}} - \boldsymbol{X_1} : \dot{\boldsymbol{a}}_{\boldsymbol{1}} - \boldsymbol{X_2} : \dot{\boldsymbol{a}}_{\boldsymbol{2}} - \frac{\partial H}{\partial p}\dot{p}. \tag{2.40}$$

2.3 Thermodynamically Consistent Recurrent Neural Networks (ThC-RNN)

2.3.1 Neural Network architecture selection

As discussed in the introduction, the MLP neural network architecture is not suitable to treat history dependant behaviors given that no direct relationship exists between the network inputs at a current time step and its outputs from the previous time steps. To overcome this challenge, the RNN architecture is introduced as a reliable model capable of handling this history dependency. Figure 2.1 represents an illustration of the RNN architecture. The set of variables (x_t, y_t, h_t) are inputs, outputs and hidden state vectors at the time step t respectively. (W_x, W_y, W_h) are the weight matrix related to each of the variables. As shown in figure 2.1, the computation of the output vector y_t at the time step t requires both variables x_t the input vector at the current step t and h_{t-1} the hidden state at the previous time step $t-1$. The hidden state variable h plays a major role by taking into account the history dependency as it allows the network to carry the information from previous time steps onto future predictions. For path dependant plasticity, we would like that h contains some information about the current plastic state of the material, in the same way internal variables like accumulated plastic strains or back-stress tensors, does in classical elastoplastic constitutive laws.

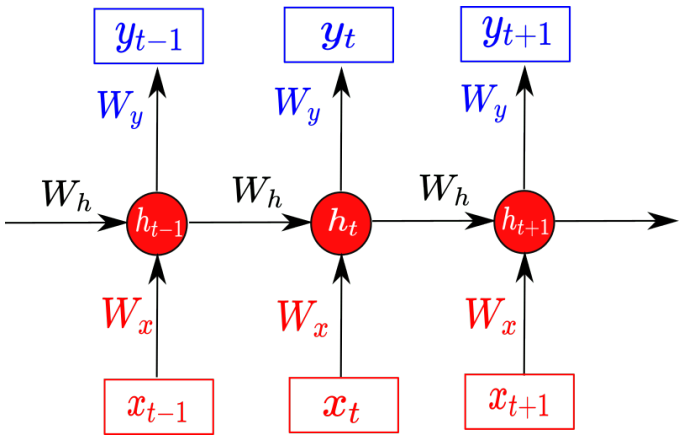


Figure 2.1: Recurrent Neural Network basic architecture

The training process of RNN is based on Backpropagation Through Time (BPTT) [Werbos 1990] algorithm. When dealing with long sequences inputs, RNN training may suffer from a common phenomenon known as vanishing and exploding gradients. To solve this problem, Long Short Term Memory (LSTM) Neural Network has been introduced in [Hochreiter & Schmidhuber 1997]. LSTM architecture has been proven to be a successful formulation of RNN as it is based on a gating mechanism, allowing the network to control the flow of information within its gates. As shown in figure 2.2, a basic LSTM cell consist on four main gates, an input gate (to update and add new information), a forget gate (to decide which information is kept or forgotten from the previous accumulated information), a memory gate (to provide the network with a Long term memory of past events) and an output gate (to predict the next hidden state). At each time step, the information flow is regulated into and out of the LSTM cell using the four gates.

The input vector to the LSTM unit at the time step t is $x_t \in \mathbb{R}^d$ and the output vector is $h_t \in \mathbb{R}^h$ where the superscripts d and h refer to the dimension of inputs and outputs features respectively. The LSTM unit is composed of a memory cell state $c_t \in \mathbb{R}^h$, an input gate $i_t \in \mathbb{R}^h$, an output gate $o_t \in \mathbb{R}^h$, a forget gate $f_t \in \mathbb{R}^h$ and a memory cell candidate $\tilde{c}_t$. The compact forms of LSTM gates are given by the following expressions :

$$f_t = \tilde{\sigma}(W_f x_t + U_f h_{t-1} + b_f), \tag{2.41}$$

$$i_t = \tilde{\sigma}(W_i x_t + U_i h_{t-1} + b_i), \tag{2.42}$$

$$o_t = \tilde{\sigma}(W_o x_t + U_o h_{t-1} + b_o), \tag{2.43}$$

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c), \tag{2.44}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t. \tag{2.45}$$

The operator $\odot$ denotes the element-wise product, $\tilde{\sigma}$ is the sigmoid function and tanh is the hyperbolic tangent function.

$$\tilde{\sigma}(x) = \frac{1}{1+e^{-x}},$$

$$\tanh(\mathrm{x}) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}.$$

W_f, W_i, W_o, W_c, U_f, U_i, U_o and U_c are weight matrix, b_f, b_i, b_o, b_c are bias vector parameters. $W \in \mathbb{R}^{h\times d}$, $U \in \mathbb{R}^{h\times h}$ and $b \in \mathbb{R}^h$ are the set of parameters which needs to be learned during backpropagation process. The output vector from the LSTM cell can finally be expressed using the following expression:

$$h_t = o_t \odot \tanh(c_t) \tag{2.46}$$

where the initial values are $c_0 = 0$ and $h_0 = 0$.

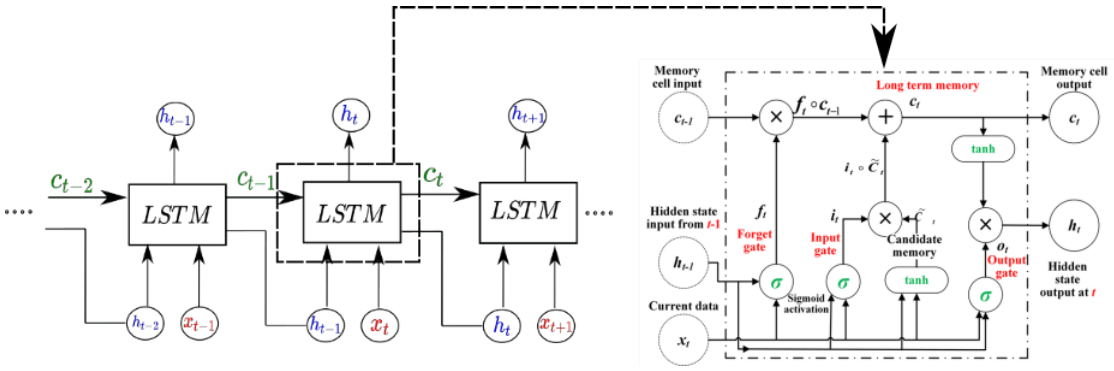


Figure 2.2: Architecture of a Basic LSTM Neural Network unrolled through time

In this study, a stacked LSTM architecture has been used instead of a model with a single LSTM cell. Adding multiple LSTM cells increase the depth of the network, thus enhancing its capabilities to capture more complex behaviors. We recall that the success of deep learning techniques is commonly attributed to the hierarchy that exists between its layers, where each layer performs a specific part of the task and passes it on to the next. Therefore, increasing the depth of the neural network appears to be important when dealing with complex non-linear problems. The selection of the suitable model has been also justified by conducting an hyperparameter study by comparing the predictive capabilities of a single cell and two cells LSTM models respectively. The obtained results in this hyperparameter analysis will be presented later in 2.4.1.1. Figure 2.3 represents an example of a Stacked LSTM architecture. The model is composed of two LSTM cells, LSTM_1 and LSTM_2, as well as a Dense layer for outputs predictions. Each LSTM layer gate is governed by the equations summarized in Table 2.1. W^i, U^i and b^i, $i \in [1, 2]$ denotes LSTM_1 and LSTM_2 gates weight matrix and bias vectors respectively. W^D and b^D are the corresponding weight matrix and bias for the Dense layer.

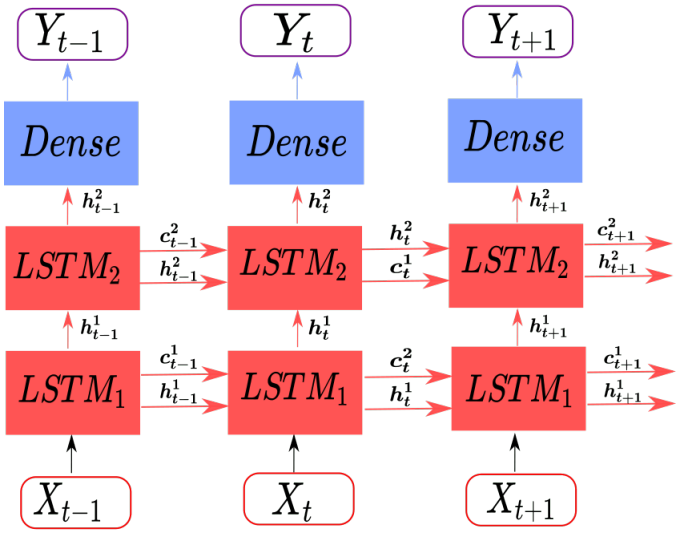


Figure 2.3: Architecture of a Stacked LSTM Neural Network with two LSTM cells unrolled through time

Table 2.1: Governing Equations in the Stacked LSTM Neural Network

Stacked LSTM Layers	**LSTM$_1$ Layer**	**LSTM$_2$ Layer**	**Dense Layer**
Input	X_t	h_t^1	h_t^2
Equations	$f_t^1 = \tilde{\sigma}\left(W_f^1 X_t + U_f^1 h_{t-1}^1 + b_f^1\right)$ $i_t^1 = \tilde{\sigma}\left(W_i^1 X_t + U_i^1 h_{t-1}^1 + b_i^1\right)$ $o_t^1 = \tilde{\sigma}\left(W_o^1 X_t + U_o^1 h_{t-1}^1 + b_o^1\right)$ $\tilde{c}_t^1 = \tanh\left(W_c^1 X_t + U_c^1 h_{t-1}^1 + b_c^1\right)$ $c_t^1 = f_t^1 \odot c_{t-1}^1 + i_t^1 \odot \tilde{c}_t^1$ $h_t^1 = o_t^1 \odot \tanh\left(c_t^1\right)$	$f_t^2 = \tilde{\sigma}\left(W_f^2 h_t^1 + U_f^2 h_{t-1}^2 + b_f^2\right)$ $i_t^2 = \tilde{\sigma}\left(W_i^2 h_t^1 + U_i^2 h_{t-1}^2 + b_i^2\right)$ $o_t^2 = \tilde{\sigma}\left(W_o^2 h_t^1 + U_o^2 h_{t-1}^2 + b_o^2\right)$ $\tilde{c}_t^2 = \tanh\left(W_c^2 h_t^1 + U_c^2 h_{t-1}^2 + b_c^2\right)$ $c_t^2 = f_t^2 \odot c_{t-1}^2 + i_t^2 \odot \tilde{c}_t^2$ $h_t^2 = o_t^2 \odot \tanh\left(c_t^2\right)$	$Y_t = \tanh\left(W^D h_t^2 + b^D\right)$

2.3.2 Constitutive modeling of dissipative materials using RNN

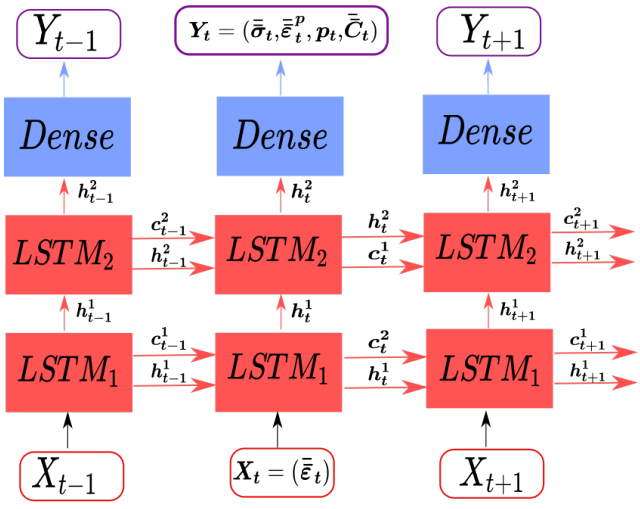


Figure 2.4: Stacked LSTM Neural Network architecture to predict *Elastoplastic with isotropic hardening* constitutive model

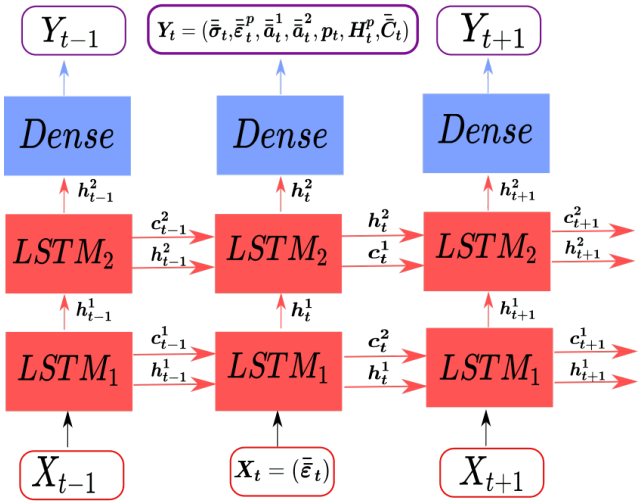


Figure 2.5: Stacked LSTM Neural Network architecture to predict *Chaboche* constitutive model

A Stacked LSTM Neural Network has been developed for dissipative materials path dependant plasticity modeling. The model have been tested on the two studied cases of constitutive laws: *Elastoplasticity with isotropic hardening* and *Chaboche plasticity model with combined isotropic hardening and two nonlinear kinematic hardening*. Figures 2.4 and 2.5 represents the Stacked LSTM architecture corresponding to each material constitutive equation. For both models, the input parameters in each time step t are the incremental strain tensor components $\mathbf{X_t} = (\bar{\bar{\varepsilon}}_t)$. At each time step t, the output $\mathbf{Y_t}$ is given by the

set of variables $\mathbf{Y_t} = (\bar{\bar{\sigma}}_t, \zeta_t, C_t)$ which includes the stress tensor components $\bar{\bar{\sigma}}_t$, the internal variables of each constitutive law ζ_t and the fourth order tangent operator C_t. For the *Elastoplastic with isotropic hardening* constitutive equation, $\boldsymbol{\zeta_t}$ contains the accumulated plastic strain p_t and plastic strain tensor components $\bar{\bar{\varepsilon}}^p_t$, thus $\boldsymbol{\zeta_t} = (\bar{\bar{\varepsilon}}^p_t, p_t)$. For the *Chaboche plasticity model*, in addition to p_t and $\bar{\bar{\varepsilon}}^p_t$, the set of internal variables $\boldsymbol{\zeta_t}$ depends on the second order tensors $(\bar{\bar{a}}^1_t, \bar{\bar{a}}^2_t)$ related to the kinematic hardening and H^p_t the derivative of the hardening function with respect to p, therefore $\boldsymbol{\zeta_t} = (\bar{\bar{a}}^1_t, \bar{\bar{a}}^2_t, \bar{\bar{\varepsilon}}^p_t, H^p_t, p_t)$.

2.3.3 Thermodynamically Consistent RNN training

The training process of RNN is based on Back Propagation Through Time (BPTT) technique which is the application of the back propagation algorithm [Lecun 2001b] to time sequences. Back propagation technique is applied to train the ANN through a chain rule method by computing the partial derivative of a loss function with respect to model's parameters (weights and bias). After each forward pass through the network, back propagation performs a backward pass while adjusting the weights and bias by gradient descent technique. As shown in Figure 2.6, BPTT algorithm for a classical RNN consists on updating model's parameters (W_x, W_h, W_y).

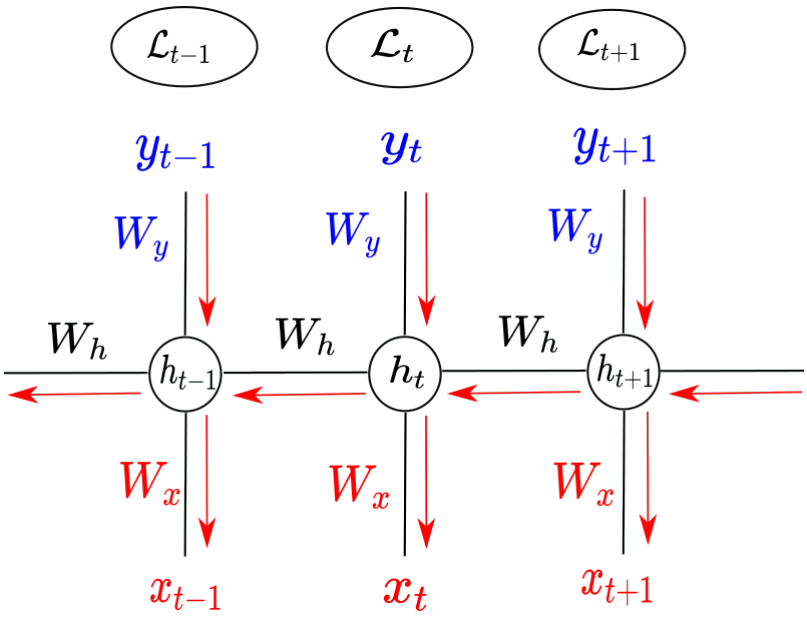


Figure 2.6: Back propagation through time of an RNN architecture

To measure the RNN predictive capabilities, a loss function $\mathcal{L}$ is introduced. For each time step t, a loss $\mathcal{L}_t$ is defined as the error between the RNN predicted values y^p and true output data y^{true}. For the studied problem, the Mean Squared Error (MSE) loss function is used:

$$\mathcal{L}_t = \frac{1}{N} \sum^{N} \left(y_t^{(p)} - y_t^{(true)} \right)^2, \tag{2.47}$$

where N is the total number of training data.

The total Loss function $\mathcal{L}$ of the entire network can be defined as the sum of each time step Loss $\mathcal{L}_t$.

$$\mathcal{L} = \sum \mathcal{L}_t. \tag{2.48}$$

Gradient descent technique consists on computing loss function gradients with respects to RNN trained parameters (W_x, W_h, W_y) and updates it after each Epoch k with a learning rate η by the following expression:

$$W_{ij}^{(k+1)} = W_{ij}^{(k)} - \eta \frac{\partial \mathcal{L}}{\partial W_{ij}}. \tag{2.49}$$

From figure 2.6, the expression of the hidden state h_t and the output y_t for a standard RNN without taking into account the bias b is given by:

$$h_t = \tilde{\sigma}(W_x x_t + W_h h_{t-1}), \tag{2.50}$$

$$y_t = \tanh(W_y h_t). \tag{2.51}$$

The computation of the partial derivatives $(\frac{\partial \mathcal{L}_t}{\partial W_y}, \frac{\partial \mathcal{L}_t}{\partial W_h}, \frac{\partial \mathcal{L}_t}{\partial W_x})$ required for the application of the gradient descent technique:

- **Computation of** $\frac{\partial \mathcal{L}_t}{\partial W_y}$:

$$\frac{\partial \mathcal{L}_t}{\partial W y} = \frac{\partial \mathcal{L}_t}{\partial y_t} \frac{\partial y_t}{\partial W y},$$
$$\frac{\partial \mathcal{L}_t}{\partial W y} = (y_t^{(p)} - y_t^{(true)}) \times (1 - tanh^2(W_y h_t)).$$

- **Computation of** $\frac{\partial \mathcal{L}_t}{\partial W_h}$:

$$\frac{\partial \mathcal{L}_t}{\partial W_h} = \frac{\partial \mathcal{L}_t}{\partial y_t} \frac{\partial y_t}{\partial h_t} \frac{\partial h_t}{\partial W_h}.$$

The computation of $\frac{\partial h_t}{\partial W_h}$ requires a special focus as the hidden state h_t depends on both W_h and h_{t-1}, and the computation of the previous hidden state h_{t-1} also depends on W_h. Thus the parameter W_h is shared across all the previous time steps. Thus, the computation of $\frac{\partial h_t}{\partial W_h}$ using the chain rule yields to the following recursive expression:

$$\frac{\partial h_t}{\partial W_h} = \frac{\partial \tilde{\sigma}(W_x x_t + W_h h_{t-1})}{\partial W_h} + \frac{\partial \tilde{\sigma}(W_x x_t + W_h h_{t-1})}{\partial h_{t-1}} \frac{\partial h_{t-1}}{\partial W_h}.$$

For the sake of simplicity, we set the following variables (a_t, b_t, c_t) where:

$$a_t = \frac{\partial h_t}{\partial W_h},$$
$$b_t = \frac{\partial \tilde{\sigma}(W_x x_t + W_h h_{t-1})}{\partial W_h},$$
$$c_t = \frac{\partial \tilde{\sigma}(W_x x_t + W_h h_{t-1})}{\partial h_{t-1}}.$$

The gradient computation satisfy the following recursive equation:

$$a_t = b_t + c_t a_{t-1}.$$

The simplification of this recursive equation leads to the final expression of a_t:

$$a_t = b_t + \sum_{i=1}^{t-1} \left(\prod_{j=i+1}^{t} c_j \right) b_i.$$

Thus, $\frac{\partial h_t}{\partial W_h}$ is expressed as follows:

$$\frac{\partial h_t}{\partial W_h} = \frac{\partial \tilde{\sigma}(W_x x_t + W_h h_{t-1})}{\partial W_h} + \sum_{i=1}^{t-1} \left(\prod_{j=i+1}^{t} \frac{\partial \tilde{\sigma}(W_x x_j + W_h h_{j-1})}{\partial h_{j-1}} \right) \frac{\partial \tilde{\sigma}(W_x x_i + W_h h_{i-1})}{\partial W_h}.$$

- **Computation of $\frac{\partial \mathcal{L}_t}{\partial W_x}$:**

$$\frac{\partial \mathcal{L}_t}{\partial W x} = \frac{\partial \mathcal{L}_t}{\partial y_t} \frac{\partial y_t}{\partial h_t} \frac{\partial h_t}{\partial W_x}.$$

The same above methodology is adopted to compute the term $\frac{\partial h_t}{\partial W_x}$:

$$\frac{\partial h_t}{\partial W_x} = \frac{\partial \tilde{\sigma}(W_x x_t + W_h h_{t-1})}{\partial W_x} + \sum_{i=1}^{t-1} \left(\prod_{j=i+1}^{t} \frac{\partial \tilde{\sigma}(W_x x_j + W_h h_{j-1})}{\partial h_{j-1}} \right) \frac{\partial \tilde{\sigma}(W_x x_i + W_h h_{i-1})}{\partial W_x}.$$

For the LSTM models or Stacked LSTM, the same reasoning behind BPPT workflow can be applied. The minimization of the loss function by gradient descent technique is based on the computation of the partial derivatives of the loss w.r.t the weights and bias associated to each cell gate.

Regarding the training process of RNN based models using BPPT, we would like to point out that the minimization of only an objective function such as MSE (See Eq 2.47) does not necessarily ensure predictions that are consistent with the knowledge of physical laws. In this study, a new physics-based loss function $\mathcal{L}_t^{\varphi}$ is proposed to ensure model's thermodynamic consistency for each time step t. $\mathcal{L}_t^{\varphi}$ is defined as follows :

$$\mathcal{L}_t^{\varphi} = \frac{1}{N} \sum^{N} (y_t^{(p)} - y_t^{(true)})^2 + \lambda \text{ReLU}(-\gamma_{loc}), \tag{2.52}$$

where γ_{loc} is the intrinsic dissipation at each time step t.
For *Elastoplastic with isotropic hardening* constitutive model, we recall that:

$$\gamma_{loc} = \boldsymbol{\sigma} : \dot{\varepsilon}^{\mathrm{p}} - Hp^n\dot{p}.$$

For *Chaboche plasticity with combined isotropic and two non-linear kinematic hardening* model:

$$\gamma_{loc} = \boldsymbol{\sigma} : \dot{\varepsilon}^{\mathrm{p}} - \boldsymbol{X_1} : \dot{\boldsymbol{a_1}} - \boldsymbol{X_2} : \dot{\boldsymbol{a_2}} - \frac{\partial H}{\partial p}\dot{p}.$$

The function *ReLU* is the Rectified Linear Unit defined as follows :

$$\mathrm{ReLU(x)} = \begin{cases} 0 \text{ for } x < 0 \\ x \text{ for } x \geqslant 0 \end{cases}$$

The total loss function $\mathcal{L}^{\varphi}$ of the model is defined as the sum of $\mathcal{L}_t^{\varphi}$ over all the time steps,

$$\mathcal{L}^{\varphi} = \sum \mathcal{L}_t^{\varphi}. \tag{2.53}$$

It can be seen from equation 2.52, that $\mathcal{L}_t^{\varphi}$ can be split into two terms : The first term $\frac{1}{N}\sum^N (y_t^{(p)} - y_t^{(true)})^2$ is the common MSE that measures the mean quadratic error between the predicted values and true values. The second term is where the thermodynamic consistency is taken into account during ANN training phase. $\lambda\mathrm{ReLU}(-\gamma_{loc})$ is a physical based term that penalize the loss when the dissipative mechanical work rate has negative value, it can be seen as a physical part that constraint the model to respect the second principle of thermodynamics during the training phase for all the time steps. λ is an hyperparameter that controls the degree of regularization in the model. The effect of the physical term during training can be summarized as follows :

- if $\gamma_{loc} > 0$ then $\mathrm{ReLU}(-\gamma_{loc}) = 0$ (No penalty on the Loss function).
- if $\gamma_{loc} < 0$ then $\mathrm{ReLU}(-\gamma_{loc}) = -\gamma_{loc}$ (Add a penalty term to constraint the loss when a physical inconsistency is occuring).

RNN seems very suited for independent time sequences of data, such as non-proportional loadings. As discussed in the introduction, F.Masi et al [Masi *et al.* 2021] have also developed a Thermodynamic-based ANN (TANN) for constitutive modeling. The main differences between the two approaches is related to the choice of the neural network architecture and to the training strategy. F.Masi et al used an MLP architecture and proceeded to the training of TANN by the minimization of the thermodynamic quantities (e.g, free energy potential) which they used its derivatives afterwards to compute the stress tensor components. In our approach, the main quantity of interest (for numerical simulation of structures) is the prediction of the mechanical fields. The thermodynamic consistency is introduced as a regularization term to ensure that the dissipated mechanical power predicted by the RNN is always positive.

2.4 Results and discussion

2.4.1 Comparison between a Standard Stacked LSTM and ThC-RNN model for constitutive modeling for *Elastoplasticity with isotropic hardening*

The following subsection is dedicated to the comparison between ThC-RNN model and a Standard Stacked LSTM model to assess the effect of incorporating a thermodynamic consistency on model's reliability. The two models share the same ANN architecture as described in figure 2.4 but differs in terms of the adopted training process. The Standard Stacked LSTM is trained by minimizing the MSE defined in equation 2.48 and ThC-RNN is trained using $\mathcal{L}^{\varphi}$ loss function expressed in equation 2.52 and 2.53. The models are tested for a case of a material obeying an *Elastoplastic with isotropic hardening* constitutive law. The tested material is a titanium alloy Ti-6Al-4V modeled using a power law hardening. The corresponding mechanical properties are given in Table 2.2.

Table 2.2: Mechanical properties of titanium alloy Ti-6Al-4V

Material parameter	**Value**
Young's modulus E	113800 MPa
Poisson's ratio ν	0.34
Yield Stress σ_Y	1000 MPa
Hardening parameter H	1600 Mpa
Plastic hardening exponent n	0.5

Afterward, the database $\overline{\mathscr{D}}$ preparation strategy is described as follows: 10000 multi-axial; non proportional loading path samples are generated using an open source software in mechanics and materials *Simcoon* [Prulière & Chemisky 2022]. The input parameters are the in-plane strain tensor components $\bar{\bar{\varepsilon}}^t = (\varepsilon^t_{11}, \varepsilon^t_{22}, \varepsilon^t_{12})$ evolving incrementally through 300 time steps. To generate non-proportional loading paths, the loading is divided into 3 blocks of 100 time steps. Each block correspond to a linear proportional loading path where the final value of each strain tensor component is chosen randomly from the interval $[-0,05, 0.05]$. Figure 2.7 shows some examples of loading paths used as training data. The generated samples are then split into training $\overline{\mathscr{T}}$, validation $\overline{\mathscr{V}}$ and test $\overline{\mathscr{T}}$ data sets. Training data (90% of total samples) are used during model's learning process. Then, to verify the training procedure accuracy, a subset (20%) is chosen randomly from training data after each learning iteration and used as a validation data. Finally, test data set (10%) are the unseen samples during the training process, the model's predictive and generalization capabilities are evaluated on this set of data.

In addition, special attention was paid to the distribution of the database to ensure that the plastic strains are much larger than the elastic strain for all the training samples, this is necessary to investigate the model approximation quality of material non-linear behavior. As shown in the histograms of figure 2.8, the computed stresses are induced on average by plastic strains during 200 time steps and by elastic strains during only 100 time steps for a loading of 300 time steps.

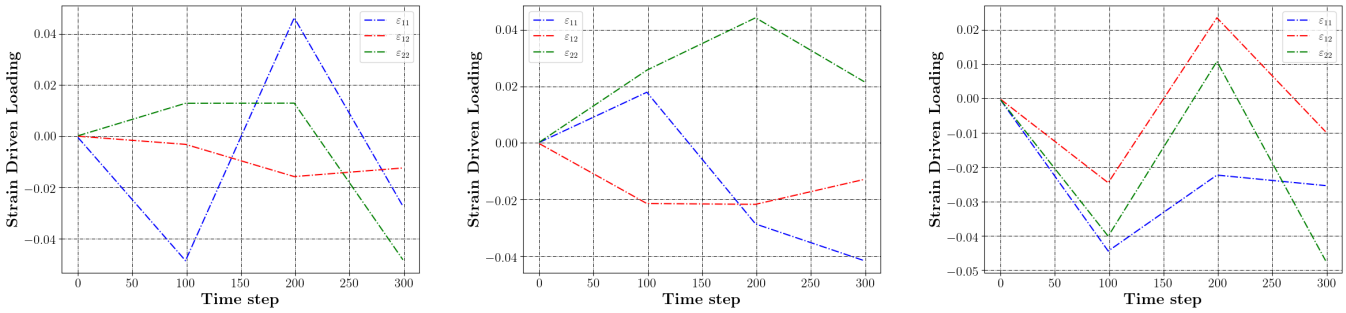


Figure 2.7: Examples of multi-axial and non proportional loadings paths samples used as training data

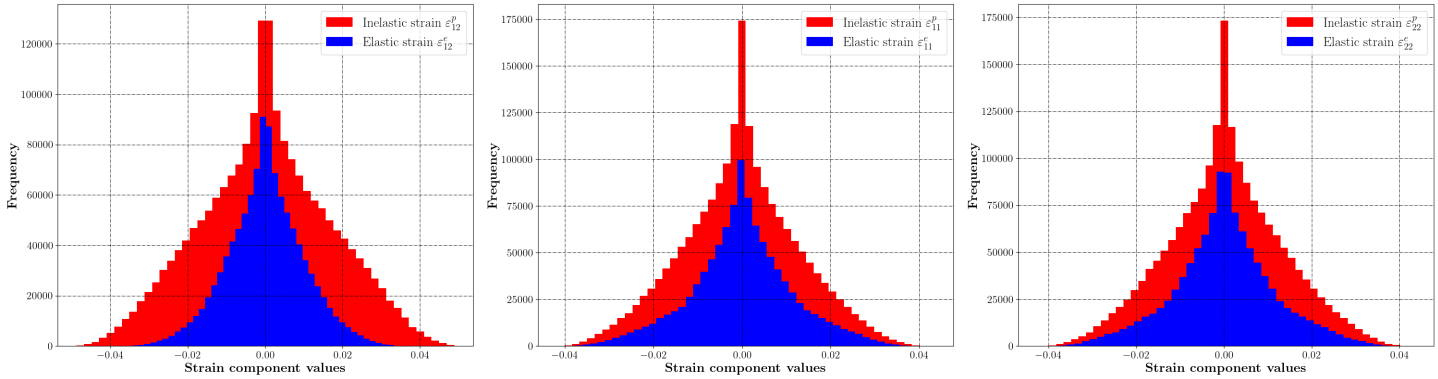


Figure 2.8: Histograms of elastic and inelastic strain components repartition on the database

2.4.1.1 Prediction of material mechanical response using a Standard Stacked LSTM

Once the database generation has been achieved, we proceeded to the training process of the model with only Mean Squared Error as a loss function. The model implementation is performed in Python with the help of Keras library [Chollet *et al.* 2015] and Google's Tensorflow API [Abadi *et al.* 2015]. Six configurations of LSTM architectures have been tested by varying two hyperparameters : first the activation function of the output dense layer and then the number of LSTM layers in the network, the main objective is to fix these two hyperparameters to obtain the most reliable model. Figure 2.9 shows the evolution of the metric Normalized Mean Squared Error (NMSE) for both training and validation data sets for all the six tested architectures. According to figure 2.9 and table 2.3, it has been found that an architecture with two LSTM layers (Stacked LSTM) and an hyperbolic tangent activation function in the output dense layer (Config 6) is the most reliable one as the training and validation errors are the lowest compared to the others configurations. Furthermore, it can be seen that Config 6 learning process can be achieved rapidly, as an exponential decrease of the errors is observed during the first epochs. This decrease was followed by a loss value stabilization after nearly 100 epochs. Finally, the obtained NMSE on the last epoch is under 1% for the training error and around $2,2\%$ for the validation error. This disparity in error values shows that the Standard Stacked LSTM model is potentially suffering from the overfitting problem, but without having a very large effect on model's performance since the NMSE values are relatively low.

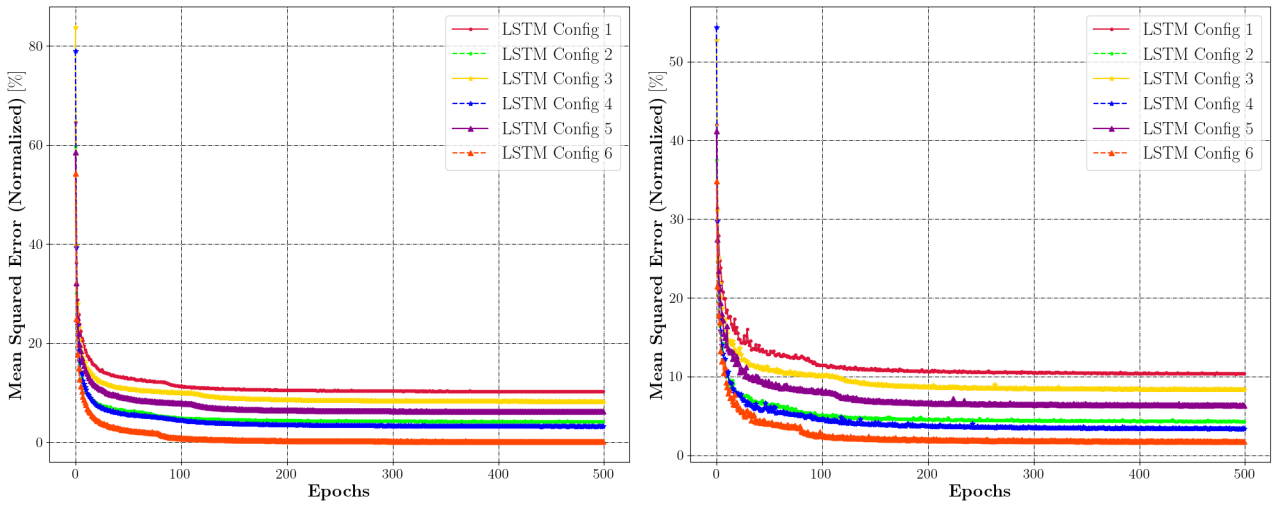


Figure 2.9: Evolution of training (left figure) and validation errors (right figure) as a function of Epochs during Stacked LSTM training for the tested LSTM configurations

Table 2.3: Evaluation of LSTM configurations training and validation performances

LSTM Configuration	**Number of hidden LSTM layers**	**Output layer activation function**	**Training loss NMSE (%)**	**Validation loss NMSE (%)**
Config 1	1	ReLU	10.20 %	10.38 %
Config 2	2	ReLU	4.13 %	4.29 %
Config 3	1	Sigmoid	8.19 %	8.36 %
Config 4	2	Sigmoid	3.21 %	3.35 %
Config 5	1	Hyperbolic tangent	6.17 %	6.43 %
Config 6	2	Hyperbolic tangent	0.41 %	2.2 %

To assess the generalization and predictive capabilities of the Standard Stacked LSTM after the training phase, the model is evaluated on new multi-axial and non proportional loading paths (test data set) to predict the resulting mechanical response. Figure 2.10 shows a comparison between model's predicted stress-strain responses and the target values obtained via *Simcoon* simulations. At first sight, it can be observed that the general trend of the stress components is well captured by the Stacked LSTM despite the complexity of the loading conditions. Note that this performance could not be achieved with basic LSTM model with a single cell, which justify the need of stacking LSTM cells for reliable predictions. The quantification of model's global predictive quality can be demonstrated with the use of the correlation plots shown in figure 2.10 where the coefficient of determination R^2 is very close to 1 for all the predicted responses. Furthermore, the model is also able to predict the change in stress-strain response from the elastic to the plastic domain. This first results proves the potential of RNN as a surrogate model capable of modeling history dependant mechanical behaviors, giving the possibility to potentially substitute time

integration schemes of systems of Partial Derivative Equations (PDE) with an AI based model.

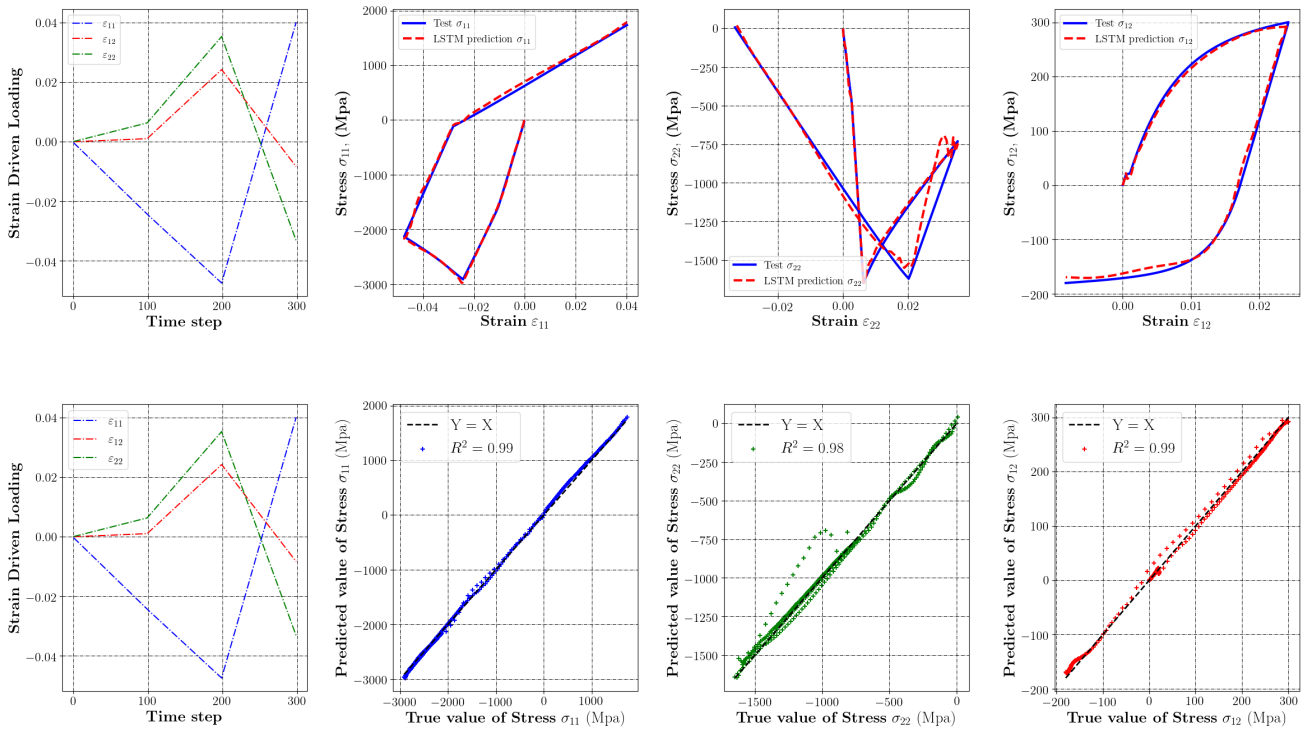


Figure 2.10: Prediction of Stress-Strain responses with a Standard Stacked LSTM model on test data set & Correlation plots between model's predicted values and target values

However, even if the coefficient of determination R^2 is close to 1 regarding the mechanical responses, we recall that this coefficient only provide a global estimation on model's reliability as it includes all the contributions from all the predicted values with respect to target values (See Equation 2.54).

$$R^2 = 1 - \frac{\sum_{k=1}^{N}(y_{true}^{(k)} - y_p^{(k)})^2}{\sum_{k=1}^{N}(y_{true}^{(k)} - \bar{y}_{true})^2}, \tag{2.54}$$

- $\boldsymbol{y_{true}^{(k)}}$: True values of test data set
- $\boldsymbol{\bar{y}_{true}}$: Mean value of y_{true}
- $\boldsymbol{y_p^{(k)}}$: Predicted values by the ANN
- N : Number of time steps

Thus, it might seem more interesting to investigate model's accuracy when it comes to local behaviors at specific increments. As shown in figure 2.10, a presence of small noise during the transition from the elastic domain to the plastic domain can be observed from most configurations on test data set, these oscillations after direction change can have a negative impact regarding the convergence of numerical simulations. Furthermore, we noticed that the prediction of the shear stress components σ_{12} were not

accurate compared to the normal stress components in certain time steps, but still follows the general trend of the response.

The verification of model's thermodynamic consistency is highlighted by predicting the various mechanical work rates regarding the *Elastoplastic with isotropic hardening* constitutive law. Figure 2.11 presents a comparison between the predicted recoverable, irrecoverable, dissipative mechanical work rates by the Standard Stacked LSTM model and the target values. As we recall that a proper formulation of dissipative materials constitutive laws must follow the thermodynamic principles, it can be observed from figure 2.11 that the predicted dissipative mechanical work rate is negative during some time steps even if the overall prediction still follows the general trend of the target dissipation. Based on the thermodynamic framework of dissipative materials developed in Section 2.2, this result contradicts the second principle of thermodynamics that state that the local generated entropy should always be greater than zero while a dissipative mechanism occur. Thus, the respect of the second principle could not be achieved using the Standard Stacked LSTM model with only a MSE as loss function. Using a physics-guided RNN model 'ThC-RNN', the following subsection aims at investigating the effect of incorporating physical based constraints during Stacked LSTM training process to potentially satisfy the governing physical laws and to achieve a thermodynamic consistency.

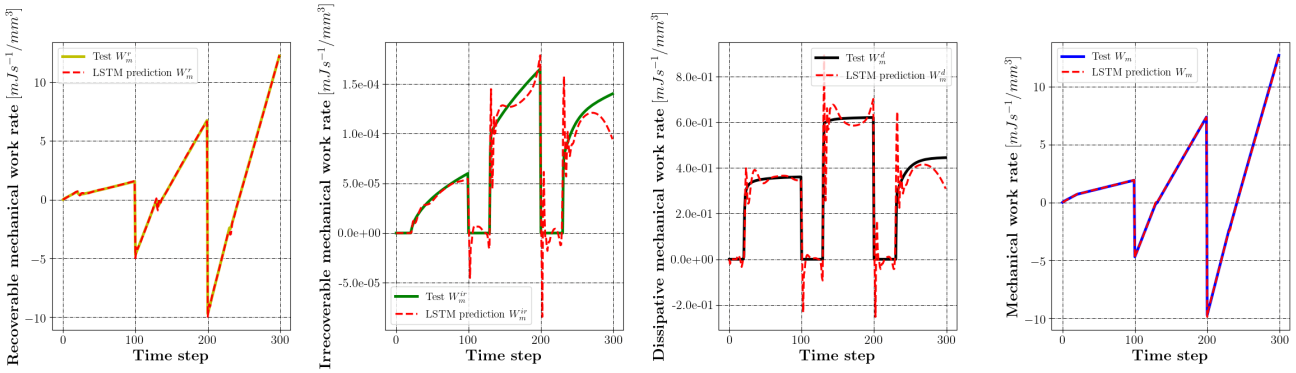


Figure 2.11: Evaluation of various mechanical work rates with a Standard Stacked LSTM model on an example of test data set

2.4.1.2 Prediction of the mechanical response using a ThC-RNN - Effect of the Thermodynamic consistency on model's accuracy

The developed ThC-RNN architecture is based on a Stacked LSTM model with a physics-based loss function whose expression in each time step t is given by Equation 2.52. ThC-RNN is trained with the same amount of data as the previous model presented in subsection 2.4.1.1. The objective being to verify ThC-RNN capabilities to accurately predict the mechanical behavior regarding the *Elastoplastic with isotropic hardening* constitutive law as well as the respect of the thermodynamic principles when computing the mechanical work rate partition (recoverable part, irrecoverable part and dissipative part).

Figure 2.12 present ThC-RNN predicted stress and the target response under a test data loading

path. Compared to the previous model, it can be deduced that ThC-RNN ensure a better performance while predicting the stress-strain responses. Using ThC-RNN model, the transition from the elastic to the plastic domain is smooth for the majority of the test data set contrary to the previous model where small noise was observed. The incorporation of thermodynamic consistency also led to stable responses after change of directions. Figure 2.13 describe the evolution of the predicted internal variable p which is the accumulated plastic strain. As shown in the correlation plot from figure 2.13, an excellent agreement is observed between ThC-RNN predicted values and the targets for p, ThC-RNN model is therefore capable of capturing accurately the evolution of this state variable. Therefore, we can assume that incorporation of the physical laws during the training phase has a beneficial effect on model accuracy. Our assumption to explain this result is that ThC-RNN is forced to regulate the predicted quantities in order to verify the second principle of thermodynamics which is introduced as a physical constraint on the loss function.

To properly highlight the effect of the physical knowledge on model's reliability, it might be more interesting to evaluate the various mechanical work rates as done previously. Figure 2.14 shows a comparison between ThC-RNN predicted mechanical work rates and the target values. Fig. 2.15 shows a comparison between the predicted mechanical work rates with a standard stacked LSTM model and with ThC-RNN on the same example of test data. Regarding the dissipative mechanical work rate, it can be noticed that the model is at least thermodynamically consistent as the intrinsic dissipation is always greater or equal to zero with the respect to all the time steps. We recall that using the first model, the

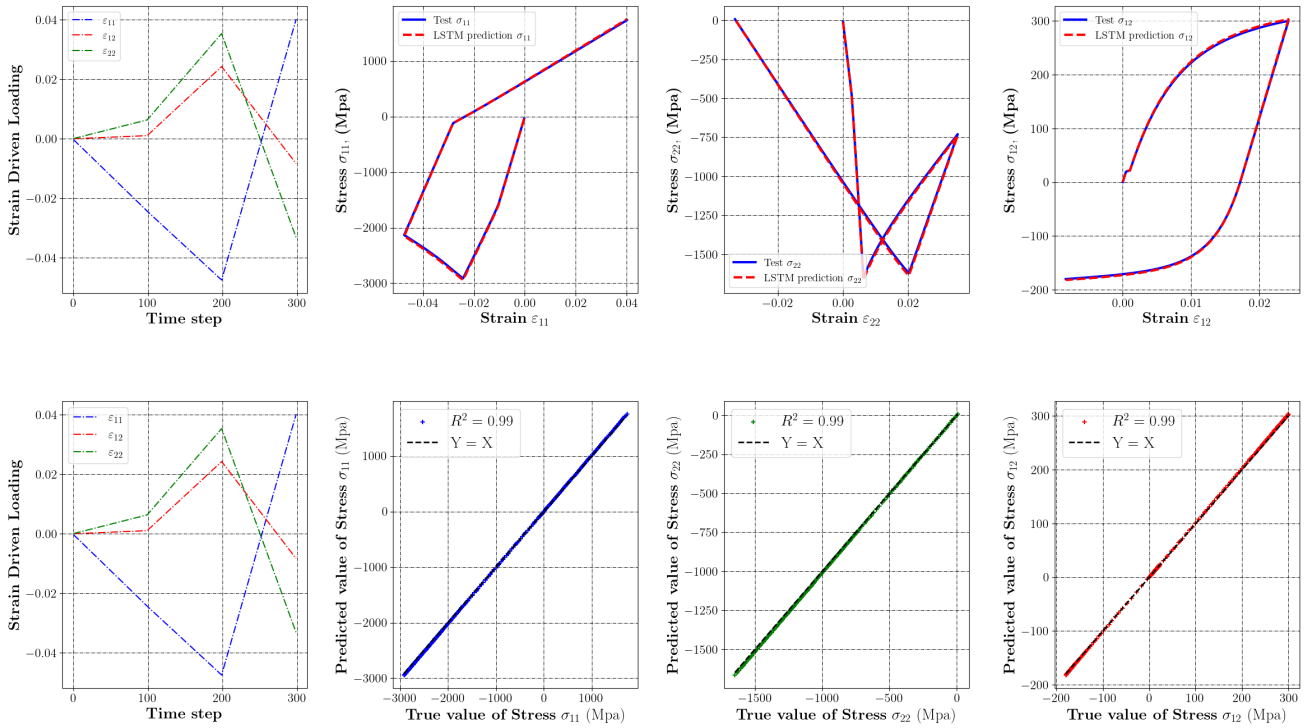


Figure 2.12: Prediction of Stress-Strain responses with ThC-RNN model on an example of test data set

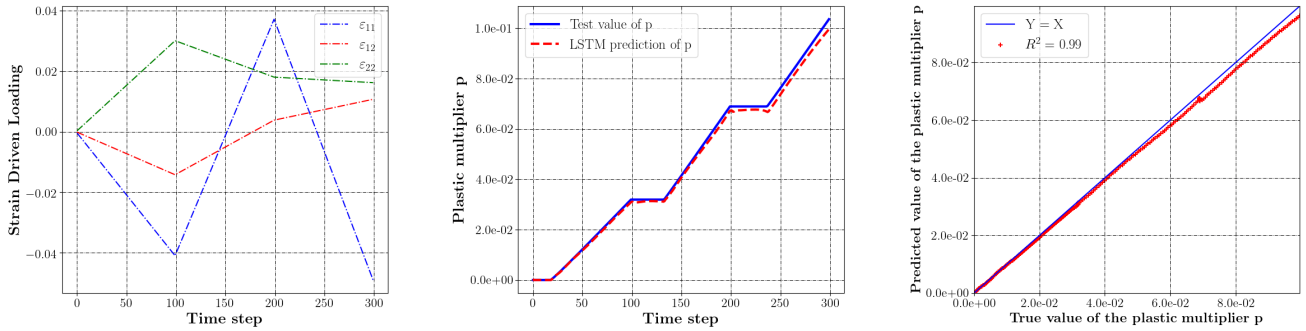


Figure 2.13: Prediction of accumulated plastic strain p using ThC-RNN on an example of test data set

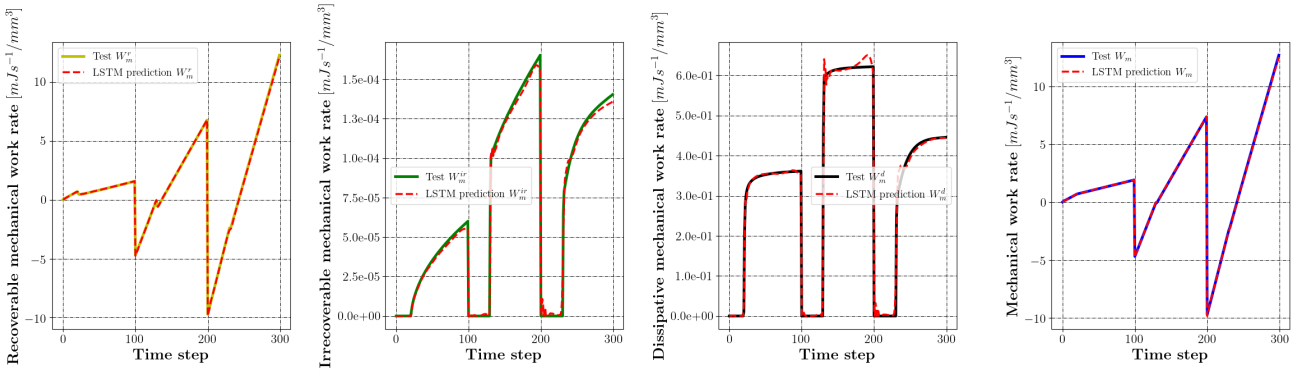


Figure 2.14: Evaluation of various mechanical work rates with ThC-RNN model on an example of test data set

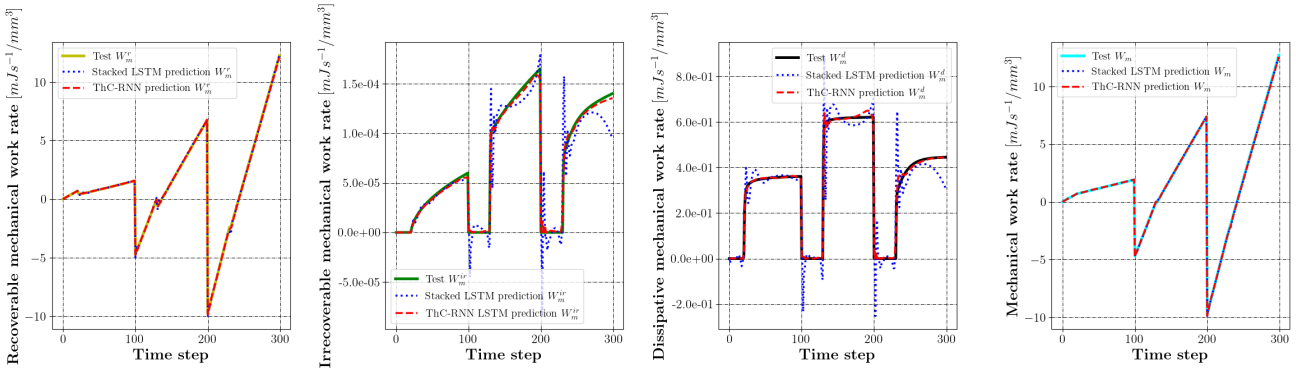


Figure 2.15: Comparison between the prediction of various mechanical work rates with Standard Stacked LSTM model and ThC-RNN on the same example of test data

second principle constraint was not always guaranteed due to the lack of any governing physical law on model's implementation. The obtained results with ThC-RNN provides evidence that the incorporation

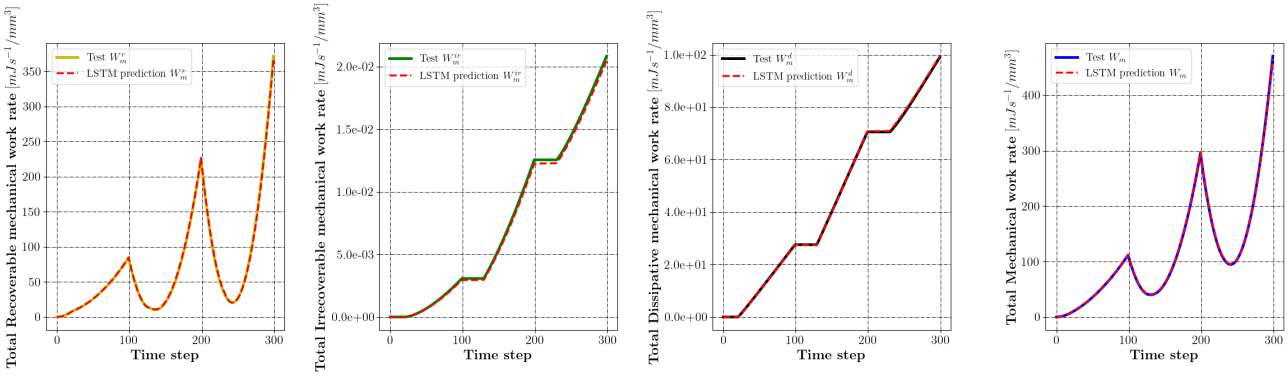


Figure 2.16: Evaluation of the total mechanical work rates with ThC-RNN model on an example of test data set

of physics based loss functions along with sufficient data can reduce significantly the impact of physical inconsistencies on model's accuracy. However, as we analyzed the prediction quality for all the tested data, we observed a presence of some discrepancies on the predicted irrecoverable and dissipative mechanical work rate during some time steps, this mismatch can also be highlighted in figure 2.16 where the error between the predicted total dissipation and the target value is around 10%. The main hypothesis to explain this difference is the propagation of errors as the computation of W_m^r and W_m^d is based on several quantities including the internal variables, plastic strain and stress tensor components. The prediction errors of those outputs, even of low values, may induce an amplification of the error when computing W_m^r and W_m^d. Therefore, the training process of ThC-RNN model using $\mathcal{L}_t^{\varphi}$ loss function certainly ensure the respect of the dissipation positivity condition but does not necessarily guarantee that the predicted dissipative mechanical work rate will match the target one. A potential solution to overcome this problem will be to add a term on the objective function $\mathcal{L}_t^{\varphi}$ that minimize the error between the predicted and the target mechanical work rates similarly to the Mean Absolute Error (MAE) or the Mean Squared Error (MSE) loss functions.

The generalization capabilities of ThC-RNN are verified by evaluating the model on a subset of the database (10%) where a new type of loading condition is tested: Non-proportional paths with non-linear steps (i.e, sinusoidal shape). We recall that the training database was only composed of non-proportional loadings with linear steps, the samples with sinusoidal shapes steps were never seen by the model during the training phase. Figure 2.17 shows some examples of the predicted stress-strain responses by ThC-RNN with this new type of loading condition. Based on the obtained results in figure 2.17, it can be observed that the model achieved a good generalization ability, the overall mechanical response are well captured by the model despite the complexity of the loading paths. However, the predicted response under sinusoidal based steps was less accurate than the results of a loading with linear steps, the obtained prediction error for the first case was 3.6% contrary to the second case where the error does not exceed 0.1% .This result is completely expected as neural networks are known to have less precise extrapolation capabilities.

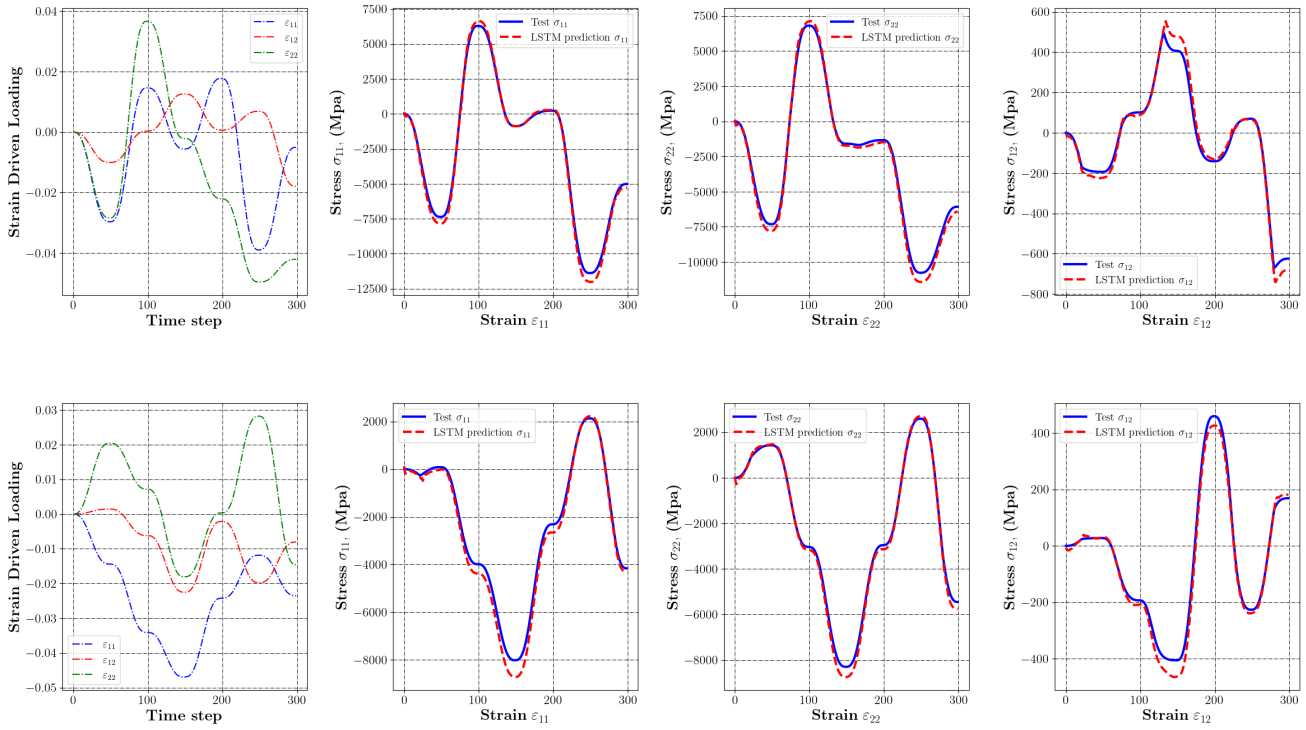


Figure 2.17: Prediction of Stress-Strain responses with ThC-RNN model on two examples of test data set : Case of non-proportional loadings with sinusoidal shape steps

2.4.2 Prediction of the mechanical behavior for *Chaboche plasticity* model using ThC-RNN

Through an example of an *Elastoplastic with isotropic hardening* material constitutive law, it has been proven in subsection 2.4.1.2 that using a physics guided Stacked LSTM model results in a significant improvement on model's performance as well as its physical consistency. Given those primary results, it might be interesting to further explore ThC-RNN capabilities when dealing with more complex mechanical behaviors involving many physical mechanisms. In that sense, *Chaboche plasticity model* can be a suitable case study as it combines two non-linear kinematic hardening mechanisms as well as an isotropic hardening. Furthermore, several internal variables are required to describe the material mechanical behavior including the accumulated plastic strain p and the back stress tensors X_1 and X_2. The main objective of this subsection is to verify ThC-RNN predictive capabilities of the mechanical responses and the internal state variables. Furthermore, a special focus will be given to the respect of the thermodynamic consistency.

For this purpose, a database containing 10000 multi-axial, non-proportional loading paths is generated with the *Simcoon* library. The same data preparation strategy is followed as the previous case example (isotropic plasticity). The mechanical properties of the tested material are summarized in Table 2.4. The identification of those parameters from experimental testing is carried out using a specific optimization

Table 2.4: Mechanical properties of the tested material

Material parameter	**Value**
Young's modulus E	140000 MPa
Poisson's ratio ν	0.3
Yield Stress σ_Y	57.86 MPa
Parameter Q	280.48 Mpa
Parameter b	7.60
Parameter C_1	31194.68 Mpa
Parameter D_1	176.12
Parameter C_2	435376.84 Mpa
Parameter D_2	5367.40

module on *Simcoon* library. ThC-RNN neural network architecture is illustrated in Figure 2.5 and the model's training process with BPTT consist on the minimization the loss function $\mathcal{L}^{\varphi}$ given in Equation 2.52 and 2.53.

After the completion of the training phase, the model is evaluated on the test data set to verify its reliability on unseen loading configurations. For the mechanical responses, it appears that ThC-RNN performs accurately when predicting the material stress state with respect to the strain driven loading as shown in figure 2.18. This result provide a first insight of ThC-RNN capabilities when dealing with complex non linear constitutive laws.

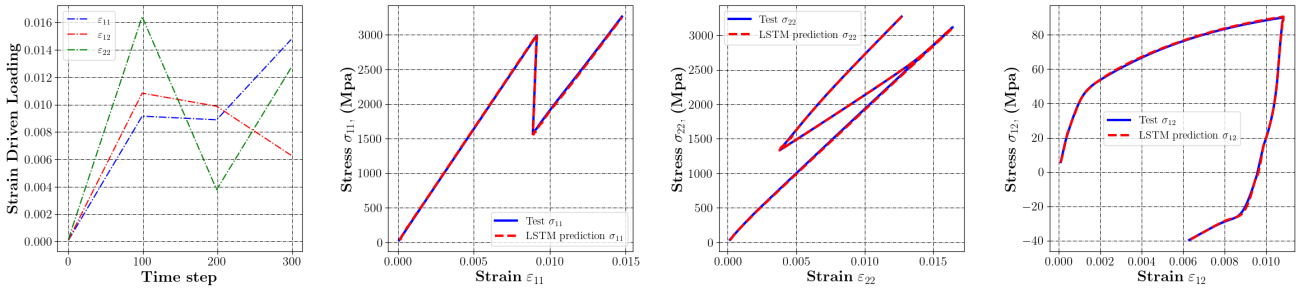


Figure 2.18: Prediction of Stress-Strain responses with ThC-RNN model on an example of test data set for *Chaboche Plasticity model*

As for the thermodynamic consistency, the same procedure is adopted as previously by comparing the predicted mechanical work rates with the the target ones. It can be seen from figures 2.19 and 2.20, that the predicted local and total mechanical work rates are very well captured by ThC-RNN and most importantly, the positivity of the intrinsic dissipation is guaranteed during all the time steps. Furthermore, as pointed in the previous case study, it can be observed that the predicted values of W_m^d can present a slight difference compared to the targets for specific time steps, however the general trend is still captured accurately. The obtained results gives a supplementary evidence that the combination of

physical knowledge with the high potential of RNN as a powerful surrogate for treating history dependant behaviors can ensure models with high accuracy even while dealing with complex mechanical behaviors.

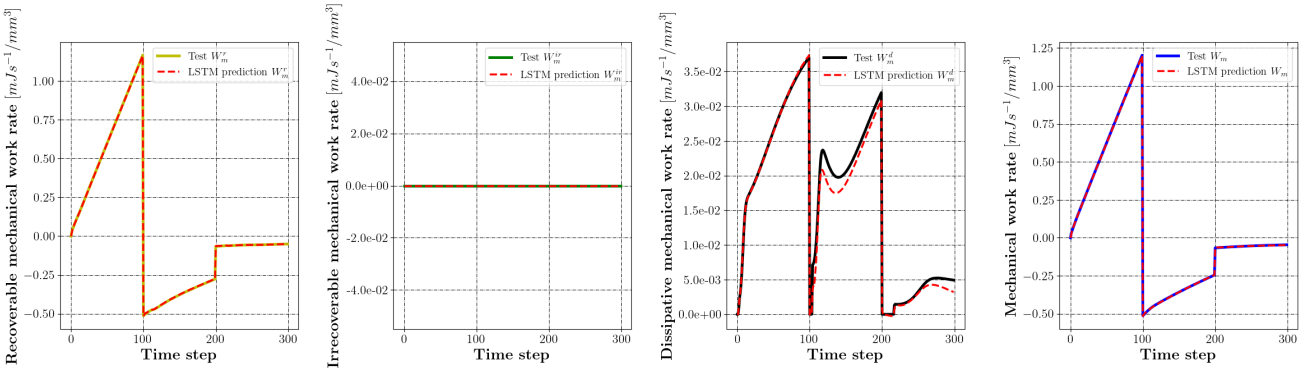


Figure 2.19: Evaluation of various mechanical work rates with ThC-RNN model on an example test data set for *Chaboche Plasticity model*

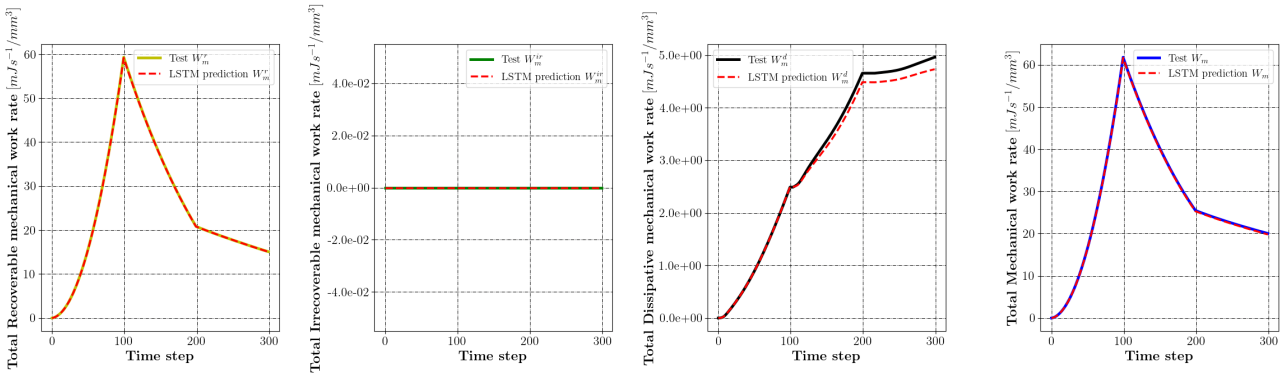


Figure 2.20: Evaluation of Total mechanical work rates with ThC-RNN model on an example of test data set for *Chaboche Plasticity model*

Finally, ThC-RNN predictive capabilities of the internal state variables such as the accumulated plastic strain p and the kinematic hardening variables X_1 and X_2 are highlighted in Figures 2.21, 2.22 and 2.23. An excellent agreement can be observed between the predictions and the targets regarding all the state variables. The evolution of the back stress tensors components related to the kinematic hardening are well captured by ThC-RNN in addition to the accumulated plastic strain p. A reliable prediction of the internal variables is important given that the computation of the mechanical work rates is mainly based on those quantities. Moreover, it can be deduced that the presence of many internal variables that describes material behavior did not have an impact on ThC-RNN reliability for the *Chaboche plastic law* which is a very interesting result that emphasize the high potential of physics-guided RNN models.

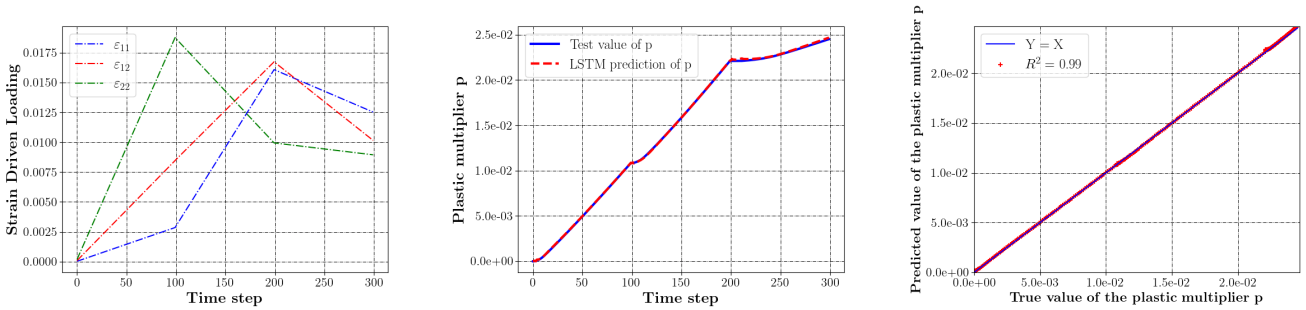


Figure 2.21: Prediction of accumulated plastic strain p using ThC-RNN on an example of test data set for *Chaboche Plasticity model*

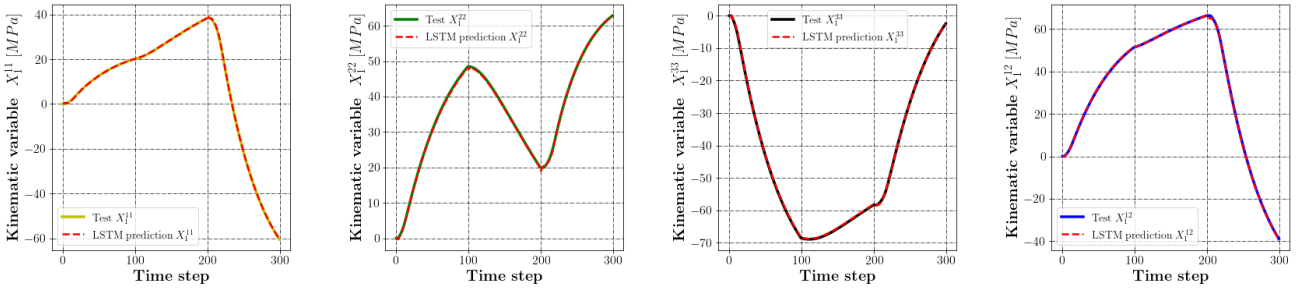


Figure 2.22: Prediction of the kinematic hardening variable X_1 components using ThC-RNN on an example of test data set for *Chaboche Plasticity model*

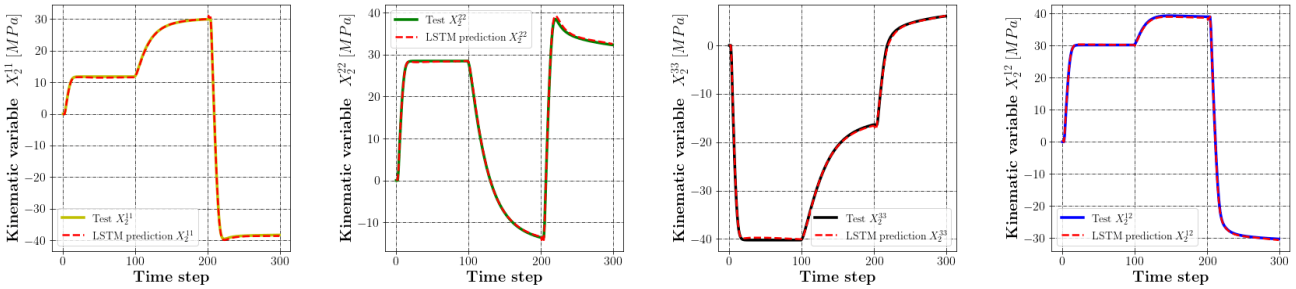


Figure 2.23: Prediction of the kinematic hardening variable X_2 components using ThC-RNN on an example of test data set for *Chaboche Plasticity model*

2.4.3 Prediction of the continuum tangent modulus components

The last subsection is dedicated to investigate ThC-RNN effectiveness to predict the tangent modulus components C_{ij}^{tan}. This analysis may prove its usefulness when this kind of models is employed as a surrogate for accelerating multiscale finite element simulations where the computation of the tangent matrix is required to obtain the homogenized responses. For this purpose, ThC-RNN model also included

the tangent matrix as an output as shown in its architecture from Figure 2.4. Note that even though we used ThC-RNN model as a surrogate for predicting C^{tan} components, the physical consistency has absolutely no effect on those predictions as no governing physical law was incorporated regarding this outputs, the training process in that case is mainly based on the data and the Stacked LSTM model capability of capturing the history dependency. Figure 2.24 illustrates an example of tangent modulus components predictions with ThC-RNN for the *Elastoplastic with isotropic hardening* constitutive law. Similarly to all the previous outputs, the model still keeps its reliability when predicting all the tangent matrix components. The transitions from the elastic to the plastic domains are very smooth and very well captured by the model without involving any significant noise that might violate the positivity condition of the matrix. This result prove that RNN based models can be potentially considered as an efficient and reliable tools for the computation of the tangent matrix, offering new possibilities of employment especially in the multiscale analysis.

Furthermore, predicting C^{tan} with RNN offers a significant advantage in terms of the computational time saving compared to the existing methods. For instance, using a non linear homogenization method such as FE^2 [Feyel 1999], the computation of the effective tangent matrix is required for coupling the micro and macro scales to predict structures macroscopic response. The determination of the macroscopic tangent matrix for periodic homogenization can be achieved for example using the FEM-based perturbation technique [Zhu *et al.* 2020]. We recall that for non linear homogenization problems with FE^2 method, C^{tan} is computed on each time increment and for each Gauss integration point, this process may potentially be costly in terms of the required computational time. Thus, the ability to predict accurately and quickly C^{tan} components will assess the possibility to train RNN based networks on unit cells FE simulations to perform efficient multi-scale structure analysis. An alternative solution to compute C^{tan} could also be the use of Automatic Differentiation Method offered by the computational graph of the neural network. Considering classical Neural Network (Multi-Layer Perceptron), the Automatic Differentiation Method could indeed provide the direct connection between the input and output through backpropagation. However, in the case of Recurrent Neural network, the computation of the jacobian matrix is more difficult to isolate due to the strong dependency of the memory carried out through the neural network. The total derivative of stress with respect to strain (including the internal variation) is much more difficult to assess since there is no criterion that one can rely to compute the relationship between internal variable variation and strain variation, like in classical elastic-plastic models (where the consistency condition is utilized).

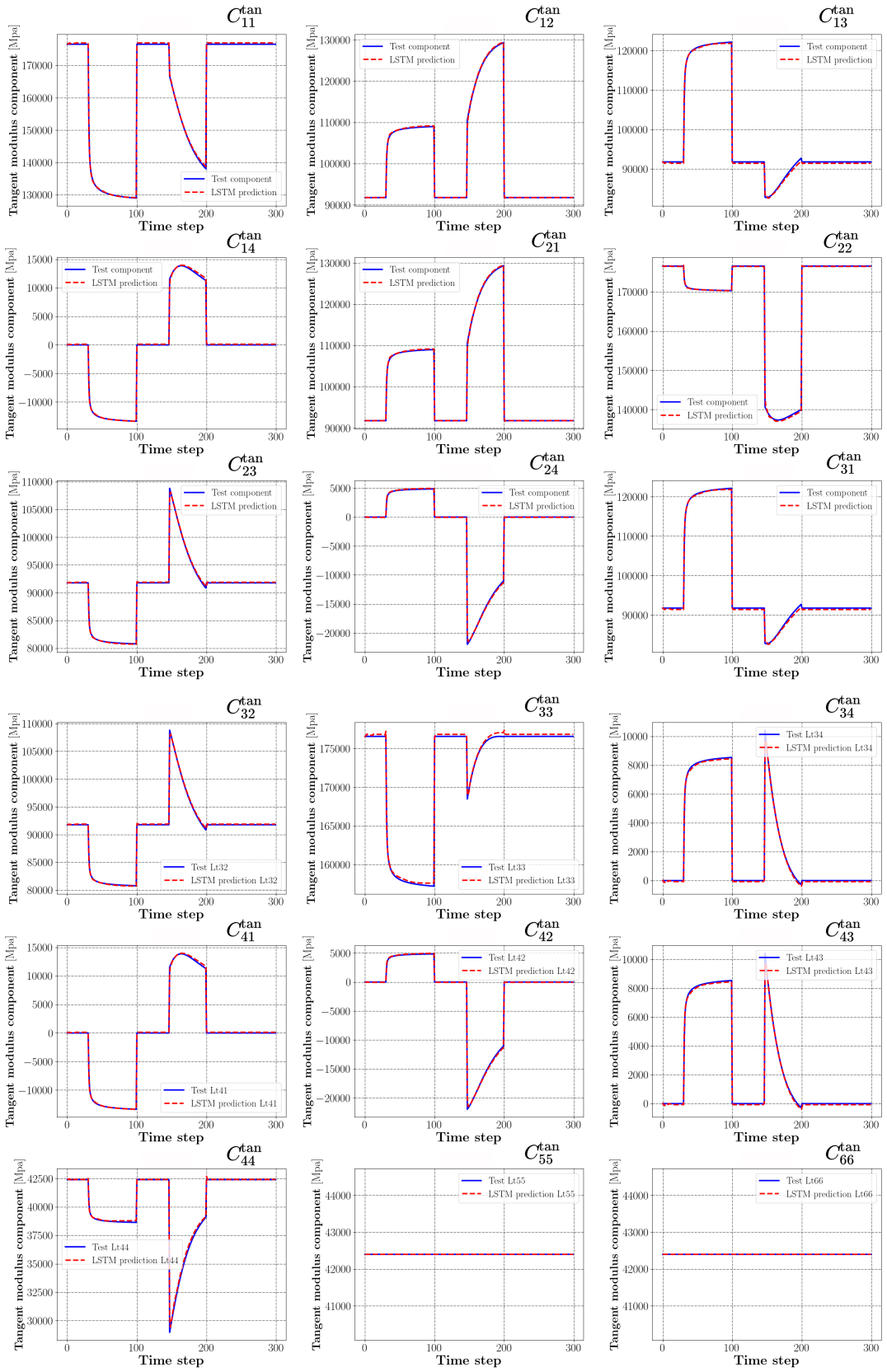


Figure 2.24: Prediction of the tangent modulus components on an example of test data set for the *Elastoplastic with isotropic hardening* constitutive law on an example of test data

2.4.4 Application to Finite Element Analysis

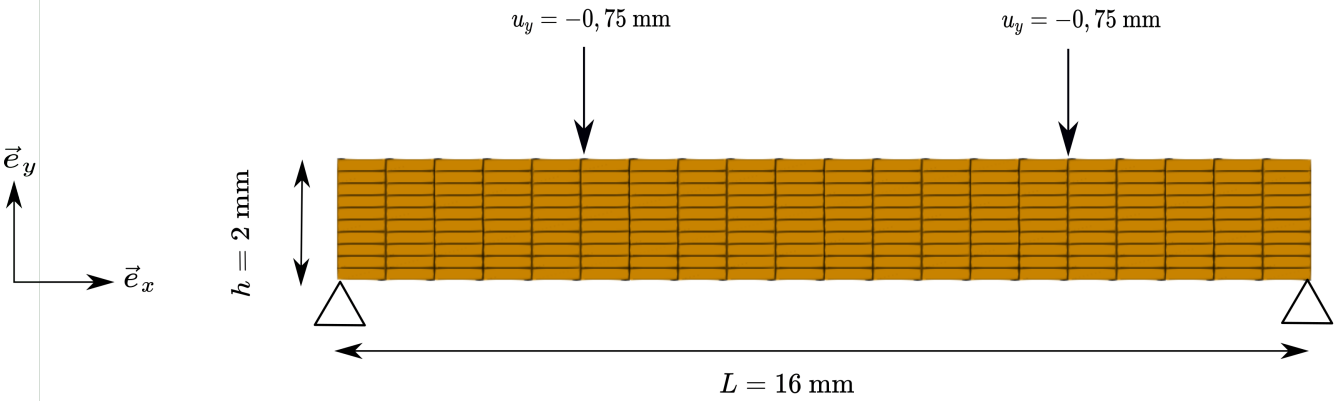


Figure 2.25: Geometry, boundary conditions and mesh of an elastoplastic 4 nodes bending beam

To assess the usefulness and the validity of the developed approach, ThC-RNN model is coupled with a Finite Element Method (FEM) framework to simulate structures considering standard loading tests. A case study of an elastoplastic 4 point bending test is considered, the mechanical properties of the tested structure are given in Table 2.2. Figure 2.25 illustrates the geometry, boundary conditions and the meshing of the tested structure. The beam is subjected to a displacement-controlled loading on its 2 top nodes at $\frac{L}{4}$ and $\frac{3L}{4}$. The incorporation of ThC-RNN model as surrogate of the material constitutive law is done in a similar way to the User Material subroutines in Finite Element codes. For each time increment t, each material Gauss integration point is subjected to an incremental strain $(\varepsilon_{ij}^t, \Delta\varepsilon_{ij})$, the material mechanical state is then stored in the memory gate c_t of the trained RNN (similarly to the internal state variables), the input gate i_t, forget gate f_t and output gate o_t are updated to compute the next hidden state h_t which contains the predicted stress components σ_{ij}^t and the tangent matrix C_{ij}^{tan}. Using the predicted outputs, the structure global equilibrium is verified using Newton-Raphson iterative scheme until the convergence of the simulation. In this study, ThC-RNN model was integrated with an open source FE code *Fedoo* [Prulière & Chemisky 2022].

In order to verify the model's reliability and efficiency when coupled with a FEM approach, the same simulation is conducted using a FE approach which uses the original material constitutive law.

The results of the FE simulation are then compared to the predicted mechanical behavior with ThC-RNN. Figures 2.26,2.27,2.28,2.29,2.30 and 2.31 shows a comparison of the Von-Mises stress σ_{VM}, stress magnitude σ_M, normal stress components σ_{XX}, σ_{YY}, σ_{ZZ} and shear stress component σ_{xy} distributions respectively obtained using pure FE simulation and by the coupling of ThC-RNN with FE framework. Several aspects have been verified and served as a basis for comparison between the two models, the addressed points are summarized as follows:

i) In terms of convergence, the simulation with ThC-RNN model achieved a numerical convergence with the same Newton Raphson tolerance as the finite element model. This result emphasizes the model's accuracy especially when predicting the tangent operators, because an error in the tangent matrix would have led to a lack of convergence of the Newton Raphson resolution.

ii) For both models, the symmetry property of the stress fields distribution is conserved since the studied problem is symmetric in terms of geometry, boundary conditions and loading conditions.

iii) As for the precision of the predicted numerical values, an excellent agreement is observed between ThC-RNN model and FE simulation for the following stress components σ_{VM},σ_M,σ_{XX},σ_{ZZ} and σ_{XY}, the obtained maximum error between the two model does not exceed 3 % for all these components. However, it can be seen from figure 2.29 that σ_{YY} is slightly less accurate compared to FE solution, an overestimation by ThC-RNN of σ_{YY} can be noticed, resulting to residuals error of nearly 6% in some Gauss points which is higher than the obtained residuals for the other stress components.

Regarding the strain fields, figures 2.32,2.33,2.34,2.35 shows the distribution of the strain magnitude ε_M, normal strain components ε_{XX},ε_{YY} and shear strain component ε_{XY} respectively obtained by FE simulation and by ThC-RNN model. The same remarks concerning the respect of the strain field symmetries and model's precision can be highlighted for all the predicted components. Overall, a very good agreement between both simulations is observed given the low values of residuals. Furthermore, it is shown from 2.33,2.34,2.35 that all the strain components remains in the range of the training data (i.e included in the interval $[-0,05, 0.05]$), if the inputs are outside of that range, this could lead to inaccurate predictions or a failure in the convergence simulation. This particular point is very important to assess as Neural Networks are known to have poor extrapolation capabilities.

To conclude this subsection, it has been demonstrated that this type of RNN based approaches can be integrated into a finite element framework provided that certain points are checked:

i) verify the generalization capability of the model and its accuracy when predicting the tangent operators and the mechanical responses.

ii) Ensure that the model inputs remain in range of training data.

iii) Incorporate physical based laws in the model training process to regularize the predicted outputs and to achieve a thermodynamic consistency.

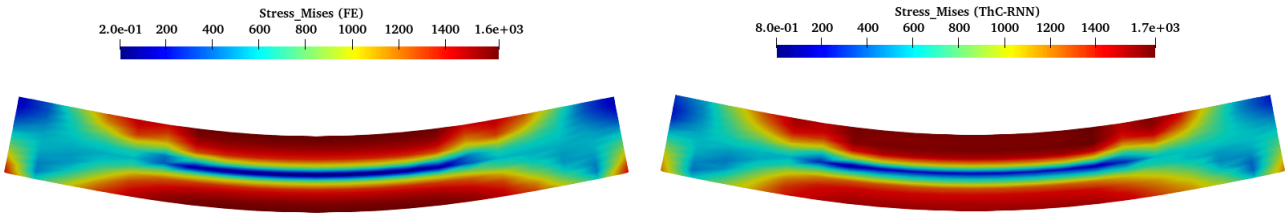


Figure 2.26: Comparison of the Von-Mises stress distribution σ_{VM} obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

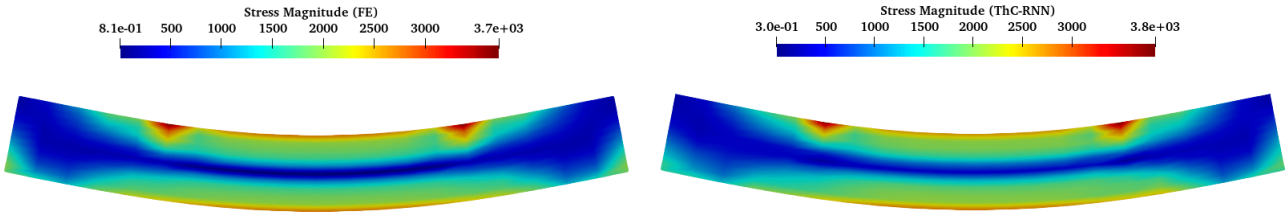


Figure 2.27: Comparison of the stress magnitude distribution σ_M obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

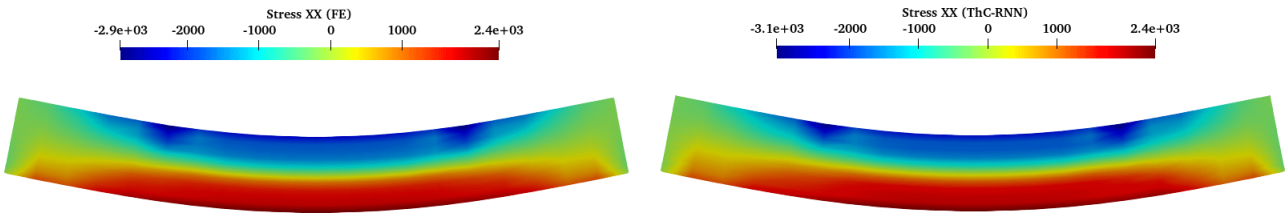


Figure 2.28: Comparison of the normal stress distribution σ_{XX} obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

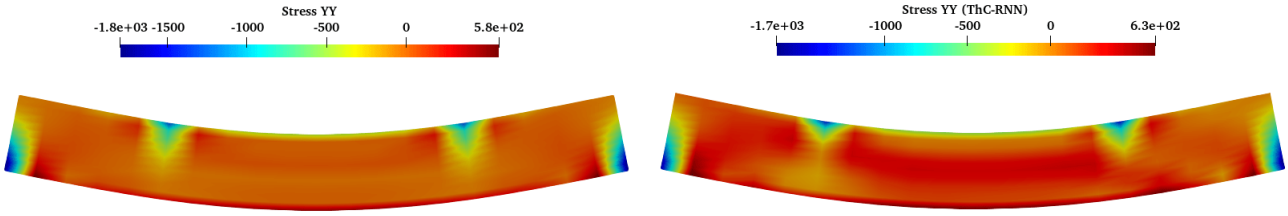


Figure 2.29: Comparison of the normal stress distribution σ_{YY} obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

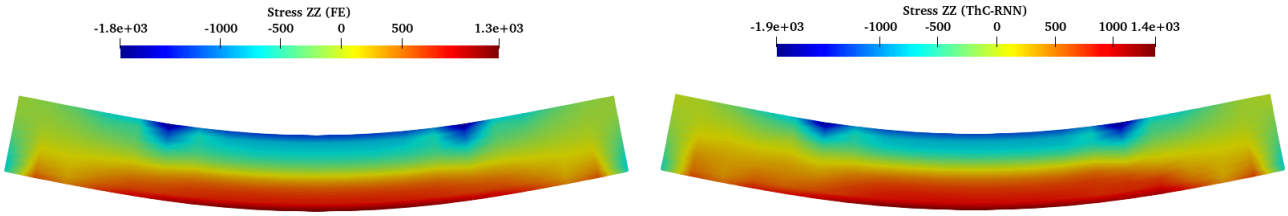


Figure 2.30: Comparison of the normal stress distribution σ_{ZZ} obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

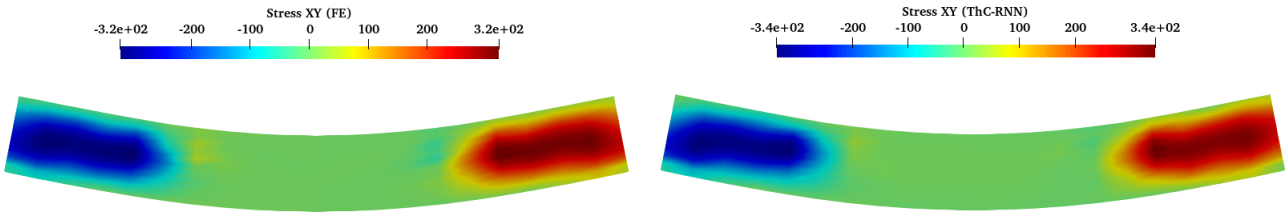


Figure 2.31: Comparison of the shear stress distribution σ_{XY} obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

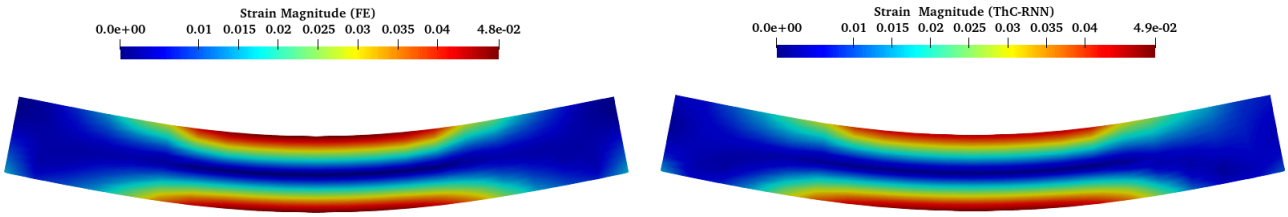


Figure 2.32: Comparison of the strain magnitude distribution ε_M obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

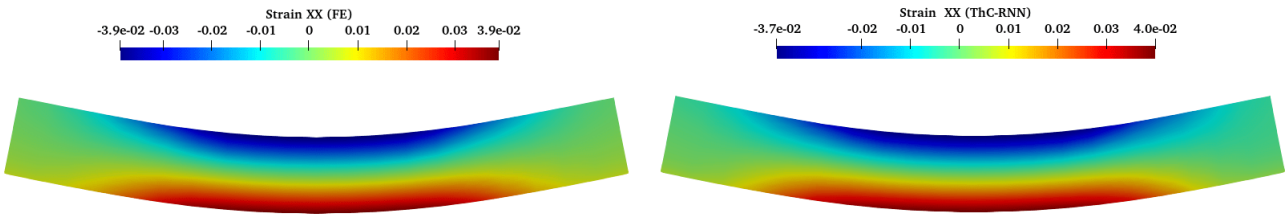


Figure 2.33: Comparison of the normal strain distribution ε_{XX} obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

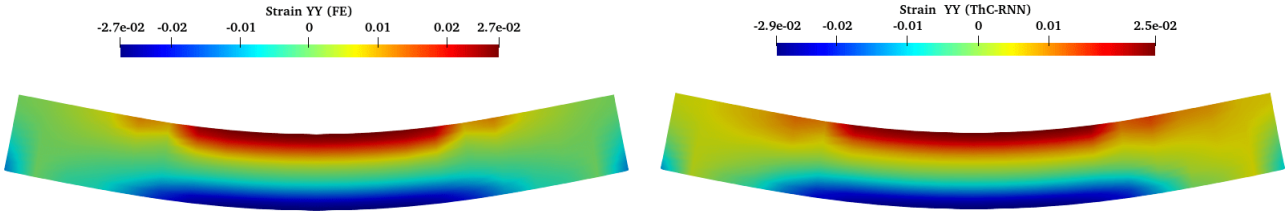


Figure 2.34: Comparison of the normal strain distribution ε_{YY} obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

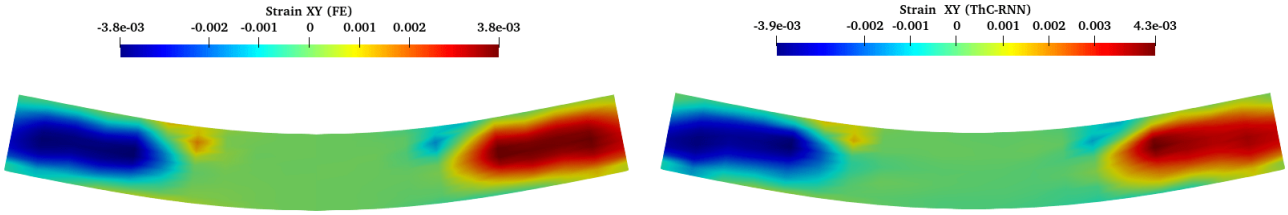


Figure 2.35: Comparison of the shear strain distribution ε_{XY} obtained by a pure FE approach that uses material original law (left figure) and by the integration of ThC-RNN as a surrogate in FE simulation (right figure)

2.5 Conclusions

Within this chapter, the potential of a hybrid physics-RNN model (ThC-RNN) is investigated to be utilized as a reliable and consistent surrogate for dissipative materials constitutive modeling. This hybrid model has been developed by combining a specific RNN architecture (Stacked LSTM) and physics based constraint resulting from thermodynamical formulation of the model. The obtained results showed the excellent predictive capabilities of ThC-RNN regarding several mechanical quantities (stress-strain responses, mechanical work rates, state variables, tangent matrix) compared to a standard RNN model used as a black-box. The robustness of ThC-RNN comes primarily from the high potential of the Stacked LSTM model which can be considered as a powerful tool to capture path dependant behaviors. On the other hand, the incorporation of the governing physical laws during the training phase provided relevant results with respect to the thermodynamic consistency. The present work provides a first insight of the achievable capabilities of physics guided deep learning techniques to treat engineering problems. While this study was only focusing on dissipative materials constitutive modeling, it might be interesting to test the proposed approach on different application fields where a relationship exist between the physical constraints and the physical quantities predicted by the surrogate model. Note that even if the computational time savings are not highlighted during this study given that the existing numerical methods for non-linear constitutive model are already fast enough, this work can be considered as proof of concept for potential future works where physics guided RNN are applied for finite elements multiscale analysis. However, an additional challenge has to be faced when using such approach for multi-scale simulation as there is no closed form of the second law the thermodynamics at the RVE scale which may also include many internal state variables. Several approaches are worth to be investigated (e.g, using averaging methods) to establish some physical constraints which are connected to the predicted average mechanical fields.

To conclude the first part of this manuscript, we first reviewed through a state of the art survey different machine learning approaches for engineering science applications. Based on recent contributions from the material science community, we have highlighted the major potential of this type of approaches to address complex mechanics of materials challenges including multifunctional material design, constitutive modeling and multiscale modeling. A particular emphasis has been devoted to Deep learning techniques, which is justified by the fact that artificial neural networks have strong capabilities to learn and perform advanced tasks with a high level of abstraction. In addition, according to the study conducted in chapter 2, the choice of deep neural networks and particularly recurrent neural networks has been proven to be the most suitable architecture to solve internal variables problems of heterogeneous materials under complex loading states, which is the main objective of this thesis. Hence, the following part of this manuscript is dedicated to multiscale modeling of heterogeneous materials using Artificial Neural Networks based approaches.

Part II

Multiscale modeling of heterogeneous materials using Artificial Neural Networks based approaches

General Introduction

Homogenization methods have been extensively studied by various researchers since the end of the 19th century to deal with the mechanical behavior of heterogeneous materials [Voigt 1887, Reuss 1929, Hill 1963, Hashin 1983]. This type of materials is commonly used in the engineering field to design for instance composite structures intended for the automotive, aeronautic or biomechanical industries. Homogenization approaches consist in identifying the overall mechanical behavior at the macro scale *i.e* the effective properties from the knowledge of microstructure characteristics (size and orientations of the particles, constitutive laws,etc). The first homogenization models date back to Voigt and Reuss work [Voigt 1887, Reuss 1929]. While these approaches lead to a simplification of the localization tensors required for the evaluation of effective properties, they generally do not provide an accurate description of material behavior as they are insensitive to heterogeneities shape and orientation. To take into account the interaction between different phases of the heterogeneous material as well as the effect of the microstructure on macroscopic responses, other more accurate approaches have therefore been proposed. Among these so-called multi-scale methods, mean field approaches and full-field approaches are commonly used.

i) Mean-field approaches as *Mori-Tanaka sheme* [Mori & Tanaka 1973] or *Self-consistent scheme* [Hill 1965] estimate the overall behavior of the heterogeneous material using average stress and strain for each material phase. These homogenization schemes rely heavily on Eshelby's single inclusion solution [Eshelby 1957, Eshelby 1959]. Eshelby's treatment of the inclusion problem is considered a milestone in the development of the micromechanics of random media. In 1957, John D. Eshelby provided a complete solution to the elastic field inside and outside ellipsoidal inclusions subjected to a prescribed eigenstrain in an infinite medium [Eshelby 1957, Eshelby 1959]. Many applications in the analysis and design of composite materials are based on Eshelby's solution, as the ellipsoidal shape is very versatile and can be used to represent various shapes of heterogeneous materials, including spheres and cylindrical wires [Withers 1989, Duan *et al.* 2006].

ii) Full-field approaches, based on the theory of periodic homogenization, estimate the overall behavior of the equivalent homogeneous material through the definition of a Representative Volume Element (RVE). Their advantages are mainly their ability to predict non-linear behaviors of structures at different scales for any type of periodic microstructure. Among these approaches, we can cite for instance the multi-scale method FE^2 developed by F.Feyel [Feyel 1999].

In Part.II of the present manuscript, we propose to incorporate AI based approaches into homogenization schemes for multiscale modeling of heterogeneous structures. For both of the aforementioned homogenization techniques, two deep learning models will be developed with different objectives. In Chapter 3, a hybrid model ANN-φ combining multilayer perceptrons and a micromechanical scheme is designed for fast exploration of the elastic response of composite materials. In Chapter 4, deep neural networks are integrated into a framework of full-field approaches to accelerate multiscale simulations of architectured materials.

Chapter 3

Estimation of the effective properties of heterogeneous materials using Artificial Neural Networks and micromechanical models

Contents

3.1 Introduction

Despite its efficiency for several microstructure often encountered in composite materials, the use of mean-field homogenization approaches to determine the effective properties of heterogeneous materials with large number of phases can lead to very long computational time. This limitation comes from the fact that Eshelby's tensors are generally estimated with numerical approaches [Gavazzi & Lagoudas 1990]. To overcome this challenge, Artificial Neural Network (ANN) can be an excellent candidate to reduce computational time. In the present work, a hybrid model combining Artificial Neural Networks and a micromechanical scheme is developed to predict the effective properties of heterogeneous materials with many phases. The choice to use a hybrid approach instead of pure numerical estimations or ANN only based modeling is justified by the following assertions:

i) As mentioned previously, a numerical estimation of the effective properties of an heterogeneous material with many phases time consuming since a numerical computation of Eshelby's tensor is required for each heterogeneity.

ii) It is difficult to design a generic model based only on Neural Networks to predict the effective stiffness tensor $\bar{L}$. Indeed, in this case, the ANN architecture and more precisely its input neurons number is a variable parameter and depends necessarily on the number of phases of the heterogeneous material. Moreover, if we assume that each phase in an heterogeneous medium with N inhomogeneity is described by m parameters including geometric and mechanical properties, the ANN will be defined by $m \times N$ inputs, and therefore an enormous amount of training data would be required for configurations with many phases. However, the proposed methodology is of utmost interest in the case of a large number of heterogeneities with different features (particles orientation, mechanical properties). Short fiber reinforced polymers parts are an illustrative example of such structure. Such material will be investigated later in this work.

The hybrid approach proposed in this study consists in splitting the problem in two parts. First, an Artificial Neural Network is built to quickly predict Eshelby's tensor S from the knowledge of ellipsoidal inclusion geometry and mechanical properties of the matrix. Then, a micromechanical homogenization scheme, here the so-called *Mori-Tanaka* [Mori & Tanaka 1973] is used to determine the effective stiffness tensor after a proper calculation of each phase localization tensor. A second objective is to study the balance of this hybrid modeling approach between computational time saving, prediction reliability and feasibility for treating heterogeneous materials with many phases. The main advantage of the proposed hybrid approach in comparison to pure ANN based material models is its ability to easily treat high dimensional inputs problems. Pure ANN approaches are generally used as surrogates to predict directly the quantities of interest. This solution is conceivable when the number of input neurons is very low for and therefore training those ANN models in a low dimensional space is feasible. In contrast, a pure ANN model can not be applied to the present problem as described in point ii), hence the use of a hybrid approach to predict first Eshelby's tensor of each heterogeneous material phase, then computing the effective elastic properties with a micromechanical homogenization scheme. In addition, this study can be viewed as a proof of concept of the use of hybrid approaches for heterogeneous materials modeling.

The present chapter is organized as follows: Eshelby's inclusion problem and Inhomogeneity problem

are presented in section 3.2. These 2 problems constitute an essential framework for the rest of the study since the applied homogenization approach depends mainly on Eshelby's solution. In Section 3.3, hybrid model ANN $-\varphi$ architecture is introduced. The building procedure of ANN $-\varphi$ model is explained. In particular, details are given about database generation, hyperparameters tuning and feature scaling. Section 3.4 presents the results of this study. First, the predictive capability of the ANN $-\varphi$ is highlighted. ANN $-\varphi$ model is then used in an illustrative example to predict the effective properties of Random Oriented Fiber Composite. A parametric analysis is finally performed to determine the effective behavior of heterogeneous materials with various ellipsoidal inclusions geometries and matrix properties. This section shows the model's efficiency in terms of reliability and computational time saving. Finally, ANN $-\varphi$ is compared to another hybrid model $\text{ANN}_2-\varphi$ where the neural network is trained on a strain interaction database instead of Eshelby's tensors. Some concluding remarks are given in Section 3.5.

3.2 Eshelby's problem

3.2.1 Eshelby's ellipsoidal inclusion problem

The Eshelby inclusion problem considers an inclusion Ω in an infinite elastic body V with elastic stiffness tensor L^0. An inclusion is a region where uniform eigenstrain ε^* appears. ε^* corresponds to the stress-free strain if Ω was separated from its surrounding elastic matrix. ε^* is considered generally inelastic, resulting for instance from thermal expansion and phase transformation. Notice that both the inclusion and the matrix have the same elastic stiffness tensor L^0 (Figure 3.1).

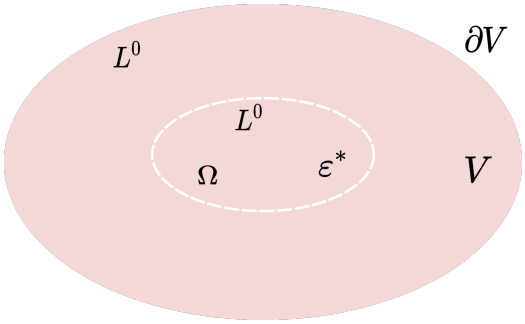


Figure 3.1: Eshelby's inclusion problem

Eshelby has proven that the strain ε field in the inclusion is uniform and can be expressed using the fourth-order Eshelby tensor S such that:

$$\varepsilon_{ij} = S_{ijkl}\varepsilon^*_{kl} \qquad \text{in } \Omega \tag{3.1}$$

Given the Eshelby tensor S, the stress field inside the inclusion σ can be obtained:

$$\sigma_{ij} = L^0_{ijkl}(\varepsilon_{kl} - \varepsilon^*_{kl}) = L^0_{ijkl}(S_{klmn}\varepsilon^*_{mn} - \varepsilon^*_{kl}) \tag{3.2}$$

The fourth order tensor S relates the eigenstrain ε^* to the constrained strain ε, which is the actual strain of the inclusion when embedded in the matrix. S depends on inclusion shapes and material properties L^0. Considering an ellipsoidal inclusion with semi-axes a_1,a_2,a_3 (Figure 3.2), the integral form of S is given by the formula [Mura 1987]:

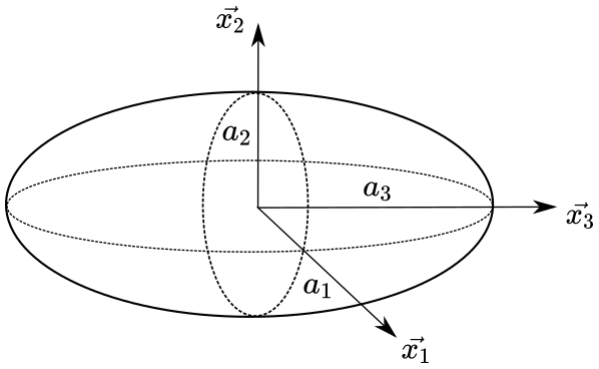


Figure 3.2: Ellipsoidal inclusion with semi-axes a_1, a_2, a_3

$$S_{ijkl} = \frac{L^0_{mnkl}}{8\pi} \int_{-1}^{1} \int_{0}^{2\pi} (G_{imjn} + G_{jmin}) d\omega d\zeta_3 \tag{3.3}$$

with

$$G_{ijkl} = \hat{\zeta_k}\hat{\zeta_l} Z^{-1}_{ij}, Z_{ik} = L^0_{ijkl}\hat{\zeta_j}\hat{\zeta_l}$$

$$\hat{\zeta_1} = \sqrt{1-\zeta_3}\frac{\cos\omega}{a_1}, \hat{\zeta_2} = \sqrt{1-\zeta_3}\frac{\sin\omega}{a_2}, \hat{\zeta_3} = \frac{\zeta_3}{a_3}$$

For special inclusion geometries and isotropic materials, analytical form of S can be found in [Mura & Barnett 1983]. For anisotropic materials and ellipsoidal inclusions, the Eshelby tensor is computed numerically using Gauss quadrature method [Gavazzi & Lagoudas 1990], Eshelby's tensor components are given using equation 3.4. For the case of isotropic ellipsoidal inclusions, a reduced elliptical integral form can be found in [Mura 1987]. In this case, if a numerical evaluation of Eshelby tensor is still required, the expected computational time required for the numerical integration is significantly reduced [Barthélémy 2020].

$$S_{ijkl} = \frac{L^0_{mnkl}}{8\pi} \sum_{p=1}^{M} \sum_{q=1}^{N} [G_{imjn}(\omega_q, \zeta_{3_q}) + G_{jmin}(\omega_q, \zeta_{3_q})] W_{pq} \tag{3.4}$$

where $M \times N$ are the number of Gauss integration points along spherical coordinates, (ω_q, ζ_{3_q}) and W_{pq} are the computed Gauss-Legendre quadrature points and weights matrix respectively [Swarztrauber 2003]. This numerical evaluation significantly increase the computational cost of S but allows to evaluate it for anisotropic inclusions.

3.2.2 Inhomogeneity problem and Eshelby's equivalent inclusion principle

The inhomegeneity problem considers a small inhomogeneity Ω with elastic stiffness tensor L^1 in an infinite elastic body V with material properties L^0. In the far field, V is subjected to uniform surface traction $\vec{T}$ such as $\vec{T} = \sigma_0 \cdot \vec{n}$. The displacement u of any material point x located in the far field can be expressed using strain tensor ε_0 such as $u = \varepsilon_0 \cdot x$. The inhomogeneity problem can thus be stated using the following set of equations :

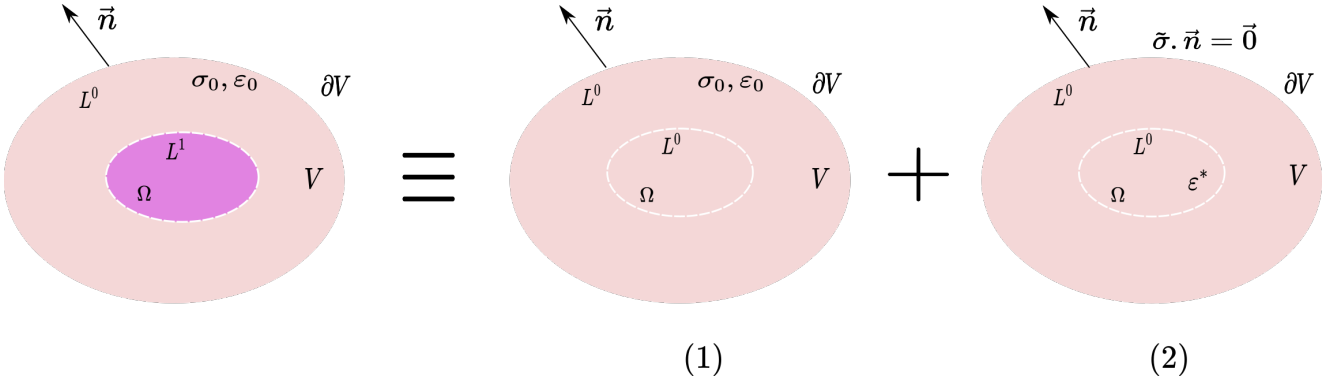


Figure 3.3: Illustration of Eshelby's equivalence principle: Inhomogeneity problem split into two problems (1) and (2)

$$\begin{aligned} &\frac{\partial \sigma_{ij}}{\partial x_j} = 0 \qquad \text{in } V \\ &\sigma_{ij} = \begin{cases} L^0_{ijkl}\varepsilon_{kl} & \text{in } V - \Omega \\ L^1_{ijkl}\varepsilon_{kl} & \text{in } \Omega \end{cases} \\ &\vec{T} = \sigma_0 \cdot \vec{n} \quad \text{and} \quad u = \varepsilon_0 \cdot x \quad \text{in far field} \end{aligned} \tag{3.5}$$

The problem (3.5) can be solved using Eshelby's equivalence principle [Eshelby 1961] shown in Figure 3.3. According to this principle, (3.5) can be expressed as the sum of two simpler problems (1) and (2) :

(1) : A boundary value problem with uniform surface traction $\vec{T}$ at the far field, where the inhomogeneity Ω has the same elastic properties L^0 as V. The solution of this problem is expressed as uniform stress σ_0 and strain ε_0 fields inside the entire body V.

(2) : A boundary value problem with zero surface tractions, where the inhomogeneity is replaced by an inclusion with the same shape but of mechanical properties L^0_{ijkl} as the rest of the body V. To compensate for the material change between the inhomogeneity and the inclusion, prescribed eigenstrain ε^* is attributed to the inclusion. According to Eshelby's inclusion problem, the strain field inside Ω can be evaluated using Eshelby's fourth tensor S:

$$\tilde{\varepsilon}_{ij} = S_{ijkl}\varepsilon^*_{kl} \qquad \text{in } \Omega \tag{3.6}$$

The principle of superposition holds in this case as all the problems stated above are elastic, which means that the solution of (3.3) is the sum of the two problems (1) and (2). Inside the inhomogeneity Ω, the total stress and strain fields are expressed,

$$\begin{aligned} \sigma_{ij} &= \sigma^0_{ij} + \tilde{\sigma}_{ij} \qquad \text{in } \Omega \\ \varepsilon_{ij} &= \varepsilon^0_{ij} + \tilde{\varepsilon}_{ij} \qquad \text{in } \Omega \end{aligned}$$

According to [Eshelby 1961], it can be proven that the total strain inside the inhomogeneity is expressed using a fourth order tensor T:

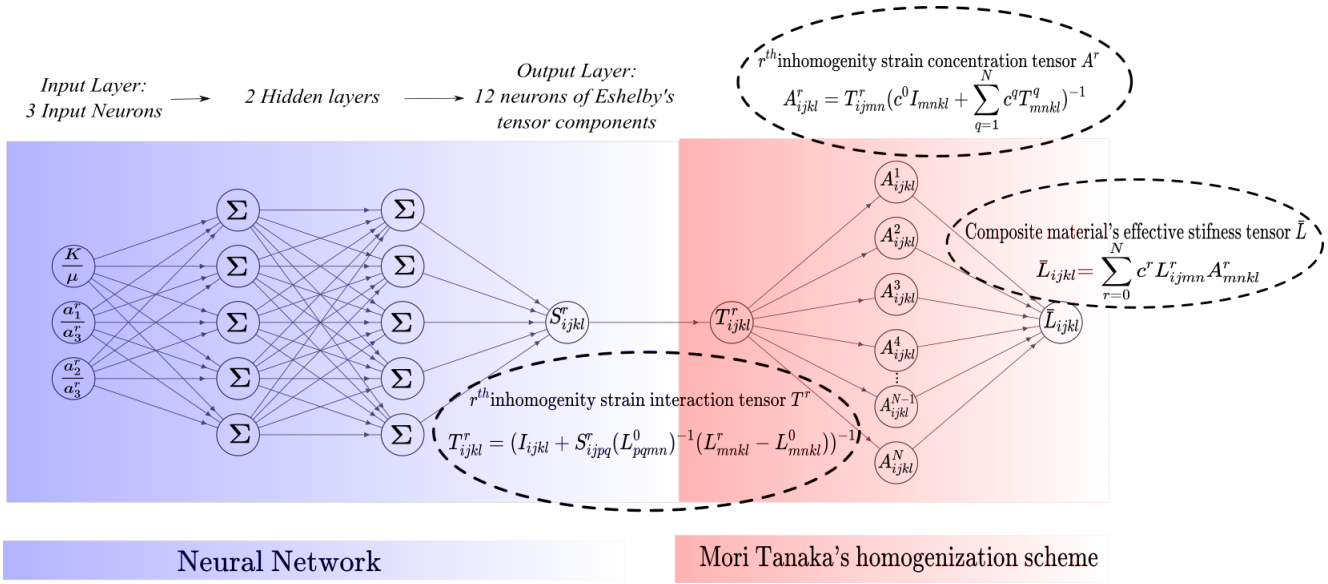


Figure 3.4: Architecture of ANN-φ : A hybrid model combining Artificial Neural Networks and micromechanical homogenization using *Mori-Tanaka* scheme

$$\varepsilon_{ij} = T_{ijkl}\varepsilon^0_{kl} \qquad \text{in } \Omega \tag{3.7}$$

where T is called strain interaction tensor. T depends on the computed Eshelby's tensor S and mechanical properties of the inhomoheneity L^1 and the matrix L^0,

$$T_{ijkl} = (I_{ijkl} + S_{ijpq}(L^0_{pqmn})^{-1}(L^1_{mnkl} - L^0_{mnkl}))^{-1} \tag{3.8}$$

where I_{ijkl} denotes the fourth order symmetric identity tensor.

3.3 ANN-φ : A hybrid model combining Artificial Neural Networks and micromechanical modeling

The architecture of the proposed hybrid model ANN-φ can be split into two parts. First, an Artificial Neural Network is built to predict Eshelby's tensor components S_{ijkl} from the knowledge of ellipsoidal inclusion geometry and mechanical properties of the matrix. Then, a micromechanical homogenization scheme such as *Mori-Tanaka* [Mori & Tanaka 1973] is used to determine the strain concentration tensors A^r associated with each inhomogeneity r of the heterogeneous medium. A^r is estimated from the predicted Eshelby's tensor S^r. Finally, the effective stiffness tensor $\bar{L}$ is evaluated using mechanical properties L^r and strain concentration tensor A^r of each phase. The Figure 3.4 shows the architecture of ANN-φ.

For the present problem, the ANN architecture shown in figure 3.5 is used. This *MLP* consists on three parts: an input layer of medium's mechanical properties and inclusions geometrical attributes, 2 hidden layers and an output layer with 12 neurons of non zero Eshelby's tensor components for isotropic

mediums. The optimal number of hidden layers, hidden neurons and activation function are obtained by performing an hyperparameter analysis which will be discussed in details in subsection 3.3.

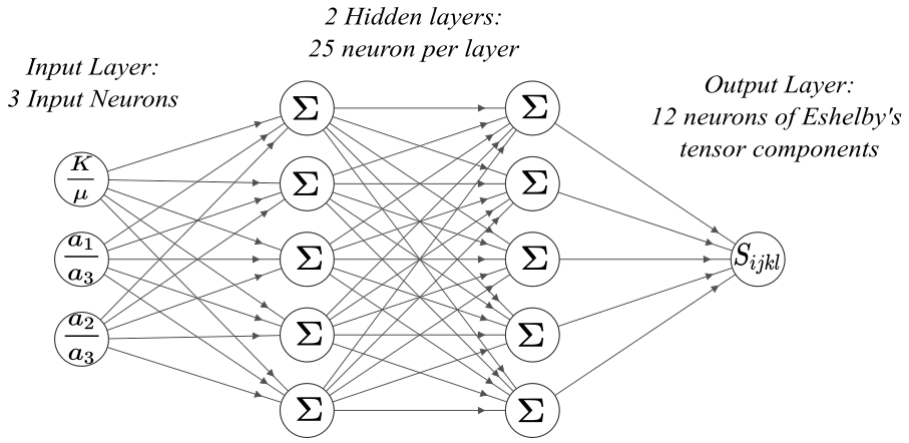


Figure 3.5: Structure of the Multi Layer Perceptron *(MLP) used in the study*

3.3.1 Database Preparation

A database containing Eshelby's tensor components corresponding to different configurations was generated using an open source software in mechanics and materials *Simcoon* [Chemisky 2018]. As mentioned in section 3.2, Eshelby's tensor S depends on the geometric parameters (a_1, a_2, a_3) of the ellipsoidal inclusion (Figure 3.2) as well as on mechanical properties of the matrix L^0. For isotropic medium, the stiffness tensor L^0 can be expressed using Bulk modulus K and Shear modulus μ which also represents the eigenvalues of L^0,

$$L^0_{ijkl} = 3K P^H_{ijkl} + 2\mu P^D_{ijkl} \tag{3.9}$$

with

$$P^H_{ijkl} = \frac{1}{3}\delta_{ij}\delta_{kl}, P^D_{ijkl} = \frac{1}{2}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) - \frac{1}{3}\delta_{ij}\delta_{kl}$$

In order to better control the number of parameters in the Neural Network's input layer, the geometric and mechanical properties are expressed in adimensional forms by introducing the ratios $\frac{K}{\mu}, \frac{a_1}{a_3}, \frac{a_2}{a_3}$. This choice is justified by the fact that Eshelby's tensor is calculated in an infinite medium, therefore an adimensional representation is more significant since the Eshelby's tensor computation will only depends on the scale ratio of each physical quantity. The ratio $\frac{K}{\mu}$ is always defined due to the positivity of L^0.

The strategy of database $\overline{\mathscr{D}}$ generation is described as follows: 1 million samples are generated using *Simcoon*. Regarding the precision of the numerical method, a maximum difference of 0.25% has been observed in the components of Eshelby tensor when computed in the case of a cylindrical geometry with 100×100 integration points in the ellipsoid (most unfavorable case). As for the numerical parameters required to generate Eshelby tensors database, 100x100 Gauss integration points have been used. Increasing the number of integration points beyond theses values didn't show any significant improvement on Eshelby's tensor components. The inputs $\frac{K}{\mu}, \frac{a_1}{a_3}, \frac{a_2}{a_3}$ are set on the interval $[10^{-2}, 10^2]$ and are chosen

randomly on a logarithmic space. Then, samples are split into training $\overline{\mathscr{T}}$, validation $\overline{\mathscr{V}}$ and test $\overline{\mathcal{T}}$ data sets. Training data are used during ANN's learning process in order to fit the parameters of the model (weights and bias). Then, to verify the accuracy of the training procedure, a subset (20%) is chosen randomly from training data after each Epoch and used as the validation data. Finally, test data set are the samples that have never been used in training, the purpose is to provide an unbiased evaluation of the model once it is trained and to verify its capability to generalize.

The last step of data pre-processing phase consist in the Feature Scaling. Rescaling the data before training is necessary since the scale and distribution of input and output variables may be distinct. This scale difference may increase the difficulty of the problem as the model may suffer from instability during training process. Several normalization techniques can be used as Min-Max Scaler or Standard Scaler [Wan 2019]. In this study, the last scaling technique is used. Standardization is a transformation that centers the data by removing the mean value m of each feature and then dividing it by its standard deviation σ.

$$z = \frac{x - m}{\sigma}$$

Standardization can drastically improve the model's performance. For instance, if a feature has a variance that is orders of magnitude larger than others, it might dominate the loss function and make Neural Network unable to learn from other features correctly as expected.

3.3.2 Hyperparameters tuning

In this study, a Grid Search with Cross validation score is applied to the model. The aim is to find the combination of hidden neurons number and activation function allowing to have the most reliable prediction. This parametric analysis has been performed on a subset of data (10000 samples) using GridSearchCV tool of Sci-kit learn library [Pedregosa *et al.* 2011]. To tune hidden neurons number, several approaches proposed in the literature have been tested, and are presented in Table 3.1 where N_i and N_o are the input and output neurons number respectively, N_p is the input sample number and N_L is the number of hidden layers. As for activations, 5 functions have been tried out (Sigmoid, Rectified Linear Unit (ReLU), SoftPlus, Hyperbolic Tangent (TanH) and Exponential Linear Unit (ELU).

Table 3.1: Various approaches for fixing hidden neurons number in Neural Networks

Method	**Number of hidden neurons**
[Tamura & Tateishi 1997]	$N_h = N_i N_o / 2$
[Zhang *et al.* 2003]	$N_h = 2^{N_i}/(N_i + 1)$
[Ke & Liu 2008]	$N_h = (N_i + \sqrt{N_p})/N_L$
[Shibata & Ikeda 2009]	$N_h = \sqrt{N_i N_o}$
[Sheela & Deepa 2013]	$N_h = (4N_i^2 + 3)/(N_i^2 - 8)$

The minimization of model's Mean Squared Error (MSE) on validation data was chosen as a criterion to select the optimal parameters. As shown in Figure 3.6, the parametric analysis shows that the

hyperbolic tangent activation function and 25 hidden neurons per layer provides a minimum value of the error for a Neural Network with 2 hidden layers. The optimal number of hidden neurons is obtained via [Ke & Liu 2008] method. This pair of parameters is further used to train the model with the entire data set.

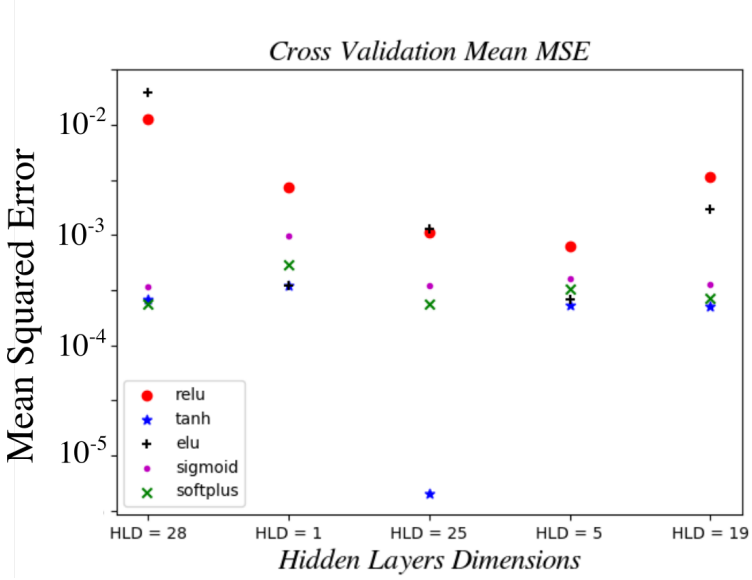


Figure 3.6: Determination of model's optimal hyperparameters using Grid Search method

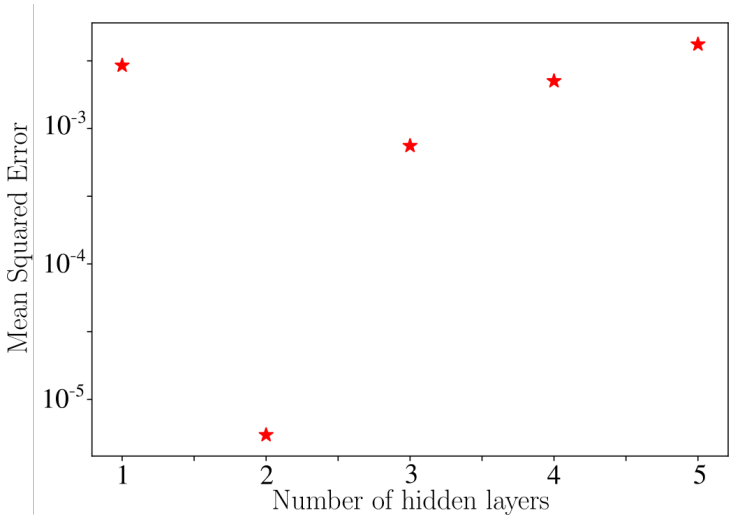


Figure 3.7: Determination of model's optimal number of hidden layers

The determination of the optimal number of hidden neurons was a subject of an another parametric study once the activation function and the method fixing hidden neurons number were found. Various number of hidden layers were tested (1 to 5 layers) and the optimal number is found to be 2 layers according to Figure 3.7. Thus, increasing hidden layers number does not necessarily improve model's

reliability because of the vanishing gradient problem. This problem is often encountered when training ANN models with gradient-based optimizers using backpropagation algorithm. As explained in subsection 3.2, backpropagation consists on updating model's parameters (weights and biases) by computing the partial derivative of the loss function with respect to the weights and biases with the chain rule method. In some cases, the gradients values can be vanishingly insignificant and therefore preventing model's parameters from changing its values. For instance, a sigmoid activation function has gradients in the range $(0, \frac{1}{2})$, multiplying N of these gradient values to update the weights and biases in an N layer neural network could prevent the first layers from changing its parameters values when N is very high as the resulting gradient values tends towards 0.

3.4 Results and discussion

3.4.1 Prediction of Eshelby's tensor S by ANN approach

Once Eshelby's tensor training database is prepared and Neural Network hyperparameters are tuned, we proceed to the training phase of the ANN. The model implementation was done in Python language with the use of the Google's TensorFlow library [Abadi *et al.* 2015]. Figure 3.8 describe the evolution of Normalized Mean Squared Error (NMSE) on training and validation set as a function of Epochs. First, we can notice a fast convergence of the ANN model and Loss stabilization after 10 Epochs. In addition, the obtained NMSE on the last Epoch are in order of $10^{-2}\%$ for training set and $10^{-3}\%$ for validation set. Those low values shows that the ANN model achieved a good performance after training and has been prevented from overfitting as the validation error is very low. Even if it is very common for the validation error to be greater than the training error for most ANN models, the error values obtained in this study (0.017% and 0.0031%) are insignificant with respect to the numerical noise. The only information to retain in that case is that the model was sufficiently well trained and prevented from overfitting as the validation error is also very low. The values indicated here cannot be interpreted to compare the training accuracy with respect to validation accuracy.

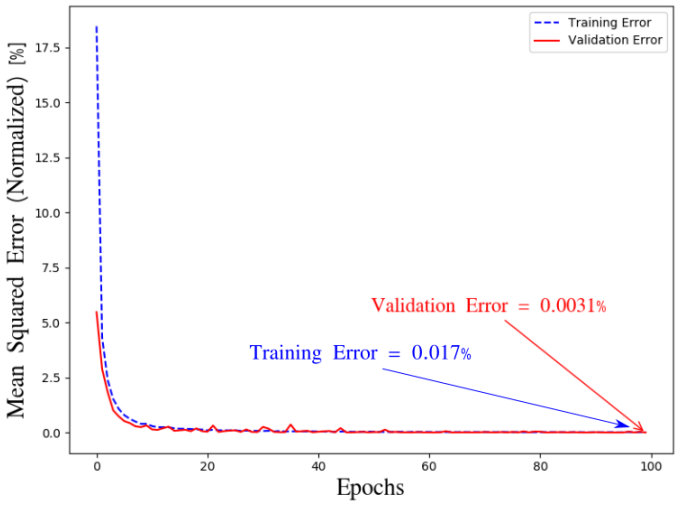


Figure 3.8: Evolution of training and validation errors as a function of Epochs during ANN training

The model was then evaluated on a test data set to verify its predictive ability on samples, other than training data. To quantify the predictive quality of the model, the coefficient of determination (R^2) introduced in Equation.2.54. This coefficient measures the accuracy of fitting by giving an indication of how well the test data are predicted by the model.

For each Eshelby's tensor test component, a correlation plot between ANN predicted values and true values is presented in Figure 3.9. An excellent agreement between predictions and targets is shown from the plots as R^2 values are very close to 1 for all the components. It is found that the designed model is very reliable and can be used to rapidly predict Eshelby's tensor for a given ellipsoid geometry and medium mechanical properties. These results can be used as inputs to the hybrid model in order to predict the effective mechanical behavior of heterogeneous materials with many phases. An illustration of this approach is provided in the next section.

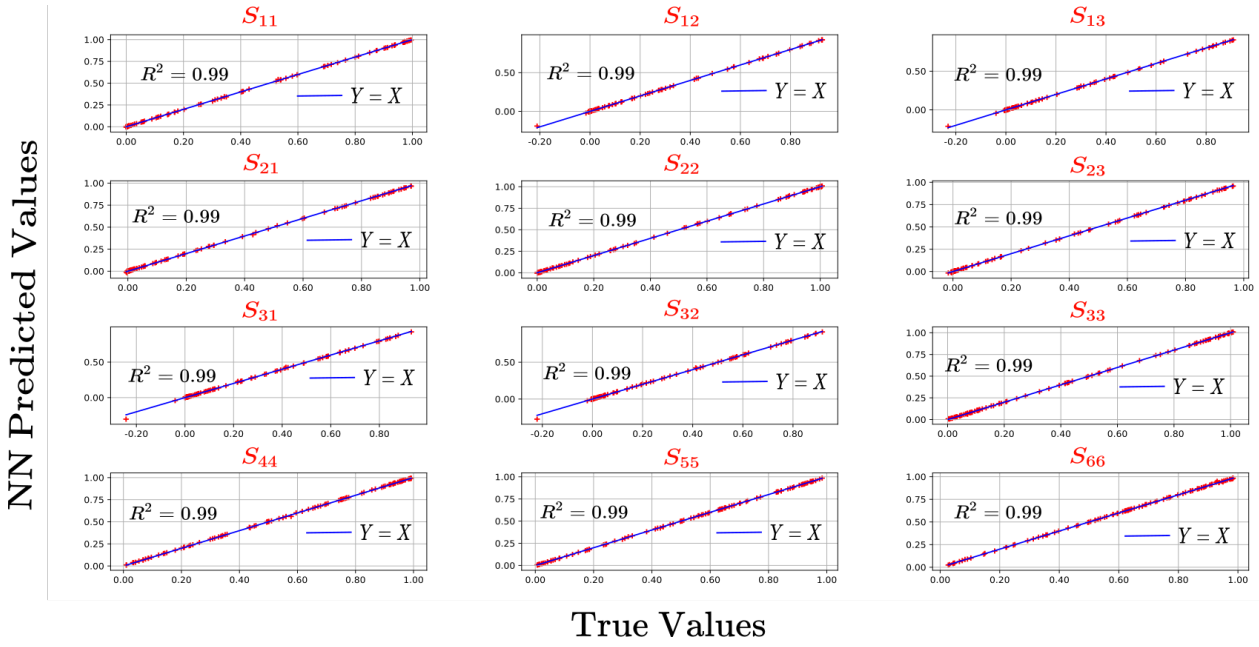


Figure 3.9: Correlation plots between ANN predicted values and target values for each Eshelby's tensor test component

3.4.2 Illustrative Example: Prediction of effective properties of Random Oriented Fiber Composite

Results of the previous section have demonstrated the predictive capability of Neural Network model. This ANN is then coupled with a physical homogenization model to predict the effective behavior of a random oriented fiber composite. As shown in Figure 3.10, the studied composite material consists of an Isotropic Epoxy matrix Ω_0 of volume V_0 and N Transversely Isotropic Carbon fibers Ω_r of volume V_r, with $r \in [1, N]$. The volume fraction c_r of each phase is defined by $c_r = \frac{V_r}{V}$ with $\Omega = \Omega_0 \cup \Omega_1 \cup_2 \cup ... \Omega_r \cup ... \Omega_N$ and $V = V_0 + V_1 + ... + V_r + ... + V_N$. The mechanical properties of the matrix and the fibers are summarized in Table 3.2. Each fiber is considered as a different phase because of the different orientation which lead to mechanical properties.

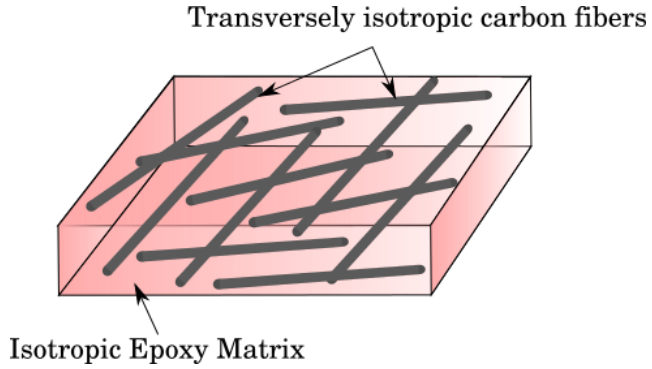


Figure 3.10: Studied Random Oriented Fiber Composite

Table 3.2: Properties of AS4 carbon fibre and Hercules 3501-6 epoxy matrix [Mccartney 2017]

Material parameter	**Value**
Matrix Young's modulus E^m	4.2 GPa
Matrix Poisson's ratio ν^m	0.34
Matrix volume fraction c_m	0.7
Fibre longitudinal Young's modulus E_L^f	225 GPa
Fibre transverse Young's modulus E_T^f	15 GPa
Fibre longitudinal Poisson's ratio ν_L^f	0.2
Fibre transverse Poisson's ratio ν_T^f	0.07
Fibre longitudinal shear modulus μ_L^f	15 GPa
Fibre transverse shear modulus μ_T^f	7 GPa
Fibre Total volume fraction c_f	0.3
Fibre Number N	1000

To determine the effective elastic stiffness tensor of the composite material, it is necessary to evaluate the strain concentration tensor A^r for each phase. We recall that A^r relates the average strain on the r^{th} inhomogeneity $\bar{\varepsilon}^r$ with the average strain of the entire composite material $\bar{\varepsilon}$:

$$\bar{\varepsilon}_{ij}^r = A_{ijkl}^r \bar{\varepsilon}_{kl}$$

With the help of the strain concentration tensors, the effective elastic stiffness tensor is given by the following expression:

$$\bar{L}_{ijkl} = \sum_{r=0}^{N} c_r L_{ijmn}^r A_{mnkl}^r \tag{3.10}$$

To determine the expressions of the strain concentration tensors, Mori-Tanaka homogenization scheme is used [Mori & Tanaka 1973]. This approach is based on the Eshelby single-inclusion solution where A^r is given by:

$$A^r_{ijkl} = T^r_{ijmn}(c_0 I_{mnkl} + \sum_{i=1}^{N} c_i T^i_{mnkl})^{-1} \tag{3.11}$$

where

$$T^r_{ijkl} = (I_{ijkl} + S^r_{ijpq}(L^0_{pqmn})^{-1}(L^r_{mnkl} - L^0_{mnkl}))^{-1}$$

The Figure 3.11 shows a comparison between the composite effective properties obtained by ANN-φ model and by numerical estimation of $\bar{L}$ (using the numerical integration proposed in [Gavazzi & Lagoudas 1990] to compute the Eshelby tensor, then computing T^r_{ijkl}, A^r_{ijkl} and $\bar{L}_{ijkl}$ from the expressions above). We can notice that ANN-φ predicted values are almost similar to the ones estimated numerically and each component relative error do not exceed 3 % as described in table 3.3. These results show that the proposed model have a strong potential to predict the homogenized behavior of heterogeneous materials.

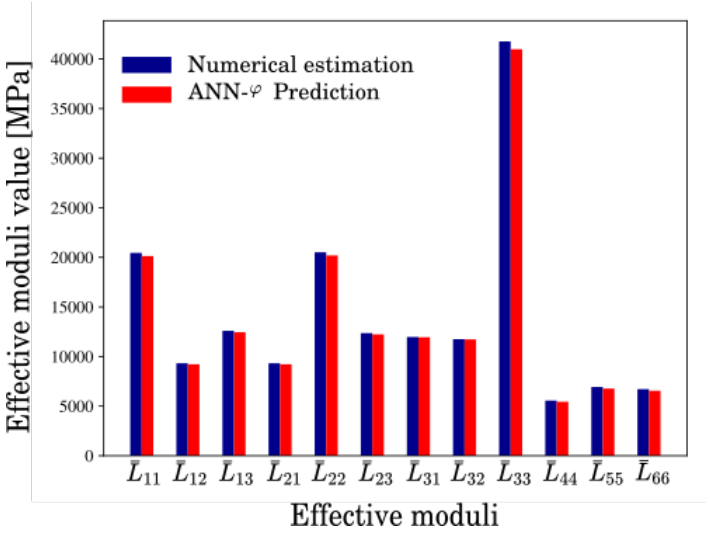


Figure 3.11: Comparison between ANN-φ predicted effective stiffness tensor $\bar{L}$ and numerical estimation of $\bar{L}$

An interesting feature of a hybrid ANN-φ model relies in its capability to predict not only the effective response, but also the average mechanical fields in each phase. The determination of local response such as phase local stress is indeed necessary during the design of composite structures to predict certain local phenomena such as damage. The model's ability to accurately predict the local stresses of each phase when the composite is subjected to a global unit uniaxial stress state is evaluated therefore:

$$\bar{\sigma} = \begin{pmatrix} \sigma_{11} = 0 \\ \sigma_{22} = 0 \\ \sigma_{33} = 1 \\ \sigma_{12} = 0 \\ \sigma_{13} = 0 \\ \sigma_{23} = 0 \end{pmatrix}$$

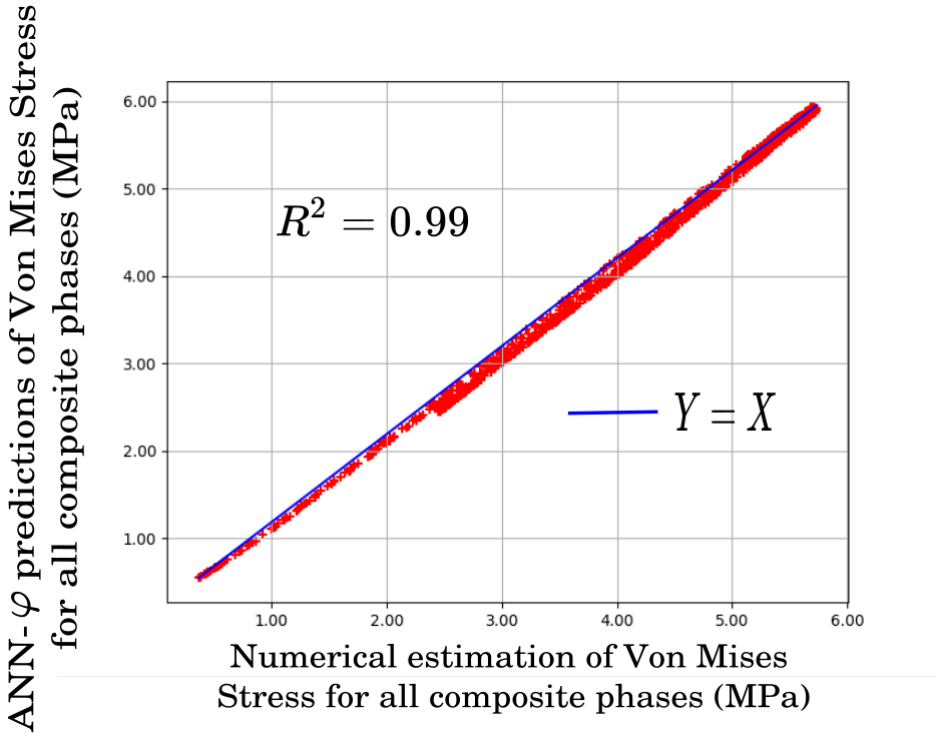


Figure 3.12: Correlation plots between numerical estimations σ_{Vm}^{Num} and ANN-φ predicted Von Mises Stress $\sigma_{Vm}^{ANN-\varphi}$ of each composite heterogeneity

Figure 3.12 shows the correlation plot between the equivalent Von Mises stress $\sigma_{Vm}^{ANN-\varphi}$ predicted by ANN$-\varphi$ and its numerical estimation σ_{Vm}^{Num}. Note that the Von-Mises stress is chosen here for illustration only, obviously not adapted to predict the failure of carbon fibers. An excellent correspondence between the two quantities is observed since the coefficient of determination R^2 is very close to 1. The model is able to accurately simulate the components of concentration tensors which are required to evaluate mechanical fields, considering mean field micro-mechanical models.

Table 3.3: Comparison between ANN-φ predicted effective stiffness tensor $\bar{L}$ and numerical estimations

Effective moduli	**Predicted values $\bar{L}_{ij}^{NN}$ with ANN-φ**	**Numerical values $\bar{L}_{ij}^{Num}$**	**RE** $= \left\lvert \frac{\bar{L}_{ij}^{NN} - \bar{L}_{ij}^{Num}}{\bar{L}_{ij}^{Num}} \right\rvert$
$\bar{L}_{11}$ (GPa)	20.13	20.45	1.58 %
$\bar{L}_{12}$ (GPa)	9.22	9.32	1.07 %
$\bar{L}_{13}$ (GPa)	12.45	12.59	1.1 %
$\bar{L}_{21}$ (GPa)	9.22	9.32	0.05 %
$\bar{L}_{22}$ (GPa)	20.19	20.51	1.61 %
$\bar{L}_{23}$ (GPa)	12.23	12.36	1.07 %
$\bar{L}_{31}$ (GPa)	11.94	11.97	0.2 %
$\bar{L}_{32}$ (GPa)	11.73	11.75	0.16 %
$\bar{L}_{33}$ (GPa)	40.98	41.77	1.81 %
$\bar{L}_{44}$ (GPa)	5.44	5.56	2.02 %
$\bar{L}_{55}$ (GPa)	6.77	6.93	2.36 %
$\bar{L}_{66}$ (GPa)	6.55	6.71	2.29%

3.4.3 Parametric analysis: Prediction of effective properties of heterogeneous materials with various ellipsoidal inclusions geometries and material properties

Through an illustrative example of a composite material, the excellent capability of the hybrid model to predict some quantities as effective homogenized properties and local stresses has been highlighted in the previous section. However, it is recalled here that for the case of inhomogeneities with cylindrical geometry (fibers), an analytical solution of Eshelby's tensor can be obtained. The previous example is considered as a test case to evaluate the proposed hybrid model reliability. In this section, more general cases where the numerical estimation of effective properties can be quite costly are investigated. These cases correspond to heterogeneous materials with various shape of ellipsoidal inclusions and where a numerical computation of Eshelby's tensor is necessarily required. For this purpose, two parametric analysis were performed to predict the heterogeneous material effective stiffness tensors using ANN-φ:

i) First, the effect of inclusions geometry on the effective properties is investigated by varying the semi-axis ellipsoids ratios a_1/a_3 and a_2/a_3 in a range between 10^{-2} and 10^2. The mechanical properties of the inhomogenities and the matrix are fixed to those of AS4 carbon fiber and Hercules 3501-6 epoxy matrix mentioned in Table 3.2. The volume fraction of inclusions is kept at $c_f = 0.3$. Figures 3.13 and 3.14 show a comparison between ANN-φ predicted effective stiffness tensor components and its numerical estimations for various ellipsoid geometries.

ii) For the second parametric study, the geometry and mechanical properties of the inclusions are fixed and matrix Bulk modulus K and Shear modulus μ vary from 10^3 Mpa to 10^5 Mpa. Similarly, the purpose is to verify ANN-φ model reliability when predicting the homogenized behavior of an heterogeneous material with different matrix mechanical properties. Figures 3.15 and 3.16 show a comparison between ANN-φ predicted effective stiffness tensor components and its numerical estimations for different values of K and μ.

It can be seen from figures 3.13,3.14,3.15 and 3.16 that effective stiffness tensor components are very well predicted by the ANN-φ model for most configurations given the low value of residuals. It can also be mentioned that the areas with high residuals in figures 3.15 and 3.16 correspond to configurations where effective stiffness tensor components have near zero values, and therefore insignificant. Few specific cases from figure 3.13 and 3.14 can be highlighted where the error is relatively large compared to other configurations (around 10%). The main hypothesis to explain these discrepancies is the propagation of errors. Eshelby's tensors prediction errors, even of low values, can propagate during the computation of the effective tensor which is a combination of each phase localization tensors. Moreover, the computation of strain interaction tensor T from predicted Eshelby's tensor S involves intermediate operations as product or inverse of matrix which may induce a propagation and an amplification of the error. Hybrid models can thus suffer from this phenomenon, hence the need to have accurate Neural Network models to reduce this effect as much as possible.

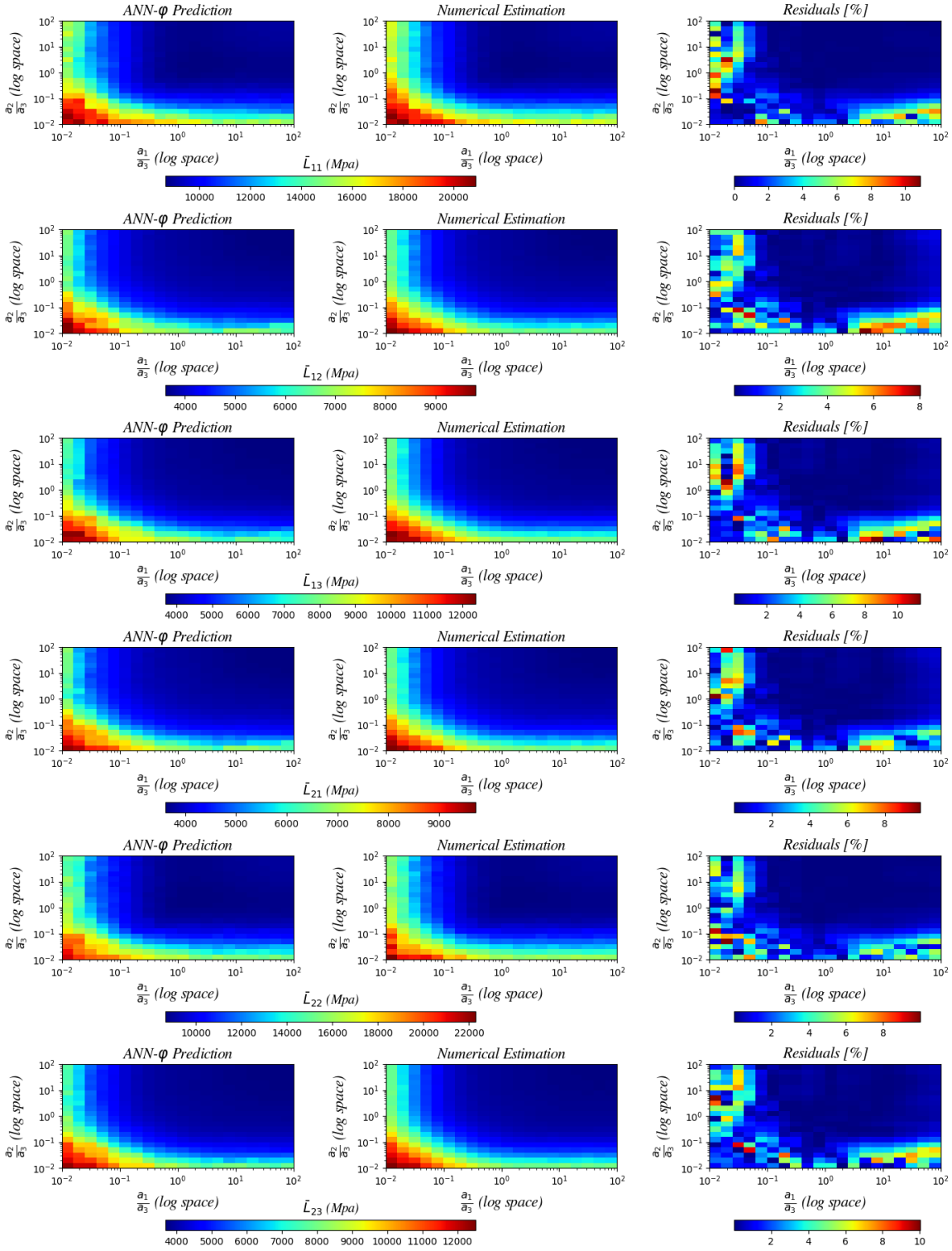


Figure 3.13: Comparison between ANN-φ predicted effective stiffness tensor components and its numerical estimation for different ellipsoidal inclusions

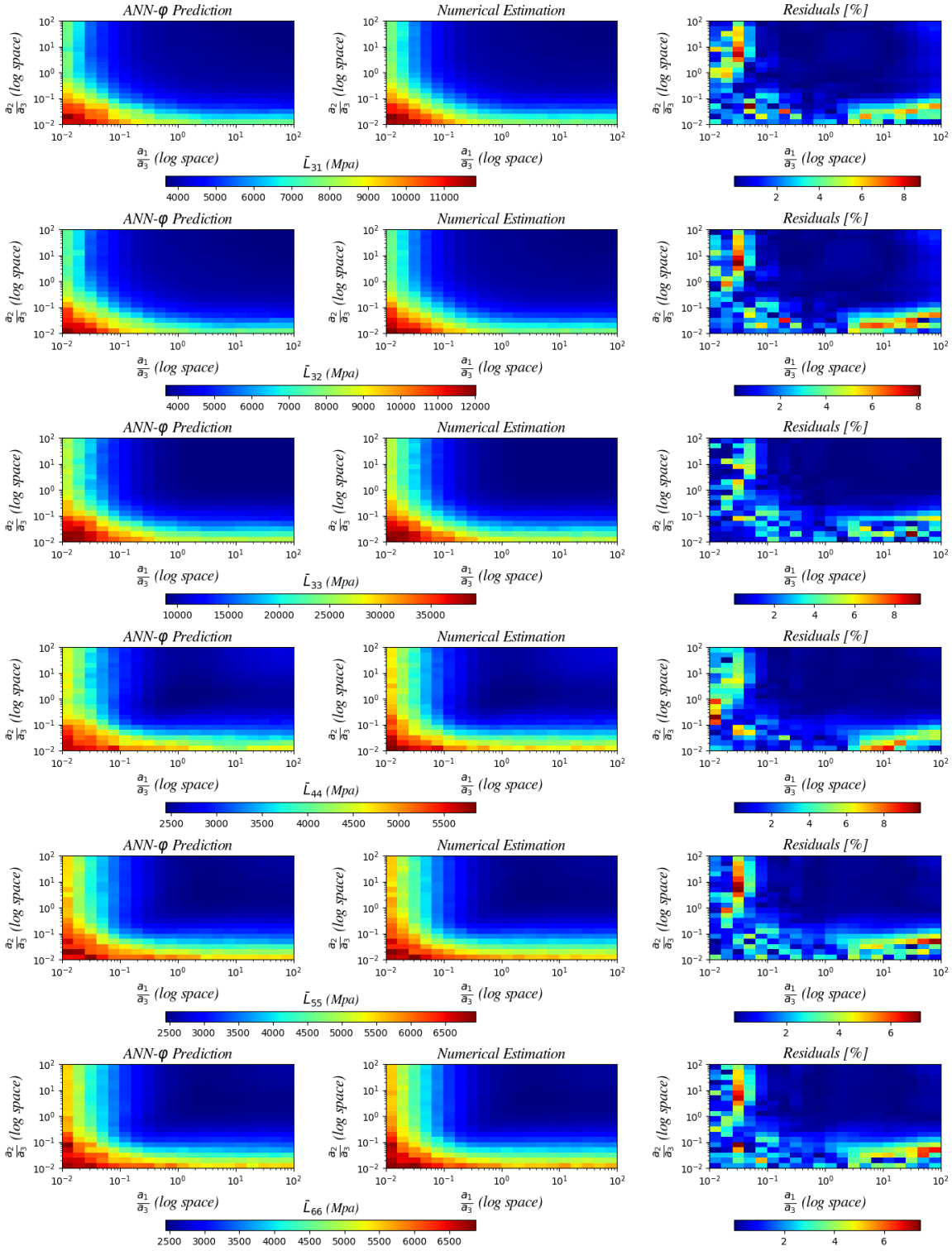


Figure 3.14: Comparison between ANN-φ predicted effective stiffness tensor components and its numerical estimation for different ellipsoidal inclusions

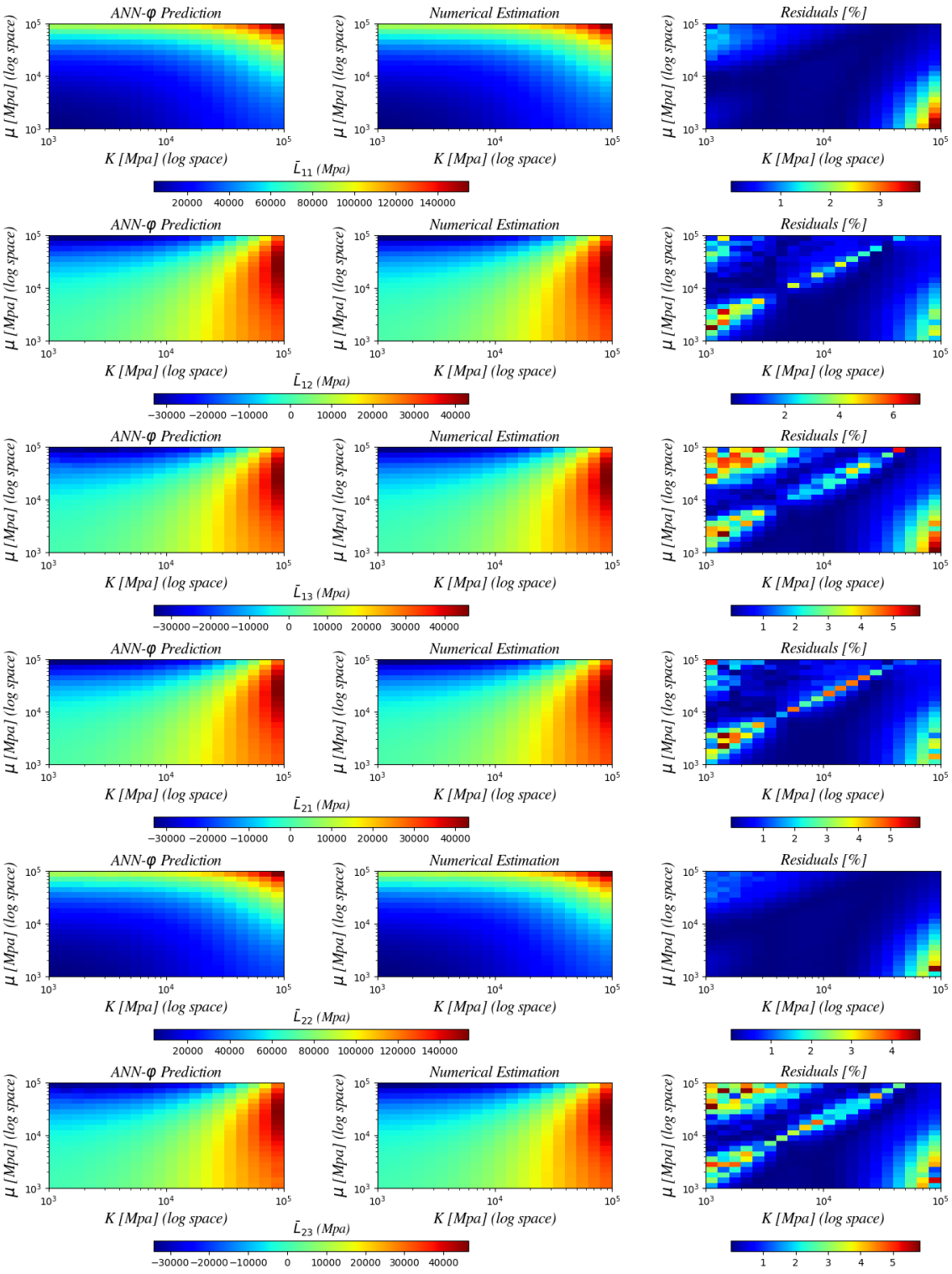


Figure 3.15: Comparison between ANN-φ predicted effective stiffness tensor components and its numerical estimation for different matrix mechanical properties

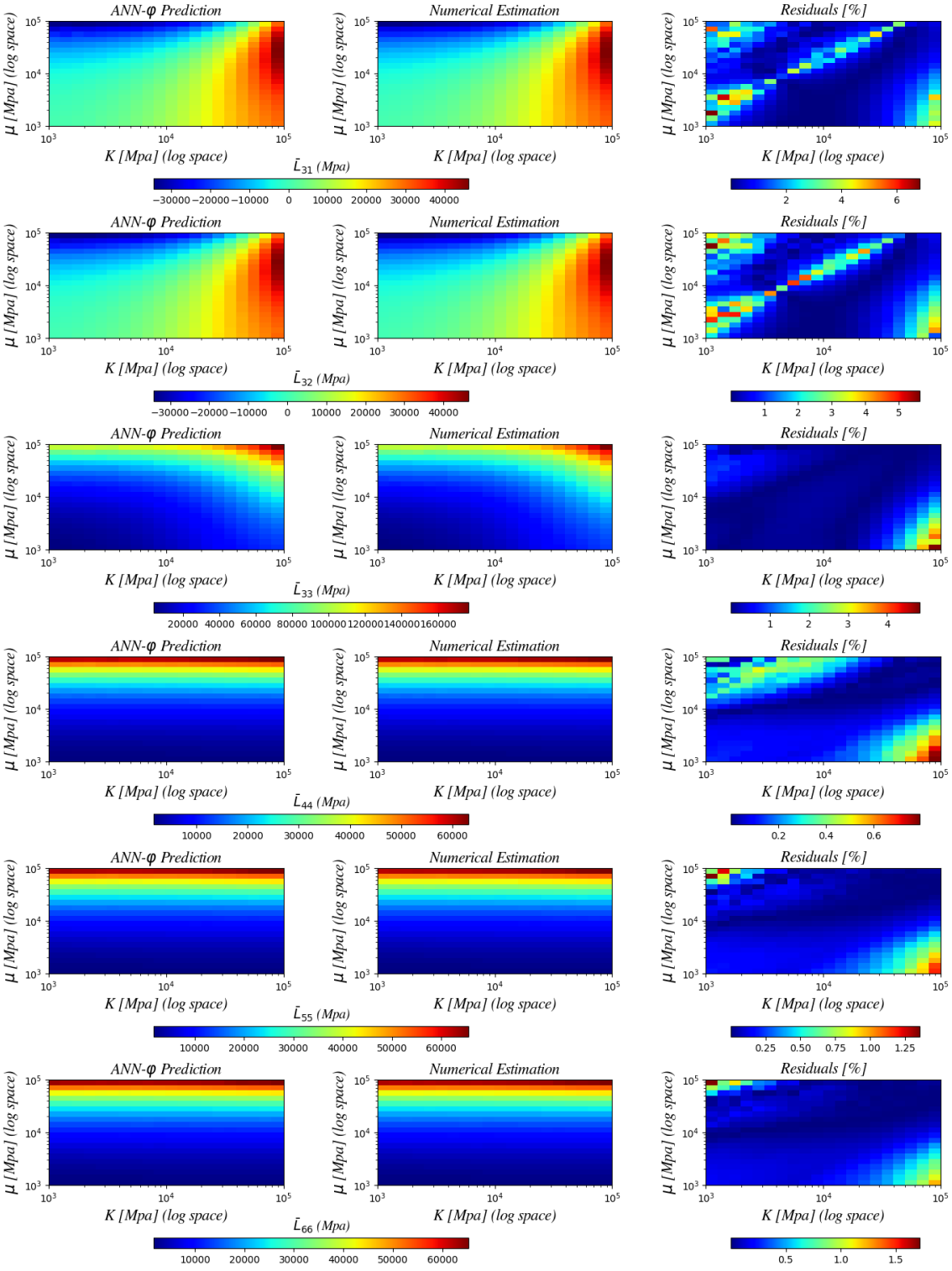


Figure 3.16: Comparison between ANN-φ predicted effective stiffness tensor components and its numerical estimation for different matrix mechanical properties

Table 3.4: Offline computational time

Offline computations	**ANN Training Time**	**Database generation**
Computational time	17 min	9.72 h

One of the main objectives of this study is computational time saving when computing the effective properties of heterogeneous materials with many different phases. For both ANN-φ and numerical estimations, figure 3.17 shows the evolution of required online computational time as a function of number of inhomogeneities in the material. It can be observed that an increase in the number of material phases N leads to significant computational times with a pure numerical approach. As for ANN-φ, running time of the proposed model is almost insensitive to the number of inhomogeneities, resulting in a online computational time savings by a factor of nearly 2000 for 10^4 phases in the heterogeneous material as shown in figure 3.18. Thus, the ANN-φ model is efficient in terms of computational time savings as well as its predictive capabilities. As for offline computations which include ANN training time and the generation of 1 million samples, the results are presented in Table 3.4. It can be seen that the total offline computational time is nearly 10 hours split between ANN training time and Database generation. At the beginning, it is very common that databases creation and model's training would take a significant amount of time for the majority of deep learning problems. Once this phase is completed, the goal is to use those models to quickly predict several quantities of interest. For instance, we recall that ANN-φ is of utmost interest when parametric study of the microstructure (by varying ellipsoids geometry) or medium mechanical properties are performed to predict the effective properties of the heterogeneous material.

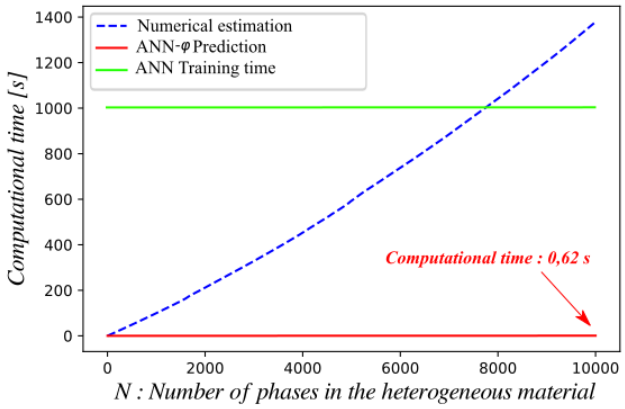


Figure 3.17: Runtime comparison between ANN-φ model and numerical estimations

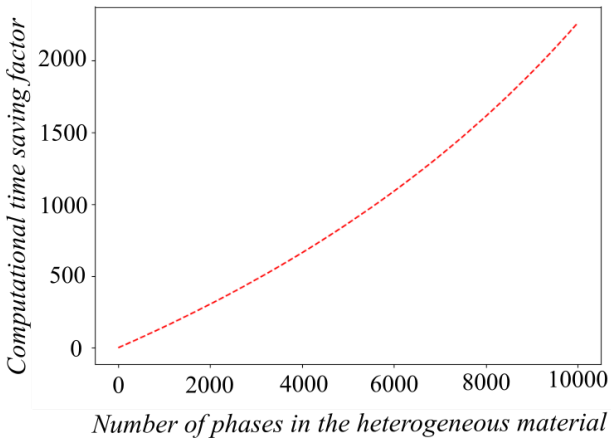


Figure 3.18: Online computational time saving factor as a function of number of phases

3.4.4 Comparison between ANN $-\varphi$ and ANN$_2-\varphi$

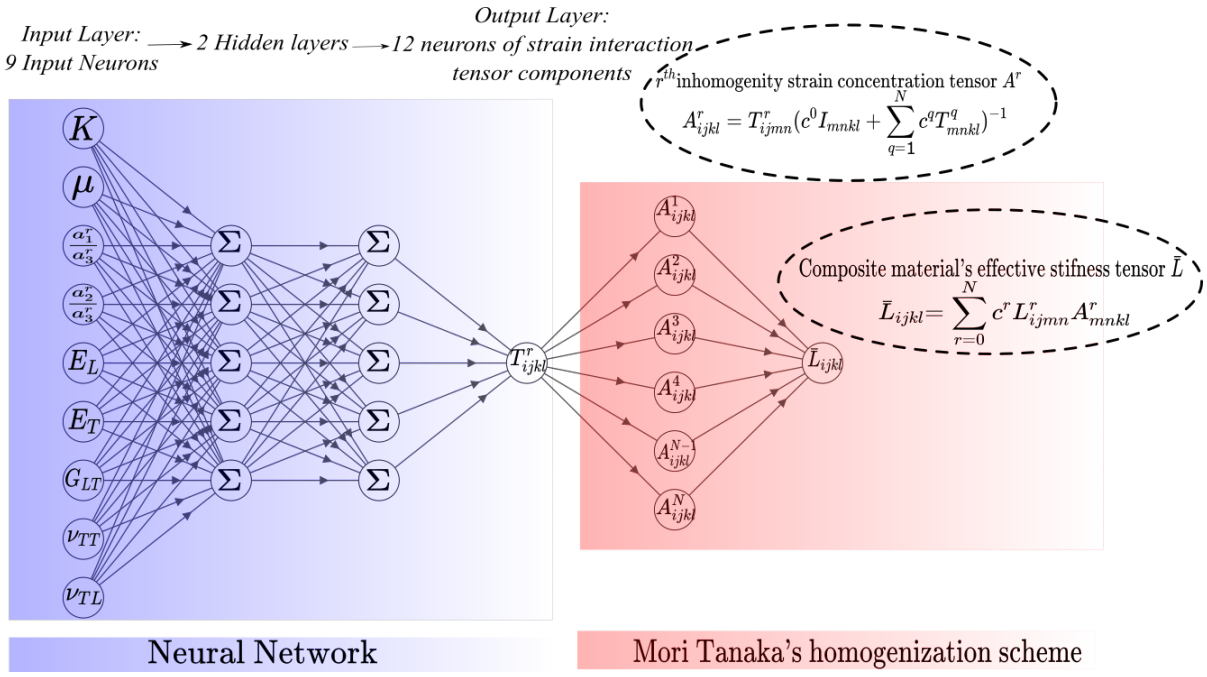


Figure 3.19: Architecture of ANN$_2-\varphi$: A hybrid model combining Artificial Neural Networks and micromechanical homogenization using *Mori-Tanaka* scheme

It has been noticed in section 3.4.3 that the relative error between ANN $-\varphi$ predicted values and numerical estimations could reach values up to 10% for some configurations of heterogeneous materials. Even if Neural Network model is very accurate in terms of predicting Eshelby's tensors components, propagation of errors can occur during the computation of the effective tensor $\bar{L}$. We recall that $\bar{L}$ depends on strain interaction tensor T. The latter is computed using the predicted Eshelby's tensor (See equation (3.8)) which involves some inverse operations that can be the source of error propagation. In this section, the aim is to investigate the possibility of reducing the propagation of errors by training a Neural Network directly on a strain interaction tensors Database T instead of Eshelby's tensors S. For this purpose, a new hybrid model ANN$_2-\varphi$ is introduced as shown in figure 3.19. ANN$_2-\varphi$ input parameters are mechanical properties of isotropic matrix (K, μ), ellipsoidal inclusions geometric ratios $(a_1^r/a_3^r, a_2^r/a_3^r)$ and inhomogeneities transversely isotropic material properties $(E_L, E_T, G_{LT}, \nu_{TT}, \nu_{TL})$. E_L and E_T are longitudinal and transverse Young Modulus respectively, ν_{TT}, ν_{TL} are Poisson's ratios for loading along the longitudinal axis and transverse axis respectively and G_{LT} is shear modulus. Note that ANN$_2-\varphi$ requires more input parameters, which substantially could complexify the training process. ANN$_2-\varphi$ model is trained on a strain interaction tensor database with 1 million samples. An hyperparameter analysis has also been conducted for ANN$_2-\varphi$ model and it has been found that the couple of parameters that minimize the validation error are an hyperbolic tangent for the activation function and a number of hidden neurons fixed by [Ke & Liu 2008] which are the same couple of hyperparameters used to train ANN $-\varphi$.

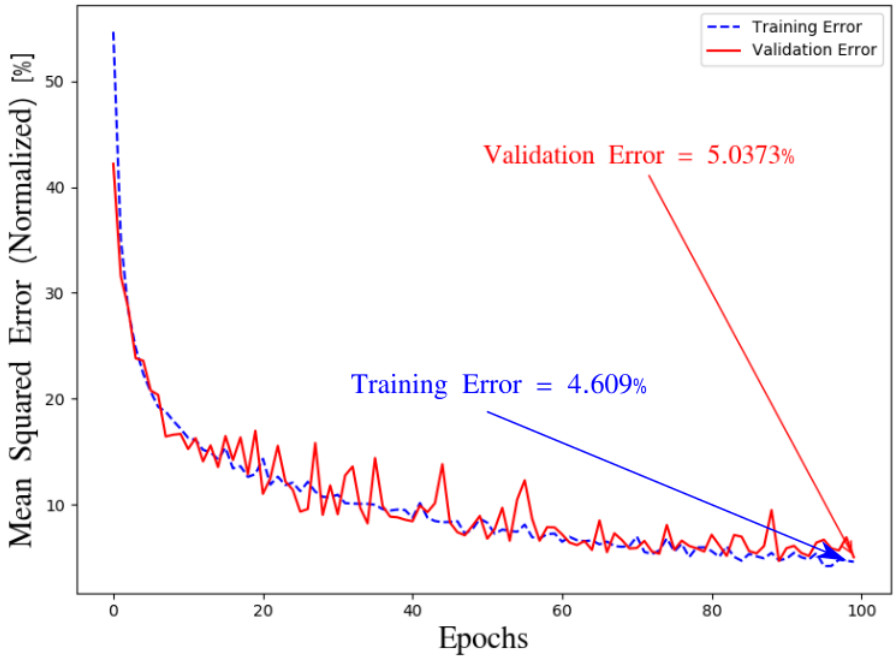


Figure 3.20: Evolution of training and validation errors as a function of Epochs during $\text{ANN}_2 - \varphi$ training

Figure 3.20 describes the evolution of training and validation NMSE during training phase. It is observed that the obtained NMSE for training and validation data during the last Epoch are 4.6% and 5.03% respectively, which are higher values compared to the $\text{ANN} - \varphi$ model where training and validation errors were of the order of $10^{-2}\%$ and $10^{-3}\%$ respectively. After training phase, the model is evaluated on the data set to check its prediction reliability. Figure 3.21 shows the correlation plots between ANN predicted values and target values. For the majority of strain interaction tensor components, it can be seen that the ANN predicted values are very well correlated with targets values. However, some important deviations from the expected result can be noticed for some values of T_{31},T_{32} and T_{33}. It is expected that these discrepancies may impact the reliability of the entire hybrid model $\text{ANN}_2 - \varphi$ as propagation of errors is more likely to occur in this case when computing the effective behavior of heterogeneous materials.

To assess the predictive capability of the entire model, $\text{ANN}_2 - \varphi$ is used to predict the effective stiffness tensor for a Random Oriented Composite material similar to the one studied in subsection 3.4.2. Figure 3.22 shows a comparison between composite effective properties obtained by $\text{ANN} - \varphi$, $\text{ANN}_2 - \varphi$ models and by numerical estimation of $\bar{L}$. We can notice major differences between predicted and numerical estimated values. As described in Table 3.5, Relative Errors (RE) for the significant terms of $\bar{L}$ reached values up to 40% contrary to the obtained results with $\text{ANN} - \varphi$ where RE did not exceed 4%.

This investigation showed that training Neural Network on a strain interaction tensors database do

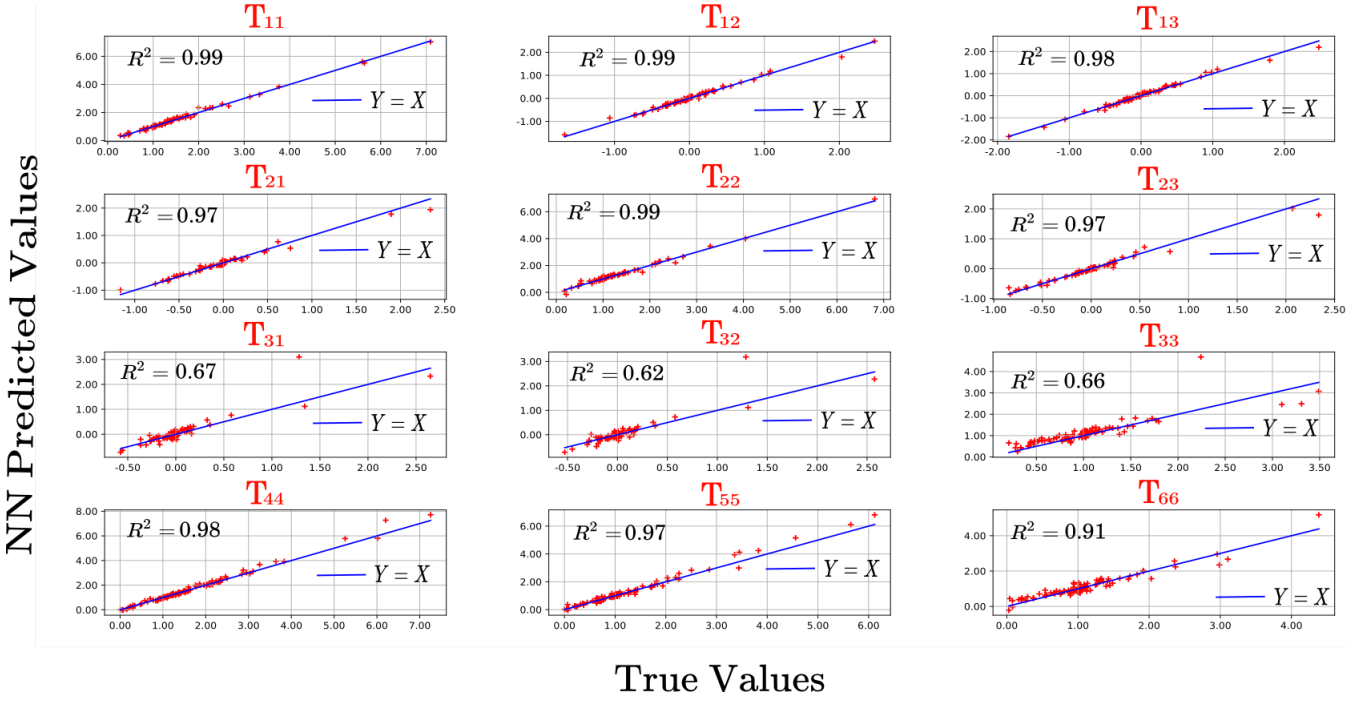


Figure 3.21: Correlation plots between ANN predicted values and target values for each strain interaction tensor test component

not necessarily improve hybrid model performance and will conversely decrease the model accuracy and its predictive capability. Several assumptions can be made to explain this results. First is the number of input parameters of $\text{ANN}_2-\varphi$ is larger than the one in $\text{ANN}-\varphi$ model. We recall that $\text{ANN}_2-\varphi$ depends on the geometry of inclusions and both mechanical properties of matrix and inhomogenities (9 parameters for an isotropic matrix and transversely isotropic inhomogenities), whereas $\text{ANN}-\varphi$ model trained on Eshelby's tensor database depends only on matrix mechanical properties and ellipsoids geometry (3 parameters for an isotropic matrix). Solving this problem in a 9 dimensional space is obviously more challenging compared to a three dimensional space, and therefore a significant amount of data may be required to improve the Neural Network reliability. The second assumption can be the behavior complexity of the strain interaction tensor which depends on one hand on the inclusions geometry and matrix material properties (Non linear term $S(L^0)^{-1}$ in equation 3.8) and on the other hand on the contrast between the two materials defined by the term $L - L^0$ in equation 3.8.

This investigation also demonstrated that $\text{ANN} - \varphi$ model trained on Eshelby's tensors database is the most suitable for a reliable prediction of the effective behavior of heterogeneous materials. Another advantage of the $\text{ANN}-\varphi$ model is that inclusions properties are not fixed as Eshelby's tensor only depends on matrix mechanical properties, which gives freedom to evaluate the effective behavior of heterogeneous materials with several inhomogenities properties from isotropic to anisotropic cases. Note that $\text{ANN}_2-\varphi$ model is nearly 8 times faster than $\text{ANN}-\varphi$, but the obtained residual error RE_2 values (See Table 3.5) renders the $\text{ANN}_2-\varphi$ model not suitable for practical engineers design applications.

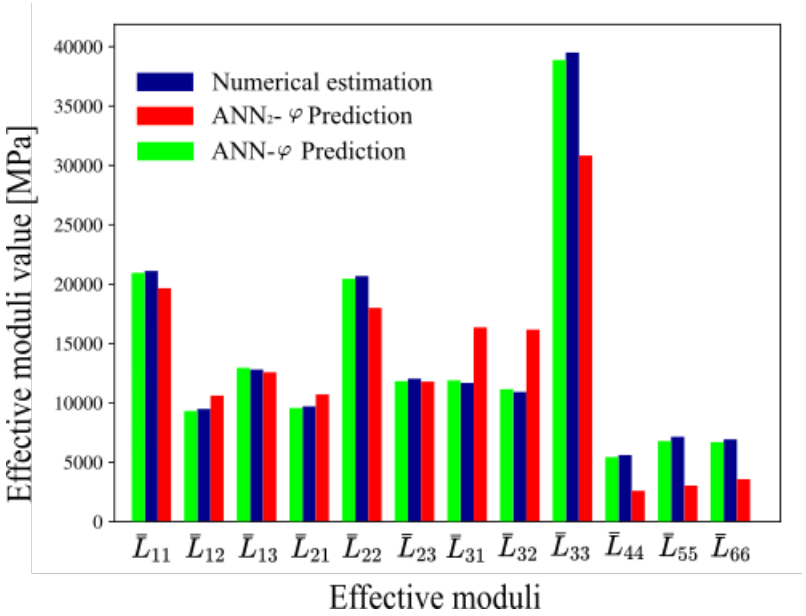


Figure 3.22: Comparison between $\text{ANN}_2-\varphi$ predicted effective stiffness tensor $\bar{L}$ and numerical estimation of $\bar{L}$

Table 3.5: Comparison between $\text{ANN}_2-\varphi$, $\text{ANN}-\varphi$ predicted effective stiffness tensors $\bar{L}_{ij}^{NN_2}$ and $\bar{L}_{ij}^{NN}$ respectively and numerical estimation of $\bar{L}$ $\bar{L}_{ij}^{Num}$

Effective moduli	$\bar{L}_{ij}^{NN_2}$	$\bar{L}_{ij}^{NN}$	$\bar{L}_{ij}^{Num}$	$\mathbf{RE_1} = \left\lvert\frac{\bar{L}_{ij}^{NN}-\bar{L}_{ij}^{Num}}{\bar{L}_{ij}^{Num}}\right\rvert$	$\mathbf{RE_2} = \left\lvert\frac{\bar{L}_{ij}^{NN_2}-\bar{L}_{ij}^{Num}}{\bar{L}_{ij}^{Num}}\right\rvert$
$\bar{L}_{11}$ (GPa)	19.29	21.51	21.82	1.42 %	11.56 %
$\bar{L}_{12}$ (GPa)	11.75	9.71	9.90	1.91%	18.73 %
$\bar{L}_{13}$ (GPa)	12.79	12.73	12.55	1.43 %	1.86 %
$\bar{L}_{21}$ (GPa)	11.47	9.87	9.95	0.80 %	15.26 %
$\bar{L}_{22}$ (GPa)	18.57	20.31	20.54	1.11 %	9.58 %
$\bar{L}_{23}$ (GPa)	12.35	12.38	12.41	0.24 %	0.49 %
$\bar{L}_{31}$ (GPa)	16.71	12.21	12.03	1.49 %	38.89 %
$\bar{L}_{32}$ (GPa)	16.63	12.03	11.78	2.12 %	41.19 %
$\bar{L}_{33}$ (GPa)	30.18	38.89	39.69	1.86 %	23.95 %
$\bar{L}_{44}$ (GPa)	3.78	5.76	5.98	3.67 %	36.78 %
$\bar{L}_{55}$ (GPa)	4.14	6.71	6.84	1.90 %	39.51 %
$\bar{L}_{66}$ (GPa)	4.10	6.66	6.79	1.91 %	39.54%

3.5 Conclusions

In this study, the predictive capability of hybrid models for homogenization purposes is investigated. Combining Artificial Neural Networks and micromechanical modeling, a hybrid model is proposed to predict effective properties of heterogeneous materials. The ANN$-\varphi$ model showed an excellent predictive capability of various quantities related to homogenization such as the effective behavior of composite structures or local stresses in different heterogeneous material phases. The robustness of this model comes primarily from Neural Network's reliability when predicting Eshelby's tensors S. Throughout this study, the obtained results with ANN$-\varphi$ model are compared with numerical estimations which are often costly in computational time. The use of a hybrid approach in this work provided a good compromise between model's reliability and computational time saving which estimated by a factor that can reach 2000 in the case of a very large number of inhomogeneities in the heterogeneous material (10^4). However, for most of practical applications, the computational time for even non optimized integration method is much faster if the ANN training process is taken into account. It is further shown that the choice of the physical quantity to be predicted by the ANN model should be selected wisely. Indeed, the avoidance of i) a large number of Neural Network's input parameters; ii) the prediction of a strongly non-linear response and iii) the presence of strong contrast in output data should always be considered. The usefulness of hybrid approaches combining ANN and micromechanical based modeling is here attested as a potential efficient and reliable tool to deal with homogenization problems. If the current investigations are performed on elastic materials without considering any mechanical properties change over time, such method would find very interesting applications in the numerical investigation of damage of heterogeneous materials. For instance, the computation of the stiffness reduction within the matrix that could be due to the appearance of cracks within the matrix could are often represented by penny-shape ellipsoid with an aspect ratio that may evolve within time, as it has been shown that the crack shape and orientation influences the stiffness tensor components [Praud *et al.* 2017]. Hybrid AI-physics micromechanical models would find interesting applications in downscaling multiscale simulation of Fiber reinforced Composites, especially considered injected or molded parts where the microstructural arrangement of reinforcements would differ depending on the position. A straightforward application is the determination of fatigue damage criteria of composite structures based on local stress criteria (e.g. the interfacial stress between the reinforcements and the matrix) considering a complex loading case of a structural part simulated using Finite Element Simulation. Such methodology has shown strong prediction capabilities [Bidaine *et al.* 2015], but requires an intense computational time. The proposed approach would allow to drastically reduce the computational time in such situations. Perspectives of the proposed work include the investigation of the benefits of such methodology in the two presented applications.

This first study performed in a context of heterogeneous materials homogenization constitutes a proof of concept of the employment of hybrid methods for multi-scale modeling. Inspired by the approach adopted in this chapter, where micromechanical models were combined with multilayer perceptron neural networks to estimate homogenized properties with mean-field methods, the next step of this work proposes to extend this methodology to address more complex problems, thus pushing further the achievable performances of such approaches. Therefore, the next chapter of this manuscript aims to combine Recurrent Neural Networks with full-field methods such as FE^2 to accelerate numerical simulations of non-linear heterogeneous materials and more specifically of architectural materials under complex loading conditions.

Chapter 4

Accelerating Multiscale simulations of architectured materials using Deep Neural Networks

Contents

4.1 Introduction

The new industrial constraints are nowadays pushing the limits of materials in terms of mechanical properties for advanced application fields (renewable energy, healthcare, transport). One solution to adapt the material properties to specific application, is to rely on architectured structures that can be manufactured from additive process. The design of such structures can be challenging given the complexity of material mechanical responses when multiple physical mechanisms are involved at different length scales. Consequently, the evaluation of the resulting mechanical behaviors requires powerful numerical tools capable of solving highly non-linear multiscale problems. Among those computational models, the Finite Element and FFT methods have been extensively developed in the literature for multiscale structural analysis. However, it is very common that the numerical simulations with finite element analysis requires high computational costs especially when the microstructure is taken into account and non-linear material behaviors are occurring. To overcome this challenge, multilevel homogenization schemes such as the FE^2 method developed by Feyel [Feyel 1999, Feyel 2003] are commonly used. The FE^2 method, based on periodic homogenization theory, consists on associating a unit cell representing the Representative Volume Element (RVE) to each Gauss integration point. The estimation of the overall behavior at the macroscale is then performed by finite elements computations at the microscale. This multiscale approach has been used for instance by Tikarrouchine et al [Tikarrouchine *et al.* 2018, Tikarrouchine *et al.* 2019] to determine the macroscopic mechanical response of composite materials exhibiting non linear elasto-viscoplastic and thermo-viscoplastic behaviors. Although the FE^2 method can contributes to reduce the required computational time compared to direct numerical simulations, the employment of such modeling strategy remains in most cases computationally expensive due to the treatment of material non-linearities such as plasticity and damage.

In the present chapter, we propose to develop a numerical modeling strategy using deep neural networks in order to overcome the aforementioned restrictions and difficulties related to multiscale modeling of heterogeneous materials. Based on the previous chapters of this manuscript, machine learning approaches and more specifically artificial neural networks have demonstrated their strong capabilities as powerful and efficient surrogates capable of capturing material highly non-linear mechanisms as well as drastically reducing simulation time costs. Therefore, the main objective of this study would be to extend and adapt ANN based methods for multiscale simulations of non-linear architectured materials. Figure 4.1 describe the overall modeling strategy for multiscale simulations of architectured structures using deep neural networks. First, a training phase on architectured mesostructures is performed to predict the homogenized mechanical responses at the meso scale. A variety of parameters can be considered as model's inputs such as the geometry of the architectured material, mechanical properties or loading history. Then, a mesostructure is associated to each macroscopic material point (or Gauss integration point) whose characteristics may vary depending on the inputs mentioned above. The ANN based model is then applied to predict the homogenized response of each unit cell which is used afterwards to obtain the macroscopic behavior of the architectured structure in a rather fast way, this process is repeated until the global equilibrium of the structure is reached. This approach can be considered as a new paradigm of heterogeneous materials multiscale simulation where the resolution of the micro problems with traditional methods is no longer required but replaced by deep neural networks, thus resulting in a significant computational cost savings in terms of execution time and computing resources (memory and CPU).

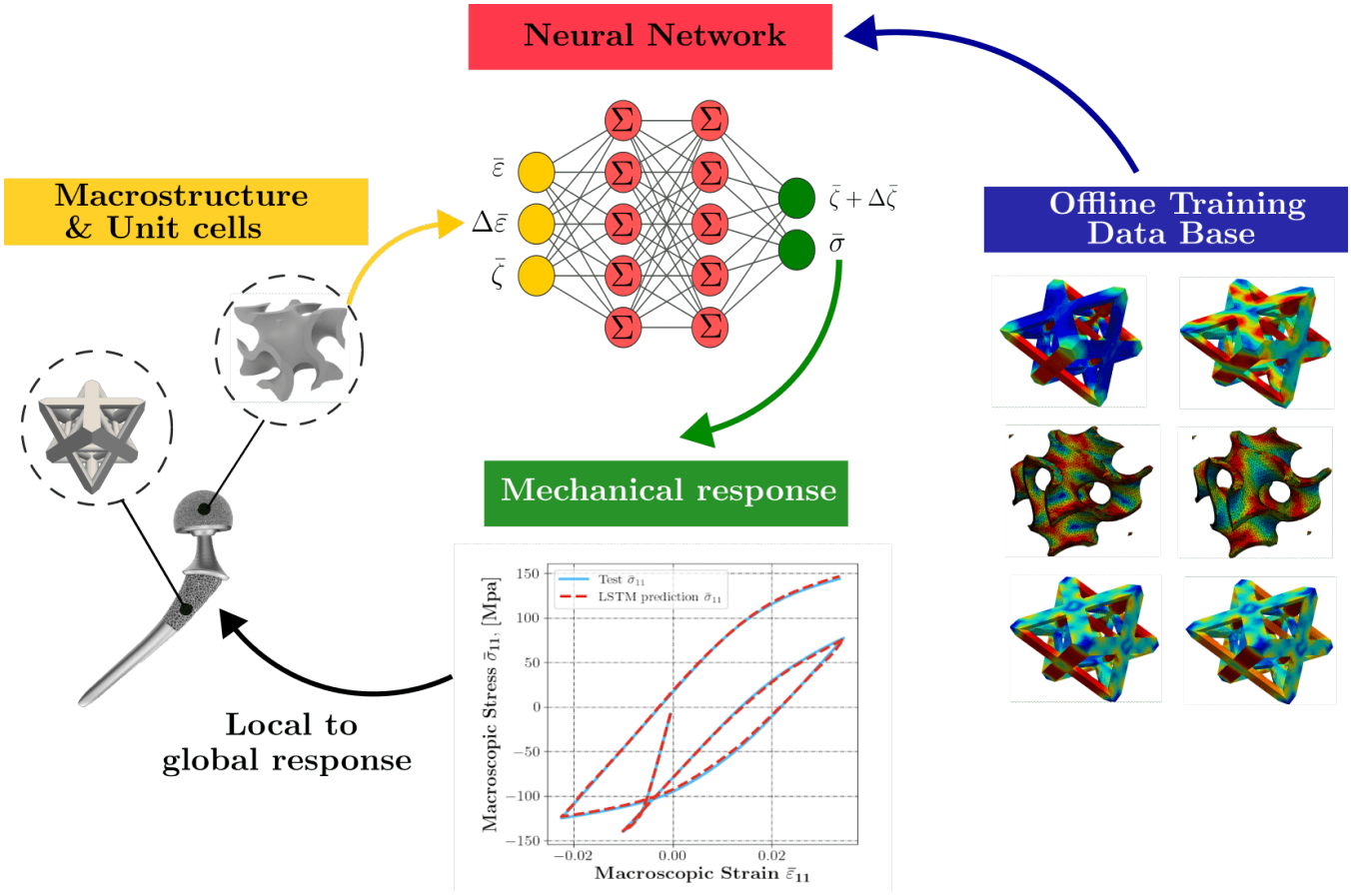


Figure 4.1: Multiscale modeling strategy of architectured structures using deep neural networks

While the general idea behind the incorporation of AI based approaches in multiscale frameworks is described in figure 4.1, specific considerations need to be addressed regarding the choice of the most suitable neural network architectures. According to the study conducted in chapter 2, the robustness of Recurrent Neural Networks has been highlighted as a reliable surrogate capable of handling path dependant behaviors. While chapter 2 was only focused on modeling non-linear mechanical behaviors of material points using RNN based models, the present chapter proposes to extend this reasoning to architectured structures under complex loading conditions (multi-axial and non proportional loading paths). As presented in the state of art chapter, some contributions based on multi layer perceptrons architectures have also been applied in a computational homogenization framework to accelerate multiscale simulations [Le *et al.* 2015, Lu *et al.* 2018, Minh Nguyen-Thanh *et al.* 2020]. However, we recall that aforementioned studies were limited to non-dissipative materials under proportional loading paths, thus leading to a significant simplification of the problem. The present work differs from the previous studies in terms of the employed ANN architecture where RNN are used instead of MLP to take into account the loading history of dissipative architectured structures.

The fourth chapter of this manuscript is structured as follows: In section 4.2, a brief introduction to the design, properties and application fields of architectured materials is presented. Then, the theoretical

framework of periodic media computational homogenization and the basic principle of the multilevel finite element method (FE^2) are recalled. The main objective of section 4.2 is to establish the theoretical foundations behind multiscale modeling approaches whose principles constitute the core of computational homogenization even when combined with deep neural networks. Section 4.3 presents the design of FE-LSTM: an approach that combines the FE method and LSTM neural networks for architectured structures multiscale modeling. The entire workflow of FE-LSTM model from database generation, hyperparameters selection, training to validation is detailed in this section. Afterwards, the application of FE-LSTM model to the multiscale analysis of architectured structures in real life condition is illustrated in section 4.4. Several examples of 3D architectured structures are considered and the obtained results with FE-LSTM model are compared with the FE^2 approach in terms of accuracy and computational costs savings (execution time, required memory, CPU usage). In section 4.5, we present a first attempt of using multi-inputs neural networks architectures combining convolutional neural networks and multilayer perceptrons to predict the effective properties of architectured materials. This investigation provide a first insight of the achievable capabilities of multi-inputs neural networks that takes into account data of different natures (images, continuous numerical data, discrete numerical data). Finally, some concluding remarks are given in section 4.6.

4.2 Theoretical framework of computational homogenization

4.2.1 Architectured materials - A brief introduction

According to the definition of *Michael Ashby* [Ashby 2013], the term *architectured* materials refers to the association of a multitude of materials, or of a unique material and space (geometry), designed in a smart way to possess attributes that cannot be achieved by the constituents alone. This hybridization process allows to expand materials property spaces by filling the empty gaps in Ashby plots [Ashby 1999]. As shown from figure 4.2 of Young's modulus - density space chart, filling the "holes" areas enables to design lightweight structures with high mechanical performances, therefore pushing the limits of using multifunctional materials for advanced industrial application while minimizing production costs. Architectured materials encompasses particulate and fibrous composites, woven and sandwich structures, foams and architected cellular materials including strut-based lattices and Triply Periodic Minimal Surfaces (TPMS) based lattices. Figure 4.3 shows some examples of architectured cellular materials such as octet-truss, hexagonal honeycomb lattices, TPMS gyroid and spherical gyroid. Architectured materials can also be categorized according to their mode of spatial organization. The associated microstructures are divided into two major categories: periodic and stochastic. Periodic microstructures are defined by a regular unit cell that is reproduced in a two-dimensional or three-dimensional spaces. Periodic architectured structures such as octet truss lattices structures are represented by the dimensions and shapes of their elementary cells. Stochastic microstructures such as foams are designed with random cell shapes that cannot be characterized by a generic pattern, but by a statistical averages of these shapes. In summary, the use of architectured materials in structures design allows to overcome the challenges related to mass minimization and production costs optimization while ensuring very high levels of requirements in terms of mechanical performance. They are particularly useful for the creation of new types of materials with interesting properties such as auxeticity [Dirrenberger *et al.* 2011] or for designing structures where the requirements in terms of mechanical properties and stiffness have already been achieved but with less material. Therefore, the development of this type of technology is crucial and essential for the industry

of the future.

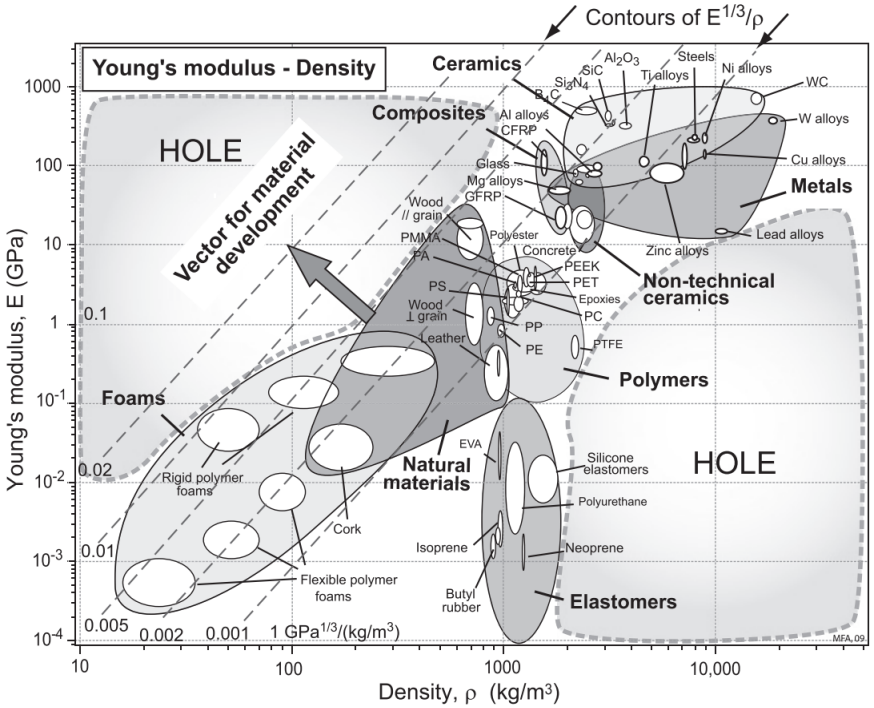


Figure 4.2: Ashby's material selection chart of Young's modulus - density [Ashby 2013]

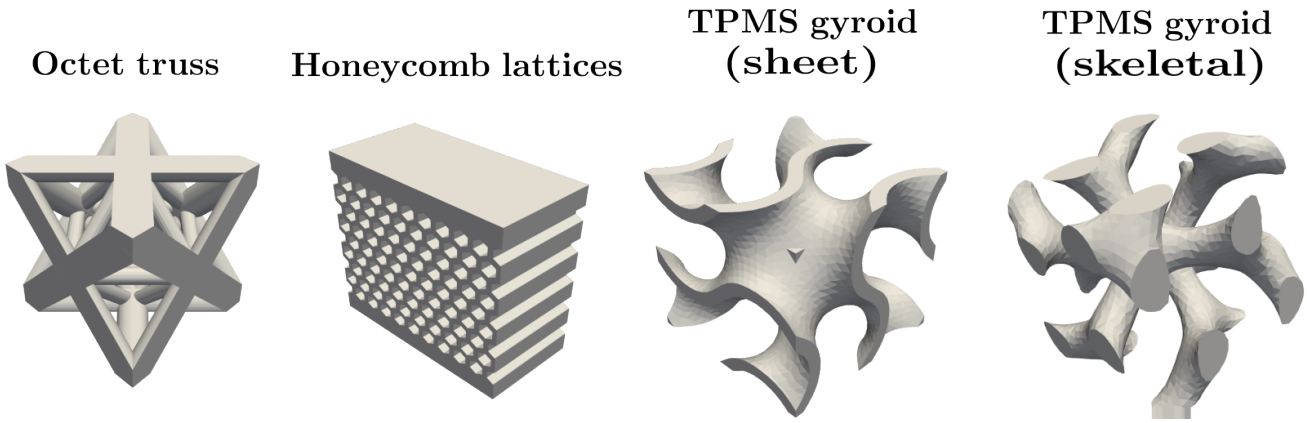


Figure 4.3: Some examples of cellular architectured materials including strut-based lattices and TPMS based lattices. Source : https://github.com/3MAH/microgen

Architectured materials are currently used in various fields of application, particularly in the automotive, aeronautics and biomechanics industries. In the field of transportation, one of the main objectives would be a to decrease part's mass in order to reduce energy consumption. By optimizing specific performances, the use of architectured materials has also a favorable environmental impact in terms of energy efficiency induced by the lightening of structures. Another interesting feature of certain types of materials, such as foams, is their ability to absorb energy during collisions and to modulate vibrations amplitudes.

In biomechanics, the current solutions to design bone prosthesis are based on metal alloys such as titanium. These alloys exhibit excellent corrosion and fatigue resistance capabilities as well as a good strength/weight ratio. However, clinical experiments have shown interface problems. In fact, the contact between the bone and the prosthesis is affected by a significant local stress gradient due to the incompatibility of their mechanical responses. This phenomenon induces a reabsorption in the bone, which causes the failure of the prosthesis. Lattice-type architectured materials are an interesting alternative of this type of application [Coelho *et al.* 2011, Fernandes *et al.* 2012]. They provide the appropriate porosity used for biological functions such as the transport of nutriments, mechanical strength of bone tissue and mechanical biocompatibility. Therefore, prostheses designed with gradient based properties architectured materials (Figure 4.4) ensure prosthesis stability and compatibility with bone tissues.

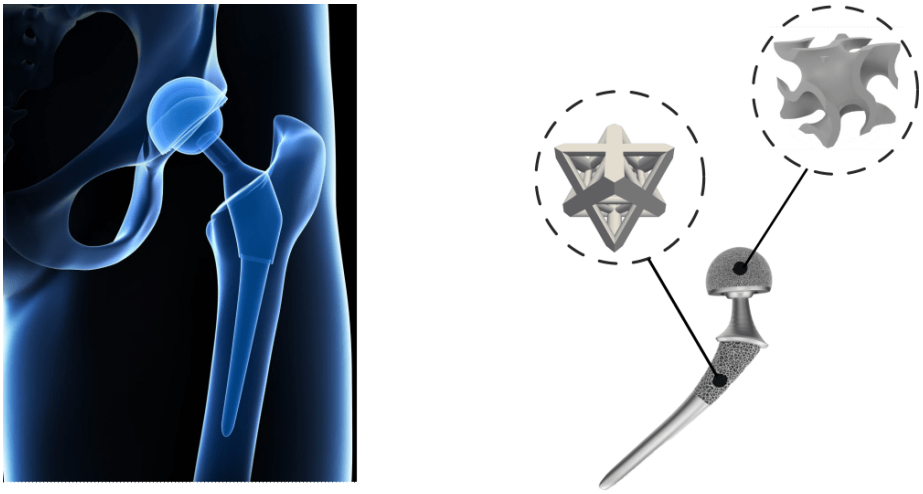


Figure 4.4: An illustration of a hip implant with a gradient based properties architectured structure. Source: https://www.orlandoortho.com/anterior-approach-total-hip-replacement-surgery-2/

4.2.2 Periodic homogenization for heterogeneous media

Homogenization techniques consists in identifying a global mechanical behavior at the macroscopic scale *i.e* the effective mechanical response given the microstructure characteristics (mechanical properties, geometry, size, orientation, volume fraction, constitutive laws). As shown in figure 4.5, the objective is to define an equivalent homogeneous medium having an identical average mechanical response (stress and strain fields) to the heterogeneous structure.

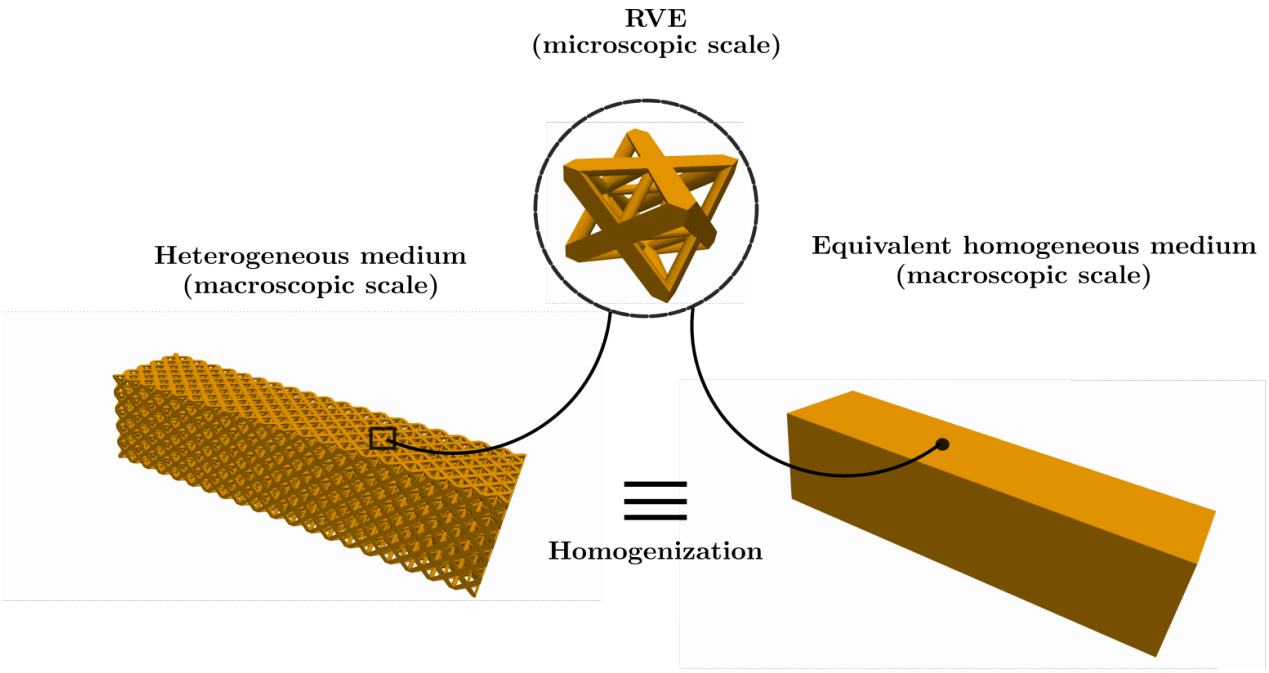


Figure 4.5: Schematic representation of the homogenization principles

An heterogeneous medium is considered triply periodic when it is defined by a repeating unit cell representative of the microstructure in all spatial directions. The theory of periodic homogenization is valid provided that scale separation conditions between the microscale and macroscale are verified. At the microscale, an RVE can be defined as a material volume sufficiently large enough to be statistically representative to the microstructure. At the macroscale, the structure is considered as a virtual homogeneous medium where each macroscopic material point can be associated to a RVE. The assumption of scale separation is valid as long as the characteristic size of the unit cell (microstructure) is several orders of magnitude smaller than the dimensions of the macrostructure.

As defined by [Hill 1967], the connection between the macroscopic quantities $(\bar{\sigma}, \bar{\varepsilon})$ and their microscopic counterparts (σ, ε) is obtained through volume averaging over the RVE Ω of volume V:

$$\bar{\sigma} = \langle \boldsymbol{\sigma} \rangle = \frac{1}{V} \int_{\Omega} \sigma \mathrm{d}V, \tag{4.1}$$

$$\bar{\varepsilon} = \langle \boldsymbol{\varepsilon} \rangle = \frac{1}{V} \int_{\Omega} \varepsilon \mathrm{d}V. \tag{4.2}$$

Under small strain theory assumptions, average stress and average strain theorems state that the stress and strain averages within the RVE are equal to the applied uniform traction and linear displacement on its boundary respectively. These theorems can be proven using the divergence theorem.

The average stress theorem can be stated as follows: Considering that a traction vector $\boldsymbol{t}(\boldsymbol{x}) = \boldsymbol{\sigma}^{\mathbf{0}} \cdot \boldsymbol{n}(\boldsymbol{x})$ is applied at each material point $\boldsymbol{x}$ of the RVE boundary $\partial\Omega$, with $\boldsymbol{n}(\boldsymbol{x})$ the outward normal vector to

$\partial\Omega$, then the volume average of stress inside the RVE is given by:

$$\frac{1}{V}\int_{\Omega}\sigma \mathrm{d}V = \boldsymbol{\sigma^0} \tag{4.3}$$

A stress field σ satisfying this condition is defined as "statically admissible" [Nguyen 1988].

The average strain theorem can be stated as follows: Considering that a linear displacement vector $\boldsymbol{u(x)} = \boldsymbol{\varepsilon^0} \cdot \boldsymbol{x}$ is applied at each material point $\boldsymbol{x}$ of the RVE boundary $\partial\Omega$, then the volume average of strain inside the RVE is given by:

$$\frac{1}{V}\int_{\Omega}\varepsilon \mathrm{d}V = \boldsymbol{\varepsilon^0} \tag{4.4}$$

A strain field ε satisfying this condition is defined as "kinematically admissible" [Nguyen 1988].

Assuming that the boundary $\partial\Omega$ of a periodic RVE is subjected to periodicity conditions, meaning that the displacement field $\boldsymbol{u}(\boldsymbol{x})$ of any material point with a position vector $\boldsymbol{x}$ can be expressed by an affine part $\bar{\boldsymbol{\varepsilon}} \cdot \boldsymbol{x}$ and a periodic fluctuating displacement field $\tilde{\boldsymbol{u}}$:

$$\boldsymbol{u}(\boldsymbol{x}) = \bar{\boldsymbol{\varepsilon}} \cdot \boldsymbol{x} + \tilde{\boldsymbol{u}}. \tag{4.5}$$

The periodic fluctuating quantity $\tilde{\boldsymbol{u}}$ remains the same for each pair of opposite nodes on the RVE boundary. As shown in figure 4.6, the displacement field of two opposite nodes i and j with a position vector $\boldsymbol{x}^+$ and $\boldsymbol{x}^-$ respectively is expressed as:

$$\boldsymbol{u}(\boldsymbol{x}^+) = \bar{\boldsymbol{\varepsilon}} \cdot \boldsymbol{x}^+ + \tilde{\boldsymbol{u}} \tag{4.6}$$

$$\boldsymbol{u}(\boldsymbol{x}^-) = \bar{\boldsymbol{\varepsilon}} \cdot \boldsymbol{x}^- + \tilde{\boldsymbol{u}} \tag{4.7}$$

In that case, each pair of opposite nodes are linked by the following kinematic equation [Suquet 1987]:

$$\boldsymbol{u}(\boldsymbol{x}^+) - \boldsymbol{u}(\boldsymbol{x}^-) = \nabla\boldsymbol{u} \cdot \boldsymbol{\Delta} x, \tag{4.8}$$

which reduces to Eq. 4.9 under small strains:

$$\boldsymbol{u}(\boldsymbol{x}^+) - \boldsymbol{u}(\boldsymbol{x}^-) = \bar{\boldsymbol{\varepsilon}} \cdot (\boldsymbol{x}^+ - \boldsymbol{x}^-), \tag{4.9}$$

Given the aforementioned periodicity conditions, the volume average of the strain inside the RVE is expressed by:

$$\frac{1}{V}\int_{\Omega}\varepsilon \mathrm{d}V = \bar{\boldsymbol{\varepsilon}} \tag{4.10}$$

The mechanical energy equivalence between the microscopic and the macroscopic scales is obtained by Hill-Mandel lemma [Hill 1967]. Considering a kinematically admissible strain $\boldsymbol{\varepsilon}$ and statically admissible stress $\boldsymbol{\sigma}$, this fundamental theorem states that micro-macro energy equivalence

$$\langle \boldsymbol{\sigma} : \varepsilon \rangle = \langle \boldsymbol{\sigma} \rangle : \langle \varepsilon \rangle = \bar{\sigma} : \bar{\varepsilon}, \tag{4.11}$$

is satisfied as long as one of the following conditions are verified on the boundary $\partial\Omega$ of the RVE: uniform tractions, linear displacements and periodicity conditions.

Hill-Mandel lemma allows to properly define an "equivalent homogeneous medium" with an effective stiffness $\bar{\boldsymbol{C}} = \langle C \rangle$ as expressed by Eq.4.12 which uses Eq.4.11:

$$\langle \boldsymbol{\varepsilon} : C : \varepsilon \rangle = \langle \boldsymbol{\varepsilon} \rangle : \langle C \rangle : \langle \varepsilon \rangle = \bar{\varepsilon} : \bar{\boldsymbol{C}} : \bar{\varepsilon} \tag{4.12}$$

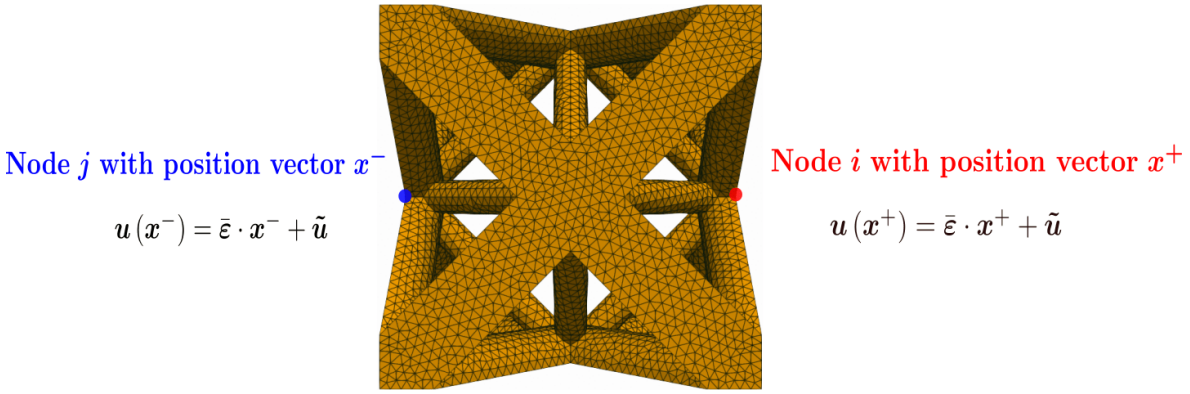
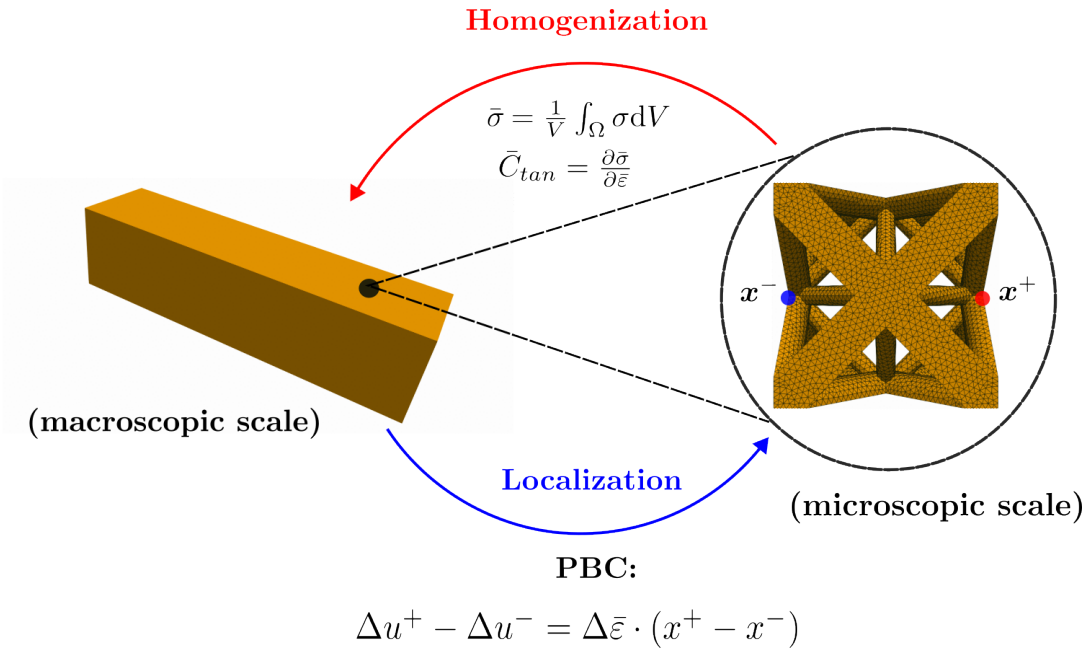


Figure 4.6: Schematic representation of two opposite nodes at the boundary of an octet-truss RVE

In summary, the determination of the effective behavior of an heterogeneous material with periodic homogenization techniques relies mainly on two steps: a localization step where adapted boundary conditions (especially periodic conditions) are applied on the RVE and an homogenization step where the macroscopic response is obtained through volume averaging. These two steps repeated at each time step allows to establish the constitutive relationship between the macroscopic stress $\bar{\sigma}$ and the macroscopic strain $\bar{\varepsilon}$. This procedure will be detailed in the next section in the context of the multilevel finite element method FE^2.

4.2.3 Multilevel finite element method (FE^2)

To compute the effective non-linear response of heterogeneous structures while taking into account microstructures mechanical behavior, multilevel homogenization schemes such as the FE^2 method is commonly applied [Feyel 1999, Feyel 2003]. This approach, based on periodic homogenization theory, consists on solving the micro and macro problems simultaneously through localization and homogenization principles. The governing micro macro problem set of equations and scale coupling are summarized in 4.1 and 4.2 respectively. As shown in figure 4.7, first an RVE is associated to each macrostructure Gauss integration point, then localization is performed to solve the periodic boundary value problem at the microscopic scale providing the RVE geometry and local constitutive equation, and finally the homogenization step is applied to obtain the macroscopic response. The main advantage such approach lies in the fact that an explicit form of the macroscopic constitutive law is not required while solving the macro problem, the relationship between $\bar{\sigma}$ and $\bar{\varepsilon}$ is established solely through scale transition by computing the appropriate average quantities and the macroscopic tangent operators.



PBC:

$$\Delta u^+ - \Delta u^- = \Delta \bar{\varepsilon} \cdot (x^+ - x^-)$$

Figure 4.7: Schematic representation of the micro-macro scale transition in FE^2 method

Table 4.1: Microscale and macroscale governing set of equations

Control equations	Macroscale problem	Microscale problem
Equilibrium	$\mathrm{div}\,\bar{\boldsymbol{\sigma}} + \bar{\boldsymbol{b}} = 0$	$\mathrm{div}\,\boldsymbol{\sigma} = 0$
Kinematics	$\bar{\varepsilon} = \frac{1}{2}\left(\nabla \bar{u} + {}^t\boldsymbol{\nabla}\bar{u}\right)$	$\bar{\varepsilon} = \frac{1}{2}\left(\nabla u + {}^t\boldsymbol{\nabla} u\right)$
Stress-Strain incremental constitutive law	$\Delta\bar{\sigma} = \bar{C}_{tan} : \Delta\bar{\varepsilon}$	$\Delta\sigma = C_{tan} : \Delta\varepsilon$

Table 4.2: Scale coupling set of equations

Control equations	Scale coupling
Periodic boundary conditions	$\Delta u^+ - \Delta u^- = \Delta\bar{\varepsilon} \cdot (x^+ - x^-)$
Homogenization	$\bar{\sigma} = \langle\boldsymbol{\sigma}\rangle = \frac{1}{V}\int_\Omega \sigma \mathrm{d}V$

The resolution of non-linear multiscale problems with FE^2 method is traditionally done in an incremental way using implicit resolution schemes such as Newton-Raphson at both scales. Figure 4.8 illustrates the key steps to compute incrementally the macroscopic fields using the concept of periodic homogenization. At each time step (n), the local equilibrium problem is solved for every macroscopic Gauss integration point where an unit cell is associated. The macroscopic strain increment $\Delta\bar{\varepsilon}$ and periodic boundary conditions are applied to the RVE (localization step). Then, the BVP problem is iteratively solved using FE method and the microscopic quantities $\left(\varepsilon^{(\mathbf{n})}, \sigma^{(\mathbf{n})}, \zeta^{\mathbf{n}}\right)$ are updated for each microscopic Gauss point. $\left(\varepsilon^{(\mathbf{n})}, \sigma^{(\mathbf{n})}\right)$ are the micro strain and stress at the time step n respectively, $\zeta^{\mathbf{n}}$ is the set of internal state variables at the time step n. Finally, the macroscopic stress $\bar{\sigma}^{(\mathbf{n+1})}$ is computed through volume averaging of the microscopic stresses (homogenization step) and the incremental macroscopic tangent operator $\overline{\mathbf{C}}_{\mathbf{tan}}^{(\mathbf{n+1})}$ is obtained with perturbation techniques for example. Once the macroscopic quantities $(\bar{\sigma}^{(\mathbf{n+1})}, \overline{\mathbf{C}}_{\mathbf{tan}}^{(\mathbf{n+1})})$ are calculated, the global equilibrium is checked. If the macroscopic convergence is not satisfied, a new macro strain increment $\Delta\bar{\varepsilon}^{'}$ is provided by the FE solver and the local problem is solved once again. This process is performed until numerical convergence is achieved at both scales. In that case the multiscale analysis proceeds to the next time increment $(n+1)$.

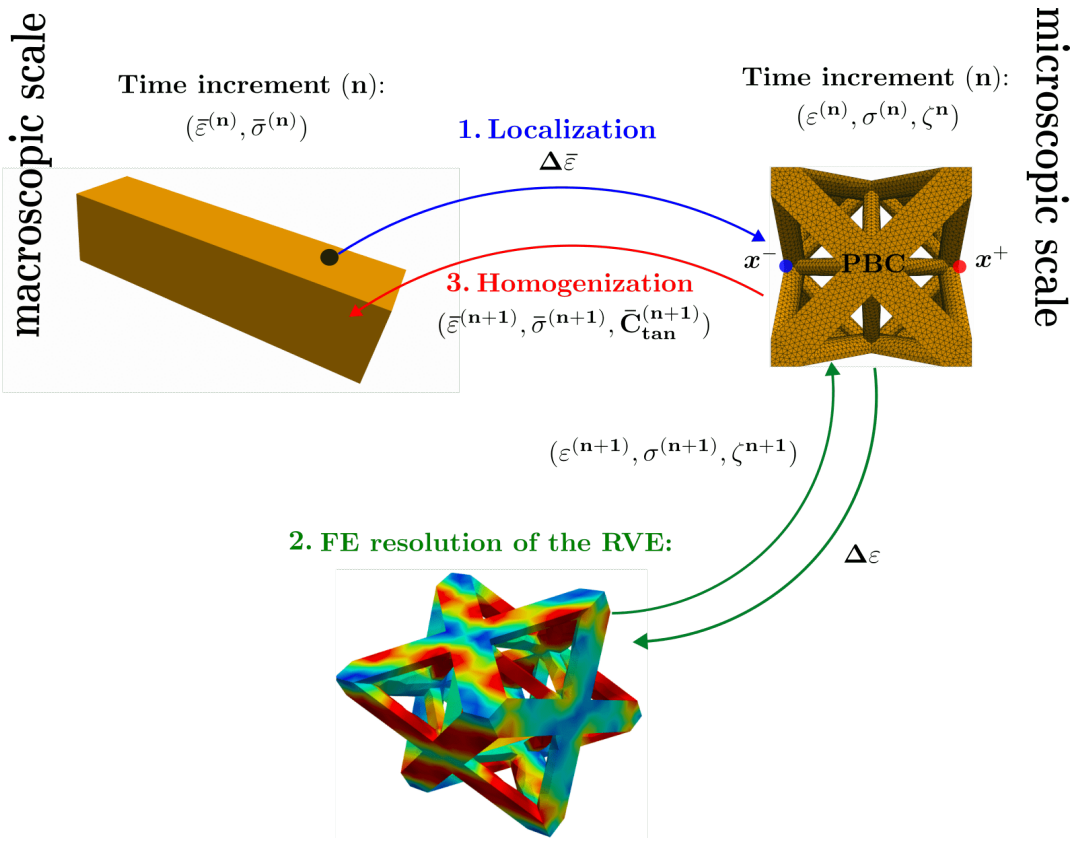


Figure 4.8: Schematic representation of the FE^2 key steps while solving multiscale problems incrementally

Although the FE^2 can be considered as an attractive alternative compared to direct FE simulations on fully meshed heterogeneous structures, it is clear that this modeling strategy also suffers from some

limitations given many factors. As described in the above mentioned FE^2 procedure, non-linear problems are solved for both the micro and macro scales using iterative schemes. This process is usually expensive due to the treatment of material non-linearities. In addition, other considerations may increase this computational time including the number of Gauss integration points, the complexity of loading conditions, the number of iterations and the meshing resolution of the RVE. Therefore, all these parameters may sometimes prevent this approach from being efficiently applied to structural analysis [Lange *et al.* 2021]. In the next section, we propose to develop a new numerical modeling strategy using a deep neural networks based method (FE-LSTM) in order to overcome the aforementioned restrictions.

4.3 FE-LSTM: A multiscale approach combining the FE method and LSTM neural networks for heterogeneous structures modeling

4.3.1 FE-LSTM architecture design

The purpose of the present work is to elaborate a novel modeling strategy to accelerate multiscale simulations of heterogeneous structures using deep neural networks. The developed approach, called FE-LSTM, combines the finite element method and an LSTM recurrent neural network to solve multiscale problems. The principle of this methodology is to treat the microscopic and macroscopic problems separately: Unlike the FE^2 method, the FE resolution of the microscopic problems is no longer required and the computation of the RVE effective response is predicted by an LSTM trained on a database of offline micro computations. Using the predicted macroscopic fields, a finite element analysis is conducted to verify the global equilibrium at the macroscale, this process is then performed until the achievement of numerical convergence. From the above description of FE-LSTM, it is obvious that the approach is heavily inspired by the FE^2 method as it mostly rely on the same workflow. However, the main advantage of FE-LSTM lies in the massive computational time savings when solving the microscopic problems. This performance can only be achieved with a reliable surrogate such as deep neural networks. Based on the studies conducted in chapter 2, the robustness of LSTM have been highlighted as a powerful tool capable of handling path dependent behaviors and accurately capturing complex and highly non-linear behaviors, hence the employment of such architecture to approximate the effective material responses in this multiscale framework.

The general workflow of the FE-LSTM approach is summarized in figure 4.9, this illustration describe the key steps to compute the macroscopic fields incrementally. First, an RVE is associated to all Gauss integration points. The macroscopic fields $\left(\bar{\varepsilon}^{(\mathbf{n})}, \bar{\sigma}^{(\mathbf{n})}\right)$ are known at the time increment (n), a macro strain increment $\Delta\bar{\varepsilon}$ is then applied to the unit cell. $\Delta\bar{\varepsilon}$ is added to $\bar{\varepsilon}^{(\mathbf{n})}$ to compute $\bar{\varepsilon}^{(\mathbf{n+1})}$ which is the input of the LSTM neural network. As shown in figure 4.10, the corresponding architecture is a Stacked LSTM composed of two LSTM layers and a dense layer at the end. This neural network takes $\bar{\varepsilon}^{(\mathbf{n+1})}$ as an input, then stores the material mechanical state in its memory gate, updates the input gate, forget gate, output gate, values to compute the next hidden state which is contains the effective stress and tangent operator $\left(\bar{\sigma}^{(\mathbf{n+1})}, \overline{\mathbf{C}}_{\mathbf{tan}}^{(\mathbf{n+1})}\right)$. Using these predicted quantities for all the Gauss integration points, the global equilibrium is checked using a FE analysis. Similarly to the FE^2 approach, if the convergence test is not successful, a new macro strain increment will be provided by the FE solver and new macroscopic fields will be predicted by the LSTM model. This process is performed until the satisfaction of numerical

convergence and then the multiscale analysis proceeds to the next time increment.

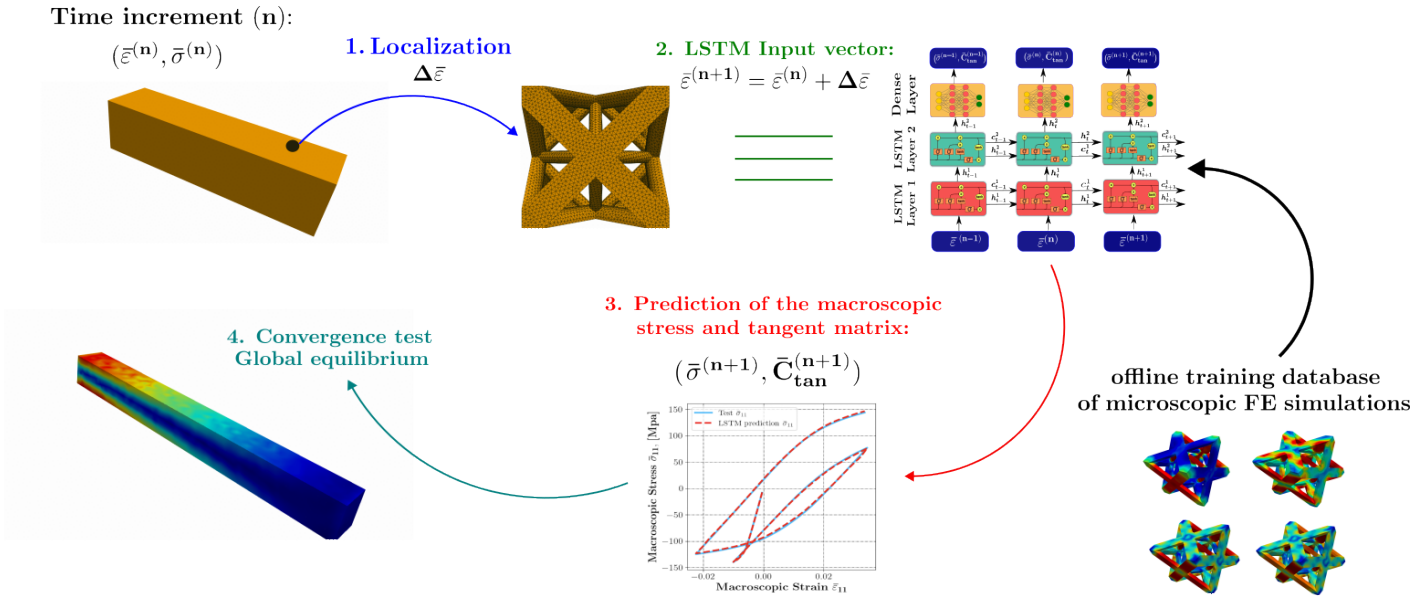


Figure 4.9: FE-LSTM general workflow to compute the effective mechanical response of non-linear heterogeneous structures

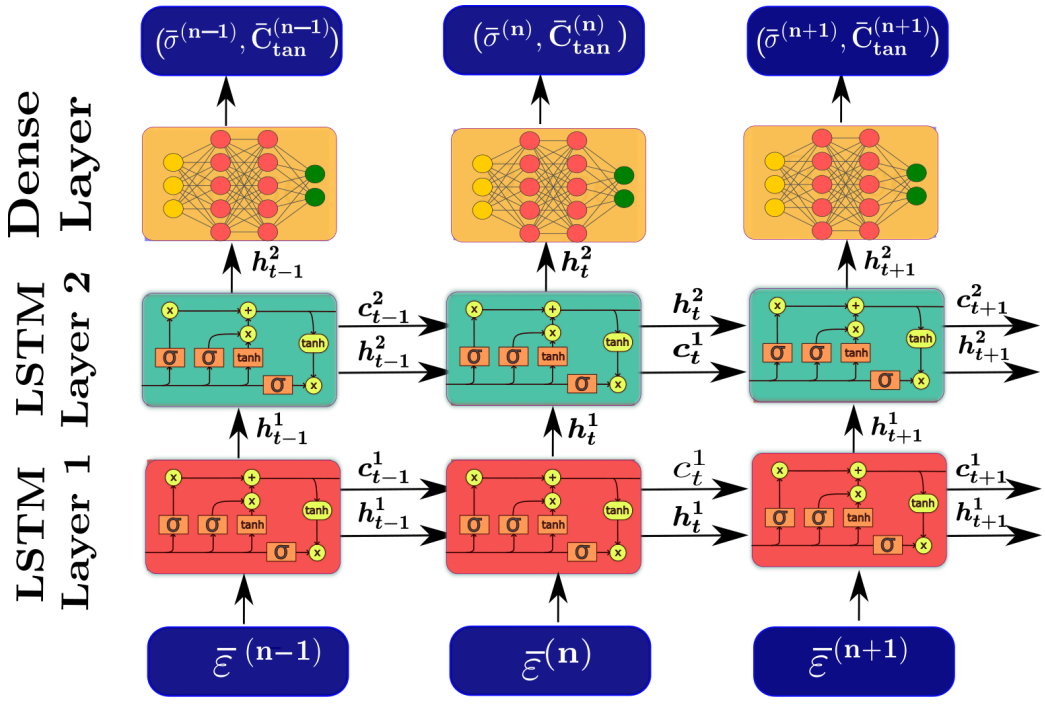


Figure 4.10: Stacked LSTM Neural Network architecture to predict the homogenized response of the RVE

4.3.2 Architectured materials database generation

The calibration of the LSTM model parameters requires an offline training phase involving a database of numerical simulations on the RVE. Before proceeding to the description of the database generation strategy, some clarifications needs to be addressed regarding the main hypothesis of this study. First, the microstructure geometry is similar everywhere in the macrostructure, so that the same RVE is associated to all Gauss integration points. The corresponding microstructure is an octet-truss architectured material whose geometry and periodic mesh are illustrated in figure 4.3.2. The geometrical and mechanical properties of the constitutive material which is a titanium alloy Ti-6Al-4V are summarized on table 4.3. The non-linear microscopic constitutive equation is an elastoplastic law with isotropic hardening. Therefore, the only varying parameters while conducting RVE simulations is the applied loading conditions which are characterised by the evolution of the macroscopic strain tensors $\bar{\varepsilon}^{(n)}$.

Table 4.3: Mechanical properties of titanium alloy Ti-6Al-4V and geometrical properties of the considered octet truss

Material parameter	**Value**
Young's modulus E	113800 MPa
Poisson's ratio ν	0.34
Yield Stress σ_Y	1000 MPa
Hardening parameter H	1600 Mpa
Plastic hardening exponent n	0.5
External cylinders radius R_e	0.1
Internal cylinders radius R_i	0.05

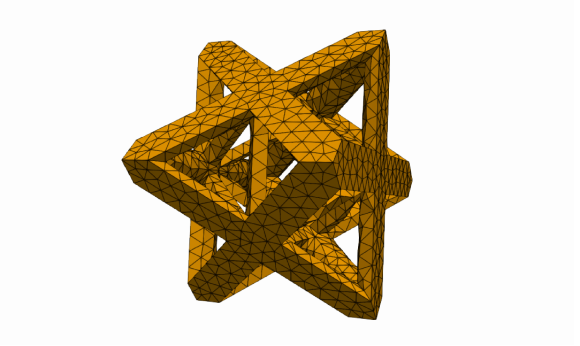


Figure 4.3.2: Geometry and mesh of the octet truss RVE (Number of nodes: 1694, Number of elements: 8284)

The adopted strategy to generate the database $\overline{\mathscr{D}}$ is described as follows: 10000 samples of microscopic RVE simulations, subjected to periodic boundary conditions and to multi-axial and non proportional loading paths, are generated using a finite element software *Fedoo* which relies on the *Simcoon* library to solve the constitutive equations [Prulière & Chemisky 2022]. The necessary resources to execute the simulations were provided by the computing facilities MCIA (Mésocentre de Calcul Intensif Aquitain) of Université de Bordeaux. The applied loading conditions on RVE boundaries are three-dimensional and can be expressed in terms of the six components of the macroscopic strain tensor $\bar{\varepsilon}^{(n)} = [\bar{\varepsilon}_{11}^{(n)}, \bar{\varepsilon}_{22}^{(n)}, \bar{\varepsilon}_{33}^{(n)}, \bar{\varepsilon}_{12}^{(n)}, \bar{\varepsilon}_{13}^{(n)}, \bar{\varepsilon}_{23}^{(n)}]$. To generate non-proportional loading paths, the total number of time steps, which is fixed at 100, is divided into 4 steps of 25 increments. This number of increments which is smaller than the one used to generate the database in Chapter 2 results in a compromise between a reasonable computational time, a good convergence of the Newton Raphson algorithm (too big increments can leads to convergence difficulties) and a sufficient number of time iterations for an efficient training. Each step is linear and the bounds values of each macroscopic strain are chosen randomly from the interval $\mathbb{I} = [-5\%, 5\%]$. Figure 4.11 shows some examples of 3D loading path configurations used as training data. Some results of unit cells finite element computations are illustrated in figure 4.12: the associated microscopic fields are subsequently homogenized to obtain the macro fields. Finally, the database $\overline{\mathscr{D}}$ is constructed using the set of macroscopic quantities $(\bar{\varepsilon}, \bar{\sigma}, \overline{\mathbf{C}}_{\mathbf{tan}})$ which are fed as time

sequences to train the LSTM model. As usually, $\overline{\mathscr{D}}$ is split into a training set $\overline{\mathscr{T}}$ (90% of total samples), a validation set $\overline{\mathscr{V}}$ (20% of $\overline{\mathscr{T}}$) and a test set $\overline{\mathscr{T}}$ (10% of total samples).

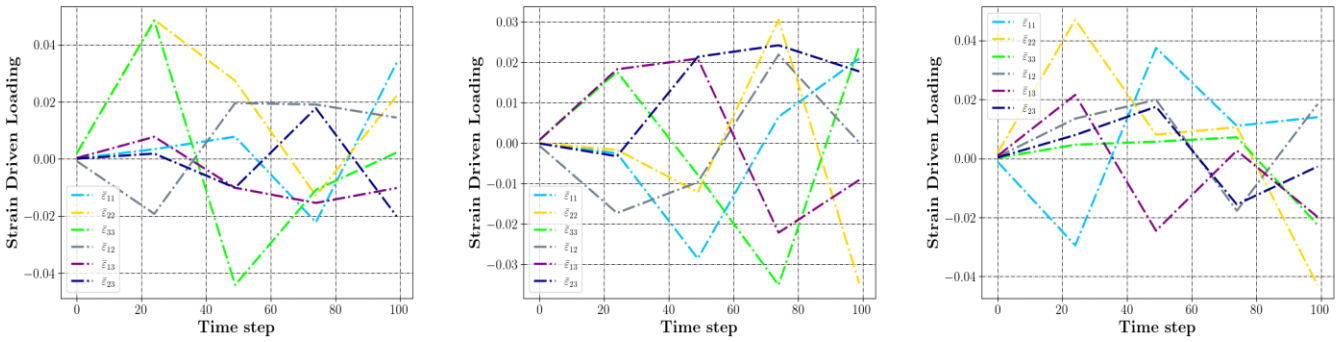


Figure 4.11: Examples of multi-axial and non proportional loading conditions for RVE finite element simulations

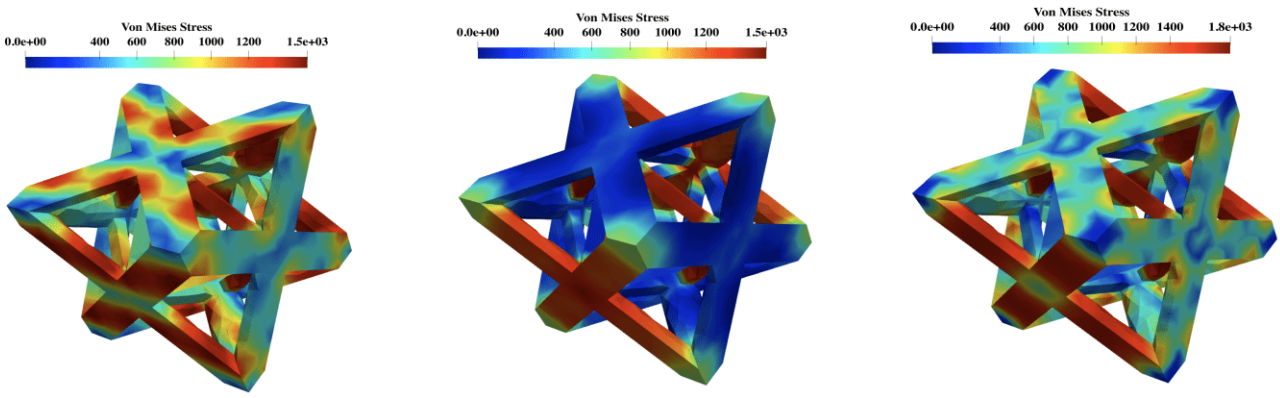


Figure 4.12: Examples of simulated octet-truss using FE analysis, the corresponding homogenized fields are used as training data

4.3.3 Model training phase and hyperparameters selection

Following the database generation and features scaling, the next step is the training process of LSTM models. For this purpose, two LSTM models have been created to predict separately each quantity of interest, i.e the macroscopic stress $\bar{\sigma}$ and the macroscopic tangent stiffness $\overline{\mathbf{C}}_{\mathbf{tan}}$. This approach is much more practical than creating the same model to predict both quantities as they are of different nature, and thus capturing the non-linear behaviors may require different parameter adjustments dependant on the complexity of each model. Models implementation was carried out using Keras library [Chollet *et al.* 2015] and Tensorflow API [Abadi *et al.* 2015]. Six different configurations of LSTM architectures were tested by varying three activation functions (ReLU, Hyperbolic tangent and Sigmoid) and the number of LSTM layers. This hyperparameter study aims to identify the most reliable model that minimizes training and

validation errors. The common Mean Squared Error (MSE) was employed as a loss function and Adam (optimizer with adaptative moment) was used as an optimization algorithm.

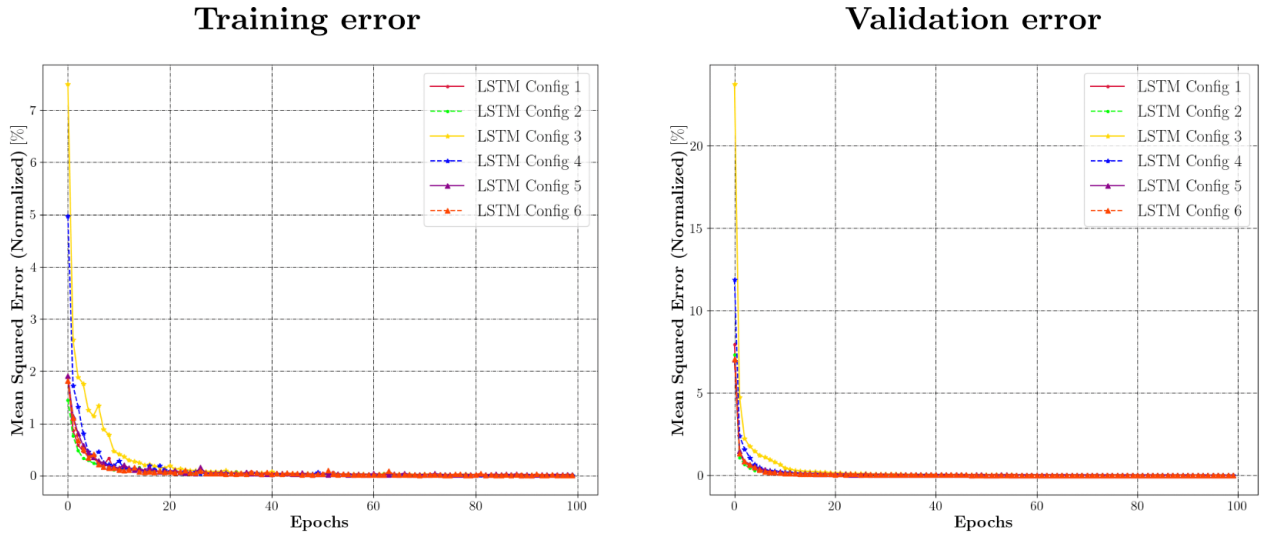


Figure 4.13: Evolution of training (left figure) and validation errors (right figure) for the tested LSTM configurations (Prediction of the macroscopic stress)

Table 4.4: Evaluation of LSTM configurations training and validation performances for the macroscopic stress-strain responses prediction

ANN Configuration	Number of hidden LSTM layers	Output layer activation function	Training loss NMSE (%)	Validation loss NMSE (%)
Config 1	2	Relu	0.011 %	0.009 %
Config 2	3	Relu	0.007 %	0.006%
Config 3	2	Sigmoid	0.013 %	0.011 %
Config 4	3	Sigmoid	0.009 %	0.008 %
Config 5	2	Hyperbolic tangent	0.013 %	0.013 %
Config 6	3	Hyperbolic tangent	0.011 %	0.010 %

For the prediction of macroscopic stress tensors, the evolution of training and validation Normalized Mean Squared Error (NMSE) metric are shown in figure 4.13 for all the six tested configurations. It can be noticed from the corresponding graphs that the learning process is achieved rapidly regardless of the choice of the activation or the number of LSTM layers. A significant decrease in the errors is observed during the first epochs, followed by a loss stabilization after nearly 20 epochs. The minimum values of training and validation loss obtained through all the learning process are summarized in table 4.4. We can observe that the error values are almost the same for all configurations (order of magnitude of 10^{-2}%). Furthermore, these low error values are an evidence that the model has achieved excellent training performances. Even though different activation functions and number of LSTM layers can be

used to approximate the macroscopic stress-strain response, the first configuration has been chosen to fix model hyperparameters. This choice is justified by the following reasons: First, two LSTM layers instead of three are sufficient to obtain a good approximation. An increase of number of layers does not necessarily improve model performance. In addition, the prediction of the quantities of interest is much faster using a small number of layers, which is consistent with the objectives of the study. For the activation function selection, the Rectified Linear Unit ReLU is computationally more efficient in backpropagation compared to Sigmoid and Hyperbolic tangent functions as it only takes the maximum between the input and zero value. The required training time with ReLU is nearly three times faster than training with Sigmoid or Hyperbolic tangent. Furthermore, ReLU activation is more suitable for overcoming vanishing gradient problems, thus allowing LSTM models to learn faster and perform better.

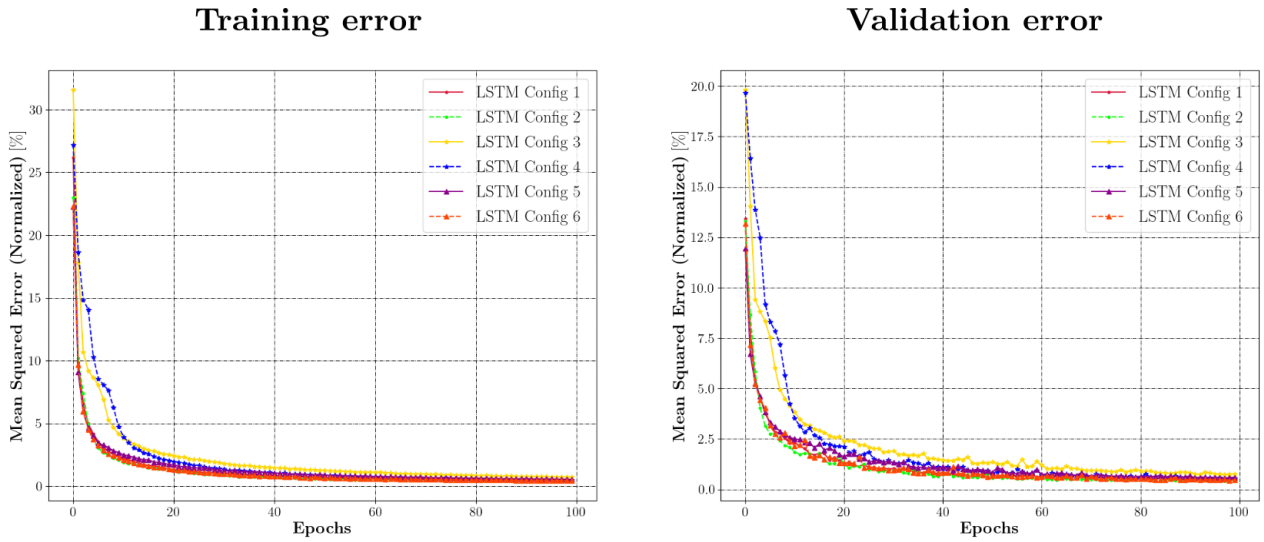


Figure 4.14: Evolution of training (left figure) and validation errors (right figure) for the tested LSTM configurations (Prediction of the macroscopic tangent matrix)

Table 4.5: Evaluation of LSTM configurations training and validation performances for the tangent matrix predictions

ANN Configuration	Number of hidden LSTM layers	Output layer activation function	Training loss NMSE (%)	Validation loss NMSE (%)
Config 1	2	Relu	0.41 %	0.42 %
Config 2	3	Relu	0.39 %	0.40 %
Config 3	2	Sigmoid	0.72 %	0.72 %
Config 4	3	Sigmoid	0.51 %	0.51 %
Config 5	2	Hyperbolic tangent	0.53 %	0.55 %
Config 6	3	Hyperbolic tangent	0.42 %	0.44 %

Regarding the training process on macroscopic tangent operators, the same observations apply to

this model about the loss evolution and the errors order of magnitude being almost the same for all configurations. However, it can be noticed that the resulting errors are much larger compared to the values obtained with the previous model. This disparity can be explained by certain assumptions, the first one being the complexity of the non-linear behavior of the tangent operators. Secondly, the number of the predicted outputs (21 components of the symmetric part of the tangent matrix) is much higher than the number of macroscopic stress components. When the number of outputs is large, it is very common to experience overfitting problems, which are induced by the fact that the model tends to attribute certain attention (larger weights) to some neurons associated to specific outputs compared to the others. Finally, the numerical noise resulting from the computation of tangent operators (especially for the insignificant components) can also be a cause of error amplification. However, it is important to note that training and validation error values regarding this quantity of interest are still very low (0.4% for the first configuration), thus not having a significant effect on the model's reliability.

4.3.4 Prediction of the macroscopic mechanical behavior of Octet-Truss structures

In order to evaluate the performance and the generalization capabilities of LSTM models, it is crucial to verify their predictive accuracy on the test dataset $\overline{\mathcal{T}}$. We recall that this batch of data (10% of total samples) was never employed during the training phase, hence it is a good indicator to determine if the model suffers from potential overfitting. Figure 4.16 present an example of the LSTM predicted macroscopic stress-strain response in comparison to the corresponding RVE finite element solution under a multi-axial and non proportional loading path shown in figure 4.15. The obtained results in this example shows an excellent agreement between the predicted macroscopic stress components and the target values. The model assess a strong capability to capture the non-linear responses despite the complexity of the loading condition. In addition, it can be seen from figure 4.16 graphs a complete absence of numerical noise during a change of direction in the stress-strain curves, which is advantageous for the convergence of multiscale simulations. To quantify the model's accuracy on the entire test data samples, we use the Mean Absolute Percentage Error (MAPE) metric expressed by Eq.4.13. The resulting MAPE for the prediction of macroscopic stress-strain responses is found to be equal to 0.27%. As a consequence the LSTM model achieved a very high accuracy rate ($\approx 99\%$). This finding emphasizes once again the potential of RNN based models as reliable surrogates capable of capturing history dependant non-linear behaviors. This preliminary outcome is very encouraging for the next step of this study where LSTM shall be integrated in a multiscale approach within the FE-LSTM framework.

$$MAPE = \frac{1}{N}\sum_{k=1}^{N}\frac{\left|y_{\text{true}}^{(k)} - y_{p}^{(k)}\right|}{|y_{\text{true}}^{(k)}|} \times 100 \tag{4.13}$$

- $\boldsymbol{y_{true}^{(k)}}$: Target values
- $\boldsymbol{y_{p}^{(k)}}$: Predicted values
- N : Total number of test samples

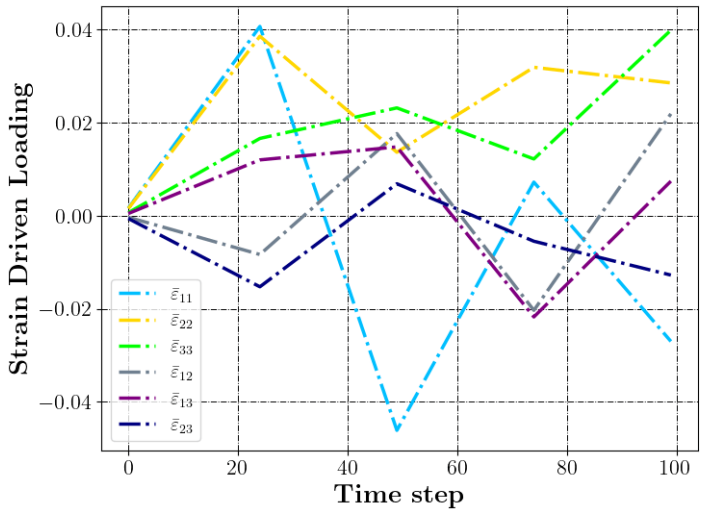


Figure 4.15: The applied multi-axial and non proportional loading path on RVE boundaries for an example of test data

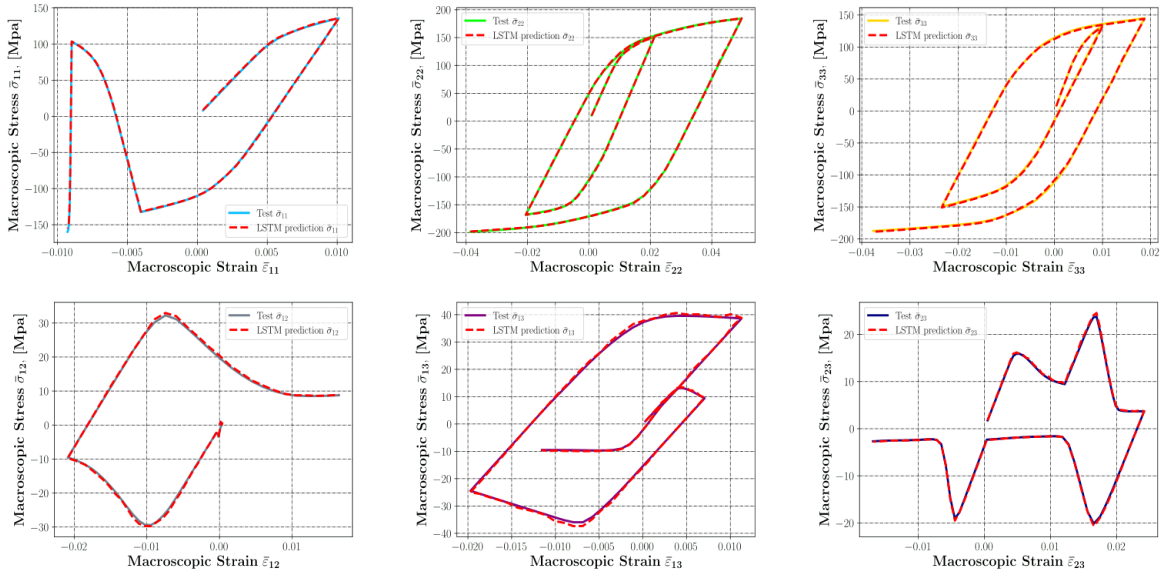


Figure 4.16: LSTM predicted macroscopic stress-strain response in comparison to the RVE FE simulation for an example of test data

We also assess the reliability of the second model for the prediction of macroscopic tangent matrix $\overline{\mathbf{C}}_{\mathbf{tan}}$. A good evaluation of the tangent operators is critical for the computation of non-linear heterogeneous structures within Newton-Raphson framework. Figure 4.17 illustrates the comparison between

tangent modulus components predictions and the target values from simulation for an example of test data. We can observe from this example a good correspondence between the two quantities, the general trend of the tangent moduli components is well captured by the LSTM during all the time steps. However, the level of accuracy obtained for this quantity of interest is not as high as the precision achieved for the stress components, which is completely expected given the discussion in the above subsection 4.3.3. In addition, it can be noticed from figure 4.17 graphs a presence of small noise especially in the first time increments, we can assume that vanishing gradient phenomena is potentially manifested in this area. Nevertheless, it is important to note that these errors do not have a large influence on the overall model reliability, the evidence is that the obtained MAPE on the test data samples is equal to 2.57%, which is still a low value.

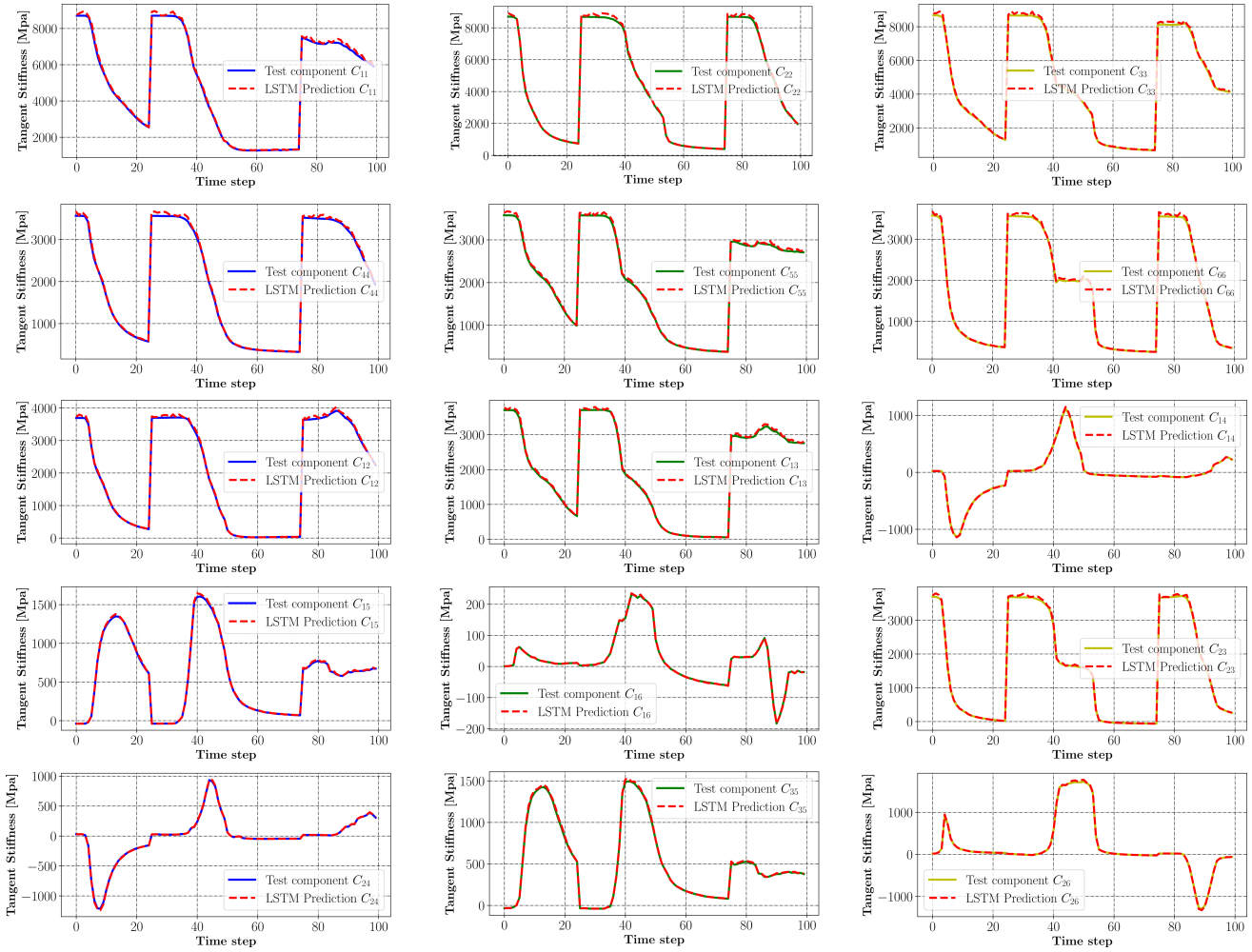


Figure 4.17: LSTM predicted macroscopic tangent matrix components in comparison to the RVE FE simulation for an example of test data (Only the symmetric part and the significant components are showed)

To conclude this section, we have presented the general framework of the FE-LSTM approach which consists in replacing the FE resolution of the microscopic problem by an LSTM surrogate. This approach

aims at accelerating multiscale simulations of non-linear heterogeneous structures. In the next section, we propose to apply this approach for the structural analysis of 3D architected structures in real life conditions. Since the RVE loading states are a priori unknown during multiscale simulations, the challenge that arises is whether LSTM models have the ability to generalize the macroscopic behavior when the RVE are subjected to loading paths that are not necessary based on linear steps employed in the training process.

4.4 Applications of FE-LSTM model in multiscale analysis of 3D architectured structures

4.4.1 First validation test: Standard architectured specimen under tensile loading

In this first example, we propose to evaluate the FE-LSTM approach on a 3D architectured structure. For this purpose, a standard specimen subjected to uniaxial tensile loading is considered as a validation test. The macrostructure geometry, mesh and boundary conditions are presented in figure 4.18. The specimen is meshed using 4-node tetrahedral elements TET4 with 4 Gauss integration points per element and the entire mesh consist of 5100 Gauss points. For the microstructure, the same octet-truss used during the training phase of LSTM models is considered as the RVE. The numerical implementation of the FE-LSTM approach was performed in Python using a dedicated script that was integrated with *Fedoo* finite element code [Prulière & Chemisky 2022]. The incorporation of LSTM models as a surrogate of the macroscopic constitutive law is done in a similar way to the User Material subroutines in FE codes. Once the LSTM training phase is completed, the corresponding trained weight matrices and bias vectors are saved for each LSTM cell gates. These quantities are then used to compute incrementally the macroscopic response within the LSTM framework by updating the appropriate cell gates and memory. Finally, the predicted outputs are processed by *Fedoo* FE solver and the structure global equilibrium is verified using Newton-Raphson iterative scheme. This process is executed until the convergence of the solution.

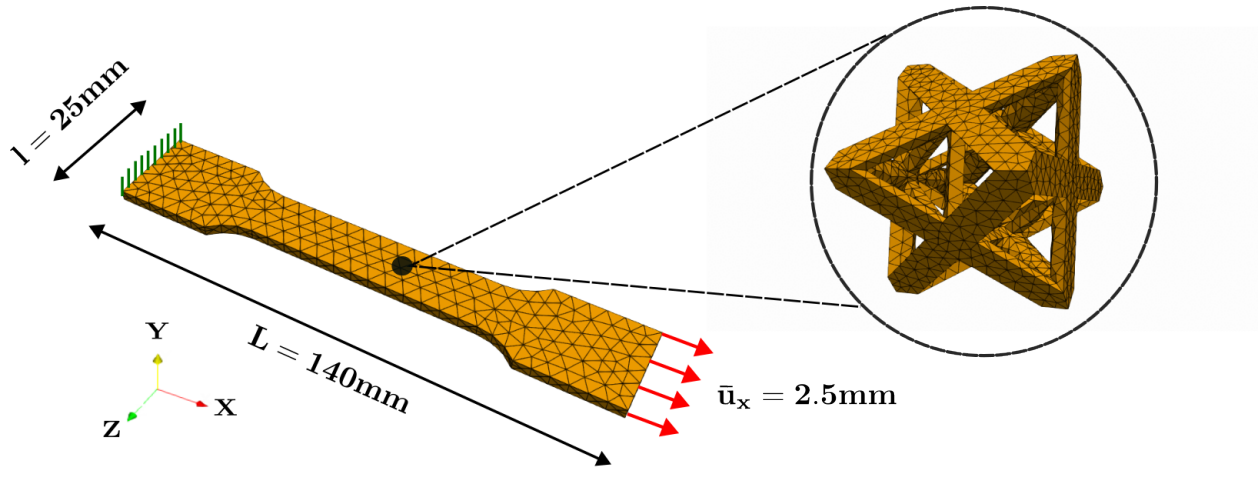


Figure 4.18: Geometry, mesh and boundary conditions of the tested architectured specimen under tensile loading

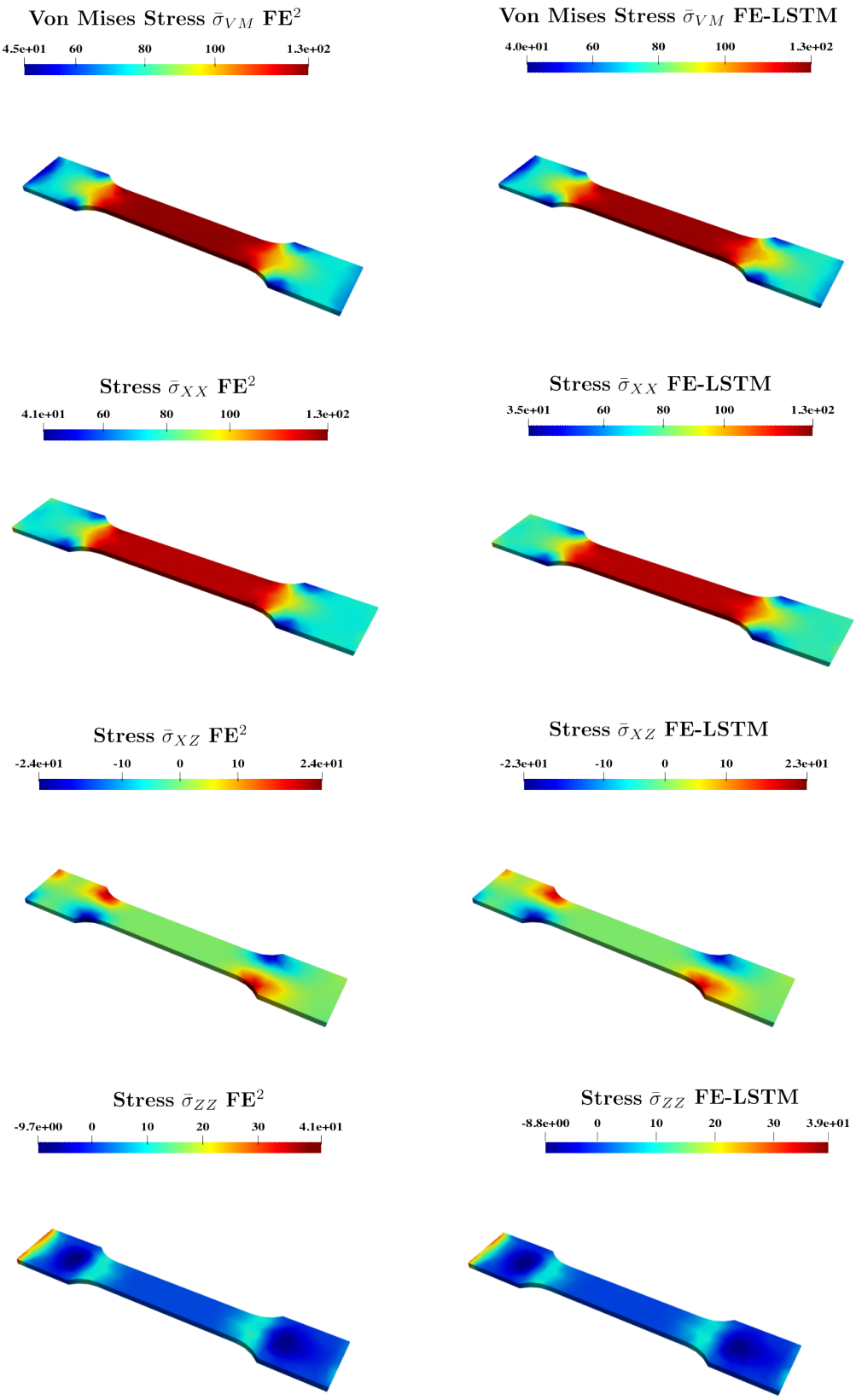


Figure 4.19: Comparison between the macroscopic stress fields $\bar{\sigma}$ obtained by FE^2 and by FE-LSTM of a standard architectured specimen under tensile loading

The FE-LSTM approach is applied to compute the effective mechanical behavior of the architectured specimen under tensile loading. The same multiscale simulation is conducted using a concurrent FE^2 method in order to validate the developed approach. Figure 4.19 illustrates the distribution of the macroscopic Von Mises stress $\bar{\sigma}_{VM}$, normal stress $(\bar{\sigma}_{XX}, \bar{\sigma}_{ZZ})$ and shear stress $\bar{\sigma}_{XZ}$ obtained by FE^2 and FE-LSTM. First of all, we would like to point out that the simulation convergence analysis has been achieved using the same Newton-Raphson tolerance as the FE^2 method. This preliminary result highlights the predictive capability of LSTM models especially when computing the macroscopic tangent modulus which are used to verify the structure global equilibrium. As for the accuracy of the predicted macroscopic fields, it can be seen from Fig. 4.19 that a very good agreement is achieved by FE-LSTM compared to FE^2 for all macroscopic stress components. The stress localization in the specimen reduced section is well captured by the FE-LSTM model and the predicted macroscopic stress fields $\bar{\sigma}_{XX}$ in the loading direction are consistent with the resulting values obtained by FE^2. In addition, we report the average normalized error (MAPE in Eq.4.13) between FE-LSTM predicted stress components and the target values using FE^2 in table 4.7. The reliability of the FE-LSTM model is clearly demonstrated in this case of application given the low MAPE values especially for the uniaxial macroscopic stress $\bar{\sigma}_{XX}$. Based on these encouraging preliminary results, we propose to apply the developed approach on a complex 3D architectured structure as detailed in the next subsection.

Table 4.6: Macroscopic stress components MAPE values

Macroscopic stress components	Stress $\bar{\sigma}_{VM}$	Stress $\bar{\sigma}_{XX}$	Stress $\bar{\sigma}_{ZZ}$	Stress $\bar{\sigma}_{XZ}$
MAPE (%)	0.02 %	0.01 %	2.04 %	0.44 %

4.4.2 Application to a complex architectured structure under proportional and non proportional loading paths

4.4.2.1 Case of a proportional loading path

In order to illustrate the flexibility of the FE-LSTM approach when applied to complex structures, two application cases of a non-standard architectured specimens under proportional and non-proportional loading are tested in this subsection. The first example consists of a 3D specimen, with a rather unusual shape containing several holes, subjected to a uniaxial tensile loading. The macrostructure geometry, mesh and boundary conditions are presented in Fig. 4.20. Dirichlet boundary conditions are applied to the the heterogeneous structure. One extremity is fixed while the other is subjected to a displacement-controlled loading. The specimen is meshed using 4-node tetrahedral elements TET4 with 4 Gauss integration points per element and the full mesh is composed of 13400 Gauss points. The computation of the macroscopic mechanical response of the heterogeneous structure is performed with FE-LSTM model and using FE^2 method. Both approaches are compared according to different criteria including the accuracy of the mechanical fields, simulation time and the required memory usage. Since the FE^2 method is very time consuming, the corresponding simulation is conducted using only 10 time increments while FE-LSTM is performed using 100 time increments. The analysis results are presented in the figures 4.21 and 4.22. Figure 4.21 illustrates the distribution of the macroscopic Von Mises stress $\bar{\sigma}_{VM}$, normal stress $\bar{\sigma}_{XX}$,$\bar{\sigma}_{YY}$

and shear stress $\bar{\sigma}_{XY}$ obtained by FE^2 and FE-LSTM. Regarding the macroscopic strain fields, figure 4.22 shows the distribution of the normal strain components $(\bar{\varepsilon}_{XX}, \bar{\varepsilon}_{YY})$ and shear strain component $\bar{\varepsilon}_{XY}$ respectively obtained by FE^2 and by FE-LSTM.

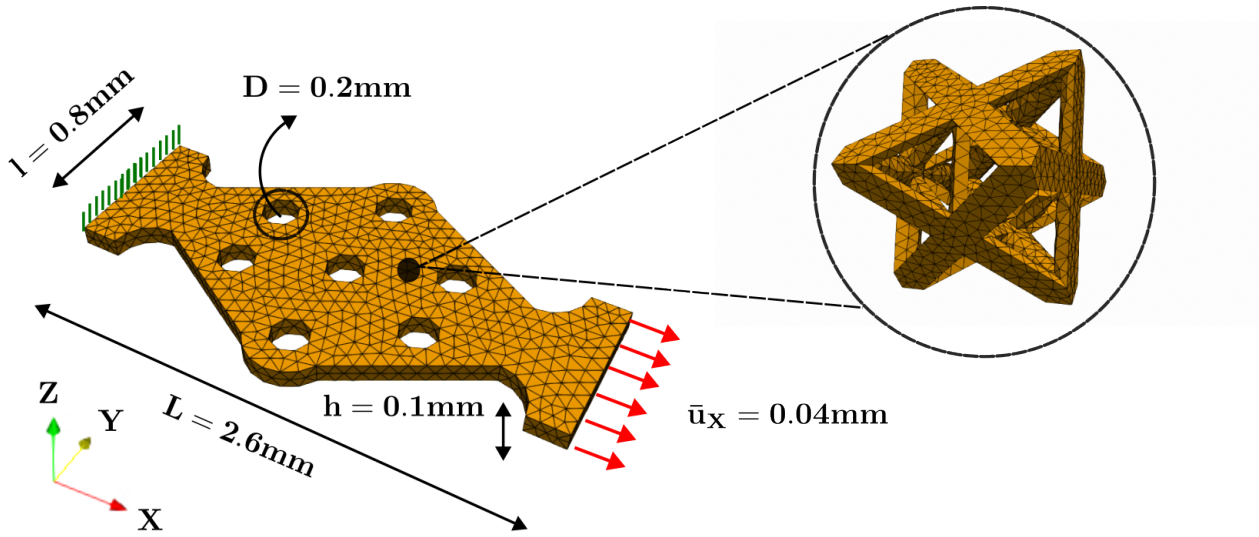


Figure 4.20: Geometry, mesh and boundary conditions of an architectured specimen with holes under tensile loading

As it can be seen from both figures, an excellent correspondence is observed between the macroscopic fields predicted by FE-LSTM model and the targets computed by FE^2. Despite the complexity of the geometry, the developed model is able to accurately predict the non-linear homogenized response of the architectured structure. Similarly to the FE^2 approach, FE-LSTM is also capable of taking into account the effect of the microstructure on the overall mechanical response of the macrostructure. In addition, we can notice from the distribution of the macroscopic stress field in the loading direction $\bar{\sigma}_{XX}$ a stress concentration effect due to the presence of holes. This stress localization is also captured by the FE-LSTM in the specimen reduced cross-sectional area and its neighboring holes, which is expected given the applied boundary conditions. In terms of values accuracy of the predicted macroscopic fields, a very good agreement between the two approaches can be observed as well, given the low error values (MAPE) reported in table 4.7. Furthermore, the maximum deviation of the macroscopic stress $\bar{\sigma}_{XX}$ between FE-LSTM and FE^2 for a Gauss point is at worst equal to 1%, which remains a reasonable error. From the distribution of the macroscopic strain fields shown in figure 4.22, we can verify that the strain components remain in the range of training data i.e in the interval $\mathbb{I} = [-5\%, 5\%]$. As neural networks are known to have poor extrapolation capabilities, it is always important to check this condition since a deformation state outside the training database could result in inaccurate predictions or a failure in convergence of the simulation.

We recall that the main objective of this study is the acceleration of multiscale simulations, therefore a breakdown of the offline and online computational costs is required to highlight the efficiency of the developed approach FE-LSTM. In table 4.8, the offline computing stage is given by the LSTM training

Table 4.7: Macroscopic stress components MAPE values

Macroscopic stress components	Stress $\bar{\sigma}_{VM}$	Stress $\bar{\sigma}_{XX}$	Stress $\bar{\sigma}_{YY}$	Stress $\bar{\sigma}_{XY}$
MAPE (%)	0.03 %	0.03 %	1.58 %	1.97 %

Table 4.8: Offline computational costs breakdown

Offline computations	**Database generation**	**LSTM training**
Computational time	70 hours	5.5 hours

phase and the database generation step. The LSTM training time (5.5 hours) remains acceptable since ReLU was used as an activation. This function is computationally more efficient in backpropagation compared to sigmoid or hyperbolic tangent, the training time is also three time faster. The database generation was carried out using the computing facilities of the cluster MCIA, 10 machines of 32 CPU each were involved in the creation of the database, parallel computing was also included in this process. The resulting computational time for this phase is 70 hours. Although this offline procedure remains expensive, major speed up can be achieved during the online computing stage. In addition, once the model is trained, it can be applied to any heterogeneous structure having the same microstructure. In contrast, the FE^2 method require solving once again non-linear problems on each macroscopic integration point at each iteration for any new simulation. FE-LSTM based approaches are therefore more interesting in the design of heterogeneous structures for examples, as they allow to quickly perform parametric analysis by varying the structure geometry or the loading conditions with respect to the desired performances.

For the online computing stage, table 4.9 summarizes the required computational time and memory usage for both approaches FE-LSTM and FE^2. The simulation of the heterogeneous structure took nearly 5 days using FE^2 while it only took 102 seconds using FE-LSTM, thus resulting in a computational time saving factor of 4235. This speed up factor is actually underestimated since only 10 time steps are used in the FE^2 simulation unlike the FE-LSTM model where 100 increments are applied. In reality, this factor can reach an order of 40 000 for two simulations performed with the same number of time steps, which is very highly significant. However, we would like to point out that the FE^2 approach used in this study has not been optimized for parallel computing, but we nevertheless believe that the speed up factor will not be affected too much, as the simulations with the developed method are performed within seconds. The achievable gains with FE-LSTM are not only related to the gain in computational time, but also to the computing resources. As shown in table 4.9, FE^2 simulation required a memory usage of 1.07 TB while the FE-LSTM only used 120 MB. The FE^2 simulation involved the use of a *bigmem* machine of 3TB RAM from the MCIA cluster, which is not always affordable. In contrast, the FE-LSTM simulation could be carried out in a desktop computer without any need for expensive computing resources.

Table 4.9: Online computational costs breakdown: The simulation of the non-linear architectured structure is performed using 100 time increments with FE-LSTM and 10 time increments with FE^2

	FE^2 simulation	**FE-LSTM simulation**	**Computational time saving factor**	**Memory usage saving factor**
Online simulation time	5 days	102 seconds	4235	–
Memory usage	1.07 TB	120 MB	–	8917

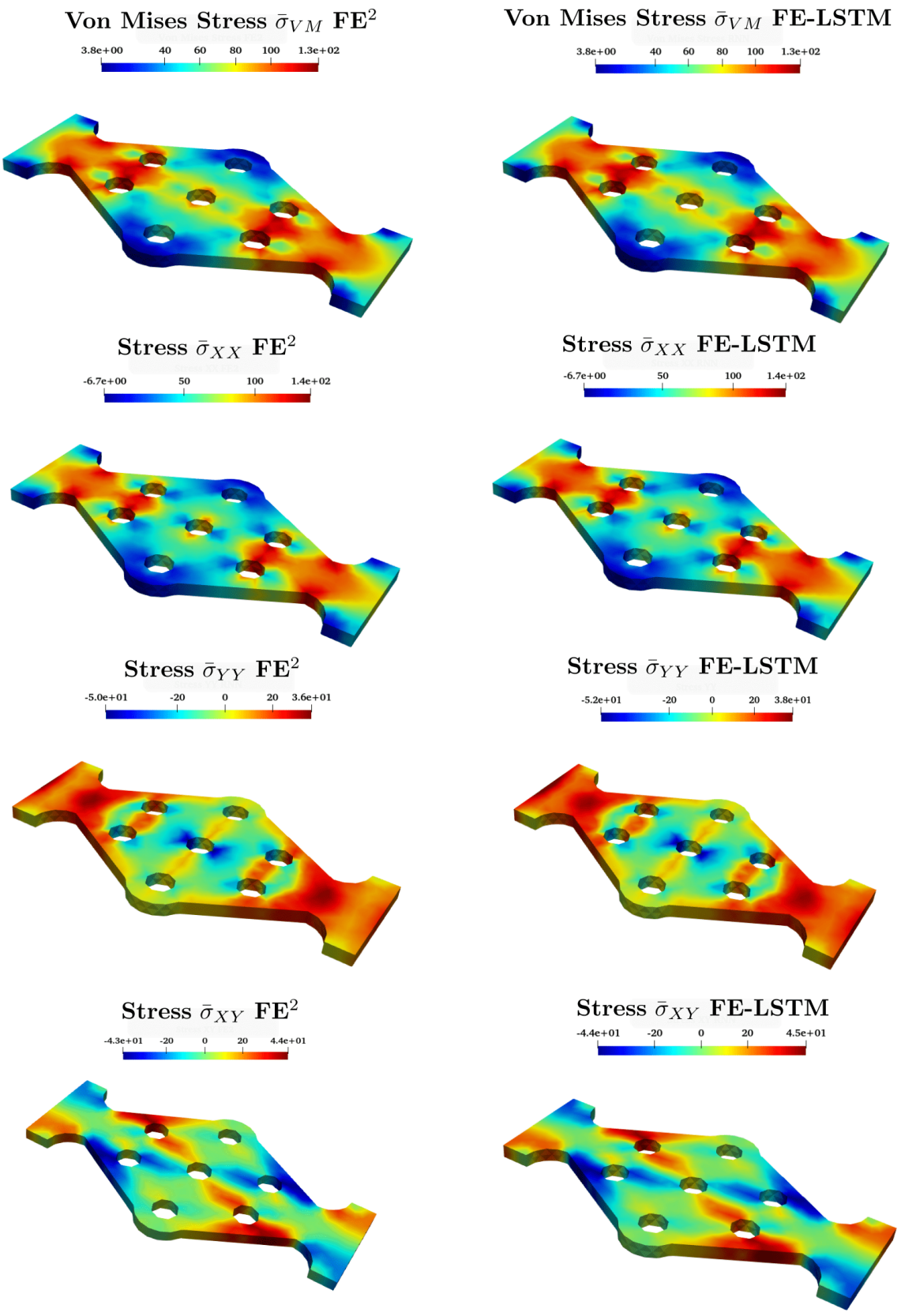


Figure 4.21: Comparison between the macroscopic stress fields $\bar{\sigma}$ obtained by FE^2 and by FE-LSTM of a standard architectured specimen with holes under tensile loading

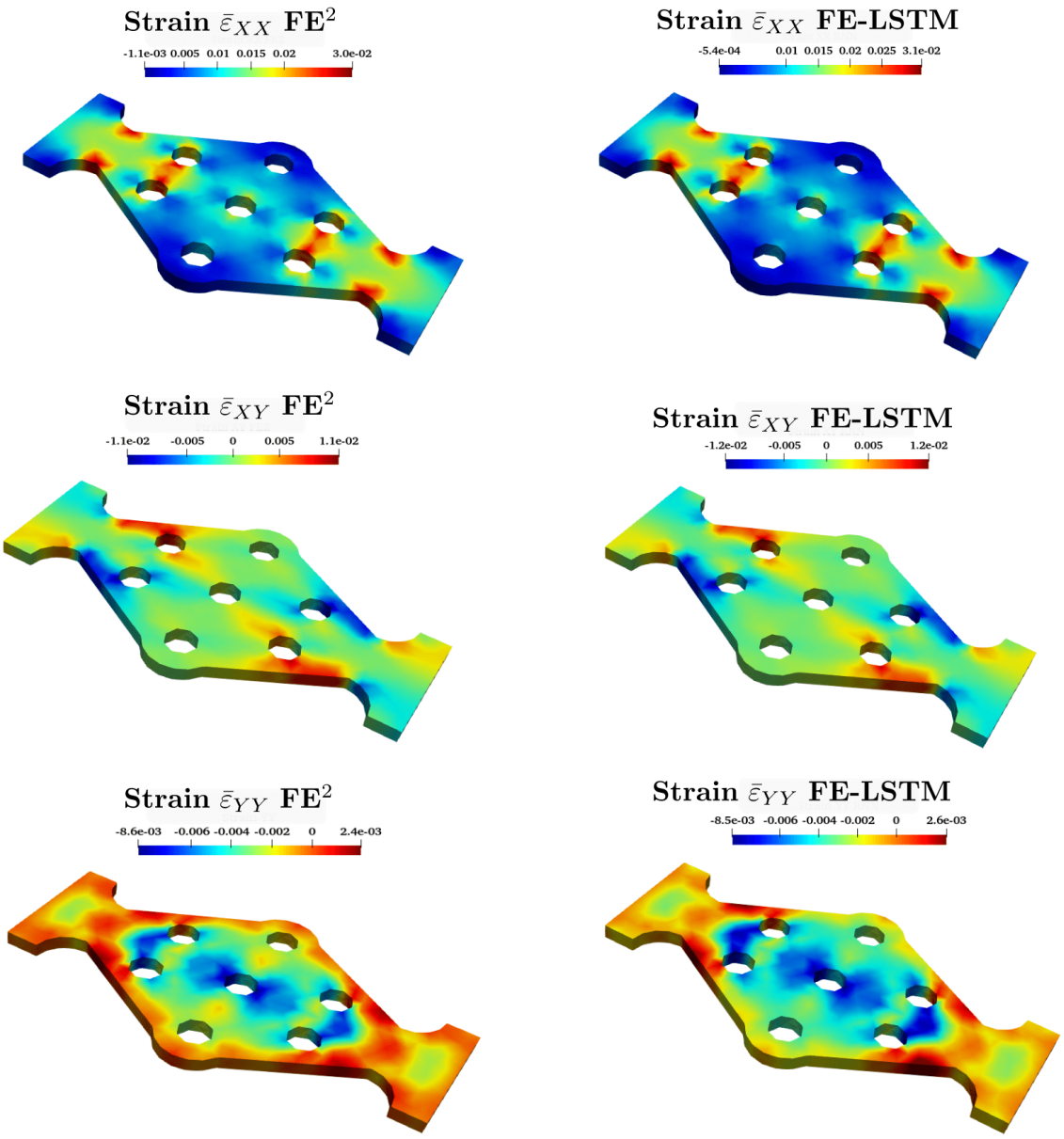


Figure 4.22: Comparison between the macroscopic strain fields $\bar{\varepsilon}$ obtained by FE^2 and by FE-LSTM of a standard architectured specimen with holes under tensile loading

4.4.2.2 Case of a non-proportional loading path

Within this second case of application, we aim to evaluate the reliability of the FE-LSTM approach for heterogeneous structures subjected to non-proportional loading. The same non-standard architectured specimen with several holes is considered as before, the only change is the applied loading conditions. The macrostructure geometry, mesh and boundary conditions are presented in Fig 4.23. Dirichlet boundary conditions are applied to the heterogeneous structure, one extremity is fixed while the other is subjected to a displacement controlled loading involving tensile and shear. This non-proportional path is composed of a tensile and a shear loading step with a final displacement value of $\bar{u}_X = 0.035mm$ and $\bar{u}_Y = 0.015mm$ respectively, followed by an unloading phase with displacement of $\bar{u}_X = 0.02mm$ and $\bar{u}_Y = 0.01mm$. The simulation of the macroscopic response of the architectured structure is only performed using FE-LSTM. The FE^2 simulation was not conducted due to computational resources limitations. In fact, the latter required a simulation time of 10 days on the MCIA cluster for a total of 20 time increments (10 for the loading step and 10 for the unloading step), unfortunately the reservation of computing machines was limited to one week especially for *bigmem* computers. However, we would like to remind that the FE-LSTM approach has already been validated previously through two case of applications.

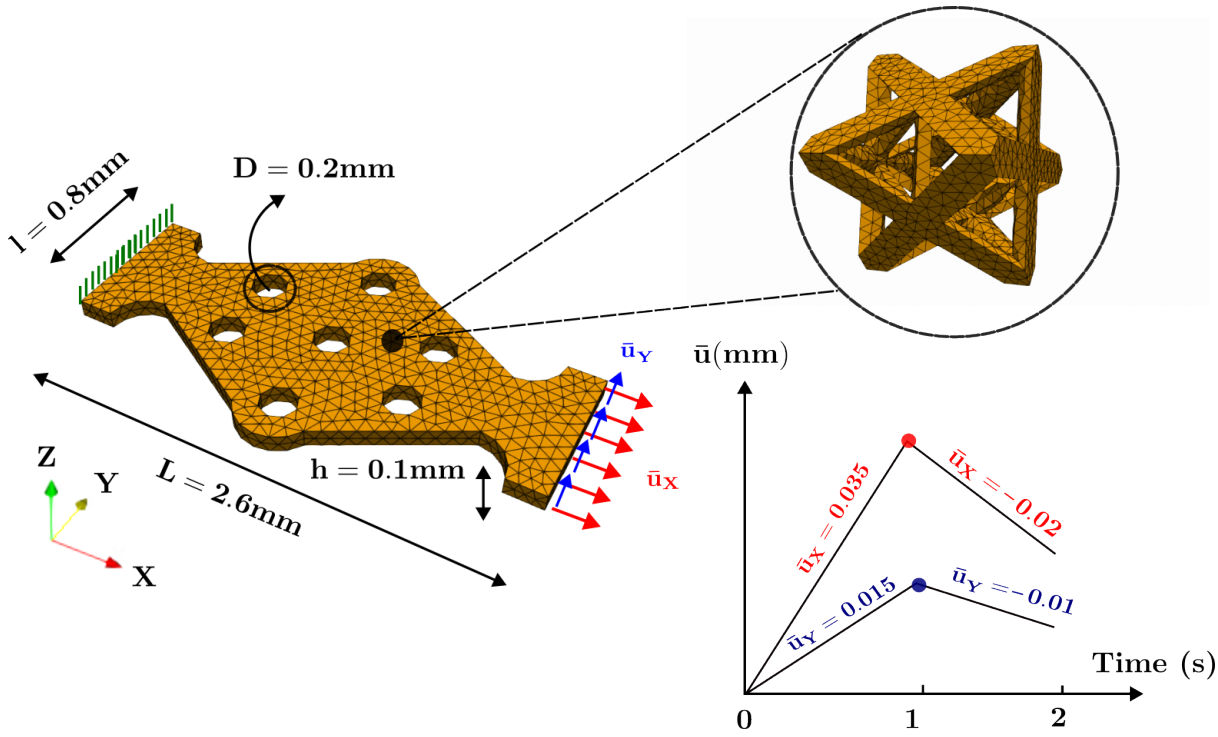


Figure 4.23: Geometry, mesh and boundary conditions of an architectured specimen with holes under tensile and shear loading/unloading conditions

The resulting macroscopic Von Mises stress $\bar{\sigma}_{VM}$, normal stress $\bar{\sigma}_{XX}$,$\bar{\sigma}_{YY}$ and shear stress $\bar{\sigma}_{XY}$ obtained with FE-LSTM are presented in Fig.4.24. The stress fields distribution on the architectured specimen are given at the end of tensile/shear loading and unloading steps. In absence of a FE^2 reference solution, we evaluate the numerical results in a qualitative way by verifying different aspects. First, the

model achieved numerical convergence with the same Newton-Raphson tolerance as used in the previous applications. This first outcome indicate that the predicted macroscopic tangent operators are potentially well predicted by LSTM model despite the complexity of loading conditions (non-proportional) compared to a simple uniaxial tensile test (proportional). For the distribution of the stress fields in the specimen, the results seems qualitatively reasonable considering the loading conditions. Due to the effect of tension, a stress concentration is observed in the specimen reduced cross-sectional area and in its neighboring holes. The effect of combining shear and tension is also captured by the model given the high stress state in zone A (see Fig.4.24), this reduced area is subject to a maximum stress as expected considering the specimen geometry and loading conditions. In the unloading phase, the stress fields are also quite smooth, thus indicating a good numerical convergence. A stress concentration is also observed around the central hole (Zone B in Fig.4.24) with a rather particular distribution due to the shear unloading effect. Based on the present findings, we can assume that FE-LSTM can also be adapted for solving history-dependent multiscale problems, provided that the loading states remains in the LSTM training range. All the examples presented in this section constitute a proof of concept of the use of FE-LSTM approaches to accelerate multiscale simulations of heterogeneous structures. However, it should be kept in mind that many parameters can influence the approach reliability, in particular the microstructure behavior complexity which sometimes may require a lot of data to train the LSTM models. In this study, only octet-truss architected structures with a plastic and isotropic hardening constitutive law were considered without geometric non linearities. Therefore it will be interesting in the future to test this approach by considering several architectured materials such as TPMS with their associate constitutive behavior and including the geometric non linearities in the model (which will highly increase the database generation cost and model complexity).

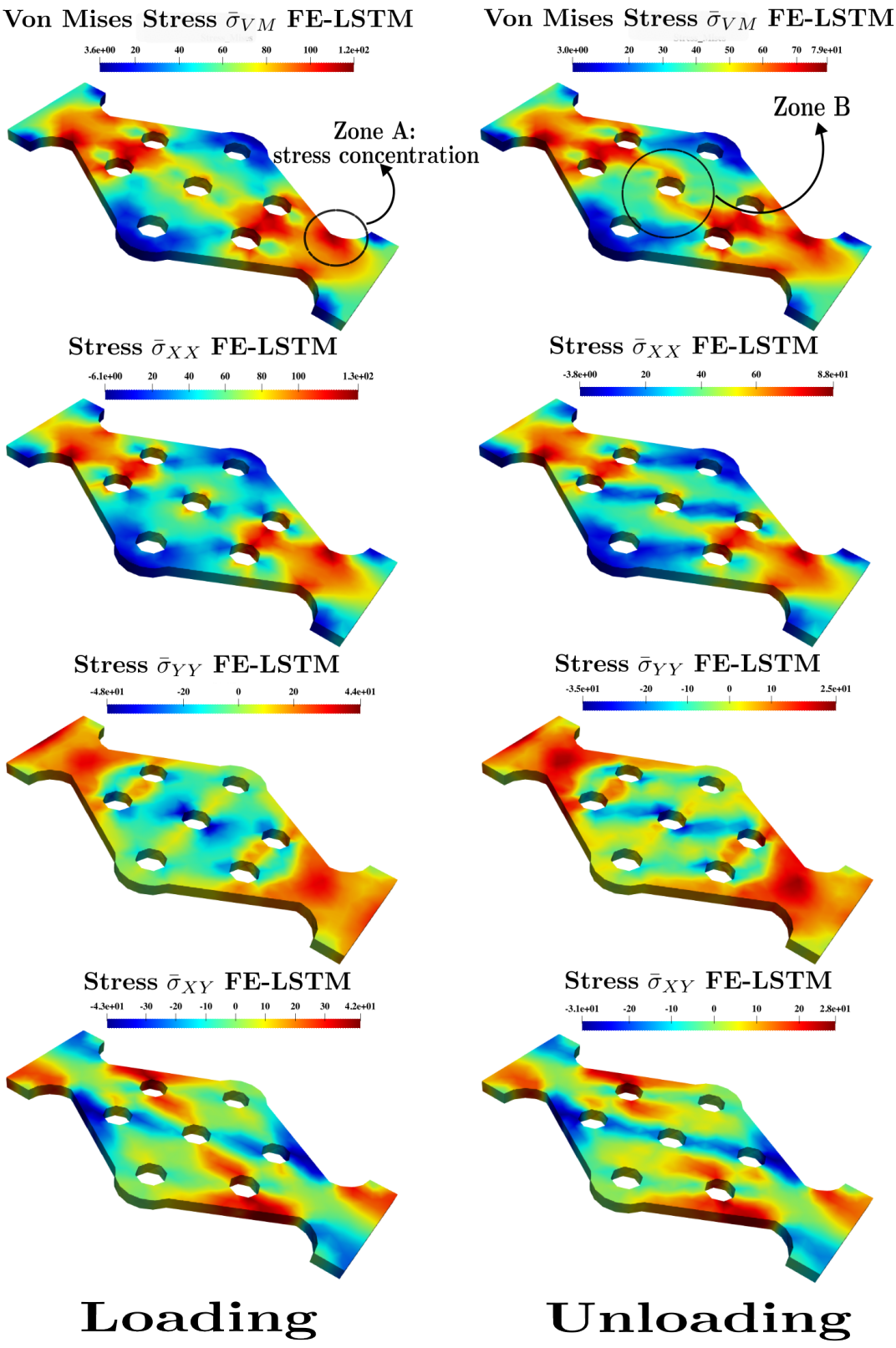


Figure 4.24: The distribution of macroscopic stress fields $\bar{\sigma}$ obtained by FE-LSTM at the end of the loading and unloading steps

4.4.3 Illustration of the achievable capabilities with FE-LSTM model for highly refined meshed architectured structures

The last example of this section is intended to illustrate the achievable performance by FE-LSTM approaches for heterogeneous structures with highly refined mesh. This investigation is motivated by the following reason: Refined meshes (sometimes involving millions of degrees of freedom) and small increments are sometimes required to ensure a numerical convergence structures with complex geometry and boundary conditions. As a consequence, the corresponding computational times can be extremely long. In addition, this kind of simulations are very demanding in terms of computing resources (memory, number of CPUs). It requires sometimes the usage of high performance machines with characteristics that are not easily accessible. We propose here, through a concrete illustrative example, to show the possibility of conducting this kind of multiscale simulations without the use of dedicated numerical resources, i.e only with a desktop computer using the FE-LSTM approach. The example considered in this study is a bracket whose geometry, associated RVE, mesh and boundary conditions are given in figure 4.25. The bracket is meshed using 4-node tetrahedral elements TET4 with 4 Gauss integration points per element. The macrostructure mesh is composed of 282824 Gauss points, which is indeed a very fine mesh. The heterogeneous structure is fixed on one extremity (Face B) and subjected to a compressive loading at the other end (Face A). In figure 4.26, the Von Mises stress $\bar{\sigma}_{VM}$ distribution on the architectured bracket is given. Qualitatively speaking, the obtained results seem very accurate and consistent given the applied loading and boundary conditions. The stress concentration around the central hole is well captured by FE-LSTM model and the distribution of the stress fields is rather smooth.

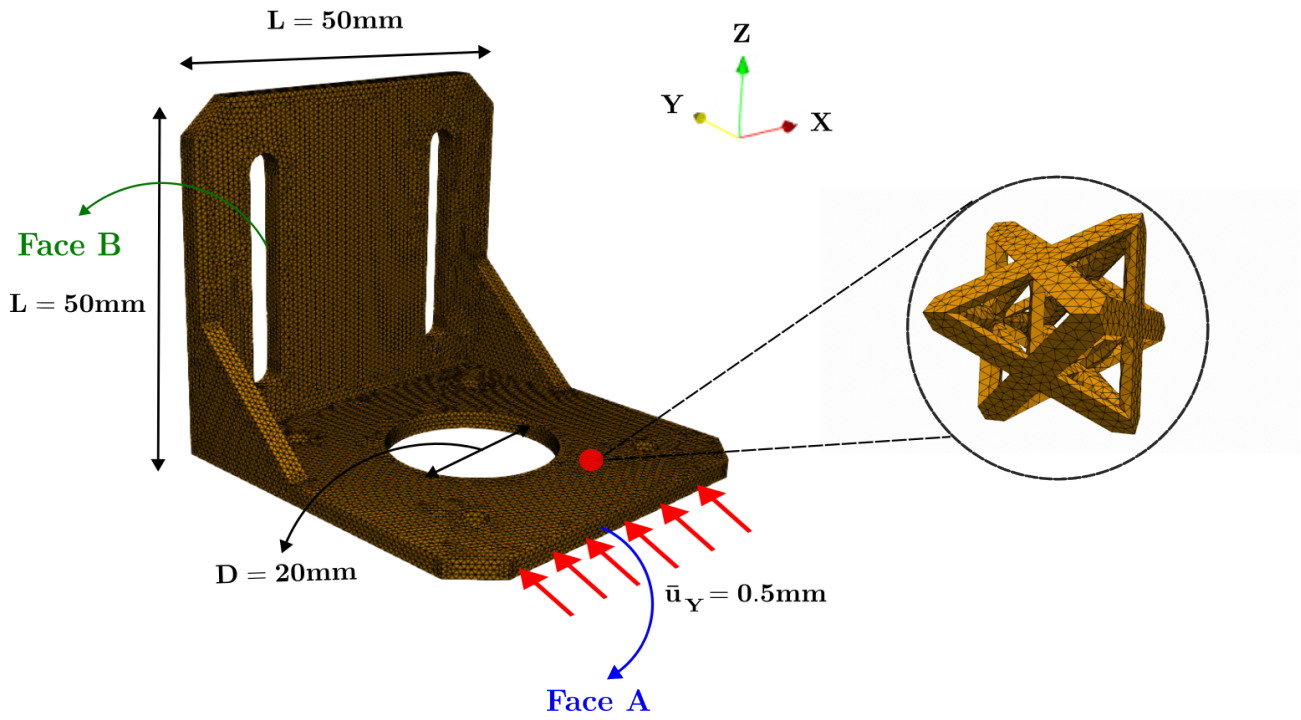


Figure 4.25: Geometry, mesh and boundary conditions of an architectured bracket under a compressive load

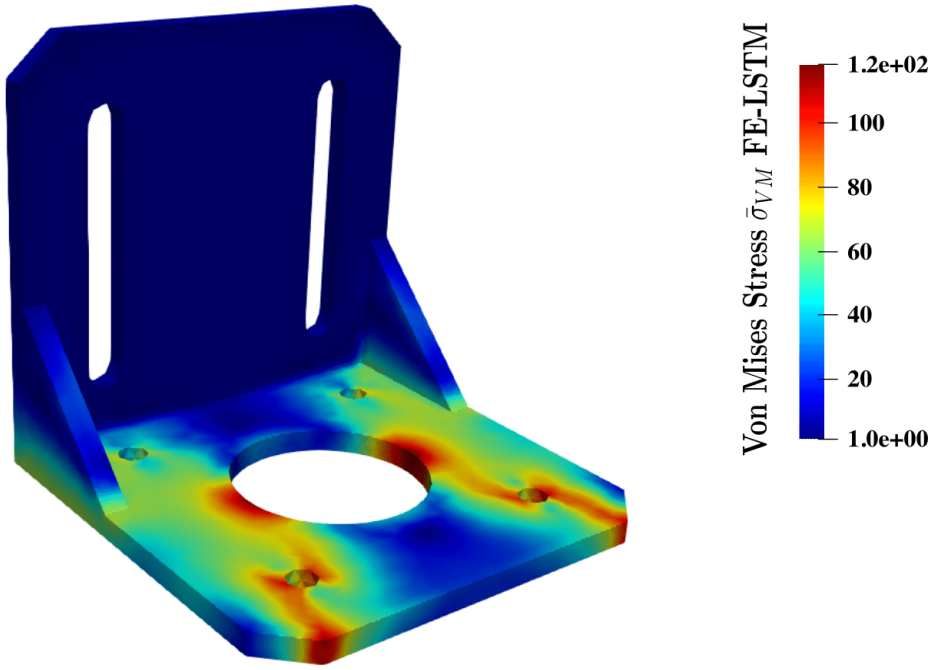


Figure 4.26: Macroscopic Von Misses stress $\bar{\sigma}_{VM}$ distribution

As shown in table 4.10, the execution of this simulation with FE-LSTM model took 35 minutes (using 100 time increments). On the other hand, the same simulation would have taken 105 days using a classic FE^2 with only 10 time steps, thus resulting in a speedup of 4320. Furthermore, if both simulations are conducted with the same number of time increments, this computational time saving factor could be estimated at nearly 40000. This factor is calculated based on the required computational time of FE^2 in the previous examples (standard specimen in subsection 4.4.1 and specimen with holes in subsection 4.4.2). Note that this factor assumes that FE^2 was not optimized for parallel computing and without specific treatment. Another interesting feature of FE-LSTM approach is that the total simulation time including database generation, training phase (5hours) and online simulation (35 min) is still insignificant compared to required FE^2 computational time (105 days). In this kind of application that involves highly refined meshes, there exists a threshold at which it is more interesting to train an LSTM model using a limited number of microscopic evaluations than to perform the entire multiscale simulation using FE^2. In terms of memory usage, the required RAM to compute the homogenized response is estimated 22.58 TB with FE^2 while FE-LSTM only need 2.5 GB, which is compatible with typical desktop computers characteristics. 22 TB of RAM makes the simulation challenging to carry out as even *bigmem* machines are usually limited to 3 TB of RAM. Conducting this type of multiscale simulation require specific treatment of the memory and usually involve domain decomposition methods. To conclude this subsection, we highlighted through an illustrative example the flexibility of FE-LSTM approach for applications involving highly refined (sometimes exaggerated) meshes. Even for these extreme cases, the FE-LSTM method is not

Table 4.10: The required computational time and memory usage to simulate the non-linear response of an architectured bracket structure using FE^2 and FE-LSTM

	FE^2 simulation	**FE-LSTM simulation**	**Computational time saving factor**	**Memory usage saving factor**
Required simulation time	105 days	35 minutes	4320	–
Memory usage	22.58 TB	2.5 GB	–	9032

affected by a significant increase in computational time: the multiscale simulations are still executed within minutes. In addition, they can be easily conducted on desktop computers without the need for clusters or high performance machines.

4.5 A first attempt of using multi-inputs neural networks architectures combining ConvNets and MLP to predict the effective properties of architectured materials

4.5.1 CNN-MLP architecture design

In the previous sections, we have highlighted the predictive capabilities of FE-LSTM based approaches to compute the non-linear behavior of heterogeneous materials. However, we recall that this framework was only applicable under specific conditions: the microstructure (octet-truss) geometry, material properties and constitutive equation was kept the same in all the macrostructure. The only changed parameter was the applied loading conditions on the RVE characterized by the evolution of the macroscopic strain tensor. Therefore, FE-LSTM framework can be applied to any heterogeneous structure provided that microstructure shares the same geometry and material properties. This can be perceived as a potential limitation since flexibility is required while designing heterogeneous structures. For instance, it might be interesting to determine the effect of varying the microstructure topology and material properties on the overall structure behavior. This type of parametric analysis requires reliable models capable of handling multiple data of different natures to quickly predict the corresponding quantities of interest. From this perspective, several deep neural network architectures, which take as inputs different types of data, have been developed in the literature for specific applications. For example, Multi Layer Perceptrons (MLP) are designed to deal with numeric/continuous values, Recurrent Neural Networks (RNN) are specialized in handling sequences and Convolutional Neural Networks (CNN or ConvNets) are intended for images processing. In this section, a first attempt of using a multi-inputs neural networks architecture is conducted to predict the effective properties of architectured materials. The objective is to develop a new model combining a ConvNet and a MLP capable of handling mixed data. To the best of our knowledge, such methodology have never been investigated before in literature for material science applications.

In this work, we aim to establish the relationship between the geometrical/material properties of an architectured material and its effective elastic properties by a CNN-MLP model. CNN-MLP architecture design and general workflow are summarized in figure 4.27. This model is composed of a convolutional neural network and two multi-layers perceptrons. The CNN takes as an input the image of an octet-truss lattice structure having different geometrical parameters such as the diameter of its constituents (inner and outer cylinders). On the other hand, the first feed forward neural network MLP_1 takes another type of data consisting of isotropic materials mechanical properties given by the Young's modulus E and Poisson ratio ν. Afterwards, CNN and MLP_1 outputs are concatenated and fed to another multi layer perceptron MLP_2 which is intended for predicting the effective stiffness tensor components. As a result, CNN-MLP aims to predict the homogenized properties of octet-truss architectured structures only from the knowledge of material properties and an image of the CAD file. By training CNN-MLP model, the meshing phase and FE simulations are no longer required to compute these quantities of interest. Through this case of application, we would like to establish a first proof of concept on the use multi-inputs deep neural networks to treat heterogeneous materials homogenization problems.

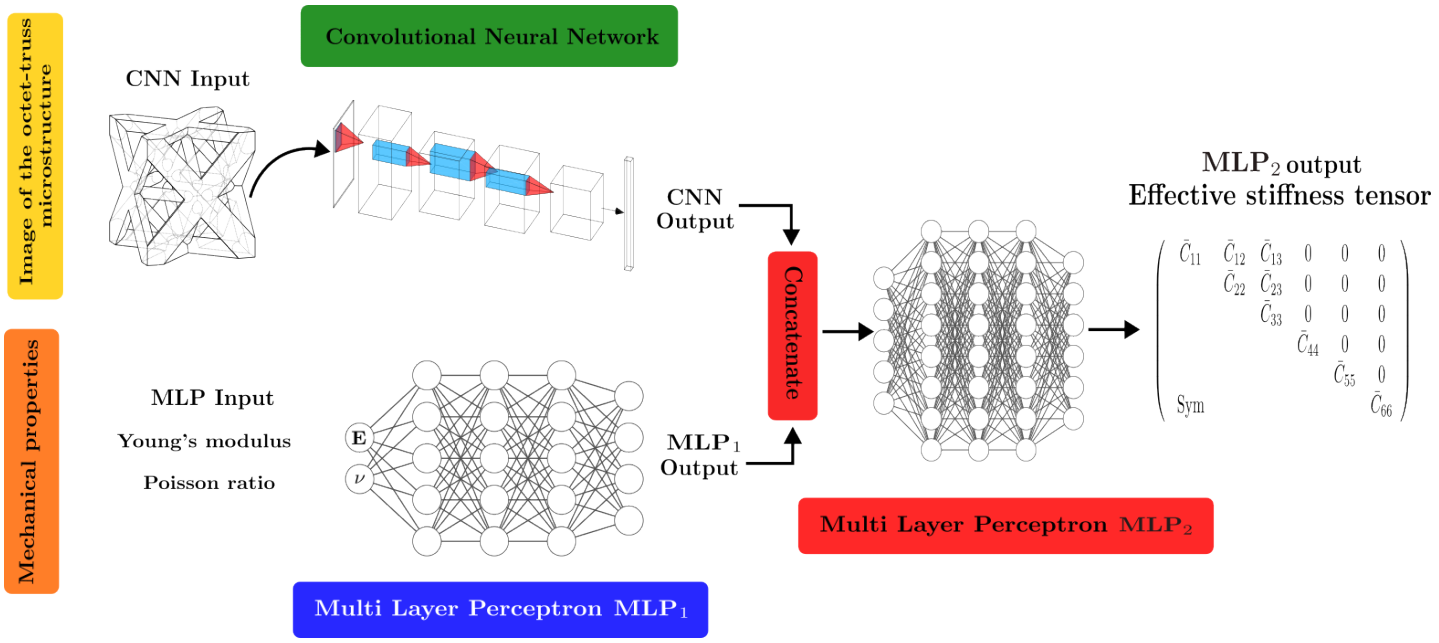


Figure 4.27: CNN-MLP architecture design to predict the effective properties of octet-truss architectured materials

Let us get into the details of each deep neural network architecture on the CNN-MLP model. The first part of CNN-MLP is composed of a convolutional neural network whose architecture including the number and types of layers are illustrated in figure 4.28. ConvNets are a type of ANN capable of extracting features and finding patterns in data that has a known grid-like topology such as images, which consist of a 2D grid of pixels. CNN are inspired by the organization of the visual cortex: its architecture can be analogously seen as neurons connectivity pattern in the human brain. ConvNets have been widely used recently in various engineering fields, particularly in computer vision and image analysis. The

significant advances in self driving vehicles, facial recognition and robotics are a consequence of the major development of AI based approaches including convolutional neural networks. In figure 4.28, the CNN takes as an input a (240×240) pixel image of an octet-truss lattice structure and then process it by applying multiple mathematical operations called convolutions and pooling operations. A CNN model is usually formed by stacking several convolutional and pooling layers whose functions are described below:

- **Convolutional layer:** Convolutional layers are designed to extract important features from the input images using a set of convolutional filters. Their role is to reduce the images into a form which is easier to process, without losing critical features or important information. In this layers, a linear mathematical operation called convolution is applied between an image $\mathbf{I}$ and a Kernel/Filter $\mathbf{K}$ of a specific size $N \times N$. The convolution operation $\mathbf{I} * \mathbf{K}$ consists on sliding the filter $\mathbf{K}$ over the image and then applying a dot product between $\mathbf{K}$ and the parts of the input image with respect to the size of the filter $N \times N$. Figure 4.29 shows an example of a convolution operation between an image of (7×7) pixels and a filter of size (3×3). After the convolution operations, non linear activation functions such as ReLU are commonly used to break up linearity. In the CNN-MLP architecture, 3 convolutional layers with 32, 64 and 64 filters of size (3×3) are used (see figure 4.28).

- **Pooling layers:** Convolutional layers are commonly followed by pooling layers. Their primary role is to optimize the number of parameters by decreasing the dimension of convoluted feature maps, this allows to reduce the computational costs. This operation can be performed using two types of pooling methods, Max Pooling and Average Pooling. An example of Max Pooling is given in figure 4.30, the maximum value of each (2×2) block is taken from the input feature to construct the pooling layer. In Average Pooling, the average is computed in each blocks. In CNN-MLP architecture, 3 Max pooling layers with (2×2) blocks are employed (see figure 4.28). The last step in ConvNets is flattening, this operation consist on taking 3D tensors resulting from convolution and pooling operations and then convert it into one dimensional array. This is important since multi layer perceptrons can only handle 1D arrays.

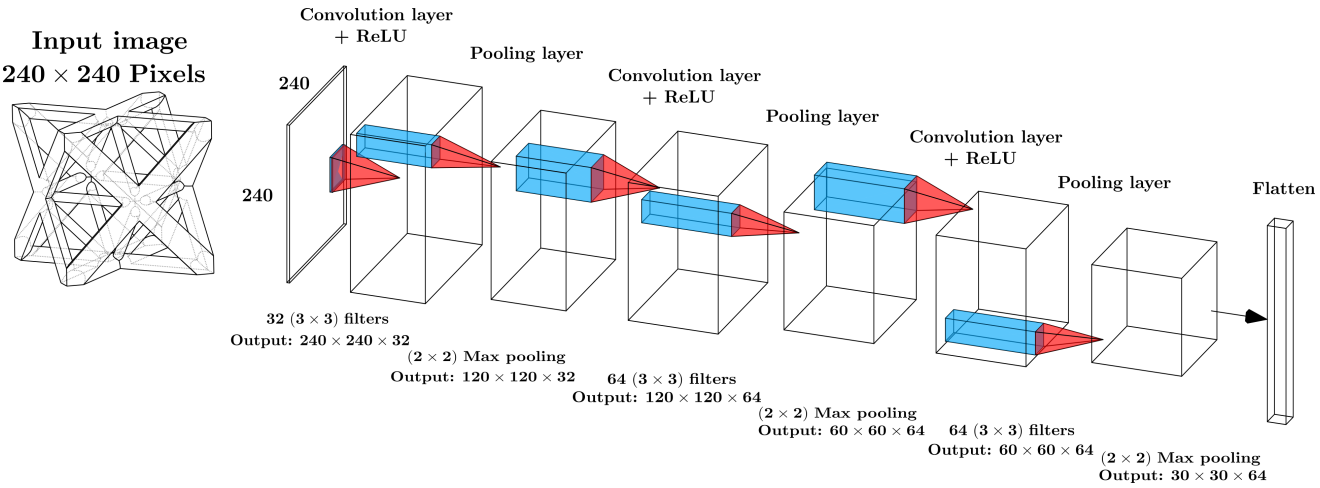


Figure 4.28: CNN architecture design in CNN-MLP model

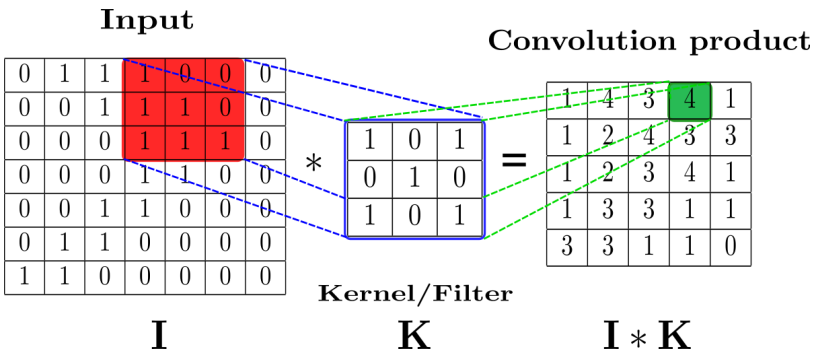


Figure 4.29: An example of a convolution operation between an image of (7×7) pixels and a filter of (3×3) size

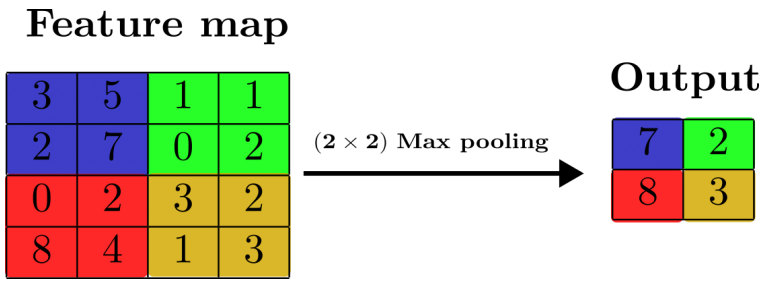


Figure 4.30: An example of (2×2) Max Pooling process

4.5.2 Database generation and training process

The database generation of CNN-MLP model is processed as follows: First, 100 octet-truss microstructures with different geometrical parameters are generated using a python library *Microgen* [Prulière & Chemisky 2022]. Using a dedicated script, *Microgen* allows to create each microstructure geometry using *CadQuery*, a python library that builds 3D CAD models. The resulting microstructures are then meshed using *GMSH* [Geuzaine, Christophe and Remacle, Jean-Francois]. In the present work, the type of architectured structure is fixed to the octet-truss lattices, the changed parameters are the radius of the external R_e and internal R_i cylinders that compose the octet-truss. R_e and R_i are chosen randomly from the interval $\mathbb{I}_R = [0.02, 0.2]$. Figure 4.31 shows examples of the architectured samples with various R_e and R_i values used as training data. Once the 3D CAD models are created, a 2D (240×240) pixels image of the microstructure is generated and saved as .png file, which is the input the ConvNet. As shown from figure 4.31 images, the octet-truss position is set in a view where maximum information about the microstructure geometry and patterns can be captured by the CNN model. The next step consists on conducting 1000 FE simulations for each generated octet-truss while varying the elastic properties E and ν. The input parameter E is chosen randomly on a logarithmic space from in interval $\mathbb{I}_E = [10^4, 10^7]$. On the other hand Poisson ratio ν follows a uniform distribution within a range of $\mathbb{I}_\nu = [0.01, 0.49]$. Using a specific module in *Fedoo* FE code, the resulting homogenized stiffness matrices are computed and stored in the database. In total, 100000 samples (100 microstructure $\times$ 1000 FE simulation) are used to train the CNN-MLP model, the images ares processed by the CNN and the material properties serve as the MLP inputs.

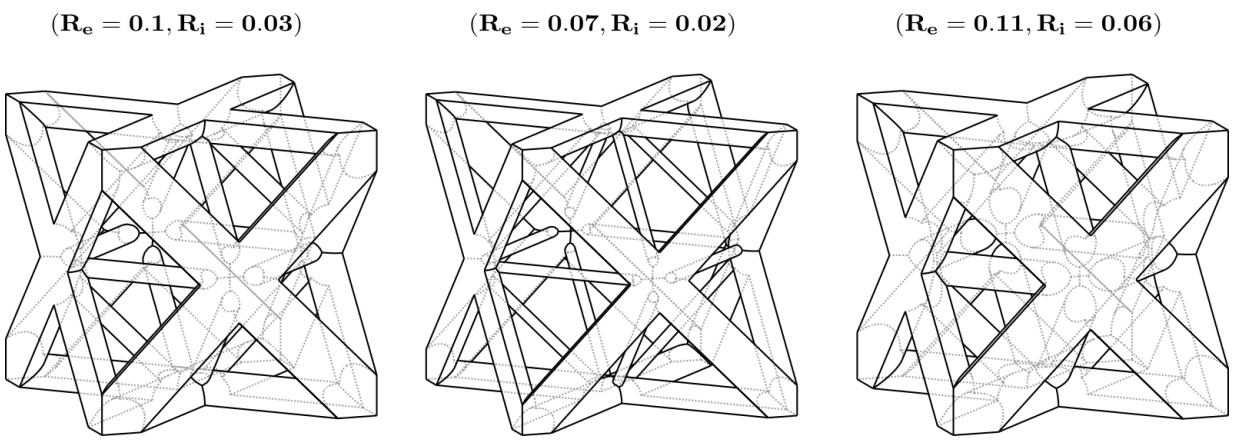


Figure 4.31: Examples of Octet-truss architectured materials with different geometrical parameters used as training images data

Following the database generation phase, we proceed to the training process of CNN-MLP model. All the programs were implemented in Python 3 with the help of Keras library [Chollet *et al.* 2015] and Tensorflow API [Abadi *et al.* 2015]. The database was split into a training set $\overline{\mathscr{T}}$ (90% of total samples), a validation set $\overline{\mathscr{V}}$ (20% of $\overline{\mathscr{T}}$) and a testing set $\overline{\mathcal{T}}$ (10% of total samples). The training phase requires a parameter calibration of each deep neural network that constitute the CNN-MLP model. For the multilayer perceptrons MLP_1 and MLP_2, the hidden layers weights matrices and bias vectors are adjusted after each epoch. Regarding the CNN, the trained parameters are the filters pixel values, which are learned by minimizing a cost function using backpropagation algorithm. In this application, the Mean Squared Error (MSE) was employed as a loss function and Adam (optimizer with adaptative moment) was used as an optimizer. Before starting the training process, $\overline{\mathscr{T}}$ was first partitioned into 90 batches of 1000 samples each. This choice is explained by the fact that it is impossible to feed CNN-MLP with the entire data set (90000 image samples) because of the high memory requirements. The data is therefore divided into several sets that will progressively enrich CNN-MLP model during the learning phase. The evolution of CNN-MLP training and validation MSE with respect to epochs is illustrated in figures 4.32 and 4.33 respectively. One can notice from figure 4.32 a regular decrease of the MSE as more batches are fed into the neural network, thus indicating that CNN-MLP is able to capture additional information and new patterns from images for each provided set. Regarding the validation error (see figure 4.33), the MSE evolution is much more noisy and exhibits a different behavior compared to the training MSE, the error decrease is not stable and consistent with respect to the supplied batches. However, this pattern only lasts during the first 50 batches where the model is attempting to adjust model parameters. In the last 40 batches, the validation MSE trend is considerably more steady and consequently it reaches an error value of order of magnitude 10^{-3} at the end of batch 90. This resulting MSE is similar to the training error, thus indicating that CNN-MLP has successfully achieved a good generalization capability and has been prevented from overfitting.

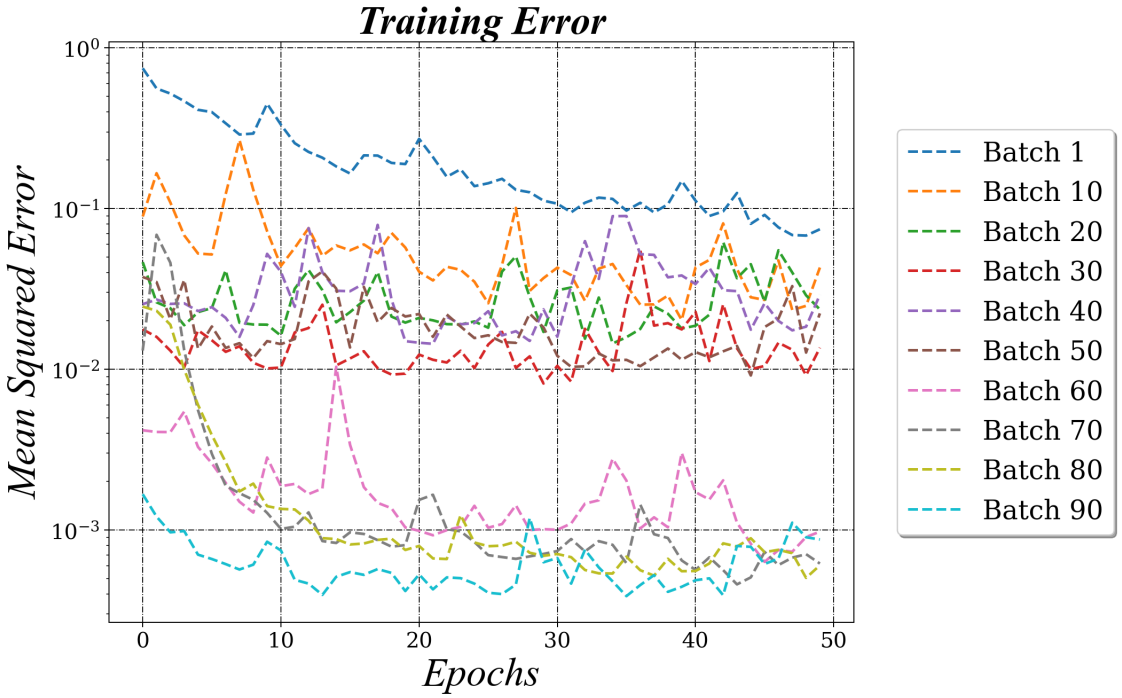


Figure 4.32: Evolution of CNN-MLP training MSE as a function of epochs for several batches of data

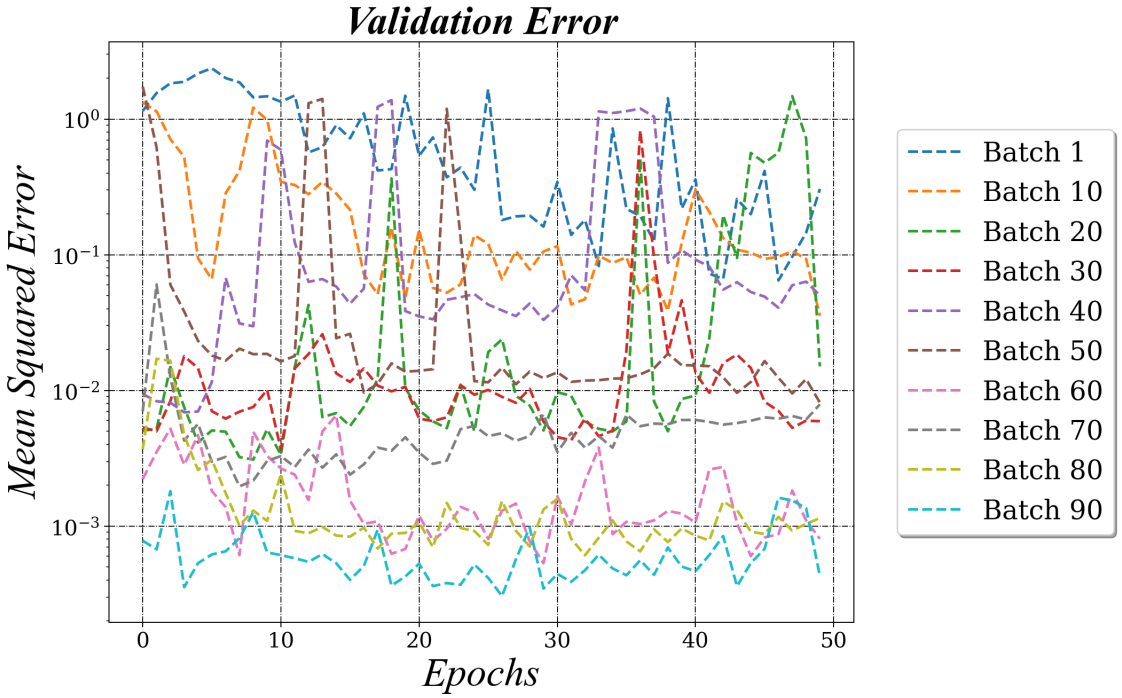


Figure 4.33: Evolution of CNN-MLP validation MSE as a function of epochs for several batches of data

4.5.3 Numerical results

In order to assess CNN-MLP reliability and generalization capabilities while predicting the effective properties of new microstructures with various material properties, it is crucial to evaluate its performance on the test dataset $\overline{\mathcal{T}}$. To quantify CNN-MLP model accuracy on all the tested data, the metric Mean Absolute Percentage Error (MAPE) is evaluated. Figure 4.34 shows the computed MAPE values for each effective stiffness tensor component predicted by the CNN-MLP. It can be observed from figure 4.34 that the errors order of magnitude is almost similar for all the components with a maximum value of nearly 8.5%. This first finding is an evidence that CNN-MLP has been potentially prevented from overfitting problems. If this were not the case, the model would have the tendency to assign certain attention (larger weights) to given neurons associated to specific outputs compared to the others, therefore large discrepancies in MAPE values would have been observed as a consequence. In addition, the resulting errors remains reasonable given the complexity of the problem. We recall that CNN-MLP involves three different neural network architectures with hundred of thousands of trainable parameters. Even though, the model still achieved an acceptable performance with a relatively small quantity of data compared to the required amount of data (sometimes involving millions of images) to train CNN models. Figure 4.35 illustrates some test data examples of the predicted effective tensor components by CNN-MLP in comparison to FE simulations. As shown in figure 4.35, a good agreement is obtained between both quantities. Through this application, CNN have shown a great potential of capturing higher-order spatial correlations and patterns from images microstructure. This outcome assess the high capabilities of ConvNets as reliable computer vision models capable of establishing structure-property linkages for architectured materials. By calibrating the appropriate pixel values of CNN filters and combining them with the fully connected neural networks MLP_1 and MLP_2, CNN-MLP demonstrated its ability to solve both a classification and a regression problem simultaneously.

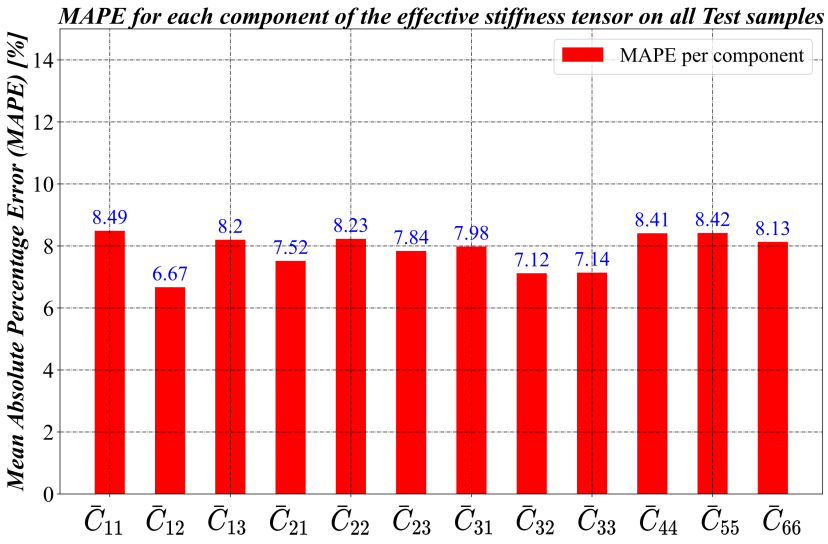
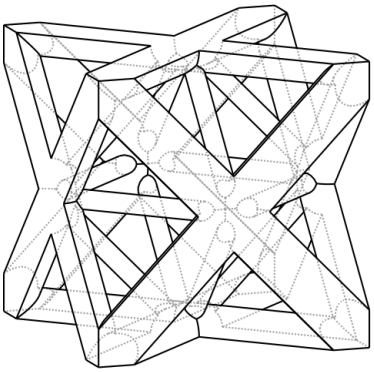
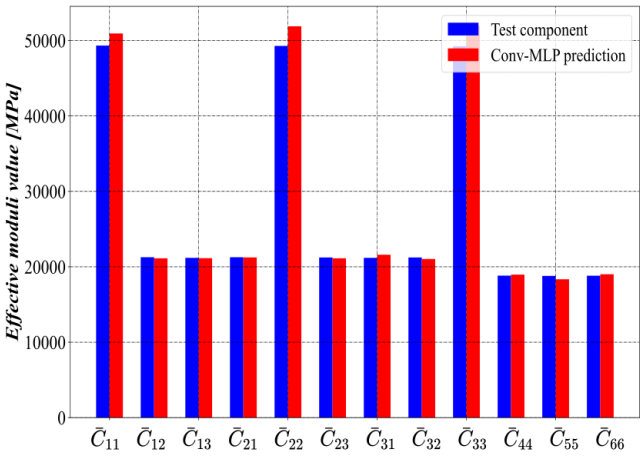
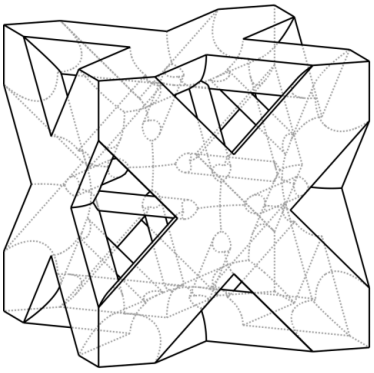


Figure 4.34: Evaluation of the MAPE values for each CNN-MLP predicted components of the effective stiffness tensor on all the test data

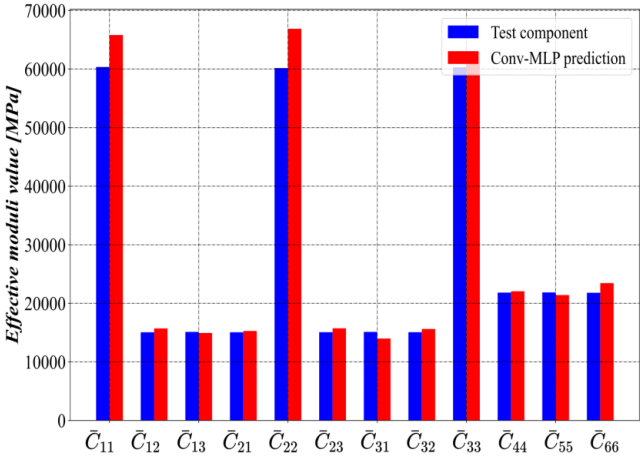


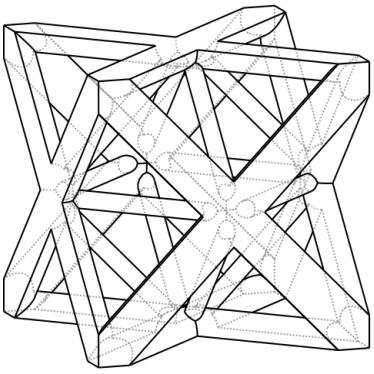
$(E = 4{,}05.10^5 Mpa,\ \nu = 0.41)$





$(E = 1{,}23.10^5 Mpa,\ \nu = 0.22)$





$(E = 1{,}95.10^6 Mpa,\ \nu = 0.05)$

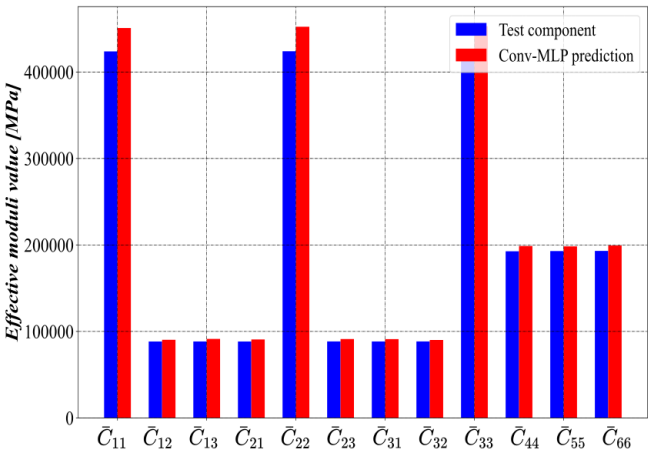


Figure 4.35: CNN-MLP predicted effective stiffness tensor components in comparison to the targets obtained by FE simulation for three examples of test data

In the last section of this chapter, we have established through a practical case of study the potential of multi-inputs deep neural networks to predict the effective properties of architectured materials. This proof of concept offers the possibility of extending this methodology to address more complex problems, thus pushing further the achievable performances with such approach. For instance, we could envision the combination of ConvNets with LSTM neural networks to simulate a wide range of architectured materials. Given an initial 2D or 3D image of the microstructure and a time sequence corresponding to the applied mechanical loading, CNN-LSTM can be used to predict the evolution of the heterogeneous material mechanical response. However, training of this type of models will necessarily require a significant amount of data. To overcome this challenge, several approaches can be adopted such as Image Data Augmentation. This technique consists on artificially increasing the size of the database by creating modified versions of the training images. By applying specific transformations such as zooms, rotation and shifts, new variations of the images can be used to expand the dataset. This means that new versions of the same images are likely to be seen by the model during the training process, and therefore CNN would be able to capture as much information about the microstructure patterns as possible. Furthermore, we recall that ConvNets are capable of learning features that are invariant to their location in the image, thus rendering the data augmentation technique very effective for convolutional neural networks.

4.6 Conclusions

The last chapter of the present manuscript was dedicated to the development of new numerical modeling strategies using deep neural networks to accelerate multiscale simulation of architectured materials. Through the design of the so called FE-LSTM approach, we have highlighted the strong capabilities of such method to accurately and quickly predict the effective mechanical response of heterogeneous structures under proportional and non proportional loading paths. In order to demonstrate the flexibility of FE-LSTM approach, several architectured structures from simple to complex macroscopic geometries have been tested. The obtained results were very promising given the excellent agreement between the predicted macroscopic fields by FE-LSTM and by FE^2 simulations. The main advantage of such approach lies in the fact that once the training process of the RNN is achieved, FE-LSTM can be applied to any heterogeneous structure having the same microstructure. In contrast, the multi-level FE^2 method requires solving once again non-linear problems on each Gauss point at each iteration for any new simulation. As a result, massive computational costs saving factors have been achieved. In terms of execution time, FE-LSTM simulations can be run within seconds compared to several days with FE^2, thus resulting in a speedup factors of nearly 40000. In addition, the simulations can be easily conducted on desktop computers without requiring high computing resources. We have demonstrated that the resulting memory usage can reach saving factors up to 10000. Finally, a first attempt of using multi-inputs neural networks have been carried out to predict the effective properties of architectured materials. A CNN-MLP model combining convolutional neural networks and multi layer perceptrons have been developed to establish microstructure-property linkages of octet-truss structures. The main motivation behind this application was to demonstrate the achievable capabilities with such methodology that involves data from different natures to predict several quantities of interest. To the best of our knowledge, such approach have never been investigated before for material science applications. Therefore, this present work findings constitute a first proof of concept of combining multiple deep neural networks architectures that takes mixed data

as inputs and predict the corresponding mechanical fields and homogenized properties as outputs.

Conclusion and perspectives

Within this thesis, we aimed at developing novel modeling strategies of the nonlinear mechanical behavior of heterogeneous materials using hybrid approaches that combines physical knowledge and deep learning based methods. This sort of techniques can be perceived as a change of paradigm compared to traditional solutions which remains sometimes limited for structural analysis applications. Various artificial neural network architectures have been explored from Multi Layer Perceptrons (MLP), Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN). Each model has been employed to address a specific challenge in the numerical simulation of heterogeneous structures.

For the sake of clarity, we present, in the form of a list, the major contributions of this thesis.

- We have proposed a new and original approach to train ANN models by introducing regularization constraints based on physical laws. In Chapter 2, a Thermodynamically Consistent Recurrent Neural Network (ThC-RNN) model has been developed to serve as a potential surrogate for dissipative materials constitutive modeling. This hybrid approach consists on combining a specific RNN architecture called Long Short Term Memory (LSTM) with thermodynamical laws. The robustness of ThC-RNN comes primarily from the powerful predictive capabilities of LSTM to capture history dependant behaviors as well as overcoming vanishing gradient problems. In addition, the incorporation of physical laws and specifically the second law of thermodynamics in the training process have provided relevant and consistent results compared to pure black-box RNN. ThC-RNN model reliability have been tested on different plasticity models including elastoplasticity with isotropic hardening and Chaboche plasticity model with a combined isotropic hardening and two nonlinear kinematic hardening. It has been shown that thermodynamic consistency have improved significantly ThC-RNN predictive capabilities considering several quantities of interest including stress and tangent stiffness tensors components, internal state variables and mechanical work rate partition (recoverable part, irrecoverable part and dissipative part). Finally, we demonstrated that ThC-RNN can be integrated into a FEM framework to simulate structures under real-life conditions provided that certain points are verified: 1) Check models generalization capabilities when predicting the tangent operators and mechanical responses, 2) Verify that ThC-RNN inputs remain in the training range to avoid inaccurate predictions or a failure in the convergence of simulation, 3) Incorporate physics based laws into model training process to regularize the predicted mechanical fields with respect to thermodynamic consistency.

- Hybrid models potential for homogenization applications have been highlighted. Through the development of ANN-φ, a hybrid approach combining feed forward neural networks and Mori-Tanaka micromechanical scheme, we aimed at evaluating the reliability of such method to predict the effective properties of heterogeneous materials. ANN-φ consisted in splitting the homogenization problem into two parts. First, a MLP is designed to quickly predict Eshelby's tensors from the knowledge of ellipsoidal inclusion geometric and material properties. The predicted outputs are used afterwards to compute localization and effective stiffness tensors within Mori-Tanaka homogenization scheme. The use of this hybrid model have provided a good compromise between predictions reliability and

computational time saving which can reach a factor of 2000 in the case of heterogeneous materials with a large number of heterogeneities.

- We elaborated a novel modeling strategy to accelerate multiscale simulations of heterogeneous materials using deep neural networks. The developed approach, called FE-LSTM, consists on combining the finite element method and LSTM recurrent neural networks to solve multiscale problems. In contrast to FE^2 method, the FE resolution of the microscopic problems is no longer required within FE-LSTM framework, the computation of RVE homogenized response is directly predicted by an LSTM trained on a database of offline micro simulations. Consequently, the main advantage of FE-LSTM lies in the massive computational time and resources saving when solving the microscopic problems. To assess the validity and reliability of the developed approach, FE-LSTM has been evaluated on several 3D architectured structures from rather simple to complex geometries under proportional and non proportional loading conditions. The obtained results have demonstrated FE-LSTM high predictive capabilities, an excellent agreement has been found between the macroscopic fields predicted by FE-LSTM and the targets computed by FE^2. The usefulness of such approach lies in the fact that one the RNN training process is performed, FE-LSTM can be applied to simulate nonlinear behaviors of any heterogeneous structure having the same microstructure used in training. By performing parametric analysis while testing a variety of macroscopic geometries and loading conditions, such approach find its interest in the design of heterogeneous structures as it allows to conduct multiple multiscale simulations in a short amount of time. In terms of execution time, we have highlighted through the previous application that FE-LSTM simulation are performed within seconds compared to days with FE^2, thus resulting in computational time saving factors of nearly **40000**. Furthermore, they can be easily conducted on desktop computers without the need for HPC clusters.

- We established a first proof of concept of using multi-inputs neural networks architectures combining Convolutional Neural Networks (CNN) and Multi Layer Perceptrons (MLP) to predict the effective properties of architectured materials. We have shown that it is possible to simulate the homogenized behavior of heterogeneous structures only from the knowledge of the microstructure image and the numerical values of the material properties, therefore without the need of a mesh or a FE simulation. To the best of our knowledge, such methodology have never been investigated before for material science applications. In chapter 4, CNN-MLP model have been developed to determine microstructure-property linkages of octet-truss structures. The ConvNet is intended for images processing of Octet-truss lattice structures having different geometrical parameters. On the other hand, the MLP part takes as inputs another type of data which is materials mechanical properties. CNN and MLP outputs are concatenated and fed to another feed forward neural network which is designed to predict the effective properties of the Octet-Truss. The obtained results were promising, the errors order of magnitude remained reasonable despite the complexity of the treated problem.

The modeling strategies developed in the present thesis have led to considerable advances in the numerical simulation of heterogeneous structures. Nevertheless, several research directions are still to be explored in order to push further the achievable performances with such methodologies. The prospects of this thesis are detailed as follows:

1. The incorporation of physical based laws in the training process of FE-LSTM approaches similarly to ThC-RNN model. Even though FE-LSTM achieved very promising performances based only on RNN high predictive capabilities and abilities to capture history dependant behaviors, such approach may potentially face some limitation when applied to complex microstructures with strongly nonlinear material laws. The introduction of physics in the training phase is therefore required for the regularization of the solution. However, an additional challenge has to be faced when using such approach for multiscale analysis as there is no closed form of the second law of thermodynamics at the RVE scale contrary to a material point case with a well defined constitutive law. A potential solution to overcome this issue is to add a new constraint on the loss function based on the macroscopic mechanical work. Using the micro-macro energy equivalence (Hill-Mandel lemma), the idea would be to constraint the quantity $\bar{\sigma} : \Delta\bar{\varepsilon}$ to be equal to $\Delta\bar{W}_m$ during the training process. $\Delta\bar{W}_m$ correspond to the incremental macroscopic mechanical work at the RVE scale which can be computed by FE analysis and stored in the data base. By forcing the connection between $\bar{\sigma} : \Delta\bar{\varepsilon}$ and $\Delta\bar{W}_m$ during the training phase, it is possible to provide an additional information about the macroscopic fields, which could result in more accurate solutions compared to pure RNN approach.

2. Although the present study focused only on octet-truss lattices with an elastoplastic and isotropic hardening constitutive law, it will be more valuable to conduct further testing on several architectured materials such as Triply Periodic Minimal Surfaces (TPMS) with their associate material laws. The generalization capabilities of LSTM neural networks are worth to be investigated for more complex behaviors including rate dependant constitutive laws such as viscoelasticity and viscoplasticity. Moreover, the incorporation of geometric non linearities in simulations is likely to lead to additional challenges regarding the database generation costs and model selection. Therefore, such cases require further investigation to properly define the domain of validity and the limitations of the developed approach. As more testing is performed, FE-LSTM reliability can be assessed further.

3. Define a quantification of uncertainties strategy for deep learning methods using Bayesian techniques similarly to Gaussian Random Process (GRP). The deployment of deep learning based approaches in structural analysis requires the development of a well defined methodology capable of measuring the uncertainties inherited from each step of the process (data collection, pre-processing, hyperparameter selection etc ...).

4. Extend the CNN-MLP approach to address more complex problems. A potential application would be to combine ConvNets with LSTM neural networks to simulate a wide range of architectured materials accounting for their non linear behavior. Given an initial 2D or 3D image of the microstructure and a time sequence corresponding to the applied mechanical loading, CNN-LSTM can be used to predict the non linear evolution of the heterogeneous material mechanical response.

Bibliography

[Abadi *et al.* 2015] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu and Xiaoqiang Zheng. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, 2015. Software available from tensorflow.org. (Cited on pages 49, 82, 113 and 136.)

[Abueidda *et al.* 2021] Diab W. Abueidda, Seid Koric, Nahil A. Sobh and Huseyin Sehitoglu. *Deep learning for plasticity and thermo-viscoplasticity.* International Journal of Plasticity, vol. 136, page 102852, jan 2021. (Cited on page 34.)

[Ali *et al.* 2019] Usman Ali, Waqas Muhammad, Abhijit Brahme, Oxana Skiba and Kaan Inal. *Application of artificial neural networks in micromechanics for polycrystalline metals.* International Journal of Plasticity, vol. 120, no. May, pages 205–219, 2019. (Cited on pages vii, 20, 21 and 34.)

[Ashby 1999] Michael F. Ashby. *Materials Selection in Materials Design.* Elsevier, 1999. (Cited on page 102.)

[Ashby 2013] Mike Ashby. *Designing architectured materials.* Scripta Materialia, vol. 68, pages 4–7, 1 2013. (Cited on pages x, 102 and 103.)

[Barthélémy 2020] J. F. Barthélémy. *Simplified approach to the derivation of the relationship between Hill polarization tensors of transformed problems and applications.* International Journal of Engineering Science, vol. 154, page 103326, sep 2020. (Cited on page 76.)

[Bebis & Georgiopoulos 1994] G. Bebis and M. Georgiopoulos. *Feed-forward neural networks.* IEEE Potentials, vol. 13, no. 4, pages 27–31, October 1994. (Cited on page 15.)

[Bergstra & Bengio] James Bergstra and Yoshua Bengio. *Random Search for Hyper-Parameter Optimization.* page 25. (Cited on page 18.)

[Bergstra *et al.*] J Bergstra, D Yamins and D D Cox. *Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures.* page 9. (Cited on page 18.)

[Bergstra *et al.* 2011] James S. Bergstra, Rémi Bardenet, Yoshua Bengio and Balázs Kégl. *Algorithms for Hyper-Parameter Optimization.* In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira and K. Q. Weinberger, editors, Advances in Neural Information Processing Systems 24, pages 2546–2554. Curran Associates, Inc., 2011. (Cited on page 18.)

[Bertoldo *et al.* 2021] João P.C. Bertoldo, Etienne Decencière, David Ryckelynck and Henry Proudhon. *A Modular U-Net for Automated Segmentation of X-Ray Tomography Images in Composite Materials.* Frontiers in Materials, vol. 8, nov 2021. (Cited on pages vii, 22 and 23.)

[Bidaine *et al.* 2015] Benoît Bidaine, Laurent Adam, Pierre Charrier, Ludovic Chauvet, Issam Doghri, Sarra Haouala, Anouar KRAIRI, Jean-François Luye, Olivier Moulinjeune, Laurent Perret, G. Robert, Denis Taveau and Roger Assaker. *Fiber Reinforced Plastic Durability: Nonlinear Multi-Scale Modeling for Structural Part Life Prediction.* 09 2015. (Cited on page 98.)

[Botev *et al.* 2016] Aleksandar Botev, Guy Lever and David Barber. *Nesterov's Accelerated Gradient and Momentum as approximations to Regularised Update Descent.* Proceedings of the International Joint Conference on Neural Networks, vol. 2017-May, pages 1899–1903, jul 2016. (Cited on page 19.)

[Breiman 2001] Leo Breiman. *Random Forests.* Machine Learning 2001 45:1, vol. 45, no. 1, pages 5–32, oct 2001. (Cited on page 25.)

[Browne 2000] Michael W Browne. *Cross-Validation Methods.* Journal of Mathematical Psychology, vol. 44, no. 1, pages 108–132, March 2000. (Cited on page 18.)

[Bulgarevich *et al.* 2018] Dmitry S. Bulgarevich, Susumu Tsukamoto, Tadashi Kasuya, Masahiko Demura and Makoto Watanabe. *Pattern recognition with machine learning on optical microscopy images of typical metallurgical microstructures.* Scientific Reports 2018 8:1, vol. 8, no. 1, pages 1–8, feb 2018. (Cited on page 27.)

[Bulgarevich *et al.* 2019] Dmitry S. Bulgarevich, Susumu Tsukamoto, Tadashi Kasuya, Masahiko Demura and Makoto Watanabe. *Automatic steel labeling on certain microstructural constituents with image processing and machine learning tools.* Science and Technology of Advanced Materials, vol. 20, no. 1, pages 532–542, dec 2019. (Cited on page 27.)

[Cai *et al.* 2021] Shengze Cai, Zhicheng Wang, Sifan Wang, Paris Perdikaris and George Em Karniadakis. *Physics-informed neural networks for heat transfer problems.* Journal of Heat Transfer, vol. 143, no. 6, jun 2021. (Cited on page 30.)

[Cai *et al.* 2022] Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin and George Em Karniadakis. *Physics-informed neural networks (PINNs) for fluid mechanics: a review.* Acta Mechanica Sinica 2021 37:12, vol. 37, no. 12, pages 1727–1738, jan 2022. (Cited on page 30.)

[Caruana *et al.* 2001] Rich Caruana, Steve Lawrence and C. Lee Giles. *Overfitting in Neural Nets: Backpropagation, Conjugate Gradient, and Early Stopping.* In T. K. Leen, T. G. Dietterich and V. Tresp, editors, Advances in Neural Information Processing Systems 13, pages 402–408. MIT Press, 2001. (Cited on page 18.)

[Cecen *et al.* 2018] Ahmet Cecen, Hanjun Dai, Yuksel C. Yabansu, Surya R. Kalidindi and Le Song. *Material structure-property linkages using three-dimensional convolutional neural networks.* Acta Materialia, vol. 146, pages 76–84, mar 2018. (Cited on page 23.)

[Chatzigeorgiou *et al.* 2018] George Chatzigeorgiou, Nicolas Charalambakis, Yves Chemisky and Fodil Meraghni. Thermomechanical behavior of dissipative composite materials. 2018. (Cited on pages 35 and 36.)

[Chemisky 2018] Yves Chemisky. *SIMCOON : OPEN SOURCE SOFTWARE IN MECHANICS AND MATERIALS | Yves Chemisky | 2 updates | Research Project.* 2018. (Cited on page 79.)

[Chen *et al.* 2020] Qiang Chen, Wenqiong Tu and Meng Ma. *Deep learning in heterogeneous materials: Targeting the thermo-mechanical response of unidirectional composites.* Journal of Applied Physics, vol. 127, no. 17, page 175101, may 2020. (Cited on page 23.)

[Chinesta *et al.* 2014] Francisco Chinesta, R. Keunings and Adrien Leygue. *The proper generalized decomposition for advanced numerical simulations : a primer.* page 117, 2014. (Cited on page 11.)

[Chollet *et al.* 2015] François Chollet*et al. Keras.* `https://github.com/fchollet/keras`, 2015. (Cited on pages 49, 113 and 136.)

[Coelho *et al.* 2011] Pedro Gonçalves Coelho, Paulo Rui Fernandes and Helder Carriço Rodrigues. *Multiscale modeling of bone tissue with surface and permeability control.* Journal of Biomechanics, vol. 44, pages 321–329, 1 2011. (Cited on page 104.)

[Cortes *et al.* 2012] Corinna Cortes, Mehryar Mohri and Afshin Rostamizadeh. *L2 Regularization for Learning Kernels.* Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence, UAI 2009, pages 109–116, 5 2012. (Cited on page 14.)

[Daw *et al.* 2017] Arka Daw, Anuj Karpatne, William Watkins, Jordan Read and Vipin Kumar. *Physics-guided Neural Networks (PGNN): An Application in Lake Temperature Modeling.* oct 2017. (Cited on page 34.)

[Dirrenberger *et al.* 2011] J. Dirrenberger, S. Forest, D. Jeulin and C. Colin. *Homogenization of periodic auxetic materials.* Procedia Engineering, vol. 10, pages 1847–1852, 1 2011. (Cited on page 102.)

[Doreswamy 2011] Doreswamy. *Linear Regression Model for Knowledge Discovery in Engineering Materials.* Computer Science Conference Proceedings, vol. 3, pages 147–156, sep 2011. (Cited on page 15.)

[Duan *et al.* 2006] H L Duan, Y Jiao, X Yi, Z P Huang and J Wang. *Solutions of inhomogeneity problems with graded shells and application to core–shell nanoparticles and composites.* Journal of the Mechanics and Physics of Solids, vol. 54, no. 7, pages 1401–1425, 2006. (Cited on page 71.)

[Duchi *et al.* 2011] John Duchi, Elad Hazan and Yoram Singer. *Adaptive Subgradient Methods for Online Learning and Stochastic Optimization.* Journal of Machine Learning Research, vol. 12, no. 61, pages 2121–2159, 2011. (Cited on page 19.)

[Eshelby 1957] J. D. Eshelby. *The Determination of the Elastic Field of an Ellipsoidal Inclusion, and Related Problems.* Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, vol. 241, no. 1226, pages 376–396, 1957. (Cited on page 71.)

[Eshelby 1959] J. D. Eshelby. *The Elastic Field Outside an Ellipsoidal Inclusion.* Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, vol. 252, no. 1271, pages 561–569, 1959. (Cited on page 71.)

[Eshelby 1961] J. D. Eshelby. *Elastic Inclusions and Inhomogeneities.* Progress in Solid Mechanics 2, pages 89–140, 1961. (Cited on page 77.)

[Fernandes *et al.* 2012] Paulo R. Fernandes, Helder C. Rodrigues, José M. Guedes and Pedro G. Coelho. *Multiscale Modelling on Bone Mechanics – Application to Tissue Engineering and Bone Quality Analysis.* IFAC Proceedings Volumes, vol. 45, pages 1013–1017, 1 2012. (Cited on page 104.)

[Feyel 1999] Frédéric Feyel. *Multiscale FE2 elastoviscoplastic analysis of composite structures.* Computational Materials Science, 1999. (Cited on pages 1, 60, 71, 100, 107 and 157.)

[Feyel 2003] Frédéric Feyel. *A multilevel finite element method (FE2) to describe the response of highly non-linear structures using generalized continua.* Computer Methods in Applied Mechanics and Engineering, 2003. (Cited on pages 100 and 107.)

[Furukawa & Yagawa 1998] Tomonari Furukawa and Genki Yagawa. *IMPLICIT CONSTITUTIVE MODELLING FOR VISCOPLASTICITY USING NEURAL NETWORKSR.* Technical report, 1998. (Cited on pages 19 and 34.)

[Gavazzi & Lagoudas 1990] A. C. Gavazzi and D. C. Lagoudas. *On the numerical evaluation of Eshelby's tensor and its application to elastoplastic fibrous composites.* Computational Mechanics, vol. 7, no. 1, pages 13–19, 1990. (Cited on pages 74, 76 and 85.)

[Geuzaine, Christophe and Remacle, Jean-Francois] Geuzaine, Christophe and Remacle, Jean-Francois. *Gmsh.* (Cited on page 135.)

[Ghaboussi *et al.* 1991] J. Ghaboussi, J. H. Garrett and X. Wu. *Knowledge-Based Modeling of Material Behavior with Neural Networks.* Journal of Engineering Mechanics, vol. 117, no. 1, pages 132–153, jan 1991. (Cited on page 19.)

[Gorji *et al.* 2020] Maysam B. Gorji, Mojtaba Mozaffar, Julian N. Heidenreich, Jian Cao and Dirk Mohr. *On the potential of recurrent neural networks for modeling path dependent plasticity.* Journal of the Mechanics and Physics of Solids, 2020. (Cited on page 34.)

[Halphen & Nguyen 1975] Bernard Halphen and Quoc Nguyen. *On Generalized Standard Materials./SUR LES MATERIAUX STANDARDS GENERALISES./.* J Mec, vol. 14, pages 39–63, 01 1975. (Cited on page 38.)

[Hashin 1983] Z. Hashin. *Analysis of composite materials - A survey.* Journal of Applied Mechanics, vol. 50, pages 481–505, 09 1983. (Cited on page 71.)

[Hill 1963] R. Hill. *Elastic properties of reinforced solids: Some theoretical principles.* Journal of the Mechanics and Physics of Solids, vol. 11, no. 5, pages 357–372, 1963. (Cited on page 71.)

[Hill 1965] R. Hill. *A self-consistent mechanics of composite materials.* Journal of the Mechanics and Physics of Solids, vol. 13, no. 4, pages 213–222, 1965. (Cited on page 71.)

[Hill 1967] R. Hill. *The essential structure of constitutive laws for metal composites and polycrystals.* Journal of the Mechanics and Physics of Solids, vol. 15, pages 79–95, 3 1967. (Cited on pages 105 and 106.)

[Hochreiter & Schmidhuber 1997] Sepp Hochreiter and Jürgen Schmidhuber. *Long Short-Term Memory.* Neural Computation, vol. 9, no. 8, pages 1735–1780, nov 1997. (Cited on page 41.)

[Hornik *et al.* 1989] Kurt Hornik, Maxwell Stinchcombe and Halbert White. *Multilayer feedforward networks are universal approximators.* Neural networks, vol. 2, no. 5, pages 359–366, 1989. (Cited on page 16.)

[Ibañez *et al.* 2018] Rubén Ibañez, Emmanuelle Abisset-Chavanne, Jose Vicente Aguado, David Gonzalez, Elias Cueto, Francisco Chinesta, Rubén Ibañ Ez, • Jose, Vicente Aguado, • David Gonzalez and • Francisco Chinesta. *A Manifold Learning Approach to Data-Driven Computational Elasticity and Inelasticity.* Archives of Computational Methods in Engineering, vol. 25, no. 1, pages 47–57, jan 2018. (Cited on page 29.)

[Jain & Medsker 1999] L. C. Jain and L. R. Medsker. Recurrent neural networks: Design and applications. CRC Press, Inc., USA, 1st édition, 1999. (Cited on page 34.)

[Jung & Ghaboussi 2006] Sungmoon Jung and Jamshid Ghaboussi. *Neural network constitutive model for rate-dependent materials.* Computers Structures, vol. 84, no. 15-16, pages 955–963, jun 2006. (Cited on page 20.)

[Ke & Liu 2008] Jinchuan Ke and Xinzhe Liu. *Empirical Analysis of Optimal Hidden Neurons in Neural Network Modeling for Stock Prediction.* In Proceedings of the 2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application - Volume 02, PACIIA '08, pages 828–832, USA, December 2008. IEEE Computer Society. (Cited on pages 80, 81 and 94.)

[Kerschen *et al.* 2005] Gaetan Kerschen, Jean Claude Golinval, Alexander F. Vakakis and Lawrence A. Bergman. *The Method of Proper Orthogonal Decomposition for Dynamical Characterization and Order Reduction of Mechanical Systems: An Overview.* Nonlinear Dynamics 2005 41:1, vol. 41, pages 147–169, 8 2005. (Cited on page 11.)

[Kingma & Ba 2014] Diederik P. Kingma and Jimmy Lei Ba. *Adam: A Method for Stochastic Optimization.* 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, dec 2014. (Cited on page 19.)

[Kirchdoerfer & Ortiz 2016] T. Kirchdoerfer and M. Ortiz. *Data-driven computational mechanics.* Computer Methods in Applied Mechanics and Engineering, vol. 304, pages 81–101, jun 2016. (Cited on pages 28 and 29.)

[Kirchdoerfer & Ortiz 2017] T. Kirchdoerfer and M. Ortiz. *Data Driven Computing with noisy material data sets.* Computer Methods in Applied Mechanics and Engineering, vol. 326, pages 622–641, nov 2017. (Cited on page 29.)

[Kirchdoerfer & Ortiz 2018] T. Kirchdoerfer and M. Ortiz. *Data-driven computing in dynamics.* International Journal for Numerical Methods in Engineering, vol. 113, no. 11, pages 1697–1710, mar 2018. (Cited on page 29.)

[Lange *et al.* 2021] Nils Lange, Geralf Hütter and Björn Kiefer. *An efficient monolithic solution scheme for FE2 problems.* Computer Methods in Applied Mechanics and Engineering, vol. 382, page 113886, 8 2021. (Cited on page 110.)

[Le *et al.* 2015] B. A. Le, J. Yvonnet and Q. C. He. *Computational homogenization of nonlinear elastic materials using neural networks.* International Journal for Numerical Methods in Engineering, vol. 104, no. 12, pages 1061–1084, dec 2015. (Cited on pages vii, 21, 22 and 101.)

[Lecun 2001a] Yann Lecun. *A Theoretical Framework for Back-Propagation.* 08 2001. (Cited on page 19.)

[Lecun 2001b] Yann Lecun. *A Theoretical Framework for Back-Propagation.* 08 2001. (Cited on page 44.)

[Lemaitre & Chaboche 1990] Jean Lemaitre and Jean-Louis Chaboche. Mechanics of Solid Materials. Cambridge University Press, feb 1990. (Cited on page 35.)

[Liu *et al.* 2015] Ruoqian Liu, Yuksel C. Yabansu, Ankit Agrawal, Surya R. Kalidindi and Alok N. Choudhary. *Machine learning approaches for elastic localization linkages in high-contrast composite materials.* Integrating Materials and Manufacturing Innovation, vol. 4, no. 1, pages 192–208, dec 2015. (Cited on page 23.)

[Liu *et al.* 2017] Ruoqian Liu, Yuksel C. Yabansu, Zijiang Yang, Alok N. Choudhary, Surya R. Kalidindi and Ankit Agrawal. *Context Aware Machine Learning Approaches for Modeling Elastic Localization in Three-Dimensional Composite Microstructures.* Integrating Materials and Manufacturing Innovation 2017 6:2, vol. 6, no. 2, pages 160–171, mar 2017. (Cited on page 23.)

[Liu *et al.* 2020] Xing Liu, Christos E. Athanasiou, Nitin P. Padture, Brian W. Sheldon and Huajian Gao. *A machine learning approach to fracture mechanics problems.* Acta Materialia, vol. 190, pages 105–112, may 2020. (Cited on pages 20 and 27.)

[Lu *et al.* 2018] Xiaoxin Lu, Dimitris G. Giovanis, Julien Yvonnet, Vissarion Papadopoulos, Fabrice Detrez and Jinbo Bai. *A data-driven computational homogenization method based on neural networks for the nonlinear anisotropic electrical response of graphene/polymer nanocomposites.* Computational Mechanics 2018 64:2, vol. 64, no. 2, pages 307–321, oct 2018. (Cited on pages 21 and 101.)

[Malik *et al.* 2021] Alexander Malik, Martin Abendroth, Geralf Hütter and Bjoern Kiefer. *A Hybrid Approach Employing Neural Networks to Simulate the ElastoPlastic Deformation Behavior of 3D-Foam Structures.* Advanced Engineering Materials, 2021. (Cited on page 31.)

[Masi *et al.* 2021] 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, vol. 147, page 104277, feb 2021. (Cited on pages 30, 34 and 47.)

[Maugin 1999] G. A. (Gerard A.) Maugin. *The thermomechanics of nonlinear irreversible behaviors : an introduction.* page 375, 1999. (Cited on page 35.)

[Mccartney 2017] L. Mccartney. *Predicting Properties of Undamaged and Damaged Carbon Fibre Reinforced Composites.* The Structural Integrity of Carbon Fiber Composites: Fifty Years of Progress and Achievement of the Science, Development, and Applications, pages 425–467, 11 2017. (Cited on pages xiii and 84.)

[Minh Nguyen-Thanh *et al.* 2020] Vien Minh Nguyen-Thanh, Lu Trong Khiem Nguyen, Timon Rabczuk and Xiaoying Zhuang. *A surrogate model for computational homogenization of elastostatics at finite*

strain using high-dimensional model representation-based neural network. International Journal for Numerical Methods in Engineering, vol. 121, no. 21, pages 4811–4842, nov 2020. (Cited on pages 21 and 101.)

[Mori & Tanaka 1973] T. Mori and K. Tanaka. *Average stress in matrix and average elastic energy of materials with misfitting inclusions.* Acta Metallurgica, vol. 21, no. 5, pages 571–574, 1973. (Cited on pages 71, 74, 78 and 84.)

[Mura & Barnett 1983] T. Mura and D. M. Barnett. *Micromechanics of Defects in Solids.* Journal of Applied Mechanics, vol. 50, no. 2, pages 477–477, 06 1983. (Cited on page 76.)

[Mura 1987] Toshio Mura. *Theory of Inclusions and Inhomogeneities.* Micromechanics of defects in solids, vol. 3, 1987. (Cited on pages 75 and 76.)

[Nguyen & Keip 2018] Lu Trong Khiem Nguyen and Marc André Keip. *A data-driven approach to nonlinear elasticity.* Computers Structures, vol. 194, pages 97–115, jan 2018. (Cited on page 29.)

[Nguyen 1988] Quoc Son Nguyen. *Mechanical modelling of anelasticity.* Revue de Physique Appliquée, vol. 23, pages 325–330, 1988. (Cited on page 106.)

[Ouyang *et al.* 2021] Boya Ouyang, Yu Song, Yuhai Li, Feishu Wu, Huizi Yu, Yongzhe Wang, Zhanyuan Yin, Xiaoshu Luo, Gaurav Sant and Mathieu Bauchy. *Using machine learning to predict concrete's strength: learning from small datasets.* Engineering Research Express, vol. 3, no. 1, page 015022, feb 2021. (Cited on page 20.)

[Pavlina & Van Tyne 2008] E. J. Pavlina and C. J. Van Tyne. *Correlation of Yield Strength and Tensile Strength with Hardness for Steels.* Journal of Materials Engineering and Performance 2008 17:6, vol. 17, no. 6, pages 888–893, dec 2008. (Cited on page 15.)

[Pedregosa *et al.* 2011] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay. *Scikit-learn: Machine Learning in Python.* Journal of Machine Learning Research, vol. 12, pages 2825–2830, 2011. (Cited on page 80.)

[Pierson *et al.* 2019] Kyle Pierson, Aowabin Rahman and Ashley D. Spear. *Predicting Microstructure-Sensitive Fatigue-Crack Path in 3D Using a Machine Learning Framework.* JOM, vol. 71, no. 8, pages 2680–2694, aug 2019. (Cited on page 23.)

[Platzer *et al.* 2021] Auriane Platzer, Adrien Leygue, Laurent Stainier and Michael Ortiz. *Finite element solver for data-driven finite strain elasticity.* Computer Methods in Applied Mechanics and Engineering, vol. 379, page 113756, jun 2021. (Cited on page 29.)

[Praud *et al.* 2017] F. Praud, G. Chatzigeorgiou, Y. Chemisky and F. Meraghni. *Hybrid micromechanical-phenomenological modelling of anisotropic damage and anelasticity induced by micro-cracks in unidirectional composites.* Composite Structures, vol. 182, pages 223–236, dec 2017. (Cited on page 98.)

[Prulière & Chemisky 2022] Etienne Prulière and Yves Chemisky. *3MAH : un ensemble de librairies pour analyser le comportement complexe de matériaux hétérogènes*. 2022. (Cited on pages 48, 62, 112, 119 and 135.)

[Quinlan 1986] J. R. Quinlan. *Induction of decision trees*. Machine Learning 1986 1:1, vol. 1, no. 1, pages 81–106, mar 1986. (Cited on page 24.)

[Raissi *et al.* 2019] M. Raissi, P. Perdikaris and G. E. Karniadakis. *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*. Journal of Computational Physics, vol. 378, pages 686–707, feb 2019. (Cited on page 29.)

[Ravinder *et al.* 2020] R. Ravinder, Karthikeya H. Sridhara, Suresh Bishnoi, Hargun Singh Grover, Mathieu Bauchy, Jayadeva, Hariprasad Kodamana and N. M.Anoop Krishnan. *Deep learning aided rational design of oxide glasses*. Materials Horizons, vol. 7, no. 7, pages 1819–1827, jul 2020. (Cited on page 20.)

[Reuss 1929] A. Reuss. *Berechnung der Fließgrenze von Mischkristallen auf Grund der Plastizitätsbedingung für Einkristalle* . ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik, vol. 9, no. 1, pages 49–58, 1929. (Cited on page 71.)

[Rosenblatt 1958] F. Rosenblatt. *The perceptron: A probabilistic model for information storage and organization in the brain*. Psychological Review, vol. 65, no. 6, pages 386–408, nov 1958. (Cited on page 15.)

[Ruder 2017] Sebastian Ruder. *An overview of gradient descent optimization algorithms*. arXiv:1609.04747 [cs], June 2017. arXiv: 1609.04747. (Cited on pages vii, 19 and 20.)

[Settgast *et al.* 2019a] Christoph Settgast, Martin Abendroth and Meinhard Kuna. *Constitutive modeling of plastic deformation behavior of open-cell foam structures using neural networks*. Mechanics of Materials, vol. 131, no. January, pages 1–10, 2019. (Cited on pages 20 and 34.)

[Settgast *et al.* 2019b] Christoph Settgast, Geralf Hütter, Meinhard Kuna and Martin Abendroth. *A hybrid approach to simulate the homogenized irreversible elastic-plastic deformations and damage of foams by neural networks*. arXiv, 2019. (Cited on page 31.)

[Sheela & Deepa 2013] K. Gnana Sheela and S. N. Deepa. *Review on Methods to Fix Number of Hidden Neurons in Neural Networks*, 2013. (Cited on page 80.)

[Shen *et al.* 2020] W. Q. Shen, Y. J. Cao, J. F. Shao and Z. B. Liu. *Prediction of plastic yield surface for porous materials by a machine learning approach*. Materials Today Communications, vol. 25, no. June, page 101477, 2020. (Cited on page 20.)

[Shibata & Ikeda 2009] Katsunari Shibata and Yuki Ikeda. *Effect of number of hidden neurons on learning in large-scale layered neural networks*. 2009 ICCAS-SICE, pages 5008–5013, 2009. (Cited on page 80.)

[Strohmann *et al.* 2019] Tobias Strohmann, Katrin Bugelnig, Eric Breitbarth, Fabian Wilde, Thomas Steffens, Holger Germann and Guillermo Requena. *Semantic segmentation of synchrotron tomography of multiphase Al-Si alloys using a convolutional neural network with a pixel-wise weighted loss function.* Scientific Reports, vol. 9, no. 1, dec 2019. (Cited on page 22.)

[Suquet 1987] PM Suquet. *Elements of Homogenization Theory for Inelastic Solid Mechanics.* Homogenization Techniques for Composite Media, pages 193–287, 1987. (Cited on page 106.)

[Swarztrauber 2003] Paul N. Swarztrauber. *On computing the points and weights for Gauss-Legendre quadrature.* SIAM Journal on Scientific Computing, vol. 24, no. 3, pages 945–954, 2003. (Cited on page 76.)

[Tamura & Tateishi 1997] S. Tamura and M. Tateishi. *Capabilities of a four-layered feedforward neural network: four layers versus three.* IEEE Transactions on Neural Networks, vol. 8, no. 2, pages 251–255, March 1997. (Cited on page 80.)

[Tibshirani 1996] Robert Tibshirani. *Regression Shrinkage and Selection via the Lasso.* Journal of the Royal Statistical Society. Series B (Methodological), vol. 58, no. 1, pages 267–288, 1996. (Cited on page 14.)

[Tikarrouchine *et al.* 2018] E. Tikarrouchine, G. Chatzigeorgiou, F. Praud, B. Piotrowski, Y. Chemisky and F. Meraghni. *Three-dimensional FE2 method for the simulation of non-linear, rate-dependent response of composite structures.* Composite Structures, 2018. (Cited on page 100.)

[Tikarrouchine *et al.* 2019] E. Tikarrouchine, G. Chatzigeorgiou, Y. Chemisky and F. Meraghni. *Fully coupled thermo-viscoplastic analysis of composite structures by means of multi-scale three-dimensional finite element computations.* International Journal of Solids and Structures, 2019. (Cited on page 100.)

[Voigt 1887] W. Voigt. *Waldemar Voigt, Theoretische Studien über die Elasticitätsverhältnisse der Krystalle. I.* Abhandlungen der Königlichen Gesellschaft der Wissenschaften in Göttingen, vol. 34, pages 3–52, 1887. (Cited on page 71.)

[Wan 2019] Xing Wan. *Influence of feature scaling on convergence of gradient iterative algorithm.* Journal of Physics: Conference Series, vol. 1213, page 32021, 2019. (Cited on page 80.)

[Wang *et al.* 2020] Yixing Wang, Min Zhang, Anqi Lin, Akshay Iyer, Aditya Shanker Prasad, Xiaolin Li, Yichi Zhang, Linda S. Schadler, Wei Chen and L. Catherine Brinson. *Mining structure–property relationships in polymer nanocomposites using data driven finite element analysis and multi-task convolutional neural networks.* Molecular Systems Design Engineering, vol. 5, no. 5, pages 962–975, jun 2020. (Cited on page 23.)

[Waszczyszyn & Ziemiański 2001] Zenon Waszczyszyn and Leonard Ziemiański. *Neural networks in mechanics of structures and materials - New results and prospects of applications.* Computers and Structures, vol. 79, no. 22-25, pages 2261–2276, sep 2001. (Cited on pages 20 and 34.)

[Werbos 1990] Paul J. Werbos. *Backpropagation Through Time: What It Does and How to Do It.* Proceedings of the IEEE, vol. 78, no. 10, pages 1550–1560, 1990. (Cited on page 41.)

[Withers 1989] P. J. Withers. *The determination of the elastic field of an ellipsoidal inclusion in a transversely isotropic medium, and its relevance to composite materials.* Philosophical Magazine A: Physics of Condensed Matter, Structure, Defects and Mechanical Properties, vol. 59, no. 4, pages 759–781, 1989. (Cited on page 71.)

[Xu *et al.* 2020] Rui Xu, Jie Yang, Wei Yan, Qun Huang, Gaetano Giunta, Salim Belouettar, Hamid Zahrouni, Tarak Ben Zineb and Heng Hu. *Data-driven multiscale finite element method: From concurrence to separation.* Computer Methods in Applied Mechanics and Engineering, vol. 363, page 112893, may 2020. (Cited on page 29.)

[Yang *et al.* 2018] Zijiang Yang, Yuksel C. Yabansu, Reda Al-Bahrani, Wei keng Liao, Alok N. Choudhary, Surya R. Kalidindi and Ankit Agrawal. *Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets.* Computational Materials Science, vol. 151, pages 278–287, aug 2018. (Cited on pages vii, 23 and 24.)

[Yang *et al.* 2019] Kai Yang, Xinyi Xu, Benjamin Yang, Brian Cook, Herbert Ramos, N. M.Anoop Krishnan, Morten M. Smedskjaer, Christian Hoover and Mathieu Bauchy. *Predicting the Young's Modulus of Silicate Glasses using High-Throughput Molecular Dynamics Simulations and Machine Learning.* Scientific reports, vol. 9, no. 1, page 8739, dec 2019. (Cited on pages vii, 15 and 27.)

[Yuan *et al.* 2022] Lei Yuan, Yi Qing Ni, Xiang Yun Deng and Shuo Hao. *A-PINN: Auxiliary physics informed neural networks for forward and inverse problems of nonlinear integro-differential equations.* Journal of Computational Physics, vol. 462, page 111260, aug 2022. (Cited on pages viii and 29.)

[Zhang & Mohr 2020] Annan Zhang and Dirk Mohr. *Using neural networks to represent von Mises plasticity with isotropic hardening.* International Journal of Plasticity, vol. 132, sep 2020. (Cited on page 20.)

[Zhang *et al.* 2003] Zhaozhi Zhang, Xiaomin Ma and Yixian Yang. *Bounds on the number of hidden neurons in three-layer binary neural networks.* Neural Networks, vol. 16, no. 7, pages 995–1002, September 2003. (Cited on page 80.)

[Zhao *et al.* 2021] Qingkun Zhao, Huiya Yang, Jiabin Liu, Haofei Zhou, Hongtao Wang and Wei Yang. *Machine learning-assisted discovery of strong and conductive Cu alloys: Data mining from discarded experiments and physical features.* Materials Design, vol. 197, page 109248, jan 2021. (Cited on page 15.)

[Zhu *et al.* 2020] Jianchang Zhu, Mohamed Ben Bettaieb and Farid Abed-Meraim. *Comparative study of three techniques for the computation of the macroscopic tangent moduli by periodic homogenization scheme.* Engineering with Computers, page 30 pp., 2020. (Cited on page 60.)

Résumé étendu en français - Extended summary in french

Contexte et problématique

L'utilisation de matériaux hétérogènes devient de plus en plus indispensable pour les applications d'ingénierie de pointe, notamment dans les domaines des énergies renouvelables, de la santé et de l'aérospatiale. Par exemple, les matériaux composites ont acquis une grande importance dans l'industrie aéronautique en raison de leurs propriétés mécaniques spécifiques, permettant de concevoir des structures légères tout en maintenant des niveaux de performances élevées et en répondant aux exigences de sécurité. En outre, l'utilisation des composites présente un impact environnemental significatif en termes d'efficacité énergétique et de réduction des émissions de CO_2. L'émergence des matériaux architecturés cellulaires a également révolutionné un large éventail de domaines d'ingénierie. En raison de leur rapport résistance/poids élevé et de leurs importantes capacités d'absorption énergétique, les structures architecturées peuvent être utilisées, par exemple, en biomécanique pour concevoir des implants orthopédiques. En optimisant la topologie des microstructures, les prothèses architecturées peuvent être conçues de manière optimale pour garantir leur compatibilité avec les os humains. De plus, ils fournissent à la fois la porosité et la résistance mécanique appropriées au tissu osseux, ainsi que la légèreté nécessaire pour permettre à la prothèse de résister à des conditions de charge très élevés. La croissance considérable de ce type de matériaux est principalement due au développement significatif des méthodes d'optimisation topologique et de fabrication additive qui permettent la production de formes assez complexes non accessibles avec les méthodes traditionnelles telles que les procédés d'usinage et de moulage.

Cependant, la simulation du comportement thermomécanique des structures hétérogènes peut s'avérer très complexe en raison de multiples facteurs: Premièrement, le comportement mécanique résultant des microstructures est très difficile à identifier en raison de la présence de différents mécanismes physiques impliqués à différentes échelles. Le comportement local est notamment régi par des phénomènes dissipatifs fortement non linéaires qui nécessitent une identification appropriée des lois constitutives. En outre, l'utilisation des stratégies de modélisation multi-échelles s'avère nécessaires pour une compréhension fine de l'effet des microstructures sur le comportement macroscopique. Par conséquent, l'évaluation de la réponse mécanique des structures hétérogènes nécessite le développement d'outils puissants et avancés de simulation numérique, capables de prendre en compte l'interaction micro-macro et de prédire avec haute fidélité la réponse globale résultante. Parmi ces approches numériques, la méthode des éléments finis au carré FE^2 [Feyel 1999] a été largement développée dans la littérature pour l'analyse des structures multi-échelle. Cette approche à champ complet basée sur la théorie de l'homogénéisation périodique, consiste à attribuer un Volume Élémentaire Représentatif (VER) à chaque point d'intégration de Gauss, puis l'estimation de la réponse globale est obtenue en résolvant les problèmes micro et macro simultanément par les principes de localisation et d'homogénéisation. Bien que FE^2 demeure plus avantageuse en termes de temps de calcul par rapport aux simulations FE directes sur des structures hétérogènes entièrement maillées, il est clair que cette stratégie de modélisation souffre également de nombreuses limitations pour les raisons suivantes: Comme décrit dans la procédure FE^2 ci-dessus, des problèmes non linéaires sont

résolus simultanément à l'échelle micro et macro en utilisant des schémas itératifs Newton Raphson. Ce processus est souvent très coûteux en temps de calcul en raison du traitement des non-linéarités pour chaque calcul microscopique et à chaque itération. De plus, d'autres considérations peuvent augmenter d'avantage les temps de simulation, notamment le nombre de points d'intégration de Gauss, la complexité des conditions de chargement, le nombre d'incréments et la résolution de maillage du VER et de la macrostructure. Par conséquent, le recours à la méthode FE^2 peut s'avérer limité en calcul de structure compte tenu de tous ces facteurs.

Au vu de la quatrième révolution industrielle (Industrie 4.0), l'innovation technologique présente actuellement un grand impact sur la société : véhicules autonomes, robotique, Internet des objets (IoT) et impression 3D. L'intelligence artificielle (IA) est l'une des technologies les plus marquantes de l'industrie 4.0 grâce à l'apprentissage automatique (Machine Learning ML) et à l'apprentissage profond (Deep Learning DL). À titre d'exemple, l'apprentissage automatique a considérablement marqué le système de santé au cours des dernières années. À travers le développement d'algorithmes ML et de logiciels exhaustifs, les médecins bénéficient désormais de l'assistance de l'IA pour le diagnostic des patients et des maladies. Les progrès remarquables réalisés dans le domaine de l'IA ne sont que le reflet de la croissance exponentielle des ressources informatiques et de la disponibilité massive de données. Dans le domaine de la science des matériaux, les développements récents des techniques de mesure expérimentale permettent la génération de bases de données importantes de comportements de matériaux. D'autre part, les progrès significatifs des capacités de calcul facilitent la production de grandes quantités de simulations numériques. Par conséquent, la disponibilité des données, considérée comme la principale source d'enrichissement des modèles IA, ne représente plus un obstacle majeur. Dans cette perspective, l'utilisation de l'apprentissage automatique et de l'apprentissage profond a considérablement suscité l'intérêt de la communauté des sciences des matériaux. L'essence des techniques ML et DL réside dans leur capacité à prédire rapidement des réponses mécaniques cibles après une phase d'apprentissage. Ces méthodes trouvent également leur utilité en mécanique numérique pour accélérer les simulations multi-échelles de matériaux hétérogènes. Comme évoqué précédemment, l'évaluation de la réponse mécanique de structures architecturées complexes représente un véritable défi, les méthodologies traditionnelles pour réaliser ces simulations telles que FE^2 sont très coûteuse en temps de calcul et nécessitent des ressources informatiques parfois inaccessibles. L'intégration d'approches d'apprentissage profond dans les solveurs numériques de mécanique des solides est une alternative prometteuse pour surmonter ces difficultés.

Dans cette optique, ce projet de thèse a pour objectif de repenser en profondeur le paradigme de la simulation multi-échelles du comportement mécanique des structures architecturés. En combinant Intelligence Artificielle (IA), thermodynamique des matériaux et simulation numérique du comportement de structures, nous avons l'ambition de développer et valider des modeles hybrides permettant de simuler, avec un gain de calcul très important, le comportement multi-échelle de structures hétérogènes. L'objectif serait donc d'effectuer en quelques minutes, sur un ordinateur de bureau, des simulations prenant plusieurs jours sur des centres de calcul intensif. Pour répondre à ces objectifs, ce manuscrit de thèse a été structuré en deux parties et quatre chapitres.

Chapitre 1: État de l'art

Le premier chapitre de ce manuscrit de thèse a été dédié à une revue bibliographique concernant l'utilisation des techniques d'apprentissage automatique et des approches dites "Data Driven" en science des matériaux et dans les applications d'ingénierie. Dans un premier temps, les principales méthodes d'apprentissage en machine learning sont présentées, notamment l'apprentissage supervisé (Supervised Learning SL), l'apprentissage non supervisé (Unsupervised Learning UL), l'apprentissage semi-supervisé (Semi Supervised Learning SSL) et l'apprentissage par renforcement (Reinforcement Learning RL). L'utilisation d'un algorithme spécifique dépend du problème à traiter et de la nature des données accessibles. Dans la deuxième section de ce chapitre, un aperçu général de l'application des méthodes basées sur l'IA en mécanique des matériaux est mis en évidence. Sur la base des contributions récentes de la communauté de la science des matériaux, de nombreux exemples pratiques incluant la modélisation constitutive et la modélisation multi-échelle sont discutés.

Chapitre 2: Prédiction du comportement mécanique non linéaire de matériaux dissipatifs sous chargements non proportionnels avec des Réseaux de Neurones Récurrents Thermodynamiquement Consistants

Dans le chapitre 2, un modèle réseau de neurones récurrent thermodynamiquement consistant (ThC-RNN) a été développé afin de servir comme métamodèle pour la modélisation constitutive de matériaux dissipatifs. Cette approche hybride consiste à combiner une architecture spécifique de RNN appelée Long Short Term Memory (LSTM) avec des lois thermodynamiques. La robustesse du ThC-RNN provient principalement des grandes capacités prédictives des LSTM pour capturer des comportements dépendant de l'histoire de chargement ainsi que pour surmonter les problèmes de vanishing gradient. En outre, l'incorporation de lois physiques, et plus particulièrement la deuxième loi de la thermodynamique, dans le processus d'apprentissage a permis d'obtenir des résultats pertinents et cohérents comparé aux RNN de type boîte noire. La fiabilité du modèle ThC-RNN a été testée sur différents modèles de plasticité, notamment l'élastoplasticité avec écrouissage isotrope et le modèle de plasticité de Chaboche avec un écrouissage isotrope combiné à deux écrouissages cinématiques non linéaires. Il a été prouvé que la consistance thermodynamique permet d'améliorer de manière significative les capacités prédictives de ThC-RNN en considérant plusieurs quantités d'intérêt, comme les tenseurs de contrainte, matrices tangentes, variables internes et la partition du puissances mécaniques. Finalement, il a été démontré que le modèle ThC-RNN est en mesure d'être intégré dans un cadre élément finis pour simuler le comportement mécanique de structures à condition de vérifier certains points : 1) Vérifier les capacités de généralisation des modèles lors de la prédiction des opérateurs tangents et des réponses mécaniques, 2) S'assurer que les données d'entrées sont situés dans la base d'apprentissage afin d'éviter des problèmes au niveau de la convergence des solution, 3) Incorporer des lois basées sur la physique dans le processus d'apprentissage pour régulariser les champs mécaniques prédits en respectant à la consistance thermodynamique.

Chapitre 3: Estimation des propriétés effectives de matériaux hétérogènes par des approches hybrides combinant réseaux de neurones artificiels et modèles micromécaniques

Ce chapitre avait pour objectif d'investiguer la capacité de modèles hybrides à prédire les propriétés effectives de matériaux hétérogènes. Un modèle hybride ANN-φ a été développé en combinant des réseaux de neurones artificiels et un schéma d'homogénéisation micromécanique. L'approche d'homogénéisation utilisée dans cette étude est principalement basée sur le problème d'inclusion d'Eshelby. Une attention particulière est portée à la construction du réseau de neurones à travers le choix des hyperparamètres du modèle (fonctions d'activation, nombre de neurones dans les couches intermédiaires, algorithmes d'optimisation). Une base de données de tenseurs d'Eshelby est générée en variant les propriétés matérielles du milieu hétérogène ainsi que les paramètres géométriques des inclusions. Le modèle ANN-φ, une fois entraîné sur une base de tenseurs d'Eshelby, a démontré d'excellentes capabilités prédictives du comportement mécanique effectif de matériaux hétérogènes. Les résultats obtenus avec le modèle ANN-φ ont été ensuite comparés pour divers cas d'étude à des estimations numériques souvent coûteuses en termes de temps de calcul. Les résultats présentés dans ce travail ont montré que le modèle hybride développé a conduit à un gain de temps de calcul significatif de l'ordre de 99,9% tout en conservant sa précision et sa fiabilité lors de l'estimation des propriétés effectives de matériaux hétérogènes.

Chapitre 4: Accélération des simulations multi-échelles de structures architecturés avec des approches basée sur les réseaux de neurones profond

Dans le dernier chapitre de ce manuscrit, nous avons élaboré une nouvelle stratégie de modélisation pour accélérer les simulations multi-échelles de matériaux hétérogènes en utilisant des réseaux de neurones profonds. L'approche développée, appelée FE-LSTM, consiste à combiner la méthode des éléments finis et les réseaux de neurones récurrents de type LSTM pour résoudre des problèmes multi-échelles. Contrairement à la méthode FE^2, la résolution éléments finis des problèmes microscopiques n'est plus requise dans le cadre de FE-LSTM, la réponse homogénéisée du VER est directement prédite par un LSTM entraîné sur une base de données offline. En conséquence, le principal avantage de FE-LSTM réside dans le gain massif au niveau des temps de calcul et de ressources lors de la résolution des problèmes microscopiques. Afin d'évaluer la validité et la fiabilité de l'approche développée, FE-LSTM a été testé sur plusieurs structures architecturées en 3D. Les résultats obtenus ont démontré les capacités prédictives élevées de la méthode, un excellent accord a été obtenu entre les champs macroscopiques prédits par FE-LSTM et les simulations par FE^2. L'utilité de cette approche réside dans le fait qu'une fois le processus d'apprentissage du RNN effectué, FE-LSTM peut être appliqué pour simuler les comportements non linéaires de toute structure hétérogène ayant la même microstructure que celle utilisée lors de l'apprentissage. En termes de temps d'exécution, nous avons mis en évidence que les simulations FE-LSTM sont réalisées en quelques secondes contre plusieurs jours avec FE^2, ce qui se traduit par des facteurs de gain de temps de calcul proches de **40000**. De plus, elles peuvent être facilement réalisées sur des ordinateurs de bureau sans avoir recours à des mésocentres de calcul intensif. Finalement, nous avons établi une première preuve de concept de l'utilisation d'architectures de réseaux neuronaux à entrées multiples combinant des réseaux neuronaux convolutionnels (CNN) et des perceptrons multicouches (MLP) pour prédire les propriétés effectives des matériaux architecturés. Nous avons montré qu'il est possible de simuler le comportement homogène

de structures hétérogènes uniquement à partir de la connaissance de l'image de la microstructure et des valeurs numériques des propriétés des matériaux. Un modèle CNN-MLP a été développé pour établit les liens entre la microstructure et les propriétés des structures de type octet-truss.

Perspectives des travaux de thèse

Les stratégies de modélisation développées dans le cadre de ce projet de thèse ont conduit à des avancées considérables dans la simulation numérique des structures hétérogènes. Néanmoins, plusieurs directions de recherche sont à explorer. Les perspectives de ce travail de recherche sont détaillées comme suit:

- L'incorporation de lois physiques dans le processus d'apprentissage du modèle FE-LSTM de façon similaire au modèle ThC-RNN. Bien que FE-LSTM ait atteint des performances très prometteuses en se basant uniquement sur les capacités prédictives des réseaux de neurones récurrents à capturer des comportements dépendant de l'histoire de chargement, une telle approche peut s'avérer très limitée en considérant des lois de comportements complexes et fortement non linéaires. L'introduction de la physique dans la phase d'apprentissage est par conséquent nécessaire pour la régularisation des champs mécaniques. Cependant, un défi supplémentaire doit être relevé lors de l'utilisation de FE-LSTM en analyse multi-échelle. En effet, il n'existe pas de forme explicite du second principe de la thermodynamique à l'échelle du VER, contrairement au cas du point matériel avec une loi constitutive bien définie. Une solution potentielle pour surmonter ce problème est d'ajouter une nouvelle contrainte sur la fonction coût en sa basant sur le travail mécanique macroscopique. En utilisant l'équivalence d'énergie micro-macro (lemme de Hill-Mandel), l'idée serait d'appliquer une contrainte sur la quantité $\bar{\sigma} : \Delta\bar{\varepsilon}$ pour être égale à $\Delta\bar{W}_m$ pendant le processus d'apprentissage. $\Delta\bar{W}_m$ correspondent au travail mécanique macroscopique incrémental à l'échelle du VER qui peut être calculé par analyse élements finis et stocké dans la base de données. En imposant la connexion entre $\bar{\sigma} : \Delta\bar{\varepsilon}$ et $\Delta\bar{W}_m$ pendant la phase d'entraînement, il est possible de fournir une information supplémentaire sur les champs macroscopiques, ce qui pourrait aboutir à des solutions plus précises par rapport à une approche RNN pure.

- Bien que cette étude ne se soit appliqué que sur des matériaux architecturés de type octet-truss avec une loi de comportement élastoplastique avec écrouissage isotrope, il serait plus judicieux d'effectuer des tests supplémentaires sur plusieurs matériaux architecturés tels que les TPMS avec leurs lois matérielles associées. Les capacités de généralisation des réseaux neuronaux LSTM sont à analyser pour des comportements plus complexes, y compris les lois constitutives dépendant de la vitesse de déformation, comme la viscoélasticité et la viscoplasticité. De plus, l'incorporation des non-linéarités géométriques dans les modèle est susceptible de conduire à des défis supplémentaires concernant les coûts de génération de bases de données. Par conséquent, ces cas nécessitent une étude plus approfondie pour définir précisément le domaine de validité et les limites de l'approche développée.

- Définir une stratégie de quantification des incertitudes pour les méthodes d'apprentissage profond en utilisant des techniques bayésiennes similaires à la méthode des processus gaussien aléatoires. Le déploiement d'approches basées sur l'apprentissage profond en analyse de structure nécessite le développement d'une méthodologie bien définie capable de mesurer les incertitudes héritées de

chaque étape du processus (collecte de la base de données, pré traitement, sélection des hyper-paramètres etc ...).

- Généraliser l'approche CNN-MLP pour traiter des problèmes plus complexes. Une application potentielle serait de combiner les réseaux de neurones de convolutions avec les réseaux neuronaux LSTM pour simuler une large gamme de matériaux architecturés en tenant compte de leur comportement non linéaire. Étant donné une image initiale 2D ou 3D de la microstructure et une séquence temporelle correspondant au chargement mécanique appliquée, le réseau CNN-LSTM peut servir à prédire l'évolution non linéaire de la réponse mécanique du matériau hétérogène.