*Acta Polytechnica CTU Proceedings* **42**:32–36, 2023
https://doi.org/10.14311/APP.2023.42.0032

Published by the Czech Technical University in Prague

# AI-ASSISTED STUDY OF AUXETIC STRUCTURES

Sergej Grednev[a,*], Henrik S. Steude[b], Stefan Bronder[a], Oliver Niggemann[b], Anne Jung[a]

[a] *Helmut Schmidt University/University of the Federal Armed Forces Hamburg, Professorship for Protective Systems, Holstenhofweg 85, 22043 Hamburg, Germany*

[b] *Helmut Schmidt University/University of the Federal Armed Forces Hamburg, Professorship for Computer Science in Mechanical Engineering, Holstenhofweg 85, 22043 Hamburg, Germany*

[*] corresponding author: sergej.grednev@hsu-hh.de

Abstract. In this study, the viability of using machine learning models to predict stress-strain curves of auxetic structures based on geometry-describing parameters is explored. Given the computational cost and time associated with generating these curves through numerical simulations, a machine learning-based approach promises a more efficient alternative. A range of machine learning models, including Artificial Neural Networks, k-Nearest Neighbors Regression, Support Vector Regression, and XGBoost, is implemented and compared regarding the aptitude to predict stress-strain curves under quasi-static compressive loading. Training data is generated using validated finite element simulations. The performance of these models is rigorously tested on data not seen during training. The Feed-Forward Artificial Neural Network emerged as the most proficient model, achieving a Mean Absolute Percentage Error of 0.367 ± 0.230.

Keywords: Auxetic structures, regression, machine learning.

## 1. Introduction

As a main characteristic auxetic structures posses a negative Poisson's ratio, which makes them a subclass of so called mechanical metamaterials. They show remarkable mechanical properties rendering them particularly useful for applications as crash absorbers. This aptitude stems from their high stiffness and low density combined with a plateau stress over a large deformation regime [1, 2]. The mechanical performance of auxetic structures is mainly governed by their geometry making structural optimization a study topic of paramount importance, which is typically facilitated through finite element simulations. Since those can be very time consuming given the structural complexity this work aims at utilizing a machine learning approach to reduce the need for lengthy simulations – a strategy commonly pursued in contemporary engineering sciences [3, 4]. To this end a multitude of machine learning algorithms and techniques are implemented and compared regarding their predictive performance.

## 2. Materials and methods

### 2.1. Auxetic structure and training data

The starting point for this study is a specific re-entrant auxetic structure previously studied by Bronder et. al. [5]. The structure is shown in Figure 1 and is originally defined by five geometry parameters: *length*, *size*, *waist*, *strut thickness* and *angle*. As part of a carried out feature engineering process an addition geometry parameter named *gap* is introduced, which is derived from the original features according to

$$\text{gap} = [1 - \tan(90^\circ - \text{angle})]\,\text{size} - \text{length}\,. \quad (1)$$

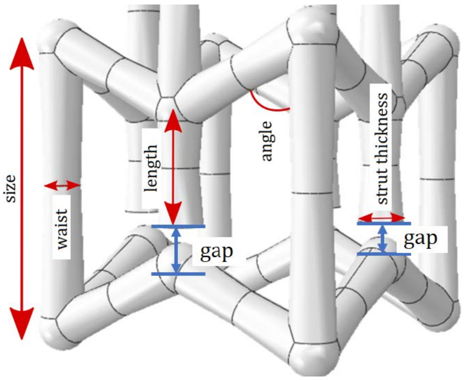

Figure 1. Auxetic unit cell with five defined geometry parameters (red), which represent the initial feature space for the modeling task at hand. An addition parameter was derived during feature engineering (blue). From Bronder et. al. [5], modified.

For this project a set of 130 stress-strain curves was available, which can be seen in Figure 2. The term *stress*, in the context of this paper, always refers to the mass-normalized stress in $\mathrm{MPa\,g^{-1}}$. From all available curves, 15 were reserved for the final performance evaluation at the very end of the project, leaving 115 example curves for model development.

The 130 curves were produced by validated finite element simulations from the software ABAQUS®. AlSi10Mg structures of $3 \times 3 \times 3$ unit cells were simulated using an explicit solver.

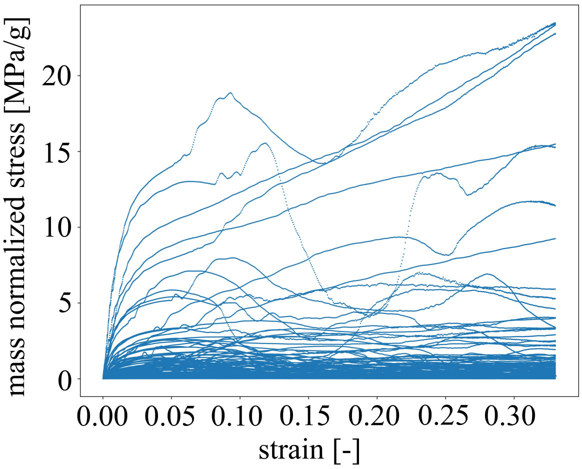

Figure 2. All 130 stress-strain curves available for the project. With lightweight applications in mind, stress values were normalized with regard to the structure's weight. For simplicity reasons compressive strain values are positive.

## 2.2. Error metric

The problem at hand is a supervised regression problem where a model predicts numerical values, in this case stresses, while minimizing the prediction error. Due to the curves potentially spreading over multiple orders of magnitude, the scale-independent Mean Absolute Percentage Error (MAPE) is chosen as the error metric, which is defined as

$$\mathrm{MAPE} = \frac{1}{N}\sum_{i}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| , \qquad (2)$$

where $N$ is the number of predictions, $\hat{y}_i$ the $i$-th predicted value and $y_i$ the corresponding $i$-th true value.

Since the MAPE becomes arbitrarily large for true target values, i.e. the variable values to be predicted, that are zero or close to zero, the metric is evaluated only for strains $> 0.01$, which corresponds to the initial linear elastic part of the curves.

## 2.3. Problem formulation

For the main study two different approaches to the problem itself are studied. In the *Single Point Approach* (SP) the models take as input the geometry parameters describing the structure as well as a certain strain value and predict the corresponding mass normalized stress value.

In contrast, using the *Whole Curve Approach* (WC) the model predicts a complete sub-sampled curve based on only the geometry parameters. Here, a sampling rate of 20 is used, reducing the number of data points per curve from 1 001 to 51. During the exploratory phase of the study different sampling rates were tested with 20 leading to the best results. In particular, better results *and* faster training times were achieved compared to the original data point density.

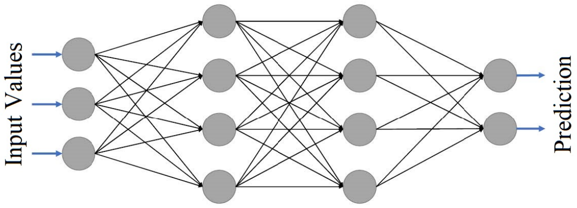

Figure 3. Topology of a Feed-Forward Neural Network. Neurons are represented by circles, while arrow indicate the direction of data transfer.

To facilitate a direct and transparent comparison of the results the predicted sub-sampled curves are linearly extrapolated to have the same point density as the original curves.

Moreover, in an additional approach, the potential of treating the entire curve as a sequence is explored. In this *Sequential Curve Approach* (SC), the model is tasked with predicting all 1 001 points of the stress-strain curve based on the geometry parameters.

## 2.4. Machine learning models

Four fundamentally different classes of machine learning models are used to make prediction in order to provide an extensive overview and comparison of methods potentially suitable for the task at hand.

Representing the class of parametric models, i.e. models that learn by fitting parameters of a mapping function to the data, Artificial Neural Networks (ANN) and Support Vector Regression (SVR) are implemented.

ANNs consist of a multitude of simple computational units, called neurons, that are arranged in a layered network. In this case, a topology is used where each neuron in one layer is connected to every neuron in the neighboring layers, forming a so called (fully connected) Feed-Forward Neural Network. Each connection between two neurons provides a degree of freedom for the fitting process [6, 7]. A schematic of the topology of such networks is shown in Figure 3.

Support Vector Regression (SVR) can be viewed – in a simplified manner – as a generalized version of linear regression that is not limited to linear problems [8]. A deeper explanation of SVR is omitted at this point, since this model proved to be not suitable for the task at hand.

K-Nearest-Neighbor Regression (KNNR) and XGBoost (XGB) represent the class of non-parametric models. These models do not fit internal parameters to the data. Rather, predictions are made based on known data directly. Therefore, changing the training data automatically changes the model's predictions.

KNNR predicts stresses for a new combination of input parameters, also called features, by identifying the $K$ closest known examples from the input space of the training data. The predicted stress value, then, is the average of these $K$ known stress values [9].

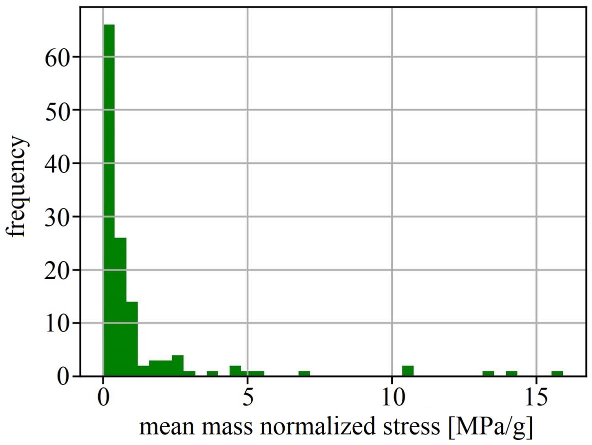

Figure 4. Distribution of mean mass normalized stress values for all 130 available curves.

XGBoost is an advanced and optimized implementation of Decision Trees [10]. Decision Trees consist of a set of "if – then" statements, also called decision rules, that divide the input space into subdomains or partitions. Predictions are then made based on the partition to which the sample at hand belongs, typically predicting the average target value of the partition [8].

For the additional SC Approach mentioned at the end of Section 2.3, Temporal Convolutional Networks (TCNs) [11] are employed to generate the sequence from the input features. TCNs are a class of neural networks that are particularly well-suited for sequence modeling tasks, as they make use of temporal convolutions to capture dependencies of different lengths in the data. Through multiple layers of dilated convolutions, TCNs can effectively learn and represent complex patterns across various time scales in the sequence, making them a promising choice for predicting the entire stress-strain curve with high resolution.

### 2.5. Course of action

To provide a starting point, first, the four main models introduced in Section 2.4 are trained on the original 5 features, namely size, waist, length, angle and strut thickness, using the SP and the WC approach. Here, a rudimentary hyperparameter optimization is carried out to obtain parameters that are close to the optimal choice. Hyperparameters are parameters describing the architecture and/or functionality of the model. Using these trained models, feature engineering is performed, which is a set of techniques aiming at creating data representations that make the learning process easier. For this sake, a selection of feature combinations is evaluated in combination with every model regarding their predictive performance. The feature combinations found to lead to best results – which can be different for different models – are then used to retrain the models and perform a more extensive hyperparameter optimization, resulting in the final optimized models.

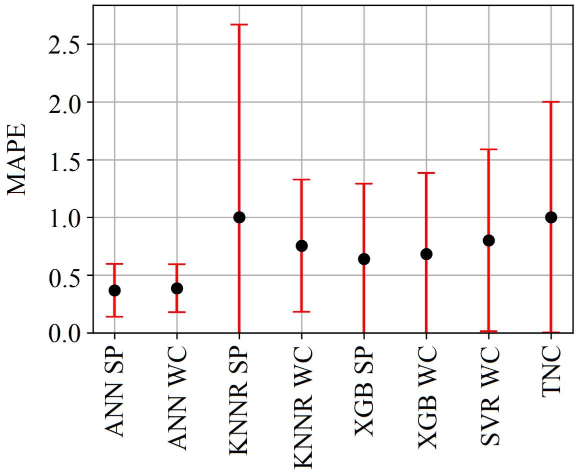

Figure 5. Performance of each model-approach combination on the final test set. Error bars denote the standard deviation with regard to individual curves.

In order to estimate model performance reliably while trying to prevent overfitting, i.e. good performance on known data and bad performance on new data, throughout the model optimization process a cross validation (CV) procedure is implemented.

The data set at hand is imbalanced with high-stress curves being highly underrepresented, see Figure 4. Therefore, the CV procedure is stratified to guarantee that the distribution of mean stress values while splitting data (train, validation and test) stays as close as possible to the distribution in the original data set. For every model, a 4-Fold-CV is implemented. The splitting procedure being repeated three times, resulting in 12 different train/validation splits with a size ratio of 75 to 25.

At the very end of the project, all optimized models are trained on all 115 curves available for the modeling process and then used to make predictions for 15 never-seen-before curves – and therefore until now unknown geometry describing input parameters – in an attempt to simulate the real application scenario.

## 3. Results and discussion

The summarized performances of all trained models can be seen in Figure 5. It is apparent that the choice of the approach, meaning SP versus WC, had no significant effect on the predictive performance of the models. The particularly bad performance of KNNR (SP) can only be explained with overfitting, even though all possible precautions were taken. Here, one particular curve of the final test set, was predicted extremely bad, drastically worsening the average performance of the model. In all prior stages of the model building process its performance was comparable to the other models, especially to KNNR (WC). SVR (SP) is not shown in the figure because this approach was discarded after the initial hyper parameter optimization stage due to extraordinarily bad performance together with unreasonably large training times.

In general, the performance of all models, with the exception of KNNR (SP), performed even better of the final test set than estimated during the model building process, all models having a MAPE of at least 0.573 in the case of ANN (SP). First of all, this shows that the efforts to provide reliable performance measures through suitable data splitting techniques generally paid off since the expectations from the model building phase could be met and even surpassed in the actual application scenario.

The overall improvement in predictive performance can be explained by the larger effective train set. During cross-validation, models were trained on sets consisting of 86 stress-strain curves (75 % of 115), whereas all 115 curves were available to train the final models. This is an increase in data amount of about 28 %. The significant improvement in performance through more data indicates that the amount of training data is at least one of the limiting factors regarding predictive capability.

Best performance was achieved by Feed-Forward Artificial Neural Networks using the Single Point Approach with a MAPE of 0.367 ± 0.230. The network used the original input variables with the exception of *angle* being replaced with *gap*, which for this model increased the predictive performance. The final network consist of 16 layers, each containing 18 neurons. Figure 6 shows examples of particularly good and bad predictions produced by the model. It is able to predict the general order of magnitude correctly, however, it struggles with predicting the actual shape, which explains the large standard deviation of the MAPE. This is not surprising since different sets of geometry parameters can lead to stress-strain curves that drastically vary in shape due to stability failure of the structure.

In the additional approach, TCNs were employed to model the stress-strain curves as sequence data. It is important to note that the TCN faced a more demanding task, as it aimed to predict the full curve consisting of 1 001 data points, which is inherently more complex compared to what the other models were required to accomplish. Consequently, the TCN did not prove to be as adept for this specific problem context as was hoped. However, exploring this method served to reinforce the decisions made in the main study, and highlighted the importance of model selection in relation to the complexity of the task.

## 4. Conclusion and future work

A model was trained that can predict stress-strain curves for a re-entrant auxetic structure, see Figure 1, based only on geometry-describing parameters, achieving a Mean Absolute Percentage error of 0.367 ± 0.230. Even though at this point the model cannot replace conventional finite element simulations, it allows to highly reduce the amount of simulations that actually need to be carried out by providing a rough estimate of the expected curve within a fraction of a second – compared to up to 48 h for the classical simulation.

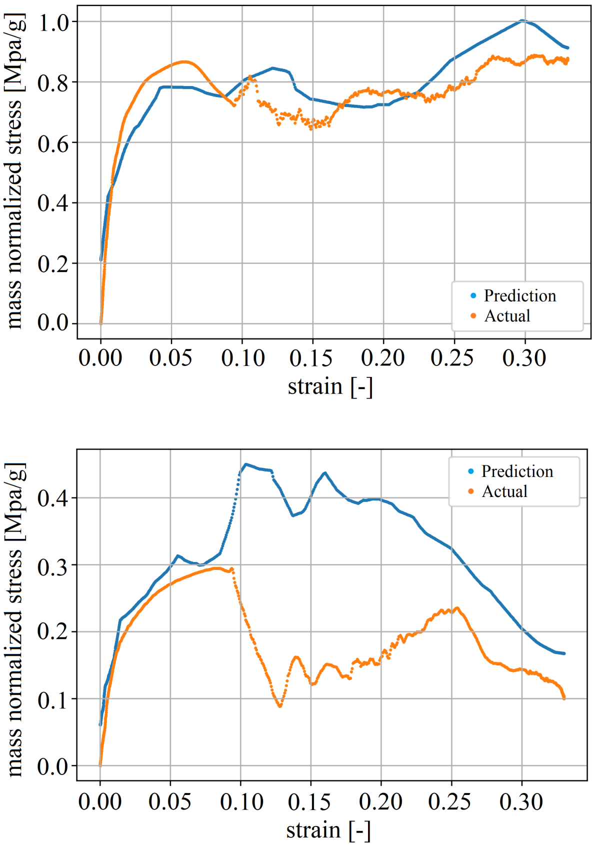

Figure 6. Particularly good (top) and particularly bad (bottom) predictions produced by the final optimized Artificial Neural Network using the SP approach.

Furthermore, it was shown that Feed-Forward Artificial Neural Networks are best suited for the task at hand, significantly outperforming all other studied options. Additionally, models like Support Vector Regression and Temporal Convolutional Networks were found to be particularly unsuitable for the studied problem.

Since an increase in training data volume enhanced the predictive performance in this study, further improvements to the models can be expected by incorporating more and more simulated curves into the data set.

Apart from a simple increase in data volume, additional models are planned to be trained predicting effective quantities like the absorbed energy during the compressive loading. Since predicting one quantity per geometry parameter combination instead of a whole curve reduces the complexity of the problem, better results are to be expected. Also, since in this case no curve shape needs to be matched, the spread in performance is expected to be reduced. Given the expected possible frequency with which predictions can be made, an extension of the framework is conceivable where the user chooses a desired specific energy absorption for a particular application and the model provides the corresponding geometry.

## References

[1] A. Bezazi, F. Scarpa. Mechanical behaviour of conventional and negative Poisson's ratio thermoplastic polyurethane foams under compressive cyclic loading. *International Journal of Fatigue* **29**(5):922–930, 2007. https://doi.org/10.1016/j.ijfatigue.2006.07.015

[2] F. Scarpa, P. Pastorino, A. Garelli, et al. Auxetic compliant flexible PU foams: Static and dynamic properties. *Physica Status Solidi (B) Basic Research* **242**(3):681–694, 2005. https://doi.org/10.1002/pssb.200460386

[3] M. Mozaffar, R. Bostanabad, W. Chen, et al. Deep learning predicts path-dependent plasticity. *Proceedings of the National Academy of Sciences of the United States of America* **116**(52):26414–26420, 2019. https://doi.org/10.1073/pnas.1911815116

[4] S. Bronder, S. Diebels, A. Jung. Neural networks for structural optimisation of mechanical metamaterials. *PAMM* **20**(1):e202000238, 2020. https://doi.org/10.1002/pamm.202000238

[5] S. Bronder, F. Herter, A. Röhrig, et al. Design study for multifunctional 3D re-entrant auxetics. *Advanced Engineering Materials* **24**(1):2100816, 2022. https://doi.org/10.1002/adem.202100816

[6] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review* **65**(6):386–408, 1958. https://doi.org/10.1037/h0042519

[7] J. Patterson, A. Gibson. *Deep Learning: A Practitioner's Approach.* O'Reilly, 2017.

[8] M. Kuhn, K. Johnson. *Applied predictive modeling.* Springer, 2013.

[9] G. James, D. Witten, T. Hastie, R. Tibshirani. *An Introduction to Statistical Learning: with Applications in R.* Springer, 2013.

[10] T. Chen, C. Guestrin. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794. ACM, 2016. https://doi.org/10.1145/2939672.2939785

[11] S. Bai, J. Zico Kolter, V. Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, [2023-06-30]. https://doi.org/10.48550/arXiv.1803.01271