A comparison of two neural network architectures for fast structural response prediction

Denny Thaler[1,*]**, Franz Bamer**[1]**,** and **Bernd Markert**[1]

[1] Institute of General Mechanics, RWTH Aachen University, Eilfschornsteinstr. 18, 52062 Aachen

In this contribution, we compare two different neural network architectures to predict the response statistics of structures. The overall goal is a significant speed-up of the numerically expensive Monte Carlo simulation. The first approach is based on a convolutional neural network that learns from the whole excitation history, whereas the second approach is based on a feed-forward network architecture learning from hand-designed features. Both procedures use supervised learning: The neural networks learn from an initial subset before the prediction of the response statistics of the Monte Carlo simulation is possible.

© 2021 The Authors. *Proceedings in Applied Mathematics & Mechanics* published by Wiley-VCH GmbH.

1 Introduction

The crude Monte Carlo simulation strategy has proven useful to estimate the probability of failure of structures subjected to ground motion [1]. Given that engineers strive for a very low probability of failure, a high number of samples must be evaluated for a reliable estimation of the response statistics. If the respective structures are complex, the computational cost to evaluate the response statistics becomes disproportionately high, which requires us to develop efficient surrogate strategies [1]. Machine learning enhanced methods reduce the computational effort significantly [2] and are applied to several problems within the field of earthquake engineering [3]. In this contribution, we compare two different machine learning approaches to predict response statistics. While the overall strategy and the goal of the proposed approaches remain the same, they differ in their input variables and their neural network architecture. By supporting the neural network with hand-designed input features, quite simple neural networks can predict the response statistic [4]. However, by including convolutional layers, neural networks can autonomously learn from the time history of the excitation [5].

2 Machine learning prediction of structural response statistics

Considerably large sample sets are necessary to perform a reliable Monte Carlo simulation. Since recorded earthquakes on the respective building site are far from enough for a Monte Carlo simulation, we generated a set of artificial earthquakes $\mathcal{S}$. As a result of this, a nonstationary Kanai-Tajimi is used to create excitation histories $\ddot{x}_g^{(i)}(t)$ which preserves the site-dependency of the original record. The finite element method is used to evaluate the response set $\mathcal{R}$:

$$\mathcal{S} = < \ddot{x}_g^{(1)}(t), \cdots, \ddot{x}_g^{(k)}(t) > \Rightarrow \ \mathcal{R} = < \mathbf{x}^{(1)}(t), \cdots, \mathbf{x}^{(k)}(t) > . \tag{1}$$

Replacing the computationally expensive calculation of nonlinear finite element simulations by neural network predictions leads to quick estimations of the response statistics. However, before the neural network can predict the structural response due to excitation, it needs to be fitted to the problem. In the two considered approaches of this comparison, supervised learning is used to train the neural network. Therefore, a set needs to be evaluated by finite element simulations. As we aim for computational benefits, this set should be small compared to the entire sample set, i.e. $n < k$.

$$\mathcal{S}_{\text{training}} = < \ddot{x}_g^{(1)}(t), \cdots, \ddot{x}_g^{(n)}(t) > \Rightarrow \ \mathcal{R}_{\text{training}} = < \mathbf{x}^{(1)}(t), \cdots, \mathbf{x}^{(n)}(t) > . \tag{2}$$

From the time history of the structural response, the story drift ratio – the drift between two floors divided by the height of the story – is calculated. The maximum drift supports the indication for structural failure in this paper. Therefore, the neural networks learn to predict peak story drift ratios ($PSDR$).

$$\mathcal{S}_{\text{NN,training}} = < \mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)} > , \ \mathcal{R}_{\text{NN,target}} = < PSDR^{(1)}, \cdots, PSDR^{(n)} >$$

$$\mathcal{S}_{\text{NN}} = < \mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(k)} > \Rightarrow \hat{\mathcal{R}}_{\text{NN}} = \hat{PSDR}^{(1)}, \cdots, \hat{PSDR}^{(k)} > . \tag{3}$$

---

* Corresponding author: e-mail thaler@iam.rwth-aachen.de, phone +49 241 80 94601, fax +49 241 80 92231

3   Comparison of representation learning and feature engineering

Neural network architectures which use convolutions in the first layers can recognize patterns in structured data such as images or time series. In this study, the convolutional neural network extracts the relevant features from the time history of the generated ground motions to learn the response behavior of the structure, which is called representation learning. The input vector $\mathbf{X}_{\text{CNN}}$ for the convolutional neural network consists of the raw data points of the generated excitation.

Simpler neural network architectures can learn from hand-designed features as input instead of the whole time history. There is extensive research on earthquake intensity measures in earthquake engineering [3]. Using these hand-designed measures as input for neural networks instead of raw data is called feature engineering. In this study, the input vector $\mathbf{X}_{\text{FFNN}}$ consists of the effective peak acceleration, the cumulative absolute velocity, the velocity spectrum intensity, the spectral acceleration at the natural period, and the spectral displacement at the natural displacement. An input parameter study has shown that this combination of intensity measures leads to good results [4]. Additional computations are necessary to extract these features from the generated data.
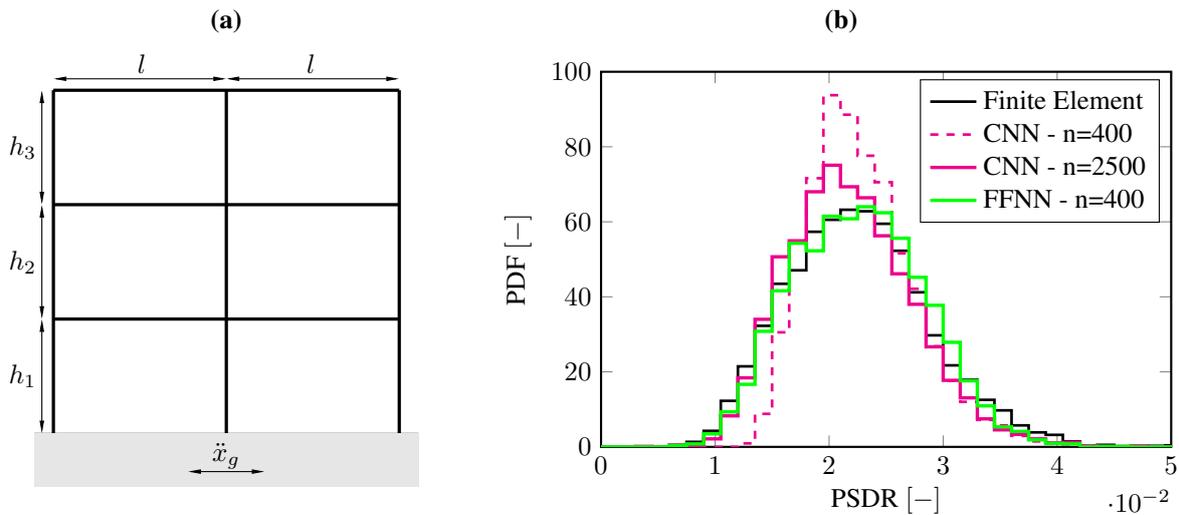
(a)   **(b)**

Fig. 1: (a) Frame structure with two bays and three stories subjected to artificial nonstationary excitation $\ddot{x}_g$; (b) probability density function of the peak story drift ratio (PSDR) of $k = 5000$ samples using the convolutional neural network approach (magenta) with 400 (dashed) and 2500 training samples and the feed forward neural network (green) with 400 training samples.

Figure 1a shows the frame structure used for the numerical demonstration of the proposed approaches. The probability distribution function (PDF) is evaluated for a set $\mathcal{S}$ with $k = 5000$ samples. The evaluation using the finite element method serves as the target for the neural network estimations. The distributions of the neural network predictions are shown in Figure 1b. Using $n = 400$ training samples, the neural network fed by pre-selected intensity measures can predict the response behavior accurately. The mean absolute error of the prediction is 0.14%. On the other hand, the convolutional layer needs a larger share of training data to find the relevant patterns from the data points of the time history. Increasing the number of samples used for the training of the convolutional neural network leads to better results. The mean absolute error of the convolutional neural network reduces from 0.37% to 0.23% by increasing the number of training samples from $n = 400$ to $n = 2500$. However, the simpler feed-forward neural network still outperforms the convolutional neural network, see Fig.1b. A higher number of samples n to enable a more accurate representation leads to negligible computational savings since only $k = 5000$ samples are used in this study.

We conclude that the feed-forward neural network benefits from the condensed information of the chosen intensity measures. Even though the preprocessing leads to some additional computational effort, the method highly benefits from the simple learning procedure. The computational saving of the machine learning procedure, including the evaluation of the training samples and the intensity measure, is mainly dependent on the amount of training data needed. Therefore, using feature engineering to predict the response statistics fast is advantageous.

References

[1] F. Bamer and B. Markert, Mechanics Based Design of Structures and Machines **45** 313-330 (2017).

[2] D. Thaler, F. Bamer and B. Markert, Proc. Appl. Math. Mech. **20** e202000294 (2021).

PAMM · Proc. Appl. Math. Mech. **21**:1 (2021)

3 of 3

[3] H. Sun, H. V. Burton and H. Huang, J. Build. Eng. **33** 101816 (2021).

[4] D. Thaler, M. Stoffel, B. Markert and F. Bamer. Earthquake Engng Struct Dyn **50** 2098-2114 (2021).

[5] F. Bamer, D. Thaler, M. Stoffel and B. Markert, Front. Built Environ.**7** 53 (2021).