Surrogate modeling of stress fields in periodic polycrystalline microstructures using U-Net and Fourier neural operators

Sarthak Kapoor
RWTH Aachen
sarthak.kapoor@rwth-aachen.de

Jaber Rezaei Mianroodi
MPIE Düsseldorf
j.mianroodi@mpie.de

Bob Svendsen
MPIE Düsseldorf, RWTH Aachen
b.svendsen@mpie.de

Mohammad Khorrami
MPIE Düsseldorf
m.khorrami@mpie.de

Nima S. Siboni
nima.siboni@gmail.com

Abstract

In this work, we implement and compare two artificial neural networks (ANNs) —U-Net and Fourier neural operators (FNO) —for surrogate modeling of stress fields in periodic polycrystalline microstructures. Both ANNs were trained on results from the numerical solution of the boundary-value problem for quasi-static mechanical equilibrium in grain microstructures under uniaxial tensile loading. More specifically, they learned mappings from the spatial fields of material properties to the equilibrium stress fields. To generate multiple output fields, one for every stress component, the networks were branched internally into parallel sub-networks at different stages, which were then trained together. We compare various such adaptations to find the best one. For the U-Net-based approach, we show that convolution with periodic padding instead of zero padding gives better accuracy along the system boundaries. We further compare the predictions from the two approaches: the FNO-based approach is more accurate than its U-Net-based counterpart; the normalized mean absolute error incurred on the predicted stress field with respect to the numerical solution is $3.5 - 7.5$ times lower for the former than the latter. In comparison to the U-Net-based approach, the errors in the FNO-based approach are restricted to grain boundaries leading to narrower error distribution.

NeurIPS 2022 AI for Science Workshop.

1 Introduction

The mechanical properties of materials depend heavily on their microstructural features. This includes the morphology of grains and sub-grains, and the presence of dispersed phase precipitates for polycrystalline materials [1, 2, 3]. But the process of experimentally finding the features that result in desired mechanical properties quickly runs into a combinatorial explosion of possible variations, making it both cumbersome and resource-draining. The last few decades saw the growth of computational methods like phase-field, finite element method (FEM), and spectral methods to simulate microstructures which significantly reduced experimental effort [4, 5, 6, 7, 8, 9]. However, being based on the numerical solution of initial boundary value problems, these methods become computationally prohibitive as more non-linear physical effects are considered or the system sizes are increased.

Recently, artificial neural networks (ANN) have attracted much attention in the field of material mechanics simulation due to their ability to learn correlative maps between spatial fields of material properties and mechanical response. Once trained, ANNs can generate response fields by performing a single pass through the network which leads to tremendous speedups in comparison to numerical solvers. U-Net and Fourier neural operators (FNO) are two such ANNs that are emerging as a popular choice given their ability to learn correlations between spatial fields with the same dimensions [10, 11, 12, 13, 14]. U-Net is a convolutional neural network which was originally developed for application in biomedical image segmentation [15]. It encodes the input n-times into lower spatial dimensions and then decodes it back n-times to recover the original spatial dimensions. Throughout the network, multiple convolution kernels learn features at different spatial dimensions/resolutions. Consequently, it learns a mapping between finite-dimensional Euclidean space [16, 17]. On other hand, FNO implements operator-based learning by learning linear transforms in Fourier space [18]. Parameterization in Fourier space allows the network to learn the underlying frequency information in the given data distribution. To much extent, this is independent of the spatial resolution of the data, provided that features to be learned are frequent enough in the given resolution, and it corresponds to learning a family of functions between infinite dimensional Banach space. However, here we only focused on inferences on training resolution to compare them with that from U-Net.

In this work, we implemented the two ANNs for surrogate modeling of stress fields in polycrystalline microstructures which are subjected to uniaxial tensile loading. We discuss two key implementation details: the extension of ANNs to generate multiple spatial fields and the incorporation of boundary conditions into the architecture of the network. We compare U-Net and FNO in terms of accuracy, sources of errors, and effective speedups in comparison to the numerical solver to generate the first Piola-Kirchhoff stress tensor from given geometry and material properties. We start with discussions on dataset generation, network architectures, and network training details, and then discuss the experiments and results followed by the conclusion.

2 Methods

2.1 Micromechanics dataset

Material microstructures found in nature vary tremendously in terms of geometrical features and composition. For sake of simplicity, we restrict ourselves to pure polycrystalline metallic materials which are composed of a large number of grains with randomly oriented crystal structures. When the material is under loading, these grains experience different stresses depending on how they align with the loading direction. To mimic this situation in 2D, we built the grain structure using periodic Voronoi tessellation of square domains and assigned different material properties (Young's modulus, Poisson's ratio, Yield stress) to each grain at random. This way each grain experiences different levels of stress. Further, the material properties were kept homogenous within the grains.[1]

We used "DAMASK —The Düsseldorf Advanced Material Simulation Kit" [19], which is a spectral-based numerical solver, to solve the boundary value problem for quasi-static mechanical equilibrium in the grain microstructures under small uniaxial tensile loading. The material was assumed to be elastoplastic with pure plasticity [20]. The first Piola-Kirchhoff stress tensor $\mathbf{P}_{ij}$ at equilibrium was

[1] corresponds to a simpler situation where all the grains within the region of interest have variation in crystallographic orientation along one axis only.

received from the solver and was post-processed to generate 2D fields of each stress component. Due to uniaxial tensile loading, only five of nine stress components had non-trivial stress distribution ($\mathbf{P}_{11}$, $\mathbf{P}_{22}$, $\mathbf{P}_{23}$, $\mathbf{P}_{32}$, $\mathbf{P}_{33}$); remaining components had homogenous response throughout the microstructure. In this work, we use ANNs to only generate non-trivial stress components.

Therefore, the dataset consisted of 2D spatial fields which inherently specified the geometry of the microstructures. Spatial fields of material properties were input to the ANN, while spatial fields of corresponding non-trivial stress components were the target labels. We generated the datasets at 64×64 and 256×256 resolutions.

2.2 Network architecture

U-Net Our implementation is based on the standard U-Net and has the following additional specification: three-channel input and five-channel output with four encoding and decoding stages, $(2, 2)$ upsampling layer with bilinear interpolation, $(9, 9)$ separable 2D convolutions with either zero or periodic padding to retain spatial dimensions. We found that not using nonlinear activation functions after convolution layers gave better accuracy for our dataset. A detailed architecture is available in the Appendix Figure A5.

Fourier neural operator We implemented Fourier neural operator (FNO) network as devised by Li et. al, [18] and extended it to generate multiple output fields. Just like U-Net, the network takes three-channel input and gives five-channel output. It uses fully connected layers to raise/decrease the number of channels of activation maps. We used a fixed number of 32 channels for every Fourier layer (four layers stacked one after the other with ReLU activation between each of them). To transform back and forth between real and Fourier space, we used fast Fourier transforms. However, not all the Fourier modes were used in the linear transform within the Fourier layer. First n-modes in each spatial dimension were preserved, while the remaining were set to zero. This mode selection is an important hyperparameter for FNO, which at large, decides the size of the parameter space. We found the test error to reduce as the number of preserved modes increased. But this is accompanied by an almost quadratic increase in parameter space. Nevertheless, this provides an easy-to-tune hyperparameter to get more accuracy at cost of computing time. We used 20 as the number of preserved modes for all the experiments. Details about the architecture can be found in the Appendix Figure A6.

Multi-output adaptation The spatial fields for different stress components have common underlying geometry defined by the grain structure, however, each of them specifies stress in different directions giving them different physical characteristics. As such, one can imagine a surrogate ANN that generates fields for multiple stress components to have two parts: the first part learns a common set of parameters for all the stress component fields while the second part specializes in each one of them. The latter can be a collection of parallel independent networks each generating one field. For both U-Net and FNO, we did this by branching out the main network from a certain layer into sub-networks. For U-Net, we experimented with three such adaptations where the point of branching the network was either the first encoding layer, the first decoding layer, or the output layer. Similarly for FNO, this was either the first Fourier layer, the third Fourier layer, or the output layer. The sooner the branching happens, the bigger the sub-networks become with more trainable parameters.

Incorporation of boundary conditions In the numerical modeling of physical systems, given that the system ceases to exist beyond its computational domain, extra conditions on top of physical assumptions are required to specify the behavior at the boundaries. These are called boundary conditions (BC). No-flux, reflective, and periodic boundary conditions are among the popular ones. Incorporating BCs specific to the dataset directly into the network architectures of surrogate ANNs can train them to generate fields that obey these conditions, even after training on small datasets. We used periodic boundary conditions to numerically solve for stress fields in periodic polycrystalline microstructures. This assumes that in every dimension, the simulated system represents one period of an infinite periodic microstructure. To incorporate this information as an inductive bias into the network, we used periodic padding in convolution operations for U-Net. On the other hand, the Fourier layers in FNO learn weights in the Fourier space which is characterized by periodic sinusoidal functions. Therefore, the inductive bias for periodicity is already in place for FNO.

2.3 Training and testing

For both networks, we used the same random split of the dataset into 800 training, 100 validation, and 100 testing cases. For U-Net, we used mean absolute error (MAE) as the loss function which was also used as a metric to monitor validation loss. The choice was based on findings by Qi et. al [21], who showed that MAE loss is more robust to noise than mean squared loss (MSE) for training deep neural networks. On the other hand, we used MSE as the loss function with random initialization of the weights for FNO, as used by Li et. al [18]. For both networks, we used Adam optimizer [22] with learning rates of 0.001. By end of the training, the validation losses for both networks were fairly saturated.

2.4 Metric for average field errors

To compare the performance of networks in the prediction of the stress response on average, we aggregated pixel errors from the entire field by averaging them. We found normalized mean absolute errors (NMAE) to be a better-suited metric than mean absolute or relative errors when working with multiple stress component predictions. As for a given test case, the range of values for each stress component is different, reporting absolute errors (like MAE or MSE) would be misleading when comparing the average errors for different components. In this case, accumulating relative errors at each pixel would be a preferred choice as it would factor out magnitudes of different stress component values. However, as the stress values range from negative to positive values, some of these are close to zero. These regions with almost zero stress values give huge contributions if we compute relative errors, irrespective of the local error incurred here. By normalizing MAE with the range of stress value, NMAE avoids both these issues. It is given by

$$NMAE = \frac{\frac{1}{mm}\sum_{i}^{mm}|X_i - Y_i|}{max(X) - min(X)}, \tag{1}$$

where X, Y are stress fields from the numerical solver and ANN prediction respectively, each being an $m \times m$ array. Here, the term in the numerator is nothing but MAE. If not mentioned otherwise, average field errors were averaged over 100 cases for stable estimates.

3 Results

3.1 Multi-output adaptation of ANNs

To compare different multi-output adaptations, we trained and tested them with 64×64 resolution dataset. As expected, the number of trainable parameters was higher for variants with early branching of the network and it also directly affects the network evaluation time (see Table 1). However, increasing the number of parameters does not always improve the stress field prediction error. As shown in Figure 1, the error for component $\mathbf{P}_{33}$ using U-Net1, where the branching happens at the first encoding stage, hence having the maximum number of parameters, is an order of magnitude higher than other adaptations. It might be because of the limitations of training data to find better optima in what is now a bigger parameter space. Nevertheless, prediction errors were also not the least for U-Net with the most delayed branching (U-Net3) due to insufficient parameter space for required learning. A trade-off between these two factors happens when U-Net is branched from the first decoding layer (U-Net2), which led to the lowest errors in four out of five components. In the case of FNO, the trend is more regular where the prediction error increased monotonically for all stress components when the branching of the network happens in earlier layers. FNO3, where the network branching happens from the output layer, gave the lowest errors with the least number of parameters among all variants. Therefore, the last fully connected layer with just 645 parameters was sufficient to distinguish between all five components with lower errors than U-Net. We selected U-Net2 and FNO3 variants for further study.

3.2 Stress field predictions

We trained and tested the networks at 256×256 resolution dataset. Both networks successfully predicted the underlying geometry for all the stress components. Further, the range of stresses for each component corresponded closely to the stress field from the numerical solver (see Figure

Table 1: Inference time (averaged over 10 cases with 5-component prediction at 64×64 resolution) and the number of parameters for multi-output variants of U-Net and FNO. The rows with bold font represent the U-Net and FNO variants selected for further study.

	Variant	Inference time (ms)	Number of trainable parameters
U-Net1	From first encoding stage	171	$329,250$
U-Net2	**From first decoding stage**	**123**	$\mathbf{245,910}$
U-Net3	From output stage	73	$72,086$
FNO1	From first Fourier layer	677	$32,811,013$
FNO2	From third Fourier layer	428	$19,695,365$
FNO3	**From output layer**	**91**	$\mathbf{6,562,821}$

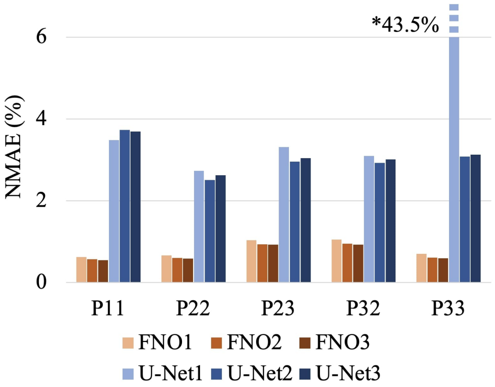

Figure 1: Normalized mean absolute error for multi-output variants of U-Net and FNO. The errors were computed for 5-stress-component prediction at 64×64 resolution.

2). FNO gave almost $3.5 - 7.5$ times lower errors than U-Net for different stress components: NMAE(%) for the former were $[0.282, 0.287, 0.389, 0.388, 0.266]$ while that of the latter were $[2.15, 1.49, 1.41, 1.42, 1.55]$, for $\mathbf{P}_{11}, \mathbf{P}_{22}, \mathbf{P}_{23}, \mathbf{P}_{32}, \mathbf{P}_{33}$ respectively. But if we investigate the error fields to compare the point-wise errors in the predicted fields in Figure 3, there is a huge deviation from the error limits typical to numerical solvers. In fact, the errors are almost in the same range as the stress values themselves.

We have broader error distribution in U-Net than FNO with significant contributions from within the grains (see Figure 3). As a given convolutional filter runs over the entire field, it might have a tendency to adapt to both grains and grain boundaries because these spatial features are locally coupled (grain boundaries are always surrounded by grain regions even for small square patches treated by kernels). This might lead to non-specialized learning for kernels, which can indiscriminately propagate errors over the entire domain. On other hand, FNO learns linear maps in Fourier space which corresponds to learning non-local features at different frequencies in real space. Some weights can specialize in high-frequency features like regions inside the grains, and some weights can specialize in low-frequency regions like grain boundaries. Therefore, errors from within the grains are minimal, leading to a much narrower error distribution.

Further, both ANNs are unable to faithfully generate regions where stress values change quickly. These regions include grain boundaries and points of stress concentration. As seen in Figure 4a, where we used a different geometry corresponding to a two-phase matrix-precipitate system, stress concentration regions on the corners of the precipitate are smudged in the ANN predictions. The inability to capture these sharp features arises from the fact that the loss function allows for smoothening of sharp gradients as far as average errors are minimized.

Boundary conditions We experimented with two types of padding for convolution layers in U-Net: zero padding and periodic padding. As discussed before, using periodic padding enforces the appropriate boundary conditions on generated stress fields. Therefore, it leads to lower average errors than zero padding: NMAE(%) for zero padding was $[2.20, 1.82, 1.80, 1.79, 1.82]$ for $\mathbf{P}_{11}, \mathbf{P}_{22}, \mathbf{P}_{23}, \mathbf{P}_{32}, \mathbf{P}_{33}$

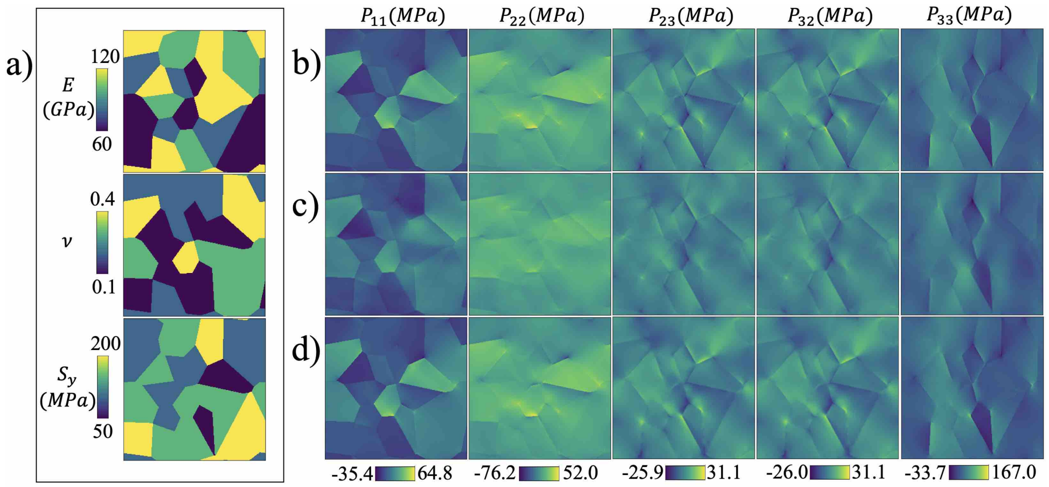

Figure 2: Network predictions on a test case with 256×256 resolution and 20 grains —**a)** Material properties as input fields to the network, **b)** simulation output of stress components with non-trivial distribution from the numerical solver and corresponding network predictions from **c)** U-Net, and **d)** FNO. Loading direction ↕.

respectively. Moreover, artifacts along the system boundaries emerge when zero padding is used. On other hand, periodic boundary conditions are fulfilled by the nature of Fourier layers in FNO. Therefore, we do not observe any additional errors around the boundaries (see Figure 4b).

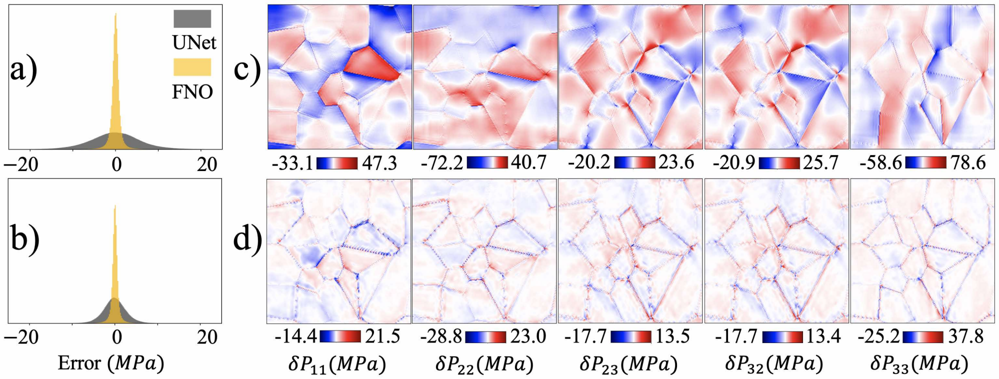

Figure 3: Error in stress field prediction for a test case with 256×256 resolution and 20 grains —distribution of errors for **a)** normal components $\mathbf{P}_{11}, \mathbf{P}_{22}, \mathbf{P}_{33}$ and **b)** shear components $\mathbf{P}_{23}, \mathbf{P}_{32}$; prediction error in all stress components with non-trivial distribution for **c)** U-Net, and **d)** FNO. Loading direction ↕.

3.3 Speedups in comparison to the numerical solver

The stress response in the numerical solver is computed for the applied load in multiple increments, and at every increment, it solves for the mechanical equilibrium of displacement fields using fixed-point iterations. On the other hand, ANNs predict the stress response by performing a single forward pass through the trained network. Here, most of the computational burden is transferred to the training of the network, while the inference from a trained network happens at a little cost. We computed the speedups for both the ANNs as compared to the numerical solver by running them on a single core of 16-Core Intel Xeon W processor clocked at 3.2GHz. Of course, using GPUs for ANNs would give even better scalability, but we did not use them so as to avoid the effects of parallelization overheads in the two approaches. Compared to the numerical solver, we observed average speedups of 2500 for FNO and 1000 for U-Net. The predicted/simulated stress fields had 256×256 spatial resolution.

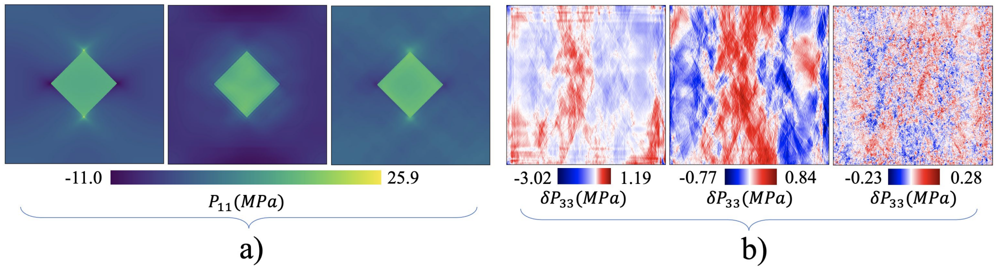

Figure 4: **a**) $\mathbf{P}_{11}$ fields for a binary matrix-precipitate system containing a single square precipitate. Fields were generated by (from left to right) the numerical solver, U-Net, and FNO. **b**) Averaged error field computed over 100 test cases with polycrystalline microstructure. ANN used for prediction —(from left to right) U-Net with zero padding convolutions, U-Net with periodic padding convolutions, and FNO. Loading direction ↕.

4 Conclusion

We adapted U-Net and FNO to generate spatial fields of five non-trivial stress components under uniaxial tensile loading, given the spatial field of material properties as input. While an early branching of ANNs to generate multiple fields led to larger parameter space, we did not find it to improve the accuracy of predicted stress fields. Both ANNs gave tremendous speedups as compared to the numerical solver and captured the underlying geometry in the stress fields. However, the errors generated were in the range of stress values themselves. Incorporating boundary conditions of the stress field in terms of periodic padding in convolution layers improved the average errors in U-Net. Errors in U-Net were generated throughout the entire field, whereas they were limited to grain boundaries in FNO. As such, the average errors were significantly lower for the latter. However, due to averaging loss functions like MSE and MAE, both approaches show limited ability to capture regions of stress concentrations where the stress value varied steeply.

References

[1] E. O. HALL. Variation of hardness of metals with grain size. *Nature*, 173(4411):948–949, 1954.

[2] T. Gladman. Precipitation hardening in metals. *Materials Science and Technology*, 15(1):30–36, 1999.

[3] K. Lu, L. Lu, and S. Suresh. Strengthening materials by engineering coherent internal boundaries at the nanoscale. *Science*, 324(5925):349–352, 2009.

[4] Nele Moelans, Bart Blanpain, and Patrick Wollants. An introduction to phase-field modeling of microstructure evolution. *Calphad*, 32(2):268–294, 2008.

[5] Julian Kochmann, Stephan Wulfinghoff, Stefanie Reese, Jaber Rezaei Mianroodi, and Bob Svendsen. Two-scale fe–fft- and phase-field-based computational modeling of bulk microstructural evolution and macroscopic material behavior. *Computer Methods in Applied Mechanics and Engineering*, 305:89–110, 2016.

[6] P. Shanthraj, L. Sharma, B. Svendsen, F. Roters, and D. Raabe. A phase field model for damage in elasto-viscoplastic materials. *Computer Methods in Applied Mechanics and Engineering*, 312:167–185, 2016. Phase Field Approaches to Fracture.

[7] Huiyu Sun, Shenglin Di, Nong Zhang, and Changchun Wu. Micromechanics of composite materials using multivariable finite element method and homogenization theory. *International Journal of Solids and Structures*, 38(17):3007–3020, 2001.

[8] P. Eisenlohr, M. Diehl, R.A. Lebensohn, and F. Roters. A spectral method solution to crystal elasto-viscoplasticity at finite strains. *International Journal of Plasticity*, 46:37–53, 2013. Microstructure-based Models of Plastic Deformation.

[9] P. Shanthraj, P. Eisenlohr, M. Diehl, and F. Roters. Numerically robust spectral methods for crystal plasticity simulations of heterogeneous materials. *International Journal of Plasticity*, 66:31–45, 2015. Plasticity of Textured Polycrystals In Honor of Prof. Paul Van Houtte.

[10] Zijiang Yang, Yuksel C. Yabansu, Dipendra Jha, Wei keng Liao, Alok N. Choudhary, Surya R. Kalidindi, and Ankit Agrawal. Establishing structure-property localization linkages for elastic deformation of three-dimensional high contrast composites using deep learning approaches. *Acta Materialia*, 166:335–345, 2019.

[11] Zhenze Yang, Chi-Hua Yu, Kai Guo, and Markus J. Buehler. End-to-end deep learning method to predict complete strain and stress tensors for complex hierarchical composite microstructures. *Journal of the Mechanics and Physics of Solids*, 154:104506, 2021.

[12] J. R. Mianroodi, N. H. Siboni, and D. Raabe. Teaching solid mechanics to artificial intelligence-a fast solver for heterogeneous materials. *npj Computational Materials*, 7:99, 2021.

[13] Meer Mehran Rashid, Tanu Pittie, Souvik Chakraborty, and N. M. Anoop Krishnan. Learning the stress-strain fields in digital composites using fourier neural operator. *https://arxiv.org/abs/2207.03239*, 2022.

[14] Vivek Oommen, Khemraj Shukla, Somdatta Goswami, Remi Dingreville, and George Em Karniadakis. Learning two-phase microstructure evolution using neural operators and autoencoder architectures. *https://arxiv.org/abs/2204.07230*, 2022.

[15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *https://arxiv.org/abs/1505.04597*, 2015.

[16] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

[17] Ding-Xuan Zhou. Universality of deep convolutional neural networks. *Applied and Computational Harmonic Analysis*, 48(2):787–794, 2020.

[18] Zongyi Li, Nikola B. Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew M. Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *CoRR*, abs/2010.08895, 2020.

[19] F. Roters, M. Diehl, P. Shanthraj, P. Eisenlohr, C. Reuber, S.L. Wong, T. Maiti, A. Ebrahimi, T. Hochrainer, H.-O. Fabritius, S. Nikolov, M. Friák, N. Fujita, N. Grilli, K.G.F. Janssens, N. Jia, P.J.J. Kok, D. Ma, F. Meier, E. Werner, M. Stricker, D. Weygand, and D. Raabe. Damask - the düsseldorf advanced material simulation kit for modeling multi-physics crystal plasticity, thermal, and damage phenomena from the single crystal up to the component scale. *Computational Materials Science*, 158:420, 2019.

[20] Joachim Rösler, Harald Harders, and Martin Bäker. *Mechanical behaviour of engineering materials: Metals, ceramics, polymers, and Composites*. Springer, 2007.

[21] Jun Qi, Jun Du, Sabato Marco Siniscalchi, Xiaoli Ma, and Chin-Hui Lee. On mean absolute error for deep neural network based vector-to-vector regression. *IEEE Signal Processing Letters*, 27:1485–1489, 2020.

[22] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *https://arxiv.org/abs/1412.6980*, 2014.

Appendix

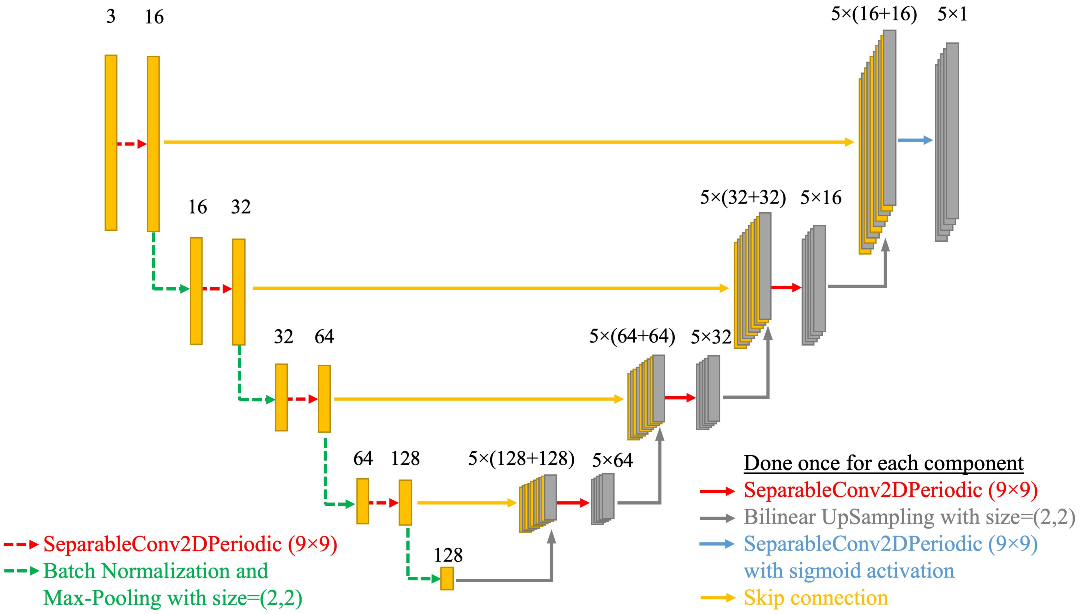
3
16
16
32
32
64
64
128
128
5×(128+128)
5×64
5×(64+64)
5×32
5×(32+32)
5×16
5×(16+16)
5×1
SeparableConv2DPeriodic (9×9)
Batch Normalization and
Max-Pooling with size=(2,2)
Done once for each component
SeparableConv2DPeriodic (9×9)
Bilinear UpSampling with size=(2,2)
SeparableConv2DPeriodic (9×9)
with sigmoid activation
Skip connection

Figure A5: Network architecture of U-Net where the branching of network happens from the first decoding stage to generate multiple output fields. Each activation map is labeled with its number of channels; spatial dimensions get halved after each encoding stage and doubled after each decoding stage.

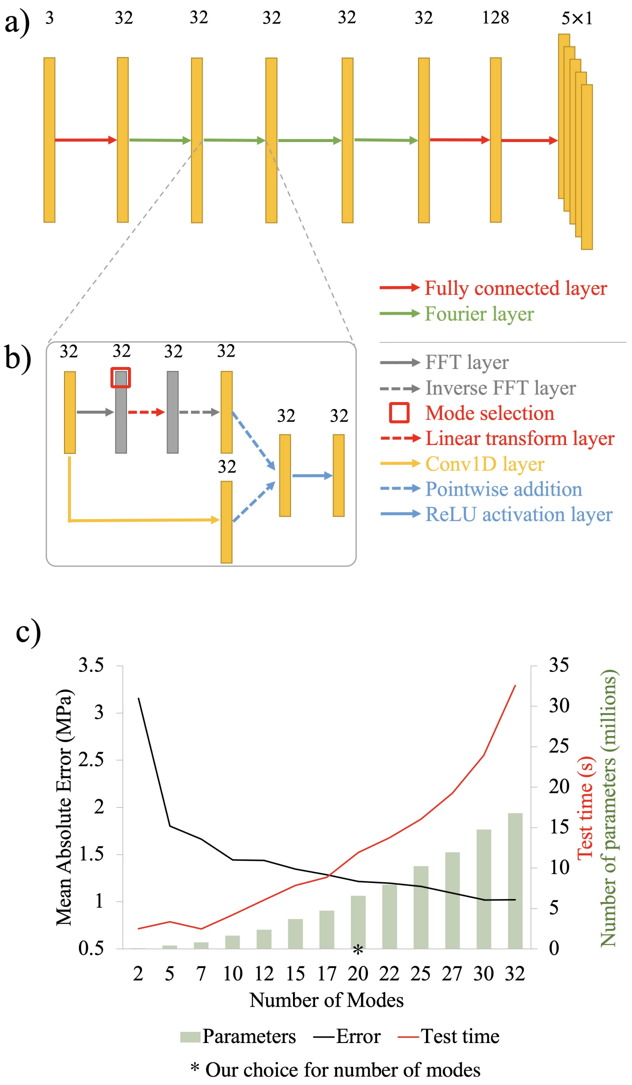

Figure A6: **a)** Network architecture of FNO where the branching of network for multiple output fields happens from the output layer. **b)** Internal working of FNO layer —here the spatial dimensions of activation maps remain the same after every layer; linear transform is done for a selected number of Fourier modes and coefficients of remaining modes are set to zero before inverse Fourier transform. **c)** Test error, test time, and number of trainable parameters for single-output FNO with varying number of modes (for $\mathbf{P}_{11}$ at 64×64 resolution)