2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)
June 28-30, 2023. Seattle, Washington, USA

Transfer Learning Enhanced Full Waveform Inversion*

Stefan Kollmannsberger[1], Divya Singh[1] and Leon Herrmann[1]

Abstract— **We propose a way to favorably employ neural networks in the field of non-destructive testing using Full Waveform Inversion (FWI). The presented methodology discretizes the unknown material distribution in the domain with a neural network within an adjoint optimization. To further increase efficiency of the FWI, pretrained neural networks are used to provide a good starting point for the inversion. This reduces the number of iterations in the Full Waveform Inversion for specific, yet generalizable settings.**

I. INTRODUCTION

Full Waveform Inversion (FWI) is a well-known tool in non-destructive testing used to detect the exact location and shapes of flaws in a given artefact. To this end, FWI solves the following inverse-problem: Given a signal emitted at the boundary of a domain of interest as well as its measured image at selected locations, find the material distribution within the domain. The solution is found iteratively by computing the corresponding forward problem: given a domain with a material distribution and signal emitted at the boundary, compute the image of the emitted signal at selected locations. A review of FWI including an extension to the detection of voids can be found in [1].

Neural networks [2] are a promising tool for solving inverse problems [3]. Two prominent methodologies exist. Firstly, the supervised learning method, where the mapping from the measured forward quantities to the inverse field is learned by interpolating labeled data. After training, cheap predictions can be made with unseen data. Applications of supervised learning in the context of ultrasonic non-destructive testing can e.g. be found in [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18]. The second methodology are physics-informed neural networks (PINNs) [19], [20], [21], where one network learns the forward field (i.e. wave field traveling through the domain) while the second learns the inverse field (i.e. the field of the material coefficients). The networks are trained on the residual of the differential equation and a measurement residual. The state of the art concerning the application of classical PINNs and FWI is reported in [22].

The supervised learning approach lacks both robustness and generalizability, while PINNs are not competitive compared to classical inversion frameworks, such as adjoint optimization [23]. However, reintroducing classical forward solvers while using a neural network as an Ansatz for the inverse field is beneficial since the neural network acts as a regularizer, as seen in [24] for the Poisson equation, and in [25] for the diffusion, wave and nonlinear Burger's equation. Compared to the classical FWI, this leads to smoother solutions without losing accuracy at material jumps i.e. flaws being located more accurately. Simultaneously, the convergence behavior and computational effort of the classical adjoint optimization are restored [26].

In this contribution we will demonstrate that using the neural network as discretization of the inverse field, as presented in [26], offers the additional advantage of introducing transfer learning to speed up convergence. This combination can outperform classical inversion frameworks in terms of computational effort by reducing the number of iterations while maintaining the same quality of reconstruction. The basic idea of this combination is to pretrain the neural network on a labeled data set in an offline phase. It is then applied in an online phase to unseen inversion scenarios where it predicts a rough estimate of the spatial distribution of the field of the material coefficients. This provides a better starting point for the subsequent gradient-based inversion and thereby accelerates convergence. The pretraining is performed in a supervised manner. The influence of the size of the pretraining data set. Although investigations are conducted on the inverse problem of full waveform inversion, the framework is applicable to other types of inverse problems as well.

II. METHOD

A. Full Waveform Inversion using Neural Networks

The general setting of the FWI problem is given in fig. 1 which depicts a domain Ω with a homogeneous material distribution and a flaw in form of a void represented by an ellipse. The task is to find the location and geometry of that void by FWI. To this end, an emitter in red is inducing a signal $u(x_s, t)$ at a specific location x_s at the boundary Γ of a computational domain. The signal is generated by a volume force $f(\boldsymbol{x}, t)$ modeled by a spatial Dirac delta using a source term $\psi(t)$.

$$f(\boldsymbol{x}, t) = \psi(t)\delta(\boldsymbol{x} - \boldsymbol{x}_s) \tag{1}$$

A common choice for the source term is a sine burst with n_c cycles, a frequency $\omega = 2\pi f_\psi$, and an amplitude ψ_0, so

The authors gratefully acknowledge the funding through the joint research project Geothermal-Alliance Bavaria (GAB) by the Bavarian State Ministry of Science and the Arts (StMWK) which finance Leon Herrmann as well as the Georg Nemetschek Institut (GNI) under the project Deep-Monitor which finances Divya Singh. Furthermore, we gratefully acknowledge the funds provided by grant No. WO 671/11-1. Without this support, this conference contribution would not have been possible.

[1] All authors are with the Technical University of Munich, Chair of Computational Modeling and Simulation, 80333 Munich, Bavaria, Germany (corresponding author: +49 89-289-25021; fax: +49 89-289-25051; e-mail: stefan.kollmannsberger@tum.de

978-1-6654-7633-1/23/$31.00 ©2023 IEEE

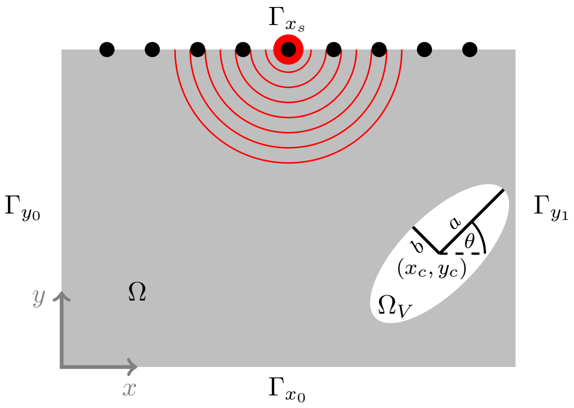


Fig. 1: Ultrasonic Nondestructive Testing on a two-dimensional rectangular domain Ω with a void Ω_V. Horizontal boundaries are defined as $\Gamma_x = \Gamma_{x_0} \cup \Gamma_{x_1}$, while vertical boundaries are defined as $\Gamma_y = \Gamma_{y_0} \cup \Gamma_{y_1}$. The ellipsoidal defect is parametrized by the two half-axes a, b, the center x_c, y_c and the rotation θ.

that

$$\psi(t) = \begin{cases} \psi_0 \sin(\omega t) \sin(\frac{\omega t}{2 n_c}), & \text{for } 0 \leq t \leq \frac{2\pi n_c}{\omega} \\ 0, & \text{for } \frac{2\pi n_c}{\omega} < t. \end{cases} \tag{2}$$

It is now assumed that the signal $u(\boldsymbol{x}, t)$ spreads throughout the medium characterized by the wave equation:

$$\gamma(\boldsymbol{x})\rho_0 u_{tt}(\boldsymbol{x}, t) - \nabla \cdot (\gamma(\boldsymbol{x})\rho_0 c_0^2 \nabla u(\boldsymbol{x}, t)) = f(\boldsymbol{x}, t) \tag{3}$$

defined in space and time $\Omega \times \mathcal{T}$. The medium is assumed to have a constant wave speed c_0 as well as a constant density ρ_0. Equation (3) is further suppield with the typical boundary and initial conditions for the wave field:

$$u_x(\boldsymbol{x}, t) = 0 \qquad \text{on } \Gamma_y \times \mathcal{T} \tag{4}$$

$$u_y(\boldsymbol{x}, t) = 0 \qquad \text{on } \Gamma_x \times \mathcal{T} \tag{5}$$

$$u(\boldsymbol{x}, 0) = u_t(\boldsymbol{x}, 0) = 0 \qquad \text{on } \Omega. \tag{6}$$

The symbols u_t and u_{tt} mark the first and the second time derivative of the signal.

Most importantly, in eq. (3) $\gamma(\boldsymbol{x})$ denotes the unknown which is to be determined in the process of FWI. The function $\gamma(\boldsymbol{x})$ is a scaling function which is constrained between $[\epsilon, 1]$, where $\epsilon \ll 1$ is a non-zero lower bound which indicates a local defect of the material in form of a void.

In the sequel we assume that the unknown scaling function $\gamma(\boldsymbol{x})$ is predicted by a neural network A_γ with trainable parameters $\boldsymbol{\theta}_\gamma$:

$$\hat{\gamma}(\boldsymbol{\theta}; \boldsymbol{x}) = A_\gamma(\boldsymbol{\theta}; \boldsymbol{x}) \tag{7}$$

The mathematical model of the wave equation given by eq. (3) is now used together with eq. (7) to predict a signal $\hat{\boldsymbol{u}}$ with the help of a deterministic operator F:

$$\hat{u}(\hat{\gamma}(\boldsymbol{\theta}; \boldsymbol{x}), t) = F(\hat{\gamma}(\boldsymbol{\theta}; \boldsymbol{x}), t) \tag{8}$$

In this contribution, the deterministic operator F is a finite difference disrectization in space and time. The prediction of the fields is denoted using the hat symbol over a variable. The indicator field over the entire domain Ω is denoted in bold as $\boldsymbol{\gamma}$. Accordingly, the wavefield over the domain spatio-temporal domain $\Omega \times \mathcal{T}$ is denoted as $\boldsymbol{u}$.

We now consider a virtual experiment with the same setting, where a known signal u is emitted at the same locations x_s as in the computational model used for the prediction of $\hat{\gamma}$. The signal is recorded by the sensors illustrated by black circles in fig. 1 and called measurement $u_{\mathcal{M}}(x_i, t)$. Its deviation from the signal predicted by the mathematical model $\hat{u}$ is measured by the following loss functional, also referred to as the cost function:

$$\begin{aligned} &\mathcal{L}_{\mathcal{M}}(\hat{u}(\hat{\boldsymbol{\gamma}}(\boldsymbol{\theta}))) = \\ &\qquad \frac{1}{2} \int_{\mathcal{T}} \int_{\Omega} \sum_{i=1}^{N_{\mathcal{M}}} \Big(\hat{u}\left(\hat{\gamma}(\boldsymbol{\theta}; \boldsymbol{x}), t\right) - u_{\mathcal{M}}(\boldsymbol{x}_i, t) \Big)^2 \\ &\qquad \delta(\boldsymbol{x} - \boldsymbol{x}_i) d\Omega d\mathcal{T}. \end{aligned} \tag{9}$$

The minimization of this cost function leads to the discovery of the scaling function $\hat{\boldsymbol{\gamma}}$ and thereby the spatial location of voids in the material[1]. The minimization procedure is depicted on the left in fig. 2 and briefly explained in the sequel.

The functional given in eq. (9) is minimized iteratively using a gradient based approach yielding an updated indicator field $\boldsymbol{\gamma}^{(j+1)}$ parametrized by the new weights $\boldsymbol{\theta}^{(j+1)}$.

$$\boldsymbol{\theta}^{(j+1)} = \boldsymbol{\theta}^{(j)} - \alpha \nabla_{\theta^{(j)}} \mathcal{L}_{\mathcal{M}}(\hat{u}(\hat{\boldsymbol{\gamma}}(\boldsymbol{\theta}^{(j)}))) \tag{10}$$

Here, α is the step size, also referred to as learning rate, the superindex $(j) = 1, 2, \ldots$ is the iteration counter. This iterative process then yields an optimized estimate of the scaling field $\hat{\boldsymbol{\gamma}}(\boldsymbol{\theta})$ which gives a direct indication of the flaw.

Equation (10) requires the computation of the gradient of the cost function with respect to the network parameters $\boldsymbol{\theta}$ and can efficiently be computed by using the chain rule:

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}(\hat{u}(\hat{\boldsymbol{\gamma}}(\boldsymbol{\theta}))) = \underbrace{\nabla_{\hat{\boldsymbol{\gamma}}} \mathcal{L}(\hat{u}(\hat{\boldsymbol{\gamma}}))}_{\text{adjoint method}} \cdot \underbrace{\nabla_{\boldsymbol{\theta}} \hat{\boldsymbol{\gamma}}(\boldsymbol{\theta})}_{\text{automatic differentiation}} \tag{11}$$

The first term on the right hand side is the outer derivative i.e. the gradient of the cost function w.r.t. the indicator field $\nabla_{\hat{\boldsymbol{\gamma}}} \mathcal{L}$. It is efficiently computed with the continuous adjoint method [1] using the Fréchet kernel K_γ:

$$K_\gamma = \int_{\mathcal{T}} \left[-\rho_0 u_t^\dagger u_t + \rho_0 c_0^2 \nabla u^\dagger \cdot \nabla u \right] dt \tag{12}$$

which is integrated in space resulting in the gradient:

$$\nabla_{\hat{\boldsymbol{\gamma}}} \mathcal{L}(\hat{u}(\hat{\boldsymbol{\gamma}})) = \int_{\Omega} K_\gamma d\Omega \tag{13}$$

See [1] for more details. To compute eq. (12), the temporal and spatial gradients of the solution field u and the adjoint field $u^\dagger$ are computed with finite differences. The second term in eq. (11) is the inner derivative, i.e. the sensitivity

[1] Note that the dependency of $\hat{\boldsymbol{\gamma}}$ on $\boldsymbol{x}$ and $\hat{\boldsymbol{u}}$ on $\boldsymbol{x}, t$ is not always explicitly given to declutter the notation.

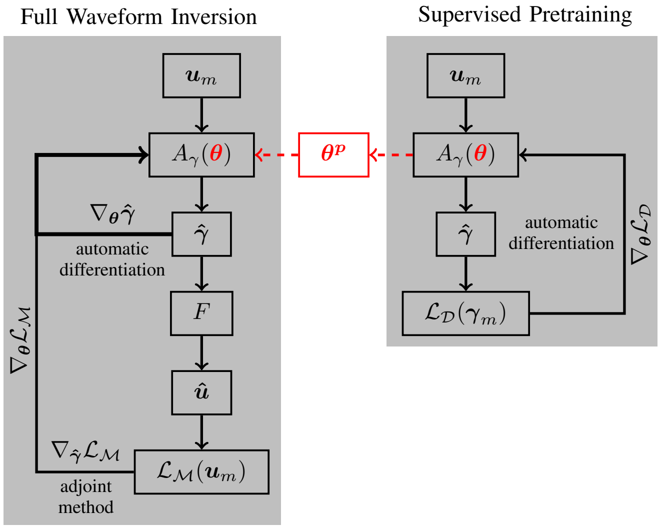

Full Waveform Inversion
Supervised Pretraining
$\boldsymbol{u}_m$
$A_\gamma(\boldsymbol{\theta})$
$\boldsymbol{\theta}^p$
$\nabla_{\boldsymbol{\theta}}\hat{\gamma}$
$\hat{\gamma}$
automatic differentiation
F
$\hat{\boldsymbol{u}}$
$\nabla_{\boldsymbol{\theta}}\mathcal{L}_\mathcal{M}$
$\nabla_{\hat{\gamma}}\mathcal{L}_\mathcal{M}$
$\mathcal{L}_\mathcal{M}(\boldsymbol{u}_m)$
adjoint method
$\nabla_{\boldsymbol{\theta}}\mathcal{L}_\mathcal{D}$
$\mathcal{L}_\mathcal{D}(\gamma_m)$

Fig. 2: Schematic of the FWI training procedure on the left and the possible supervised pretraining on the right to accelerate the convergence with good starting weights $\boldsymbol{\theta}^p$. In the FWI, first the sensitivity of the cost function $\mathcal{L}$ w.r.t. the predicted indicator field $\hat{\gamma}$ is computed with the adjoint method using eqs. (12) and (13). The sensitivity of the indicator field $\hat{\gamma}$ w.r.t. the neural network parameters $\boldsymbol{\theta}$ is then found with automatic differentiation in the form of backpropagation frequently used to train neural networks. This yields the desired gradient $\nabla_{\boldsymbol{\theta}}\mathcal{L}$ by using the chain rule eq. (11). The supervised pretraining is conducted in a conventional manner with labeled data as e.g. elaborated in [2].

of the scaling function w.r.t. the neural network parameters $\nabla_{\boldsymbol{\theta}}\hat{\gamma}$. This derivative is obtained via automatic differentiation in the form of backpropagation that is frequently used to train neural networks [2], in this case $A_\gamma(\boldsymbol{\theta})$.

The presented framework for the detection of flaws by using a scaling function discretized by neural networks as briefly recalled above was already explained in a more elaborate form in [26]. Therein, it was shown to deliver more accurate results than classical methods. This can be attributed to the favourable properties of A_γ. Although the neural network as discretization adds complexity to the FWI optimization in terms of the required automatic differentiation, the additional cost is negligible next to the evaluation of the forward operator F for a large number of timesteps. Thus similar computational times and number of iterations are observed as in classical FWI. Yet, the presented framework has many advantages, one of which is the possibility of reducing the number of required iterations and is elaborated next.

B. Transfer Learning in Full Waveform Inversion

As in any gradient based approach, the choice of the starting point $\hat{\gamma}^{(0)}$ decisively determines the number of necessary iterations. The closer this starting point is located to the optimal $\hat{\gamma}$, the less iterations are needed. This observation offers the chance to exploit the concept of transfer learning [27] for a better starting point $\hat{\gamma}^{(0)}$ obtained by better starting weights $\boldsymbol{\theta}^{(0)}$. The neural network is pretrained in a supervised manner [18], [17] with data pairs $(\boldsymbol{u}_\mathcal{M}, \boldsymbol{\gamma}_\mathcal{M})$ using a data-driven cost function defined in terms of the mean squared error

$$\mathcal{L}_\mathcal{D}(\hat{\gamma}(\boldsymbol{\theta})) = \frac{1}{2}\sum_{i=1}^{N_\mathcal{D}} \int_\Omega \left(\hat{\gamma}_i(\boldsymbol{\theta};\boldsymbol{x}) - \gamma_{\mathcal{M}_i}(\boldsymbol{x})\right)^2 d\Omega \tag{14}$$

The supervised pretraining depicted on the right in fig. 2 yields a good starting weight $\boldsymbol{\theta}$ to be saved as $\boldsymbol{\theta}^p$ and transferred via A_γ to the FWI framework, where it provides a good approximation to $\hat{\gamma}^{(0)}$ simply by evaluating the pretrained network $A_\gamma(\boldsymbol{\theta}^p)$:

$$\hat{\gamma}^{(0)} = A_\gamma(\boldsymbol{\theta}^p) \tag{15}$$

The starting point $\hat{\gamma}^{(0)}$ can then directly be used as the starting point for the iteration as depicted on the right in fig. 2. This is possible due to the identical network architecture used in the supervised training and FWI in the approximation of the field $\hat{\gamma}(\boldsymbol{\theta};\boldsymbol{x})$ as given in eq. (7). This enables the direct use of the network parameters $\boldsymbol{\theta}^p$ as a starting point for the inversion, see eq. (10) and fig. 2.

C. Network Description

Section III investigates the improvement shown by a pretrained Neural network A_γ, for carrying out the FWI. For this purpose we use a network comprised of an encoder and a decoder which was first proposed in [28], [29] for language translation. The architecture of the neural network is shown in fig. 3. The input to this encoder-decoder network is chosen as the last quarter of the signal since it contains the most information about the damage present in the domain due to multiple reflections. This input data is first reshaped from a 1D vector of 512 data points to a 2D matrix of 32×16 before being fed to the neural network.

The first half of the neural network consists of the encoder layers that are made up of a combination of convolution and average pooling. The activation function used for these layers is the PReLU function [30]. Followed by this, the decoder network scales up the compressed input received from the encoder with the help of a combination of convolutional and upsampling' layers (with nearest-neighbour interpolation to retain the discontinuities). The same activation function was used as before except for the last layer where an adaptive sigmoid function [31] is used to ensure that the output is between 0 and 1. This was done in order to match the values of the field $\hat{\gamma}(\boldsymbol{\theta};\boldsymbol{x})$ which are 1 at undamaged locations and 0 at the damaged location. The optimization is performed with the Adam [32] optimizer. The initial weights are set with Glorot initialization [33].

The dataset $(\boldsymbol{u}_\mathcal{M}, \boldsymbol{\gamma}_\mathcal{M})$ is generated with elliptical defects parametrized by the center x_c, y_c, the semi-major and semi-minor axes a, b, and the rotation θ, as illustrated in fig. 1. The wave measurements $\boldsymbol{u}_\mathcal{M}$ are then obtained by a forward

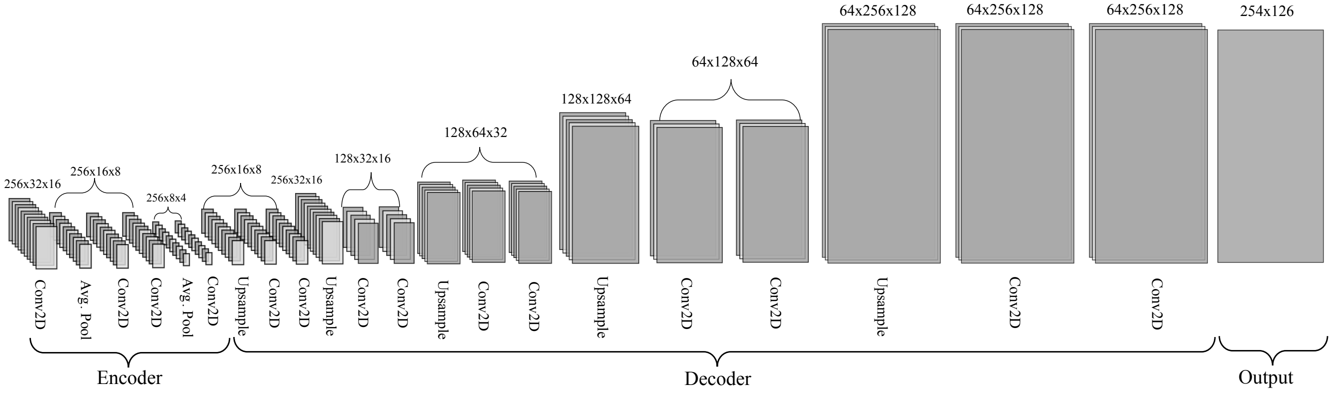


Fig. 3: The encoder-decoder neural network architecture A_γ discretizing the material $\hat{\gamma}$ given a signal $\boldsymbol{u}_\mathcal{M}$.

solution with finite differences[2]. In total 2100 samples are obtained and used throughout the upcoming studies. Up to 2048 samples are used for pretraining, while the last 62 are always unseen before application of the FWI. These are thus reserved as validation.

III. Results

The potential benefit of transfer learning is established with a basic investigation using the ellipsoidal dataset. The convergence behavior is compared in terms of the learning histories throughout the FWI. Both the cost function $\mathcal{L}_\mathcal{D}$ and average precision [35] w.r.t. the true material distribution $\gamma_\mathcal{M}$ are considered. The average precision P is a preferred metric in object detection and segmentation, which is similar to the task of void detection. See the APPENDIX for a detailed description. Pretrained models with different levels of pretraining are compared to a baserate obtained without pretraining. The models differ in terms of amount of data used during the supervised pretraining, which is $N_\mathcal{D} = 2, 8, 32, 128$. As usual in transfer learning, the learning rate has to be decreased in the subsequent training, i.e. the FWI. Interestingly, a larger $N_\mathcal{D}$ leads to a smaller optimal learning rate, obtained by hyperparameter tuning. Additionally some layers of the pretrained network are typically frozen, which in this work was determined empirically to the first 15 layers. The comparison of the learning histories is illustrated in fig. 4, where the mean over 15 different inversions is presented.

A suitable recovery of the void with limited artefacts is typically reached when the average precision is larger than $P > 1 - 10^{-4}$, indicated by the red dotted line in 4b. The pretrained networks reach this threshold at around 20 epochs, while no pretraining leads to a recovery after 30 epochs. Thus a speed-up of about 1.5 is achieved. Interestingly, a larger dataset does not seem to yield an additional benefit. This might in later work be alleviated by improved hyperparameters or network architectures.

A representative prediction history is illustrated in fig. 5 showing the learned state at every 4 epochs starting from epoch 0 until epoch 20. The pretrained model is trained on $N_\mathcal{D} = 32$ samples. The red boxes indicate the epochs at which the desired average precision of $P > 1 - 10^{-4}$ is achieved. This shows, that the pretrained model identifies the void about 4 epochs earlier. In later epochs the average precision, depicted in white in the right upper corners, remains significantly lower in the pretrained case. This highlights the additional benefit of using transfer learning when using a neural networks as an Ansatz of the inverse field.

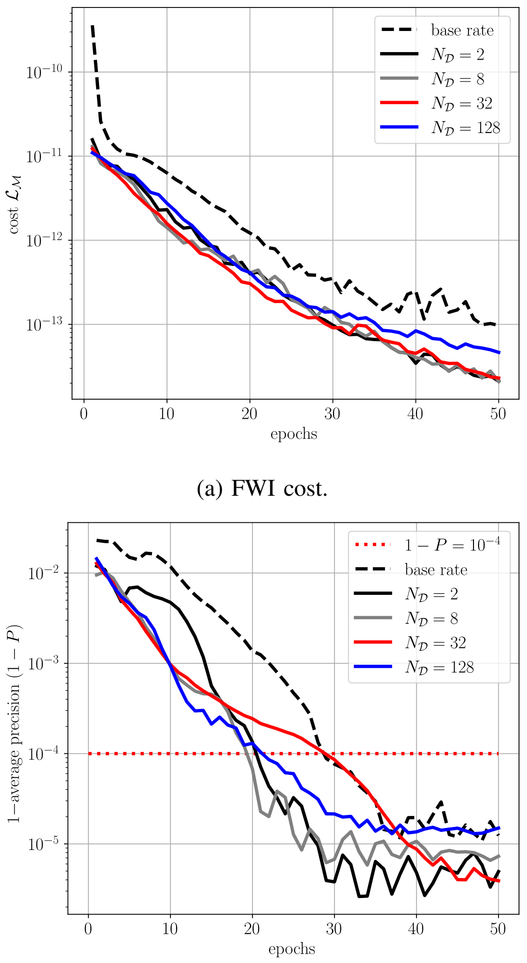
IV. Outlook

Although, the obtained speed-up is relatively modest, the results support the proof-of-concept of using transfer learning within a neural network Ansatz in FWI. The lack of improvement of dataset sizes larger than two indicate an unoptimal loss landscape when the problem is transferred from the supervised pretraining to the FWI. This is caused by the rapid change from the data-driven loss function in eq. (14) to the FWI loss functional from eq. (9). This problem might be alleviated by improved network architectures or hyperparameters, where the level of pretraining, the amount of regularization, e.g. L^2, the number of frozen layers, and the learning rate after pretraining are the key-parameters to be tuned. An alternative is to avoid the rapid shift in the optimization landscape by pretraining the network not with labeled data, but with multiple FWI cycles with the FWI loss funcitonal (eq. (9)). As the understanding of the change in loss landscape is essential in the design of an FWI enhanced by transfer learning, studying its loss landscape with visualization tools such as [36] is the key tool.

V. Conclusions

The aim of this work was to establish a proof-of-concept of using transfer learning within the optimization framework of adjoint-based FWI optimization. This was enabled by the neural network Ansatz of the inverse field, previously established as beneficial regularizers [26], [25], [24]. A supervised pretraining was employed before the application to the FWI. Although modest speed-ups of only 1.5 were achieved, the

[2] Note, that this is an inverse crime [34], as the reference data is generated with the same numerical method used in the FWI. For the sake of simplicity and to illustrate the approach, this is however tolerated and not deemed to be an obstacle to the proposed method.



(a) FWI cost.

(b) Average precision.

Fig. 4: Influence of using transfer-learning on convergence.

method shows potential. With further investigations into the optimization landscape of FWI with neural networks for the inverse field, significant improvements might be possible w.r.t. the convergence properties.

APPENDIX

To apply the average precision metric [35] to the given problem, it has to be considered a classification problem, in this case a binary classification problem, i.e. either a void $\gamma(\boldsymbol{x}) = 0$ or undamaged material $\gamma(\boldsymbol{x}) = 1$. This enables the use of true positives T_P, false positives F_P, true negatives T_N and false negatives F_N. The average precision takes both the precision p and the recall r into account. The precision p is the number of true positives over all positives, i.e. true and false positives

$$p = \frac{T_P}{T_P + F_p}, \tag{16}$$

while the recall r relates the true positives to the true positives and false negatives.

$$r = \frac{T_P}{T_P + F_N} \tag{17}$$

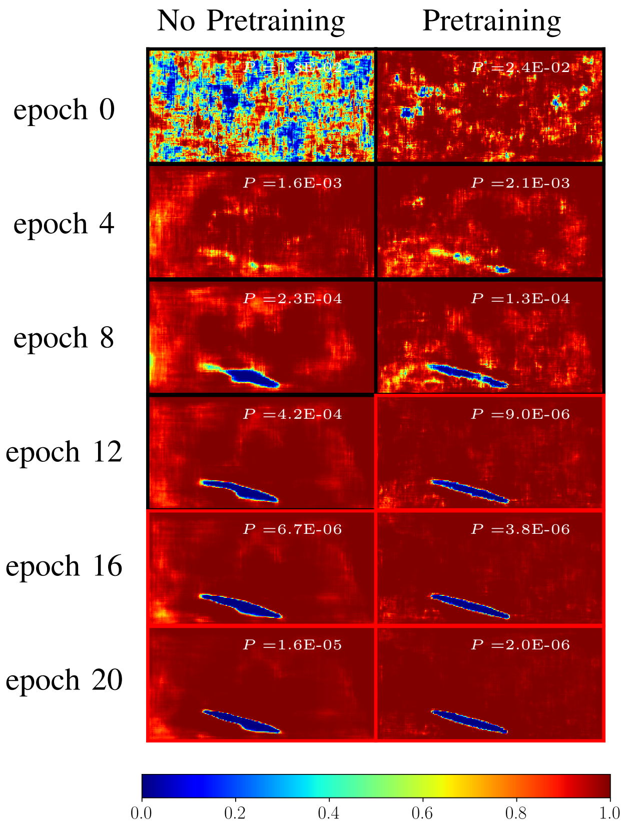


Fig. 5: Representative prediction histories starting at epoch 0 until epoch 20. On the left without pretraining and on the right with pretraining using $N_{\mathcal{D}} = 32$ samples. The red boxes highlight the epochs, where the desired threshold of 10^{-4} is breached.

Thus, the precision p represents the level of the false positive rate and recall represents the level of false negative rate. Determining if a value is true (if $\gamma(\boldsymbol{x}) \leq t$) or false (if $\gamma(\boldsymbol{x}) > t$), depends on a threshhold t, which can be varied, such that the precision and recall become functions $p(t)$ and $r(t)$. These can be plotted against each other, yielding the so-called precision-recall curve, which showcases the tradeoff between precision and recall. The average precision is then obtained by integrating the precision-recall curve, which can be approximated by the following summation of discrete threshold intervals $[t-1, t]$.

$$P = \sum_t \frac{R_t - R_{t-1}}{P_t} \tag{18}$$

The average precision provides a more accurate estimate of the quality of the prediction compared to the mean squared error, as it is alleviates imbalances in the field, i.e. only small voids in a large intact domain. For implementational details, c.f. to the scikit-learn implementation and documentation [37] also used in this work.

ACKNOWLEDGMENT

This article is dedicated to Bill and Renate Steen. A special thanks goes to Prof. Barbara Wohlmuth for her support and

interest. May many fruitful joint contributions emerge.

References

[1] T. Bürchner, P. Kopp, S. Kollmannsberger, and E. Rank, "Immersed boundary parametrizations for full waveform inversion," *Computer Methods in Applied Mechanics and Engineering*, vol. 406, p. 115893, Mar. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782523000166

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[3] S. Kollmannsberger, D. D'Angella, M. Jokeit, and L. Herrmann, *Deep Learning in Computational Mechanics: An Introductory Course*, ser. Studies in Computational Intelligence. Cham: Springer International Publishing, 2021, vol. 977. [Online]. Available: https://link.springer.com/10.1007/978-3-030-76587-3

[4] G. Fabien-Ouellet and R. Sarkar, "Seismic velocity estimation: A deep recurrent neural-network approach," *GEOPHYSICS*, vol. 85, no. 1, pp. U21–U29, Jan. 2020. [Online]. Available: https://library.seg.org/doi/10.1190/geo2018-0786.1

[5] F. Yang and J. Ma, "Deep-learning inversion: A next-generation seismic velocity model building method," *GEOPHYSICS*, vol. 84, no. 4, pp. R583–R599, July 2019. [Online]. Available: https://library.seg.org/doi/10.1190/geo2018-0249.1

[6] W. Wang and J. Ma, "Velocity model building in a crosswell acquisition geometry with image-trained artificial neural networks," *GEOPHYSICS*, vol. 85, no. 2, pp. U31–U46, Mar. 2020. [Online]. Available: https://library.seg.org/doi/10.1190/geo2018-0591.1

[7] W. Wang, F. Yang, and J. Ma, "Velocity model building with a modified fully convolutional network," in *SEG Technical Program Expanded Abstracts 2018*. Anaheim, California: Society of Exploration Geophysicists, Aug. 2018, pp. 2086–2090. [Online]. Available: https://library.seg.org/doi/10.1190/segam2018-2997566.1

[8] S. Li, B. Liu, Y. Ren, Y. Chen, S. Yang, Y. Wang, and P. Jiang, "Deep-Learning Inversion of Seismic Data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 3, pp. 2135–2149, Mar. 2020.

[9] Y. Zheng, Q. Zhang, A. Yusifov, and Y. Shi, "Applications of supervised deep learning for seismic interpretation and inversion," *The Leading Edge*, vol. 38, no. 7, pp. 526–533, July 2019. [Online]. Available: https://library.seg.org/doi/10.1190/tle38070526.1

[10] M. Araya-Polo, S. Farris, and M. Florez, "Deep learning-driven velocity model building workflow," *The Leading Edge*, vol. 38, no. 11, pp. 872a1–872a9, Nov. 2019. [Online]. Available: https://library.seg.org/doi/10.1190/tle38110872a1.1

[11] B. Mao, L.-G. Han, Q. Feng, and Y.-C. Yin, "Subsurface velocity inversion from deep learning-based data assimilation," *Journal of Applied Geophysics*, vol. 167, pp. 172–179, Aug. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0926985118308905

[12] B. Wu, D. Meng, L. Wang, N. Liu, and Y. Wang, "Seismic Impedance Inversion Using Fully Convolutional Residual Network and Transfer Learning," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 12, pp. 2140–2144, Dec. 2020.

[13] V. Das, A. Pollack, U. Wollner, and T. Mukerji, "Convolutional neural network for seismic impedance inversion," *GEOPHYSICS*, vol. 84, no. 6, pp. R869–R880, Nov. 2019. [Online]. Available: https://library.seg.org/doi/10.1190/geo2018-0838.1

[14] M. J. Park and M. D. Sacchi, "Automatic velocity analysis using convolutional neural network and transfer learning," *GEOPHYSICS*, vol. 85, no. 1, pp. V33–V43, Jan. 2020. [Online]. Available: https://library.seg.org/doi/10.1190/geo2018-0870.1

[15] M. Araya-Polo, J. Jennings, A. Adler, and T. Dahlke, "Deep-learning tomography," *The Leading Edge*, vol. 37, no. 1, pp. 58–66, Jan. 2018. [Online]. Available: https://library.seg.org/doi/10.1190/tle37010058.1

[16] Y. Kim and N. Nakata, "Geophysical inversion versus machine learning in inverse problems," *The Leading Edge*, vol. 37, no. 12, pp. 894–901, Dec. 2018. [Online]. Available: https://library.seg.org/doi/10.1190/tle37120894.1

[17] J. Rao, F. Yang, H. Mo, S. Kollmannsberger, and E. Rank, "Quantitative reconstruction of defects in multi-layered bonded composites using fully convolutional network-based ultrasonic inversion," *Journal of Sound and Vibration*, vol. 542, p. 117418, 2023.

[18] J. Ye and N. Toyama, "Automatic defect detection for ultrasonic wave propagation imaging method using spatio-temporal convolution neural networks," *Structural Health Monitoring*, vol. 21, no. 6, pp. 2750–2767, Nov. 2022.

[19] D. C. Psichogios and L. H. Ungar, "A hybrid neural network-first principles approach to process modeling," *AIChE Journal*, vol. 38, no. 10, pp. 1499–1511, Oct. 1992. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/aic.690381003

[20] I. Lagaris, A. Likas, and D. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 987–1000, Sept. 1998.

[21] 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, pp. 686–707, Feb. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999118307125

[22] M. Rasht-Behesht, C. Huber, K. Shukla, and G. E. Karniadakis, "Physics-informed neural networks (pinns) for wave propagation and full waveform inversions," *Journal of Geophysical Research: Solid Earth*, vol. 127, no. 5, p. e2021JB023120, 2022, e2021JB023120 2021JB023120. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2021JB023120

[23] A. Fichtner and F. Bleibinhaus, *Full seismic waveform modelling and inversion*, ser. Advances in geophysical and environmental mechanics and mathematics. Berlin Heidelberg: Springer, 2011.

[24] J. Berg and K. Nyström, "Neural networks as smooth priors for inverse problems for PDEs," *Journal of Computational Mathematics and Data Science*, vol. 1, p. 100008, Sept. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2772415821000043

[25] K. Xu and E. Darve, "The Neural Network Approach to Inverse Problems in Differential Equations," Jan. 2019, arXiv:1901.07758 [math]. [Online]. Available: http://arxiv.org/abs/1901.07758

[26] L. Herrmann, T. Bürchner, F. Dietrich, and S. Kollmannsberger, "On the use of neural networks for full waveform inversion," *ARXIV*, Mar. 2023. [Online]. Available: https://arxiv.org/abs/2303.03260

[27] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[28] N. Kalchbrenner and P. Blunsom, "Recurrent continuous translation models," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA: Association for Computational Linguistics, Oct. 2013, pp. 1700–1709. [Online]. Available: https://aclanthology.org/D13-1176

[29] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," Feb. 2015, arXiv:1502.01852 [cs] version: 1. [Online]. Available: http://arxiv.org/abs/1502.01852

[31] A. D. Jagtap, K. Kawaguchi, and G. E. Karniadakis, "Adaptive activation functions accelerate convergence in deep and physics-informed neural networks," *Journal of Computational Physics*, vol. 404, p. 109136, Mar. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0021999119308411

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[33] S. K. Kumar, "On weight initialization in deep neural networks," *arXiv preprint arXiv:1704.08863*, 2017.

[34] A. Wirgin, "The inverse crime," Jan. 2004, arXiv:math-ph/0401050. [Online]. Available: http://arxiv.org/abs/math-ph/0401050

[35] E. Zhang and Y. Zhang, "Average Precision," in *Encyclopedia of Database Systems*, L. LIU and M. T. ÖZSU, Eds. Boston, MA: Springer US, 2009, pp. 192–193. [Online]. Available: https://doi.org/10.1007/978-0-387-39940-9_482

[36] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the Loss Landscape of Neural Nets," Nov. 2018, arXiv:1712.09913 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1712.09913

[37] 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, pp. 2825–2830, 2011.