Wavefield reconstruction inversion via machine learned functions

Chao Song and Tariq Alkhalifah, King Abdullah University of Science and Technology.

SUMMARY

Wavefield reconstruction inversion (WRI) is a PDE-constrained optimization problem that aims to mitigate cycle skipping in full-waveform inversion (FWI) among other potential features. WRI is often implemented in the frequency domain, and thus, requires expensive matrix inversions to reconstruct the wavefield. A recently introduced machine learning (ML) framework, called physics-informed neural networks (PINNs), is used to predict PDE solutions by setting the physical laws as loss functions. These NNs have shown their effectiveness in solving the Helmholtz equation specifically for the scattered wavefield. By including the recorded data at the sensors' locations as a data constraint, the NNs can predict the wavefields which simultaneously fit the recorded data and satisfy the Helmholtz equation for a given initial velocity model. Using the predicted wavefields, we build another independent NN to predict the velocity that fits the wavefield. In this new NN, we use spatial coordinates as input to the network, and use the scattered Helmholtz wave to define the loss function. After we train this deep neural network, we are able to predict the velocity in the domain of interest. We demonstrate the potential of the proposed method using a square anomaly model and a simple layered model, and the initial results considering single frequency data show that the ML-based WRI is able to invert for reasonable velocity models.

INTRODUCTION

Full-waveform inversion (FWI) is popularly used to retrieve high-resolution physical properties of the subsurface. However, direct applications of FWI usually suffer from cycle skipping due to sinusoidal nature of the wavefield and the complex scattering embedded in it. Wavefield reconstruction inversion (WRI) uses a partial differential equation (PDE), which is the wave equation, as a regularization term to partially mitigate the nonlinearity of this seismic inverse problem (van Leeuwen and Herrmann, 2015). The key objective of WRI and wavefield-reconstruction based methods is to reconstruct a frequency-domain wavefield that fits the data and the wave equation (Leeuwen and Herrmann, 2013; Alkhalifah and Song, 2019; Song and Alkhalifah, 2020a). Since, the reconstruction process involves inverting an augmented wave equation, the cost and complexity increases for large models, like 3D, or complex physics, like anisotropy.

With the rapid developments in computer capability and the rampant availability of data, machine learning (ML) is gaining a lot of attention in many fields. ML has shown large potential in geoscience applications, like fault detection, low-frequency extrapolation, image quality and FWI improvements (Wu et al., 2019; Ovcharenko et al., 2019; Song and Alkhal-

ifah, 2020b; Li et al., 2020; Sun and Alkhalifah, 2020a,b). Most of the previous work used deep neural networks (DNN) to build connections between input and output data. Instead of a simple data-fitting objective, Raissi et al. (2019) recently proposed physics-informed neural networks (PINNs) to solve partial differential equations (PDEs). In PINNs, they use physical laws in the loss function to reconstruct NN based functional solutions to the PDE. With the help of automatic differentiation (Baydin et al., 2017), PINNs can easily produce the partial derivatives of these NN functions with respect to the spatial and temporal coordinates. These NN based functional solutions have already shown their potential in solving the isotropic P-wave eikonal equation (Waheed et al., 2020) and the Helmholtz equation (Alkhalifah et al., 2020). In both applications, the spatial coordinate values are used as input data, and the velocity is considered as a complementary parameter in the loss function. After these networks are trained, we can evaluate the traveltime or the wavefield at any point in the domain of interest.

In solving the Helmholtz equation using PINNs, and to avoid the point source singularity, Alkhalifah et al. (2020) proposed to solve for the scattered wavefield instead of the full wavefield. They specifically used a simple homogeneous model as background, which allowed for analytical solutions of the background wavefield with very little additional cost. Since the NN based functions are learning to satisfy the wave equation, we can easily teach it to satisfy the data at the boundary as well, yielding a WRI like implementation. After we reconstruct the wavefield reasonably well, we build another independent NN based function to represent the velocity. In this new network, we still use the spatial coordinate values as input data, but the output, this time, is the velocity that satisfies the wave equation for the NN predicted wavefield at the input points.

In this abstract, we first review the concept of PINNs. Then we explain how we build two independent PINNs to achieve ML-based WRI by estimating the scattered wavefield and inverting for the velocity, sequentially. Finally, we show the effectiveness of the proposed method on a square anomaly and layered models.

THEORY

Neural network functions

Conventional pure data-mapping based machine learning methods, like supervised learning, tend to perform poorly when the available data for training is limited. To achieve the goal of training a neural network under this condition, any prior knowledge of the data constraint is useful to help in the training convergence. In many cases, the problems we aim at solving follow specific principled physical laws, which can be used as a loss function or an additional constraint. By using automatic differentiation (Baydin et al., 2017), any nonlinear

Wavefield reconstruction inversion via ML

partial differential equation can be easily calculated in the loss function of the network, and this kind of network is referred to as physics-informed neural networks (Raissi et al., 2019).

To simulate acoustic wave propagation, we use an acoustic wave equation with a constant density expressed in the frequency domain:

$$\omega^2 m u(\mathbf{x}, \omega) + \nabla^2 u(\mathbf{x}, \omega) = s(\mathbf{x}_s, \omega), \quad (1)$$

where ω is the angular frequency, m denotes the squared slowness, $u(\mathbf{x}, \omega)$ is the pressure wavefield, and $s(\mathbf{x}, \omega)$ is the source function. $\mathbf{x}_s$ represents the spatial source coordinates. This frequency domain wave equation can be written in a compact form as $L(m)u = s$ with $L(m) = \omega^2 m + \nabla^2$ representing the modeling operator. We define $f(\mathbf{x}) = L(m)u + s$ to represent the physics-constrained loss function in a neural network to approximate the wavefield $u(\mathbf{x}, \omega)$. The mean squared error loss of the physics constraint is given by:

$$MSE_f = \frac{1}{N_f} \sum_{i=1}^{N_f} |f(\mathbf{x}_f^i)|_2^2, \quad (2)$$

where $\{\mathbf{x}_f^i\}_i^{N_f}$ specify the selected input points to the network so the outputs fit the physics constraint, and N_f is the number of selected points. If boundary conditions are required to specify a unique solutions for $u(\mathbf{x})$, we need to define the boundary-constrained mean squared error, which is given by:

$$MSE_u = \frac{1}{N_u} \sum_{i=1}^{N_u} |u(\mathbf{x}_u^i) - u^i|_2^2, \quad (3)$$

where $\{\mathbf{x}_u^i, u^i\}_i^{N_u}$ denote the boundary training data on $u(\mathbf{x})$, and N_u is the number of training data. The total mean squared error loss is defined as: $MSE = MSE_f + MSE_u$.

The augmented wave equation

To mitigate the point source singularity, the scattered wavefield, $\delta u = u - u_0$, is used as an alternative parameter to obtain the wavefield solution (Alkhalifah et al., 2020). Theoretically, the scattered wavefield is calculated using:

$$\omega^2 m \delta u + \nabla^2 \delta u = -\omega^2 \delta m u_0, \quad (4)$$

where u_0 represents the background wavefield. The squared slowness perturbation is defined as $\delta m = m - m_0$, and m_0 represents the background squared slowness of a homogeneous model. If the true squared slowness is provided, we can predict the scattered wavefield δu by just using the physics-constrained MSE_f , which is defined as:

$$MSE_f = \frac{1}{N_f} \sum_{i=1}^{N_f} \left| \omega^2 m_f^i \delta u_f^i + \nabla^2 \delta u_f^i + \omega^2 \delta m_f^i u_{0f}^i \right|_2^2. \quad (5)$$

However, we often do not know the true velocity. So when only an initial velocity is available (squared slowness m_1), we can solve the augmented wave equation in WRI to reconstruct

the wavefield (Leeuwen and Herrmann, 2013). The WRI objective function is given by:

$$E(m, u_{is}) = \min \sum_{is} \frac{1}{2} |d_{is} - u_{is}(\mathbf{x} = \mathbf{x}_r)|_2^2 + \frac{\alpha}{2} |L(m)u_{is} - s_{is}|_2^2. \quad (6)$$

where is and N_s represents the source index and source number, respectively. d denotes the recorded data, and $\mathbf{x}_r$ are the receiver locations, so $u(\mathbf{x} = \mathbf{x}_r)$ denotes the predicted data. In this formulation, the wave equation is added as an additional regularization term, scaled by α . The α used for both examples in this abstract is 0.00001, which is chosen through trial and error tests. The WRI objective function in equation 7 fits the framework of PINNs well by using the data fitting term as the boundary-constrained mean squared error. The only thing that we need to adjust is to use the scattered wave equation, as well as the scattered data δd , which is obtained by subtracting the analytical background data $u_0(\mathbf{x} = \mathbf{x}_r)$ from the recorded data d . Thus, the loss function to reconstruct the scattered wavefield using NNs is given by:

$$MSE = \frac{1}{N_r} \sum_{i=1}^{N_r} |\delta d^i - \delta u(\mathbf{x}_r^i)|_2^2 + \quad (7)$$

$$\frac{\alpha}{N} \sum_{i=1}^N \left| \omega^2 m_1^i \delta u^i + \nabla^2 \delta u^i + \omega^2 (m_1^i - m_0^i) u_0^i \right|_2^2,$$

where N_r is the number of receivers, and N is the training points for the physics constraint. In this network, the input parameters for training are the spatial coordinate values $\mathbf{x} = \{x, z\}$. The output parameters are the real and imaginary parts of the scattered wavefield $\delta u = \{\delta u_r, \delta u_i\}$. All the other parameters used in equation 7 are available prior to evaluate the loss function. The network architecture used to evaluate the scattered wavefield function is shown in Figure 1. In this network, the activation function is an inverse tangent function. We choose to optimize the loss functions using L-BFGS, a quasi-Newton approach, with a full-batch gradient-based optimization algorithm (Liu and Nocedal, 1989). It contains 8 fully connected hidden layers and 20 neurons in each layer.

Figure 1: The NN architecture used to represent the scattered wavefield solution.

The velocity inversion

After obtaining the ML-reconstructed scattered wavefield for a given initial velocity, we build another independent NN to

Wavefield reconstruction inversion via ML

estimate the velocity model. The input parameters for training are spatial coordinate values. In this case, the output parameter we pursue is the squared slowness. As the scattered wavefield becomes a known parameter, we can use the scattered wave equation as the loss function. We add a total-variation regularization term to the loss function to preserve the sharp edges of the estimated velocity model (enhance resolution). Finally, the loss function to train the network used to learn the velocity is given by:

$$MSE = \frac{1}{N} \sum_{i=1}^N \left| \omega^2 m^i \delta u^i + \nabla^2 \delta u^i + \omega^2 (m^i - m_0^i) u_0^i \right|_2^2 \quad (8)$$

$$+ \varepsilon \sqrt{\frac{\partial m^i{}^2}{\partial x} + \frac{\partial m^i{}^2}{\partial z}}.$$

The ε we use for both examples in this abstract is 0.1. The input spatial coordinate values and the complementary variables u_0 and δu correspond to all the sources, so N equals to grid points number multiplying the source number. We show a 5-layer fully connected network used to estimate the velocity satisfying all the sources in Figure 2, and each layer has 10 neurons. The activation function and optimizer used in this network is the same as the previous one.

Figure 2: The NN architecture used to represent the velocity solution.

Finally, like WRI, we repeat the scattered wavefield prediction and the velocity prediction until convergence is achieved or until we are satisfied with the velocity model.

NUMERICAL EXAMPLES

A square anomaly example

We first consider a large square perturbation embedded in an otherwise homogeneous background velocity model (1.5 km/s), as shown in Figure 3a. We highly smooth the true velocity model with a Gaussian window to represent the initial velocity model, as shown in Figure 3b. The size of the model is 100×100 with a grid spacing of 25 m in both vertical and horizontal directions. To simulate a cross-well experiment, we place five sources on the left-hand side of the model as reflected by the * sign, and 100 receivers at right-hand side of the model, represented by the $\triangle$ symbol. We use a delta function as the source function, and thus, the real part of the scattered wavefield (wavefield difference between true wavefield solved numerically and background wavefield solved analytically) corresponding to the middle source for 4 Hz is shown in

Figure 4a. After 20000 epochs of training using all the regular grid points at input, the real part of the reconstructed scattered wavefield using the ML-based method is shown in Figure 4b, which has mild differences from the true scattered wavefield using a numerical method, as shown in Figure 4c. Using the ML-reconstructed scattered wavefield shown in Figure 4b, we train the other network (Figure 2) to predict the velocity function for 200 epochs using all the regular grid points. By inputting the regular grid into the trained network, the inversion result after the first iteration using the ML-based method is shown in Figure 5a. We observe that the general shape of the large perturbation is recovered in the inverted velocity. After four iterations, the inverted velocity is shown in Figure 5b, and the perturbation in the center is further recovered. We show one vertical velocity profile at location 1.25 km in Figure 6. Clearly, the more iterations we use, the better the inverted velocity we obtain.

Figure 3: The true square anomaly model (a), and initial model (b).

Figure 4: The real part of the scattered wavefield corresponding to the middle source for 4 Hz using numerical (a), ML-based (b) methods, and their difference (c)

Figure 5: The inverted models using ML-based WRI after one iteration (a), and four iterations (b).

A layered model example

We further test the proposed method on a layered model, which is extracted from the left part of the Marmousi model. We show the true and initial velocity models in Figures 7a and 7b, respectively. The initial velocity is obtained by highly smoothing the true velocity model. The size of the model is 100×100 with a grid interval of 25 m in both directions.

Wavefield reconstruction inversion via ML

Figure 6: One vertical velocity profile from location 1.25 km.

Figure 7: The true (a), and initial (b) velocity models.

We still use a delta function as the source function, and we place five sources evenly distributed on the surface and recorded data at all grid points on the surface. The homogeneous background velocity used to get the background wavefield is 1.7 km/s. We first show the scattered wavefield of 5 Hz corresponding to the middle source in Figure 8a. In this example, we extend the network to 8 layers, and the neuron number in each layer is 64, 64, 32, 32, 16, 16, 8, 8. We select 2000 random points, and feed their spatial coordinate values into the network. After 20000 epochs of training, we use the coordinate values of all the regular grid points as input. The real part of the ML-reconstructed scattered wavefield corresponding to the middle source is shown in Figure 8b. We show the real part of scattered wavefield difference between the numerical solutions and the ML solutions in Figure 8c. We observe small differences in most of the areas, especially in the shallow part, where is closer to the receivers on the surface. The ML-reconstructed wavefields are generally smooth, and lack the detailed information from the small scatterers. These smooth reconstructed wavefields are beneficial to build smooth initial velocity models.

Figure 8: The real part of the scattered wavefield using numerical (a), and ML-based (b) methods, and their difference (c).

Then we train the network in Figure 2 to predict the velocity using the ML-reconstructed scattered wavefields. After 200 epochs of training, we input the regular grid points into the network and get the predicted velocity model. After 8 iterations,

Figure 9: The inverted velocity using ML-based WRI (a), conventional FWI (b), and WRI (c).

the inverted velocity is shown in Figure 9a. We show FWI and WRI results of 5 Hz after 100 iterations in Figures 9b and 9c, respectively. Clearly, ML-based WRI performs better in reconstructing the high velocity layer between depths 1.5 km and 2.0 km and low velocity beneath it than the other methods. In the vertical velocity profile at location 1.25 km shown in Figure 10, it clearly shows that the ML-based method is able to generate a smooth velocity model, and it can be used as a good initial velocity for FWI.

Figure 10: One vertical velocity profile from location 1.25 km.

CONCLUSIONS

We used two independent physics-informed neural networks to solve the augmented wave equation and estimate the velocity, sequentially. In the first network, used to reconstruct the scattered wavefield, we combined the physics- and data-constrained mean squared error (MSE) as the loss function. After predicting the scattered wavefields, we used the scattered wave equation as a physics-constrained MSE to train the network to predict the velocity. Finally, we input the regular grid points into the trained network, and get the predicted velocity. Applications on a square anomaly model and a layered model show that the ML-based WRI can invert for a reasonably smooth velocity.

ACKNOWLEDGMENTS

We thank KAUST for its support and the SWAG group for the collaborative environment. This work utilized the resources of the Supercomputing Laboratory at King Abdullah University of Science and Technology (KAUST) in Thuwal, Saudi Arabia, and we are grateful for that.

REFERENCES

- Alkhalifah, T., and C. Song, 2019, An efficient wavefield inversion: Using a modified source function in the wave equation: *Geophysics*, **84**, R909–R922.
- Alkhalifah, T., C. Song, U. b. Waheed, and Q. Hao, 2020, Wavefield solutions from a machine learned partial differential equation: 82nd EAGE Conference and Exhibition.
- Baydin, A. G., B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, 2017, Automatic differentiation in machine learning: a survey: *The Journal of Machine Learning Research*, **18**, 5595–5637.
- Leeuwen, T. V., and F. Herrmann, 2013, Mitigating local minima in full-waveform inversion by expanding the search space: *Geophysical Journal International*, **195**, 661–667.
- Li, Y., T. Alkhalifah, and Z. Zhang, 2020, High-resolution regularized elastic full waveform inversion assisted by deep learning: 82nd EAGE Conference and Exhibition.
- Liu, D. C., and J. Nocedal, 1989, On the limited memory bfgs method for large scale optimization: *Mathematical programming*, **45**, 503–528.
- Ovcharenko, O., V. Kazei, M. Kalita, D. Peter, and T. Alkhalifah, 2019, Deep learning for low-frequency extrapolation from multioffset seismic data: *Geophysics*, **84**, R989–R1001.
- Raissi, M., P. Perdikaris, and G. E. Karniadakis, 2019, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations: *Journal of Computational Physics*, **378**, 686–707.
- Song, C., and T. Alkhalifah, 2020a, Efficient wavefield inversion with outer iterations and total variation constraint: *IEEE Transactions on Geoscience and Remote Sensing*, 1–11.
- , 2020b, Identifying microseismic events in time-reversed source images using support vector machine: 82nd EAGE Conference and Exhibition.
- Sun, B., and T. Alkhalifah, 2020a, A data-driven choice of misfit function for fwi using reinforcement learning: 82nd EAGE Conference and Exhibition.
- , 2020b, MI-misfit: learn a robust misfit function for full-waveform inversion using machine learning: 82nd EAGE Conference and Exhibition.
- van Leeuwen, T., and F. J. Herrmann, 2015, A penalty method for pde-constrained optimization in inverse problems: *Inverse Problems*, **32**, 015007.
- Waheed, U. b., H. Ehsan, T. Alkhalifah, C. Song, and Q. Hao, 2020, Eikonal solution using physics-informed neural networks: 82nd EAGE Conference and Exhibition.
- Wu, X., L. Liang, Y. Shi, and S. Fomel, 2019, Faultseg3d: Using synthetic data sets to train an end-to-end convolutional neural network for 3d seismic fault segmentation: *Geophysics*, **84**, IM35–IM45.