Wavefield solutions from machine learned functions that approximately satisfy the wave equation

Tariq Alkhalifah[1], Chao Song[1], Umair bin Waheed[2], and Qi Hao[2]

ABSTRACT
Solving the Helmholtz equation provides wavefield solutions that are dimensionally compressed, per frequency, compared to the time domain which is useful for many applications, like full waveform inversion (FWI). However, the efficiency in attaining such wavefield solutions depends often on the size of the model, which tends to be large at high frequencies and for 3D problems. Thus, we use here a recently introduced framework based on neural networks to predict such solutions through setting the underlying physical equation as a loss function to optimize the neural network parameters. For an input point in the model space, the network learns to predict the wavefield value at that point, and its partial derivatives using a concept referred to as automatic differentiation, to fit, in our case, a form of Helmholtz equation. We specifically seek the solution of the scattered wavefield considering a simple homogeneous background model that allows for analytical solutions. Feeding a reasonable number of random points from the model space will ultimately train a fully connected 8-layer deep neural network with each layer having a dimension of 20, to predict the scattered wavefield function. Initial tests on a two-box-shaped scatterer model with a source in the middle, as well as, a layered model with a source on the surface demonstrate the successful training of the NN for this application and provide us with a peek into the potential of such an approach.
Introduction

Solving the wave equation is one of the most (if not the most) fundamental problems we face as we try to illuminate the Earth using recorded seismic data. It constitutes the majority of the computational cost and complexity in applications like seismic imaging and waveform inversion. Solving the wave equation in the time domain dominates seismic applications. However, frequency-domain solutions, providing a reduction in dimensionality, recently gained additional attention with the rise of waveform inversion (Pratt, 1999). Such solutions are obtained by inverting the stiffness matrix of the Helmholtz equation. However, the cost and complexity of such a matrix inversion are intolerable as the model size increases, like for high frequencies or 3D applications (Clément et al., 1990). Thus, we explore the machine learning (ML) world for a more practical solution. Recently, Rizzuti et al. (2019) built on the ML ability to predict the sequence evolution of iterative solvers in representing the Krylov-based iterative solver for the Helmholtz equation. For a large model size, their stiffness matrix based approach can be costly. Within the framework of utilizing deep neural networks as universal function approximators (Liu and Nocedal, 1989) and under the banner of physics-informed neural networks, Raissi et al. (2019) demonstrated the network’s flexibility in learning how to extract desired functional solutions to non-linear partial differential equations, utilizing the concept of automatic differentiation (Baydin et al., 2015). We utilize this approach to learn how to solve the Helmholtz equation (McFall and Mahan, 2009). However, to avoid the need for adaptive training points for the neural network (NN) to handle the expected source singularity bias, we solve for the scattered wavefield in the frequency domain, and thus, utilize the corresponding Born equation as the loss function to train a deep fully connected neural network with inputs given by (randomly chosen) points in space (within the domain of interest) and outputs given by the complex wavefield at these points. We test the performance of the NN on a two box-shaped scatterer model, as well as a simple layered model.

The Helmholtz wave equation

As a result of its linear nature, the wave equation can be easily formulated in many domains, including the very useful frequency domain. The resulting Helmholtz equation in an acoustic, isotopic, constant density medium, described by the velocity, \( v \), is given by:

\[
\left( \nabla^2 + \frac{\omega^2}{v^2} \right) u(x) = f(x).
\]  

(1)

In this case, the solution of such an equation is a complex wavefield, \( u = \{u_r, u_i\} \), defined in the Euclidean space, with \( x = \{x, z\} \) as a function of the angular frequency, \( \omega \). As a result, our time-domain solution is nothing but a superposition of such solutions for fixed frequencies (inverse Fourier transform). The often point source nature of the source function, \( f \), makes the Helmholtz equation dependent, as we will see later, on how well we represent and sample the source region. To avoid this limitation, we instead invert for the scattered wavefield, \( \delta u = u - u_0 \), where \( u_0 \) is the background wavefield satisfying the same wave equation 1 for the background velocity, \( v_0 \). Defining the perturbation, \( \delta m = \frac{1}{v^2} - \frac{1}{v_0^2} \), the scattered wavefield satisfies

\[
\left( \nabla^2 + \frac{\omega^2}{v^2} \right) \delta u = -\omega^2 \delta m u_0.
\]  

(2)

For the scattered wavefield, the source function is no longer confined in space, like in the case of a point source. It, now, depends on the perturbation, which may expand the full space domain.

The neural network solution

Based on the framework introduced by Raissi et al. (2019), we utilize a neural network architecture using fully connected layers to approximate a function. This function is the scattered wavefield solution for equation 2. The input to the network, like a function, is a location in space, given here in 2D by \( x \) and \( z \) values. The output of the network are the real and imaginary values of the complex scattered wavefield at the location. Figure 1 shows such a network and the dimension of each layer is 20. The activation function between layers is an inverse tangent. We use the network to evaluate the wavefield and its second-order partial derivatives in \( x \) and \( y \), which is needed to evaluate the loss function. Thus,
to train the network, with equation 2 as the physics constraint, we use the following loss function:

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

(3)

where $m_0 = \frac{1}{v_0^2}, N$ is the number of training samples, and $i$ is the sample index. For the loss function, we chose the background model to be simple enough (like homogeneous) so that the background wavefield can be evaluated analytically on the fly. In this early stage of tests, we use an 8-layer deep fully connected neural network with 20 neurons per layer. We chose to optimize the loss function using an L-BFGS full-batch gradient-based optimization algorithm (Liu and Nocedal, 1989). We train the network for 20000 epochs. Next, we test the algorithm on two models. The objective of the tests is to evaluate the ability of the neural network to learn the scattered wavefield function free of the need to perform costly matrix inversions.

**Figure 1** The neural network architecture with inputs $x$ and $z$ (a location in space) and outputs given by the real and imaginary parts of the (scattered) wavefield. It comprises of 8 hidden layers with 20 neurons in each layer.

**A two-scatter model**

In the first model, we place two box-shaped scatterers in an otherwise homogeneous background as shown in Figure 2a. The model has 100 samples in both the $x$ and $z$ directions, with a sampling interval of 20 m. The corresponding (real part) of the 5 Hz wavefield for a point source (a delta function, one sample) in the center of the model is shown in Figure 2b. The background model is given by a constant velocity of 2 km/s, and the corresponding wavefield for the same source and frequency is shown in Figure 2c. If we subtract the two wavefields, we obtain the true scattered wavefield shown in Figure 3a, where the energy, as expected, reflects scattering from the two scatterers. Using the loss function in equation 3, we randomly chose 5000 samples from the space domain $(x_i, z_i)$ for the training. This number represents half of the grid samples used to solve the Helmholtz equation and it was necessary to arrive to the scattered wavefield solution shown in Figure 3b. The difference between the true scattered wavefield and the NN predicted one is shown in Figure 3c. There are differences, but they are generally mild and somewhat random. The imaginary part of the scattered wavefield, not shown here, had similar accuracy. To justify inverting for the scattered wavefield instead of the wavefield directly using the Helmholtz equation, we repeated the exact experiment with the same number of randomly chosen training samples. The resulting predicted wavefield solution was inaccurate mainly because our training set was random and did not provide more sampling of source region.

**Figure 2** a) A two-scatter model. b) The real part of a 5 Hz wavefield for the velocity in Figure 2a for a source in the middle. c) The real part of the 5 Hz wavefield for the background velocity model given by a velocity 2 km/s.

**A simple layered model**

Now, we test the utilization of NN PDE in solving for the wavefield for a simple layered model (Figure 4a), extracted from the left side of the Marmousi model and slightly smoothed. A point source is...
Figure 3 a) The scattered wavefield given by the difference between the two wavefields in Figures 2b and 2c (True and background wavefields). b) The NN predicted scattered wavefield on a regular grid. c) The difference between the actual and predicted scattered wavefields.

placed, this time, on the surface at location 1.25 km. The model has, again, 100 samples in both the $x$ and $z$ directions, but now the sampling interval is 25 m. We solve the Helmholtz equation numerically to obtain the 5 Hz frequency wavefield shown in Figure 4b. The background model is homogeneous with a velocity of 1.5 km/s in which we can solve the wavefield analytically, and it is shown in Figure 4c. The difference between the true and the background wavefields is shown in Figure 5a, and it represents the scattered wavefield. The background wavefield and the model perturbations (difference between the true model and the homogeneous background) are used in the cost function given by equation 3 to invert for the NN parameters. We use the same network as previous, but this time we find that we only need 2000 training samples for similar convergence (loss value). After training, we plot the scattered wavefield extracted using the NN on a regular grid and it is shown in Figure 5b. The difference between the true scattered wavefield and the NN predicted one is shown in Figure 5c. The difference is generally small again and seems somewhat random. However, the difference seems to increase as we go deeper, or away from the source. Additional training data might improve the result, but for applications like gradient calculation for velocity model update, a perfect scattered wavefield is not necessary. Note also that the NN predicted wavefield seems smoother, which reflects the compromise nature of the NN solutions, and the role of activation functions, between training points, and for this limited size NN, we clearly avoided over fitting. We will share additional insights on the network and the training in the presentation of the work.

Figure 4 a) A simple layered model. b) The real part of a 5 Hz wavefield for the velocity in Figure 4a for a source on the surface at location 1.25 km/s. c) The real part of the 5 Hz wavefield for the background velocity model given by a velocity 2 km/s.

Discussions

In using the Born version of the wave equation instead of the Helmholtz solver, we avoided the bias needed to sample the source region well in the training of the network to capture its singularity. So by using a homogeneous background model in which the wavefield can be solved analytically (and instantly), the perturbations (difference between true and background models) will often extend the model domain, and no bias in sampling will be required. This feature is observed in the number of random space samples needed for the training in the case of the two-scatterers and the layered models. Since in the case of the two-scatterers model, the perturbation is localized at the scatterers, we needed far more random samples to properly develop the scattered wavefield, and specifically its NN function. Though, at this early stage of testing this solution function-ML, the examples used are simple and the
Figure 5 a) The scattered wavefield given by the difference between the two wavefields in Figures 4b and 4c (True and background wavefields). b) The NN predicted scattered wavefield on a regular grid. c) The difference between the actual and predicted scattered wavefields.

perturbations are small, the objective here is to test the concept, which we hope will be applied to for solve more complex wavefields in 3D, like those for orthorhombic anisotropy, even if the perturbations are small. Conventional frequency domain solutions for such complex physics are hard and somewhat beyond our capability, especially if the model size is large. Thus, for complex physics, our loss function will change, and possibly include more terms, but the mechanism is the same and does not involve solving a large matrix inverse problem. Another feature of using a loss function for NN training, like in Raissi et al. (2019), we can fit the data as part of the objective, and thus, include two terms or more in the loss function. The network, which tries to predict a function, and thus, is based on developing the wavefield by inputting a location in space, allows for easy adaptation to any model shape, including irregular topography. In summary, there is plenty of room for improvements and we will discuss those in the presentation of the work at the meeting, as well as, show more realistic 3D results.

Conclusions

We shared our initial results in utilizing a neural network trained to provide functional solutions to the Helmholtz equation. The fully connected network takes in space coordinates within the domain of interest and outputs the real and imaginary parts of the scattered wavefield. With automatic differentiation, the network is capable, as well, of evaluating the partial derivatives of the wavefield necessary to evaluate the loss function as it is used to update the network parameters. With a scattered wavefield corresponding to perturbations expanding the space domain, the training of the network can be performed with less random samples. The output wavefield is somewhat smoother than the exact one, and this can be attributed to the compromise feature of our relatively small 8-layer, 20 neurons per layer, network and this feature might be useful for applications like waveform inversion.

REFERENCES