DIRECT IMAGING USING PHYSICS INFORMED NEURAL NETWORKS

Tariq Alkhalifah and Xinquan Haung

King Abdullah University of Science and Technology
Physical Sciences and Engineering Division

ABSTRACT

Imaging is a crucial inversion-based task in fields ranging from medical to structure investigations, and Earth discovery. The exploding reflector assumption provides a direct imaging approach for zero-offset (coincident source-receiver) data, like ground penetrating radar (GPR) data. In imaging, however, we face aliasing problems when the data are coarsely sampled. Formulating the corresponding frequency-domain wavefield as a neural network (NN) function of the lateral and depth coordinates, as well as frequency, allows us to use the physics-informed neural network (PINN) framework to obtain images of the subsurface. In this case, we use a modified Helmholtz equation that incorporates the data on the Earth surface (hard constraint) as the loss function to optimize the NN function. This modified Helmholtz formulation allows us to avoid the inherent weaknesses that PINN has in handling boundary conditions, like the data on the surface as an additional loss term. The frequency dimension allows for image reconstruction by directly summing the wavefield over frequencies (the zero-time imaging condition). This zero-offset implementation serves as a proof of concept for later extensions to prestack data using the double square-root equation.

Index Terms—Waveform imaging, Neural networks, partial differential equations

1. INTRODUCTION

Prestack wavefields are representations of the data for virtual sources and receivers that may occupy any location in the domain of interest [2], and they are described by the double square-root (DSR) dispersion formulation [8]. A special case of prestack wavefields is the zero-offset component, when the sources and receivers coincide, and we have a reduction in dimensions. In this case, the DSR equation reduces to a single square-root equation, which was utilized heavily in phase shift migration [10]. The framework for imaging such wavefields is referred to as the exploding reflector. Whether prestack or zero offset, the involved operators provide a platform for direct imaging [19, 8]. However, the conventional imaging process often requires dense recording of the waveform behavior on the exposed surface [26].

Lately, researchers in our field have utilized machine learning algorithms to perform direct imaging and inversion, image enhancement, among other applications in inversion and imaging. Within this context, [4] applied seismic tomography using machine learning. [1] shared a general overview of deep learning applications to accelerate such inversion tasks. Direct mapping of full seismic waveforms to vertical velocity profiles by deep learning was applied on real data [14]. These supervised methods, however, often require large datasets for training. Considering that deep neural networks can act as universal function approximators [17], [20] demonstrated the network’s flexibility in learning how to extract desired functional solutions to nonlinear partial differential equations, and labeled the process as physics-informed neural networks (PINNs). The functional representation have shown considerable flexibility in handling irregular domains and providing compact solutions [3, 24]. So, here, we will utilize PINNs for direct imaging.

So, the objective of this paper is to introduce a framework for direct imaging using neural network (NN) wavefield functions, in which the image is a slice of this wavefield. Specifically, we outline our approach to represent the frequency-domain wavefield with a fully connected neural network function of three inputs (in 2D): the lateral location, depth and frequency. We train the neural network function by using a modified version of the Helmholtz wave equation as a loss function, and the boundary condition (the recorded data) will be imposed on the PDE as a "hard constraint". We will show that stable and flexible images are possible using PINNs, and that the implementation does not require fine sampling of sensors.

2. THE GOVERNING EQUATIONS

Frequency domain representations of the wavefield offer a reduction in dimensionality, and allows us to avoid the need for dense time sampling of the wavefield. However, such wavefields are guided by the Helmholtz equation, where solutions often require the inversion of an impedance matrix that can be of high dimensions, especially in 3D cases or for high frequencies. The neural network way offers a path free of matrix inversions. The wavefield, \( u(x, z, \omega) \), in the frequency domain in a 2D acoustic and isotropic medium, with spatial
Random \( (x, z) \), satisfies the following Helmholtz equation:

\[
\omega^2 u + v^2 u \left( \nabla_x^2 u + \nabla_z^2 u \right) = 0,
\]

(1)

where \( \omega \) is the angular frequency, and \( v \) is the velocity. For exploding reflector imaging applications, we replace the velocity with half of its true value to compensate for the two-way path nature of our zero-offset data [9, 10].

### 2.1. Incorporating the recorded data in the wave equation

Since the data in imaging applications are represented by a Dirichlet boundary condition, we can incorporate it as a “hard” constraint [6, 15, 21] in the wave equation. Considering the recorded frequency-domain zero-offset data \( D(x, \omega) \), we define the wavefield as

\[
u(x, z, \omega) = D(x, \omega) + z \tilde{u}(x, z, \omega),
\]

(2)

in which we force \( u \) to equal \( D \) at zero depth \((z = 0)\). This change of variables will allow us to formulate the neural network to predict \( \tilde{u} \), and require only the governing PDE as a loss function. The resulting PDE, obtained by substituting equation 2 into equation 1, is given by

\[
\hat{F} = z \omega^2 \hat{u} + z v^2 \nabla^2 \hat{u} + 2 v^2 \nabla_z \hat{u} + \omega^2 D + v^2 \nabla_z^2 D = 0,
\]

(3)

where \( \nabla^2 = \nabla_x^2 + \nabla_z^2 \). Note that in equation 3 the data contribution is acting as a source function in solving for \( \hat{u} \).

### 2.2. The reference frequency implementation

For imaging, we will need multiple frequencies to properly develop the reflectivity. Training a network to store the wavefield over many frequencies will require the network to learn the wavefield over multiple scales. [12] suggested an approach to train such a high-dimensional network over multiple frequencies by injecting a reference frequency into the Helmholtz formulation. Such modifications can be easily incorporated into equation 3.

### 3. THE NEURAL NETWORK SOLUTION

In this section, we first describe the general setup for physics-informed framework (PINN) for our problem. We then share our approach to incorporating the data in this setup, and finally describe the extended frequency framework needed for imaging.

#### 3.1. PINNs

In the physics-informed framework (PINN), a neural network (usually a simple fully connected version) is used to represent the functional solution of a partial differential equation (PDE) [20]. So the input to the network are the function variables, and in our 2D case, they are \( x, z, \) and \( \omega \), and the output is the wavefield, \( \hat{u} \); specifically, its real and imaginary parts, as two outputs, representing its complex value nature. In this case, we train an NN function, \( \hat{u}_\theta(x, z, \omega) \), with the our modified Helmholtz formulation as the loss function needed to update the NN parameters, \( \theta \). The training is done over a collocation of samples \( \{x_i, z_i, \omega_i\} \) covering the region of interest, where \( i \) is the index of the samples considering \( N \) samples. The loss function is, specifically, given by equation 3, as follows

\[
L(\theta) = \frac{1}{N} \sum_{j=1}^{N} |F(\hat{u}_\theta(x_i, z_i, \omega_i))|^2.
\]

(4)

Figure 1 summarizes the proposed network components, some of which will be discussed next.

#### 3.2. The data

Incorporating the boundary conditions (like recorded data) into the PDE for neural network based implementations has found a footing under the theory of functional connections [6, 15, 21]. Since the boundary, in this case, is given by Delchielt condition, we can incorporate it into the PDE, as in equation 3. For PINNs, this will reduce the loss terms to one, instead of two in including the boundary as a soft constraint, which allows us to avoid the need to sample the boundary in the training and worry about the appropriate weighting for the boundary term. [16] has also shown that soft boundary constraints admits less than optimal solutions. As suggested in [21], we represent the recorded data on the surface with a
simple one-hidden layer fully connected neural network as a function of location and frequency. This representation admits two features: it allows us to handle irregular acquisition (sensor) geometry as we utilize the power of neural networks to interpolate [18], and it allows us to robustly calculate the derivatives needed by equation 3 using automatic differentiation [5]. As shown in Figure 1, the input to this network is the $x$ and $\omega$ and the output is the real and imaginary parts of the data values at the input points. Thus, we first Fourier transform the recorded data to discrete frequency samples. Then, we train a neural network, $D_\theta(x, \omega)$ using the following data loss function:

$$L_d(\theta) = \frac{1}{M \times K} \sum_{j=1}^{M \times K} |D(s_j, r_j, \omega_j) - D_\theta(s_j, r_j, \omega_j)|^2,$$

where $M$ is the number of recorded traces, and $K$ is the number of frequencies used in the training set. The number of frequencies maybe guided by the Fourier transform sampling or we may use a random subset of the them.

### 3.3. The extended frequency and other settings

We use the extended frequency approach to train the network first for a small range of frequencies at the low end, and then extend the range at the high end using an approach introduced in PINNup [11], which utilizes neuron splitting to use the trained parameters from the smaller frequency range network, as initial values for a wider network to handle the wider frequency. We apply such training on the reference frequency form of the Helmholtz equation 3, with hard constraints. In our implementation, and to stabilize the training, we utilize a positional encoding representation of the input data [13]. The positional encoding replaces the scalar input with a vector of size, in our case here, 9 to represent the scalar value. Such a representation will help the network deal with complex variations in the wavefield between neighboring points. This feature is provided by the fact that small changes in the scalar representation of location, like 0.01 km, in an input position can be projected into a vector that injects a multi dimensional vector change into the network. As [13] suggested, the positional encoding considerably helped the convergence of the network. We also use a sine activation function in every layer other than the last layer, which is linear. The sine activation function has favorable features for wavefield representation [22, 23]. The NN functional provides a continuous representation of the wavefield and the image, as opposed to grid based representations, and such a continuous representation offers many benefits. We can attain the solution at any point, no interpolation is needed.

### 4. TESTING THE NN

As a proof of concept, we consider the zero-offset case, given by the exploding reflector imaging framework. The image in this case is given by summing mono-frequency images over frequency (the zero-time imaging condition). We use a simple layered model to evaluate the performance and accuracy of the approach. We consider a 2.5 by 2.5 km square piece from the left side the Marmousi model (Figure 2a). Using a numerical Helmholtz solver, we generate synthetic data for sources and receivers on the surface, both spaced 25 m apart. The zero-offset component of these data represent the recorded data for our test, and accordingly, the observed data spacing will be 25 m. Figure 2b shows the zero-offset data, which generally replicates the model in structure. We then Fourier transform the data to the frequency domain in preparation for imaging. We will develop the image for frequencies ranging from 3Hz to 12Hz.

### 4.1. Fitting the data

As described above, we first train a data network to fit the data, here over space and frequency. To demonstrate the versatility of the approach in handling irregular sampling, we randomly drop parts of the data to replicate the case of missing data, or irregular acquisition. The network for the data fitting includes a positional encoding representation of the input of length 18, 9 for each of the lateral position and frequency. In addition, only one hidden layer of length 128 was used. We train on 4186 samples covering the locations of the receivers over the range of frequencies of the inverse Fourier transform. The training of the data network required 15000 epochs, and considering the small size network, the cost of this training is cheap.

Figure 3a shows a comparison between the 3Hz observed data, and those learned by the data function using 10-100% of the original data in the training. The prediction looks overall fine down to training with only 30% of the data. Since, also, the second derivative with respect to $x$ of this data function is needed for the PINN training, we evaluate the accuracy of the second derivative. As reference, we calculate the second derivative from the observed data using a second-order finite-difference approximation. The second derivatives for the NN data function are evaluated using automatic differentiation (AD). Figure 3b shows that the accuracy is acceptable down to training with 40% of the original data. As expected, the higher-order derivatives expose more of the inaccuracies
For the wavefield function, we use a slightly larger network than the one used to fit the data. Now the positional encoding is of length 27, 9 elements for each of the input: \( x \), \( z \), and \( \omega \). We have two hidden layers each 128 neurons wide, admitting the real and imaginary parts of the wavefield. We now use 45000 random samples of the inputs to train this network for 60000 epochs (Figure 4a) to fit equation 3 over three ranges of frequency (3-6 Hz, 3-9 Hz, and finally 3-12 Hz). After training, we predict the zero-offset wavefield on a regular grid in space, and sum the predictions over frequency to obtain the image shown in Figure 4b. The image reflects the structure of the original velocity model considering the limited frequency band used in the imaging of 3-12 Hz.

Fig. 3. a) The real part of the observed data at 3Hz, as compared to the NN data functional predictions, when trained with 10-100% of the data. b) The corresponding second derivative computed from the observed data, and compared to those evaluated using automatic differentiation for the various training options in a).

4.2. Imaging

We share, here, initial results from our attempt to implement direct imaging using neural network functions under the framework of physics informed neural networks. Such functional solutions, as continuous representations of the wavefield, offer all the functional flexibility in sampling and in domain coverage. Incorporating the hard constraint fits perfectly with the concept of imaging as we tend to downward continue or back propagate our recorded data from the recording surface. It also allowed us to avoid a fundamental weakness PINN has in training with boundary conditions [16, 25]. It is often hard to figure out the proper weighting of the boundary condition loss term. Also, considering that the loss is applied on a boundary, it tends to have limited impact on the solution space compared to the hard constraint in which its impact, as a source term in the PDE, is felt in the full solution domain.

Though, we have shared the approach for an exploding reflector imaging program, the method can be directly applied to many problems in which we need to develop the wavefield from the data, including inversion, but more directly, source imaging in microseismic location applications, to replace time reversal methods. In this case, we will also need a range of frequencies to properly image the source. The implementation here also provides a spring board to applications in the more interesting prestack direct imaging problem in which its impact, as a source term in the PDE, is felt in the full solution domain.

5. DISCUSSIONS AND CONCLUSIONS

The resulting zero offset image.

![Image](image.png)

Fig. 4. a) The loss curve over 60000 epochs of training. b) The resulting zero offset image.

We share, here, initial results from our attempt to implement direct imaging using neural network functions under the framework of physics informed neural networks. Such functional solutions, as continuous representations of the wavefield, offer all the functional flexibility in sampling and in domain coverage. Incorporating the hard constraint fits perfectly with the concept of imaging as we tend to downward continue or back propagate our recorded data from the recording surface. It also allowed us to avoid a fundamental weakness PINN has in training with boundary conditions [16, 25]. It is often hard to figure out the proper weighting of the boundary condition loss term. Also, considering that the loss is applied on a boundary, it tends to have limited impact on the solution space compared to the hard constraint in which its impact, as a source term in the PDE, is felt in the full solution domain.

Though, we have shared the approach for an exploding reflector imaging program, the method can be directly applied to many problems in which we need to develop the wavefield from the data, including inversion, but more directly, source imaging in microseismic location applications, to replace time reversal methods. In this case, we will also need a range of frequencies to properly image the source. The implementation here also provides a spring board to applications in the more interesting prestack direct imaging problem in which we need to use the double square-root (DSR) equation. The prestack version of the exploding reflector concept is challenging considering the complexity of the equations involved [2]. However, the framework introduced here may allow us to overcome such challenges. Besides, the NN data function will be far more interesting for prestack data, especially even 3D. In this case, the data NN will be a function of five variables, two for each of the sources and receivers, and of course frequency. One of the biggest challenges of the DSR equation is its need for dense sampling of sources and receivers. This requirement can be alleviated using NNs to learn the data function with limited sources and receivers. We saw evidence of such capability in this work.
6. REFERENCES


