Sebu, C. (2019). *Xjenza Online*, 7:1–2.

Xjenza Online - Science Journal of the Malta Chamber of Scientists
www.xjenza.org
DOI: 10.7423/XJENZA.2019.1.01

Research Article

Learning to Invert Pseudo-Spectral Data for Seismic Waveforms

Christopher Zerafa[*1], **Pauline Galea**[1] **and Cristiana Sebu**[2]
[1]*Department of Geosciences, University of Malta, Msida, Malta*
[2]*Department of Mathematics, University of Malta, Msida, Malta*

Abstract. Full-waveform inversion (FWI) is a widely adopted technique used in seismic processing to produce high resolution Earth models, that fully explain the recorded seismic data. FWI is a local optimisation problem which aims to minimise, using a least-squares approach, the misfit between recorded and modelled data. The inversion process begins with a best-guess initial model which is iteratively improved using a sequence of linearised local inversions to solve a fully non-linear problem. Deep learning has gained widespread popularity in the new millennium. At the core of these tools are Neural Networks (NN), in particular Deep Neural Networks (DNN), which are variants of these original NN algorithms with significantly more hidden layers, resulting in efficient learning of a non-linear function between input and output pairs. The learning process within DNN involves repeatedly updating network neuron weights to best approximate input-to-output mappings. There is clear similarity between FWI and DNN as both approaches attempt to solve non-linear mapping in an iterative sense. However, they are fundamentally different in that FWI is knowledge-driven, whereas DNN is data-driven. This article proposes a novel approach which learns pseudo-spectral data-driven FWI. We test this methodology by training a DNN on 1D multi-layer, horizontally-isotropic data and then apply this to previously unseen data to infer the surface velocity. Results are compared against a synthetic model and success and failures of this approach are hence identified.

Keywords: Deep Neural Networks, Full-waveform Inversion, Machine Learning, Computational Geophysics, Pseudo-Spectral Inversion

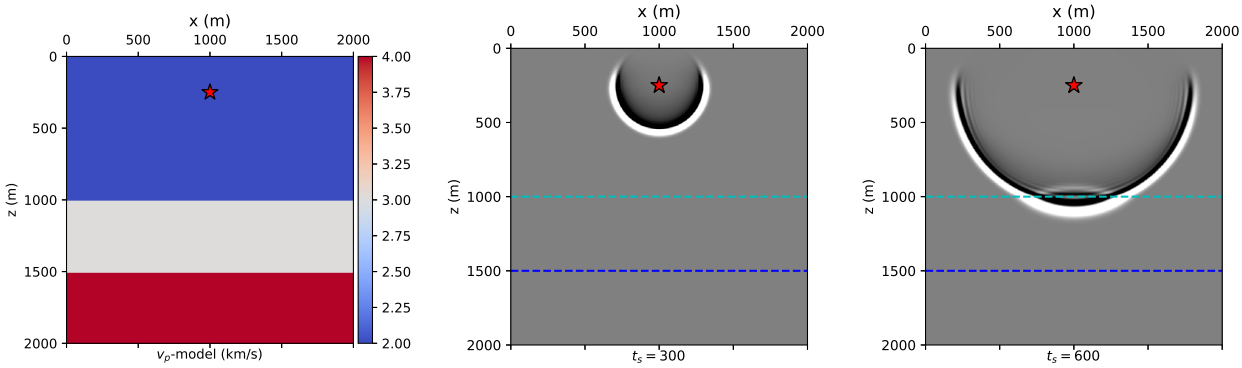
1 Introduction

1.1 Preliminaries

The seismic reflection method uses artificially generated seismic waves that excite the Earth and propagate through the subsurface. They are attenuated by interactions with their medium of propagation and are partially reflected back across a high contrasting acoustic impedance layer. A simple 2D two-layer example of an acoustic forward propagation through the subsurface is given in Fig. 1. The model contains a high acoustic impedance layer between 1 and 1.5 km depth. When hitting the interface between different velocity layers, the wave is reflected back to the surface and recorded by receivers (geophones or hydrophones) located at or close to the surface. The internal structure of the subsurface can then be inferred from the total travel time of the recorded wave.

Full-waveform inversion (FWI) is a technique which attempts to exploit the information contained in the reflected seismic wave-field as much as possible. It goes beyond refraction and reflection tomography techniques, which only use the travel time kinematics of the seismic data. It honours the Physics of the finite-frequency wave equation and uses the additional information provided by the amplitude and phase of the seismic waveform (Tarantola, 1987). FWI seeks to achieve a high-resolution geological model of the subsurface through application of multivariate optimisation to the seismic inverse problem (Lailly, 1983; Tarantola, 1984; Virieux & Operto, 2009). The inversion process begins with a best-guess initial model which is iteratively improved using a sequence of linearised local inversions to solve a fully non-linear problem. Fig. 2 illustrates the imaging uplift which is achievable through FWI. In situations of more complex structures, such as complicated salt structures with convoluted ray-paths in the overbur-

**Correspondence to*: Christopher Zerafa (christopher.zerafa.08@um.edu.mt)



(a) Synthetic p-wave velocity model. (b) Time-step 300 ms. (c) Time-step 600ms.

(d) Time-step 700 ms. (e) Time-step 800 ms. (f) Time-step 950 ms.

Figure 1: Simple 2D two-layer model used for forward propagation of seismic waves. The red star marks the source location at time-step 0 ms. Figure (a) is the ground truth velocity. Figures (b) to (f) illustrate the propagation of an acoustic wave through (a).

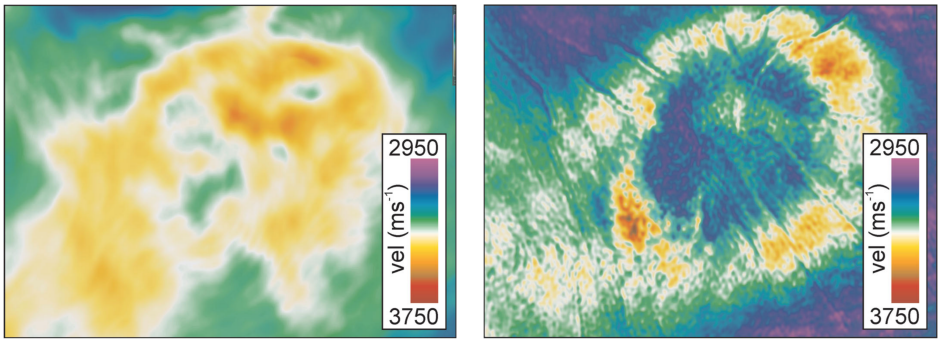
den, the inversion becomes more difficult and computationally more expensive. Fig. 3 illustrates an example of FWI on the 2004 BP synthetic data. The zoomed sections in Fig. 3(d) clearly illustrate a lack of resolution of FWI.

1.2 Aims & Objectives

Optimization theory is fundamental to FWI. The parameters of the system under investigation are reconstructed from indirect observations that are subject to a forward modelling process (Tarantola, 2005). The accuracy of this forward modelling depends on the validity of physical theory that links ground-truth, to the measured data (Innanen, 2014). Moreover, solving for this inverse problem involves learning the inverse mapping from measurements to the ground-truth which is based on a subset of degraded best-estimated data (Tarantola, 2005; Tikhonov & Arsenin, 1977). Two limitations within inverse theory can be identified: (i) solving the forward problem and (ii) training the data.

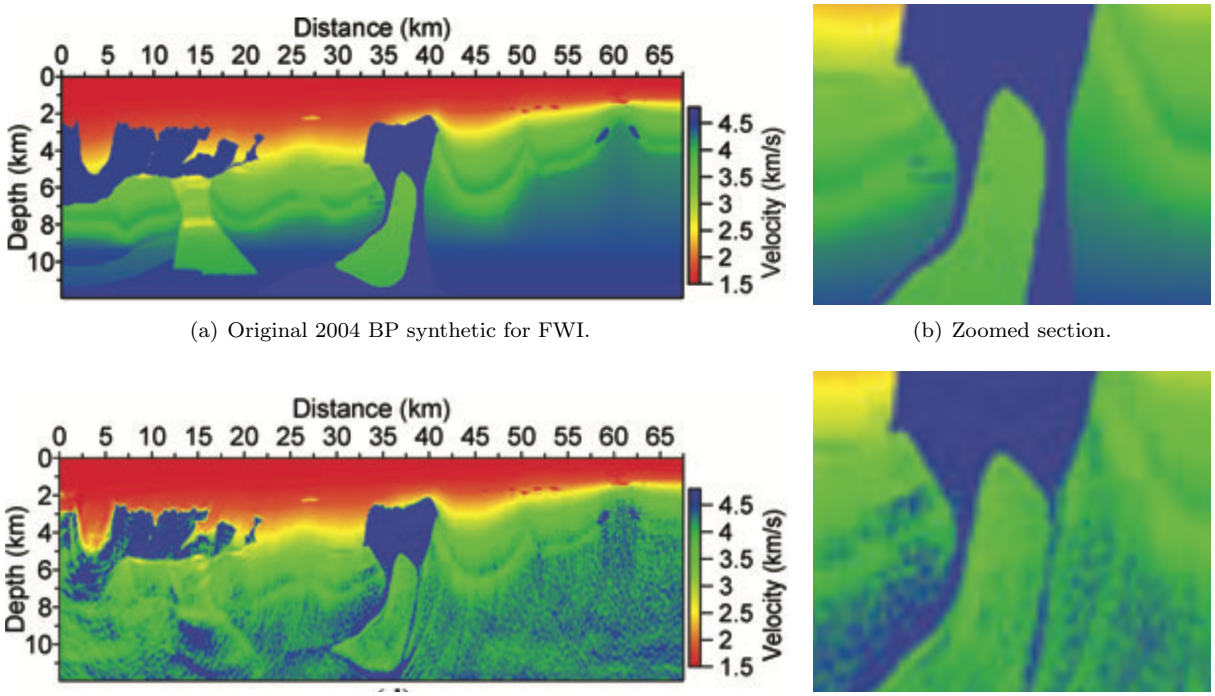
Choice of the numerical method used to solve the forward problem will crucially impact the accuracy of the FWI result. Challenging environments require more complex assumptions to explain the physical link between data and observations, with not necessarily improved levels of accuracy (Morgan et al., 2013). Secondly, the data being used to reconstruct the mapping of measurements for the ground-truth are not optimal. Very wide angle and multi-azimuth data are required to enable full inversion (Morgan et al., 2016); this information might not necessarily have been recorded in the acquisition stages of the data. Furthermore, pre-conditioning of data is a necessity prior to FWI in order to induce well-posedness (Kumar, Ramrez & Butt, 2012; Mothi, Schwarz & Zhu, 2013; Peng, Wang, Chazalnoel & Gomes, 2018; Warner et al., 2013). However, if done incorrectly this can degrade the inversion process (Lines, 2014). Indeed, Lines (2014) shows how FWI remains robust to both random and coherent noise, and his work indicates that with the inclusion of multiple data, FWI proves useful at estimating a better solution in some situations.

Recently, deep learning (DL) techniques have emerged as excellent models and gained great popularity for their widespread success in pattern recognition (Cireşan,



(a) Conventional method. (b) <10 Hz FWI velocity model result.

Figure 2: Horizontal slices though the Samson Dome at 1350 m. From Morgan et al. (2013).



(a) Original 2004 BP synthetic for FWI. (b) Zoomed section.

(c) 2D FWI result. (d) Lack of resolution.

Figure 3: Limitations of FWI in complicated geology. From Shin, Koo, Cha and Park (2010).

Meier, Masci & Schmidhuber, 2012, 2011), speech recognition (Hinton et al., 2012) and computer vision (Krizhevsky, Sutskever, Hinton, Tasci & Kim, 2015; Deng & Yu, 2013). The use of Deep Neural Networks (DNN) to solve inverse problems has been explored by Elshafiey (1991), Adler and Öktem (2017), Chang, Li, Póczos, Kumar and Sankaranarayanan (2017), as well as Wei, Fai and Carin (2017), and has achieved state-of-the-art performance in image reconstruction (Kelly, Matthews & Anastasio, 2017; Petersen, Bölcskei, Grohs & Kutyniok, 2017; Adler, Ringh, Öktem & Karlsson, 2017), super-resolution (Bruna, Sprechmann & LeCun, 2015; Galliani, Lanaras, Marmanis, Baltsavias & Schindler, 2017) and automatic-colorization (Larsson, Maire & Shakhnarovich, 2016).

In Geophysics, the applications of DL techniques have focused on the identification of features and attributes in migrated seismic sections, with few studies looking into velocity inversion. Zhang, Frogner, Araya-Polo and Hohl (2014) used a kernel regularized least-squares method for fault detection from seismic records on numerical experiments. W. Wang, Yang and Ma (2018) employed a fully convolutional neural network (FCN) to perform salt-detection from raw multi-shot gathers,

which was found to be much faster and more efficient than traditional migration and interpretation. Lewis and Vigh (2017) combined DL and FWI to improve the performance for salt inversion, by generating a probability map from learned abstractions of the data and incorporating these in the FWI objective function. These tests results showed promise for automated salt body reconstruction using FWI. Mosser et al. (2018) used a generative adversarial network (Goodfellow, Bengio & Courville, 2016) with cycle-constraints to perform seismic inversion, by reformulating the inversion problem as a domain transfer problem. The mapping between post-stack seismic traces and p-wave velocity models was approximated through DL. More recently, Yang and Ma (2019) developed a supervised FCN for velocity-model, building directly from raw seismograms using a DNN architecture based on U-Net (Ronneberger, Fischer & Brox, 2015). Their training data was obtained from modelling of the acoustic wave equation via a time-domain staggered-grid finite-difference scheme, with numerical experiments showing good potential of DL for seismic velocity inversion.

In this work, we are re-casting the mathematical formulation of FWI within a DL framework. The conventional least-squares formulation of FWI can be expressed as:

$$\min_{\mathbf{m}} J(\mathbf{m}) = ||\mathbf{d} - F(\mathbf{m})||_2^2, \tag{1}$$

where $\mathbf{m} \in M$ is the subsurface model, $F : M \rightarrow D$ is the forward wave equation model, and $\mathbf{d} \in D$ is the observed data. This inversion is non-linear and ill-posed since $\mathbf{d}$ does not contain all subsurface information to define a velocity model explicitly (Biondi, 2006). Based on the Universal Approximation Theorem (Hornik, Stinchcombe & White, 1989), a DNN can be used to approximate the non-linear inverse operator $F^{-1} : D \rightarrow M$ by a pseudo-inverse operator or mapping function g_θ which minimizes the functional:

$$J(\mathbf{m}) = ||\mathbf{m} - g_\theta(\mathbf{d})||^2, \tag{2}$$

where θ is a large simulated dataset of pairs $(\mathbf{m}, \mathbf{d})$ used for learning the process function g_θ (Hastie, Friedman & Tibshirani, 2001). In particular, based on the work of Falsaperla, Graziani, Nunnari and Spampinato (1996), DNN utilizing pseudo-spectral transformed data $\mathcal{F}$, facilitates the learning process due to better sparsity in the transformed domain, as compared to the time domain. The novelty of this approach is the combination of both DL, signal processing and inverse theory for subsurface velocity inversion. This paper aims to prove this theoretical potentially viable solution via a practical implementation to a 1D synthetic model.

The structure of this manuscript is as follows. Section 1 introduces the subject of FWI and its importance within current workflows for seismic exploration. Limitations within the current formulation are identified and a novel approach to devise better velocity models of the subsurface is proposed. In Section 2, mathematical fundamentals for FWI and DNN are derived respectively. These are then compared and their differences are highlighted. In particular, FWI is recast as a DL problem. Based on the derived formulation in Section 2, numerical results of this novel approach are presented in Section 3 and a 1D synthetic highlights the success and failures of this approach. In Section 4, concluding remarks are presented.

2 Theoretical Framework and Methodology

2.1 Inverse Problem Formulation

The aim of inversion is to estimate the parameters of a physical system based on the measurements available. In the case of Geophysics, the physical system is the Earth and data are the recorded wave-field.

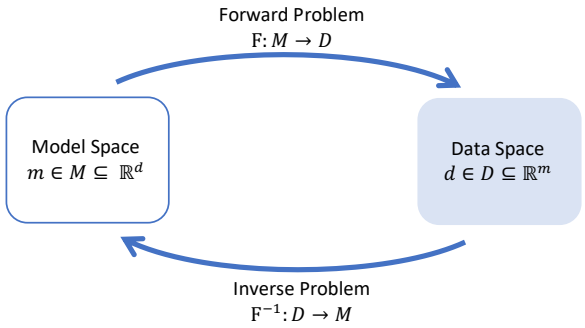


Figure 4: Visual representation of the mapping between the Forward and the Inverse problem.

The recorded wave-field is known, while the physical properties of the medium which the wave-field propagated through are the unknowns. The wave-field will be a function of these medium properties and the function for the forward problem can be as expressed as:

$$\mathbf{d} = F(\mathbf{m}), \tag{3}$$

where $F : M \rightarrow D$, $F \in \mathbb{R}^{(m \times d)}$ is the operator applied on the model space $\mathbf{m} \in M \subseteq \mathbb{R}^d$ to recover measurements $\mathbf{d} \in D \subseteq \mathbb{R}^m$. The forward problem is well-posed, that is, a unique solution exists that depends continuously on the model in some appropriate topology.

The opposite to forward modelling is the inversion. This involves making assumptions on the physical properties of the object we want to image, to be able to compute the wave-field at any given time and location to a certain degree of accuracy. If F is invertible, the inverse problem is given by:

$$\mathbf{m} = F^{-1}(\mathbf{d}), \tag{4}$$

This aims to extract all the information contained within the data.

2.2 FWI as Local Optimisation

Lailly (1983) and Tarantola (1984) re-cast the migration imaging principle introduced by Claerbout (1971) as a local optimisation problem. The forward problem is based on the wave equation, which is one of the most fundamental equations in Physics used for the description of wave motion. It is a second order, partial differential equation involving both time and space derivatives.

The particle motion for an isotropic medium is given by:

$$\frac{1}{c(\mathbf{m})^2}\frac{\partial^2 p(\mathbf{m},t)}{\partial t^2} - \nabla^2 p(\mathbf{m},t) = s(\mathbf{m},t), \tag{5}$$

where $p(\mathbf{m},t)$ is the pressure wave-field, $c(\mathbf{m})$ is the acoustic p-wave velocity and $s(\mathbf{m},t)$ is the source (Igel, 2016). To solve the wave equation numerically, it can be expressed as a linear operator. Although the data $\mathbf{d}$ and model $\mathbf{m}$ are not linearly related, the wave-field $p(\mathbf{m},t)$ and the sources $s(\mathbf{m},t)$ are linearly related by the equation:

$$\mathbf{A}p(\mathbf{m},t) = s(\mathbf{m},t), \tag{6}$$

where $p(\mathbf{m},t)$ is the pressure wave-field produced by a source $s(\mathbf{m})$ and $\mathbf{A}$ is the numerical implementation of the operator:

$$\frac{1}{c(\mathbf{m})^2}\frac{\partial^2}{\partial t^2} - \nabla^2, \tag{7}$$

A common technique employed within the forward modelling stage is to perform modelling in a pseudo-spectral domain ($\mathcal{F}$) rather than the time domain ($\mathcal{T}$). The most common domain is the Fourier domain (Igel, 2016); computational implementation is generally achieved via the Fast Fourier Transform (FFT) developed by Cooley and Tukey (1965), as it utilises the fact that $e^{-2\pi i/N}$ is an N-th primitive root of unity and allows for the reduction of computational costs from $O(N^2)$ to $O(N \log N)$.

After forward modelling the data in a pseudo-spectral domain, the objective is to seek to minimize the difference between the observed data and the modelled data. The metric for the difference or misfit between the two datasets is known as the misfit-, objective- or cost-function J. The most common cost function is given by the L_2-norm of the data residuals:

$$J(\mathbf{m}) = \frac{1}{2}||d - F(\mathbf{m})||_D^2, \tag{8}$$

where D indicates the data domain given by n_s sources and n_r receivers (Igel, 2016). The misfit function J can be minimized with respect to the model parameters d if the gradient is zero, namely:

$$\nabla J = \frac{\partial J}{\partial \mathbf{d}} = 0, \tag{9}$$

Minimising the misfit function is generally achieved via a linearised iterative optimisation scheme, based on the Born approximation in scattering theory (Born & Wolf, 1980; Clayton & Stolt, 1980). The inversion algorithm starts with an initial estimate of the model $\mathbf{m}_0$. After the first pass via forward modelling, the model is updated by the model parameter perturbation $\Delta\mathbf{m}_0$. This newly updated model is then used to calculate the next update and the procedure continues iteratively until the computed model is close enough to the observations, based on a residual threshold criterion. At each iteration k, the misfit function $J(\mathbf{m}_k)$ is calculated from the previous iteration model $\mathbf{m}_{k-1}$ by:

$$J(\mathbf{m}_k) = J(\mathbf{m}_{k-1} + \Delta\mathbf{m}_0), \tag{10}$$

Assuming that the model perturbation is small enough with respect to the model, Eq. (10) can be expanded via Taylor series up to second orders as:

$$\begin{aligned} J(\mathbf{m}_k) &= J(\mathbf{m}_{k-1} + \Delta\mathbf{m}_0) \\ &= J(\mathbf{m}_{k-1}) + \delta\mathbf{m}_{k-1}^T \frac{\partial J}{\partial \mathbf{m}_{k-1}} + \frac{1}{2}\delta\mathbf{m}_{k-1}^{2T}\frac{\partial^2 J}{\partial \mathbf{m}_{k-1}^2}, \end{aligned} \tag{11}$$

Taking the derivative of Eq. (11) and minimizing to determine the model update leads to:

$$\delta\mathbf{m}_{k-1} \approx -\mathbf{H}^{-1}\nabla_{\mathbf{m}_{k-1}} J, \tag{12}$$

where $\mathbf{H} = \frac{\partial^2 J}{\partial \mathbf{m}_{k-1}^2}$ is the Hessian matrix and $\nabla_{\mathbf{m}_{k-1}} J$ the gradient of the misfit function. The Hessian matrix is a symmetric matrix of size $N \times N$ where N is the number of model parameters and represents the curvature trend of the quadratic misfit function.

FWI is an ill-posed problem, implying that an infinite number of models that fit the observations exist. Well-posedness can be introduced with the addition of Tikhonov L_2-norm regularization (Tikhonov, 1963, 3; Tikhonov & Arsenin, 1977):

$$J(\mathbf{m}) = \frac{1}{2}\left[||d - F(\mathbf{m})||_D^2 + \lambda\,||\mathbf{m}||_M^2\right], \tag{13}$$

where λ is the regularization parameter which signifies the trade-off between the data and model residuals.

2.3 FWI Algorithm Summary

A summary of FWI as a local optimisation problem is given in Algorithm 1 and a schematic is illustrated in Fig. 5.

Algorithm 1 FWI as a local optimisation problem

(I) Choose an initial model $\mathbf{m}_0$ and source wavelet $s(\mathbf{m})$.
(II) For each source location, solve the forward problem $F : M \rightarrow D$ using pseudo-spectral forward modelling everywhere in the model space to get a predicted wave-field $\mathbf{d}_k$. This is sampled at receivers $r(\mathbf{m})$.
(III) At every receiver $r(\mathbf{m})$, data residuals are calculated between the modelled wave-field $\mathbf{d}_k$ and the observed data $\mathbf{d}$.
(IV) These data residuals are back-propagated from the receivers to produce a back-propagated residual wave-field.
(V) For each source location, the misfit function $J(\mathbf{m})$ is calculated for the observed data and back-propagated residual wave-field in order to generate the gradient ∇J required at every point in the model.
(VI) The gradient is scaled based on the step-length α, applied to the starting model and an updated model is obtained $\mathbf{m}_{(k+1)}$.
(VII) The process is iteratively repeated from Step 2 until the convergence criterion is satisfied.

2.4 Deep Neural Networks for FWI

Neural Networks (NN) are a subset of tools in machine learning, which when applied to inverse problems, can approximate the non-linear functional of the inverse problem $F^{-1} : D \rightarrow M$. That is, using a NN, a non-linear mapping can be learned to minimize

$$||\mathbf{m} - g_\theta(\mathbf{d})||^2, \tag{14}$$

where θ the large data set of pairs $(\mathbf{m}, \mathbf{d})$ used for the learning process (Lucas, Iliadis, Molina & Katsaggelos, 2018).

The most elementary component in a NN is a neuron. This receives excitatory input and sums the result to produce an output or activation, representing a neuron's action potential which is transmitted along its axon (Raschka & Mirjalili, 2017). For a given artificial neuron, consider n inputs with signals m and weights w. The output d of the k^{th} neuron from all input signals is given by:

$$d_k = \sigma\left(b + \sum_{j=0}^{m} w_{kj} m_j\right), \tag{15}$$

where σ is the activation function and b is a bias term enabling the activation functions to shift about the origin. When multiple neurons are combined together they form a NN. The architecture of a NN refers to the number of neurons, their arrangement and their connectivity (Šíma & Orponen, 2003). The initial layer of nodes $\mathbf{m}$ are referred to as the Input Layer. These are connected to a sequence of hidden layers of neurons. The final layer of the neurons is not a hidden layer and is referred to as the Output Layer. Communication proceeds layer by layer from the input layer, via the hidden layers, up to the output layer. If a NN has two or more hidden layers, it is called a DNN. Fig. 6 shows a NN

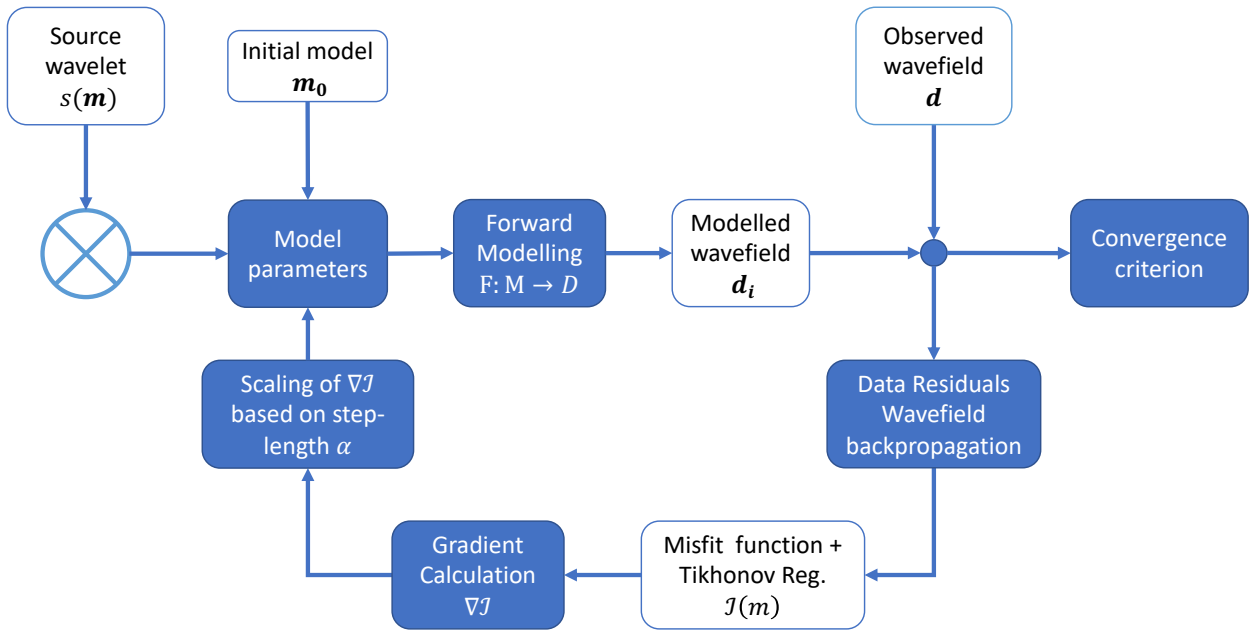


Figure 5: Schematic of a FWI workflow solved as an iterative optimisation process.

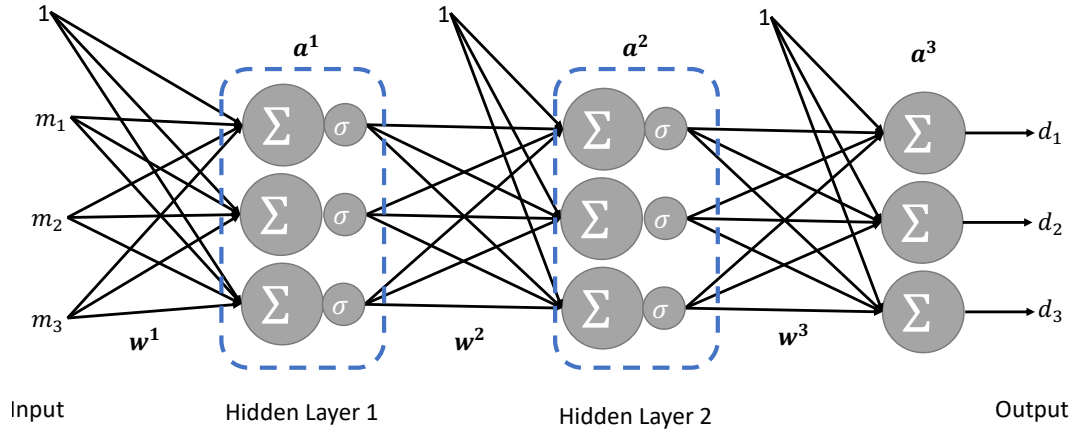


Figure 6: An example of a fully connected NN with 2 hidden layers. All weights w and bias b are learned during the training phase. The 1's connected to each hidden layer represents bias nodes which help the NN learn patterns by allowing the output of an activation function to be shifted. Adapted from Lucas, Iliadis, Molina and Katsaggelos (2018).

consisting of 2 hidden layers. The output of the unit in each layer is the result of the weighted sum of the input units, followed by a non-linear element-wise function. The weights between each unit are learned as a result of a training procedure.

When training a DNN, the forward propagation through the hidden layers from input $\mathbf{m}$ to output $\mathbf{d}$ needs to be measured for its misfit. The most commonly used cost function is the Sum of Squared Errors (SSE), defined as:

$$J(\mathbf{m}) = \frac{1}{2}\sum_{i=1}^{J}\left(\mathbf{m} - g_\theta(\mathbf{d}^{(i)})\right)^2, \qquad (16)$$

where $\mathbf{d}$ is the labelled true dataset, $\mathbf{d}^{(i)}$ is the output from the i^{th} forward pass through the network and the summation is across all neurons in the network. The objective is to minimize the function J with respect to the weights w of the neurons in the NN. Employing the Chain Rule and after a series of recursive formulations, the error signals for all neurons in the network can be recursively calculated throughout the network and the derivative of the cost function with respect to all the weights w can be calculated. Training of the DNN is then achieved via a Gradient Descent algorithm, referred to as back-propagation training algorithm (Rumelhart, Hinton & Williams, 1985). The reader is referred to Goodfellow et al. (2016) and citations therein for a full mathematical formulation.

2.5 Outline for Solving FWI Using DNN

Algorithm for training of a DNN for FWI is given in Algorithm 2 and a schematic is given in Fig. 7.

Algorithm 2 FWI as a DNN problem

(I) Setup a deep architecture for the NN.
(II) Initialise the set of weights w^l and biases b^l.
(III) Forward propagate through the network connections to calculate input sums and activation function for all neurons and layers.
(IV) Calculate the error signal for the final layer δ^L by choosing an appropriate differentiable activation function.
(V) Back-propagate the errors (δ^l) for all neurons in layer l.
(VI) Differentiate the cost function with respect to biases $\left(\frac{\partial J}{\partial b^l}\right)$.
(VII) Differentiate the cost function with respect to weights $\left(\frac{\partial J}{\partial w^l}\right)$.
(VIII) Update weights w^l via gradient descent.
(IX) Recursively repeat from Step 3 until the desired convergence criterion is met.

3 Numerical Example

3.1 Experiment Setup

The hypothesis we would like to prove is as follows:

"Given a seismic trace in the time domain, invert for the seismic velocity (v_p) via a DNN which transforms the input data into pseudo-spectral domain and learns to invert for a velocity estimate."

3.2 Training Data

Learning of the inversion from time to pseudo-spectral domain requires a training dataset which maps time to Fourier components of magnitude and phase, and their respective velocity profile. For our numeric example, 500,000 randomly generated mappings from time ($\mathcal{T}$) to Fourier components ($\mathcal{F}$) for a 2000 ms time window

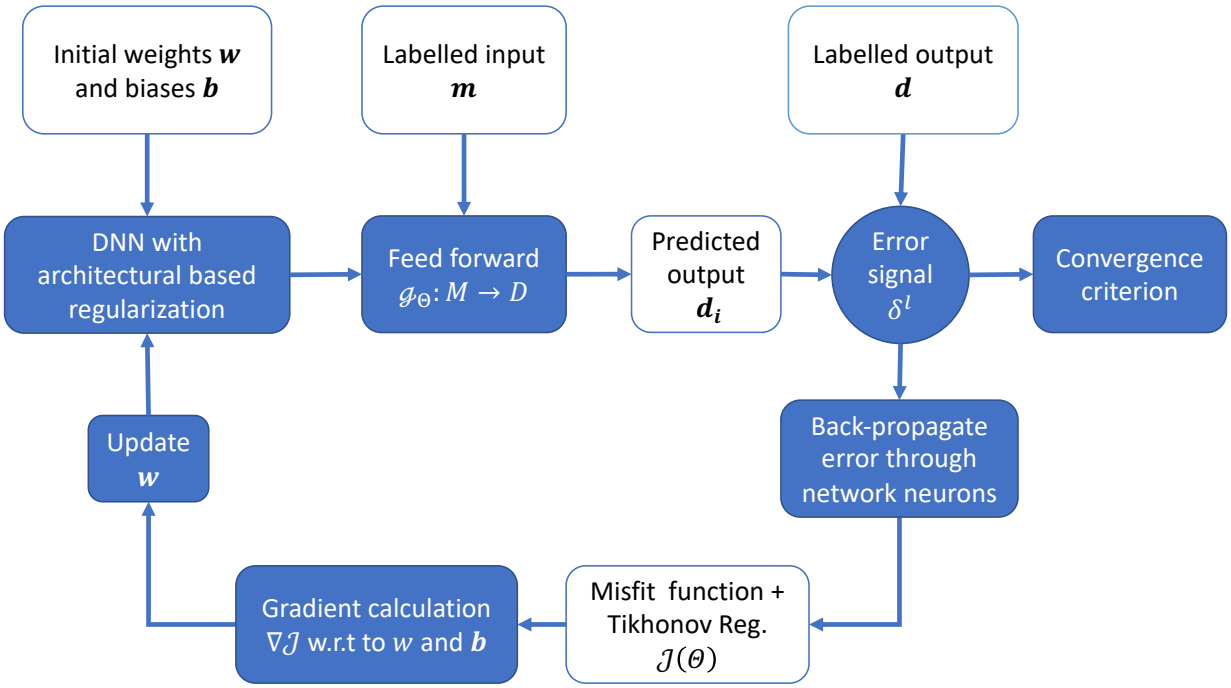


Figure 7: Schematic of a FWI workflow solved as learned optimisation process.

were created. The steps involved in the creation of the synthetic are shown in Fig. 8 for a sample velocity profile, and the steps involved in creating the dataset are given as:

i Randomly create a v_p velocity profile for a 2000 ms time duration, with values ranging from 1400 ms^{-1} to 4000 ms^{-1}. The lower bound of 1400 ms^{-1} was selected as in normal off-shore seismic exploration conditions, the smallest observed velocity is that of the water which ranges from 1450 ms^{-1} to 1460 ms^{-1} (Cochrane & Cooper, 1991). Following the assumption that limestones, carbonates and salt deposits are not present in the subsurface model being inverted, as these have velocity ranges in excess of 4000 ms^{-1}, the upper bound of 4000 ms^{-1} was selected as this is the upper limit of velocity in porous and saturated sandstones (Lee, Hutchinson, Collett & Dillon, 1996).

ii Calculate the density ρ based on Gardner's equation (Gardner, Gardner & Gregory, 1974) given by $\rho = \alpha {v_p}^{\beta}$ where $\alpha = 0.31$ and $\beta = 0.25$ are empirically derived constants that depend on the Geology.

iii At each interface, calculate the Reflection Coefficient $\mathcal{R} = \frac{\rho_2 v_{p_2} - \rho_1 v_{p_1}}{\rho_2 v_{p_2} + \rho_1 v_{p_1}}$ where ρ_i is density of medium i and v_{p_i} is the p-velocity in medium i.

iv For each medium, calculate the Acoustic Impedance $\mathcal{Z}_i = \rho_i v_{p_i}$.

v Define a wavelet $\mathcal{W}$. This was selected to be a Ricker wavelet at 10 Hz (Ricker, 1943). The Ricker wavelet is a theoretical waveform that takes into account the effect of Newtonian viscosity and is representative of seismic waves propagating through visco-elastic homogeneous media (Y. Wang, 2015), thus making it ideal for this numerical simulation. Based on literature results, the central frequency of 10 Hz was chosen as a nominal value to be representative of normal FWI conditions (Morgan et al., 2013). Beyond 10 Hz would be considered to be super-high-resolution FWI (Mispel, Furre, Sollid & Maaø, 2019), which goes beyond the scope of this manuscript.

vi The Reflection Coefficient and wavelet are convolved to produce the seismic trace $\mathcal{T}$

vii Fourier coefficients for magnitude $\mathcal{F}(\zeta)$ and phase $\mathcal{F}(\phi)$ are derived based on the FFT.

3.3 DNN Architecture

Fig. 9 illustrates the NN architecture used to first invert for the Fourier coefficients from the time domain and then invert for velocity profile. The complete workflow had 5 modules, with each module consisting of NN with 5 fully-connected hidden layers. The layer distributions consisted of an input layer of 2000 neurons, then a set of 5 hidden layers of sizes 1000, 500, 250, 500, 1000 neurons, and an output layer of 2000 neurons. This hour-glass design can be considered representative of multi-scale FWI (Bunks, Saleck, Zaleski & Chavent, 1995) since at each hidden layer, the NN learns an ab-

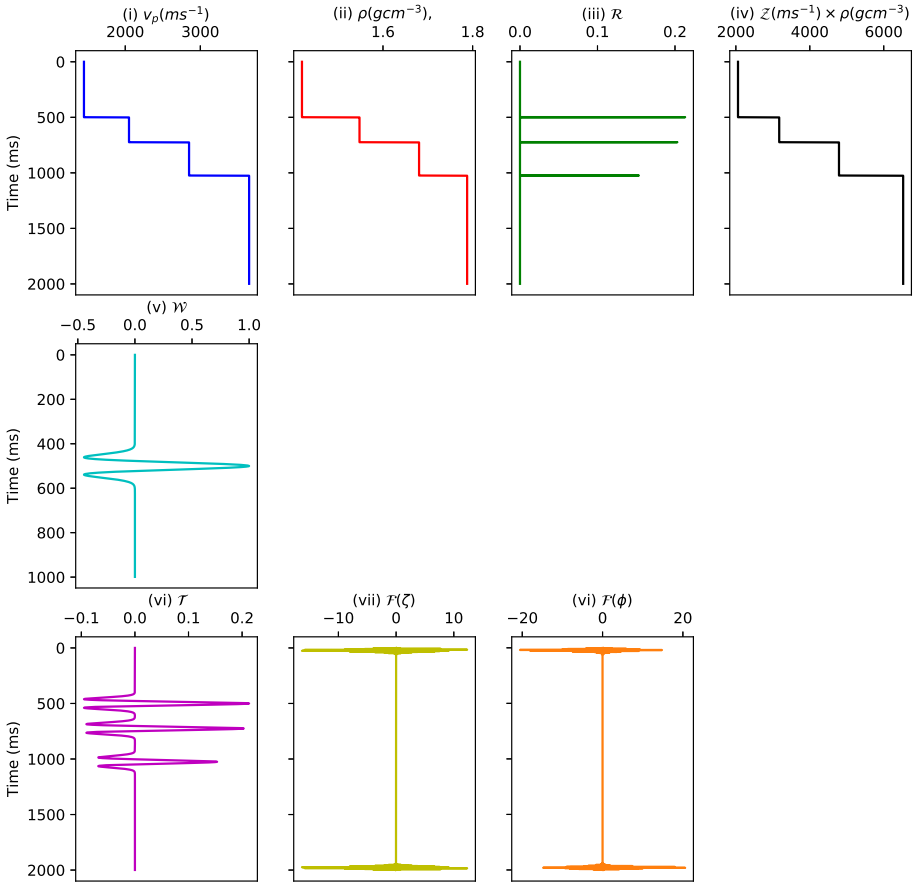


Figure 8: Workflow for creating a pseudo-spectral synthetic trace. This was repeated 500,000 times with random parameters generated within the pre-defined limits stated in Section 3.2 in order to create the learning dataset.

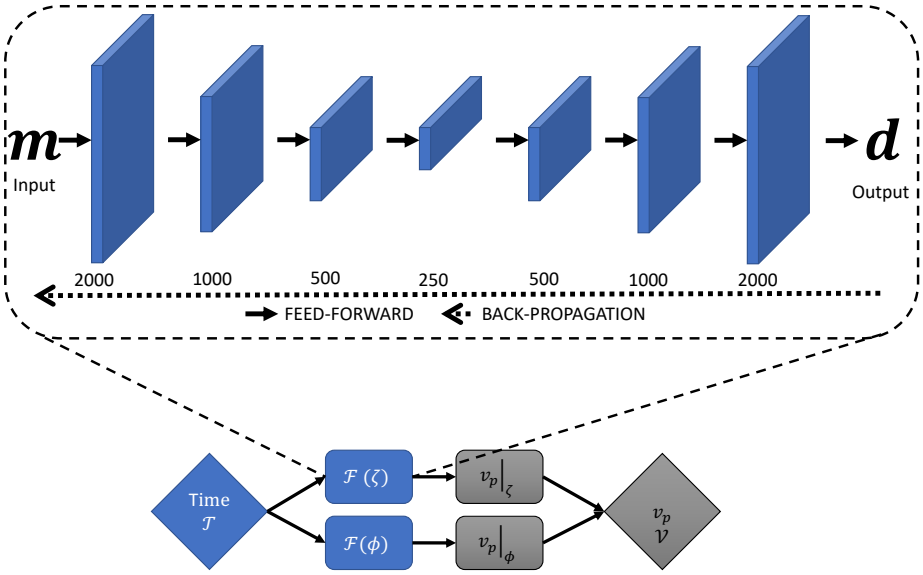


Figure 9: Pseudo-spectral FWI DNN architecture. The highlighted section indicates the set-up employed in each of the 5 modules. Each network has an hour-glass shape with layers of sizes 2000-1000-500-250-500-1000-2000 neurons which can be related to multi-scale FWI. The bottom section illustrates the DNN workflow, where $\mathcal{T}$ is the input time domain, $\mathcal{V}$ is the output v_p velocity and $\mathcal{F}$ is the Fourier domain, with magnitude ζ and phase ϕ.

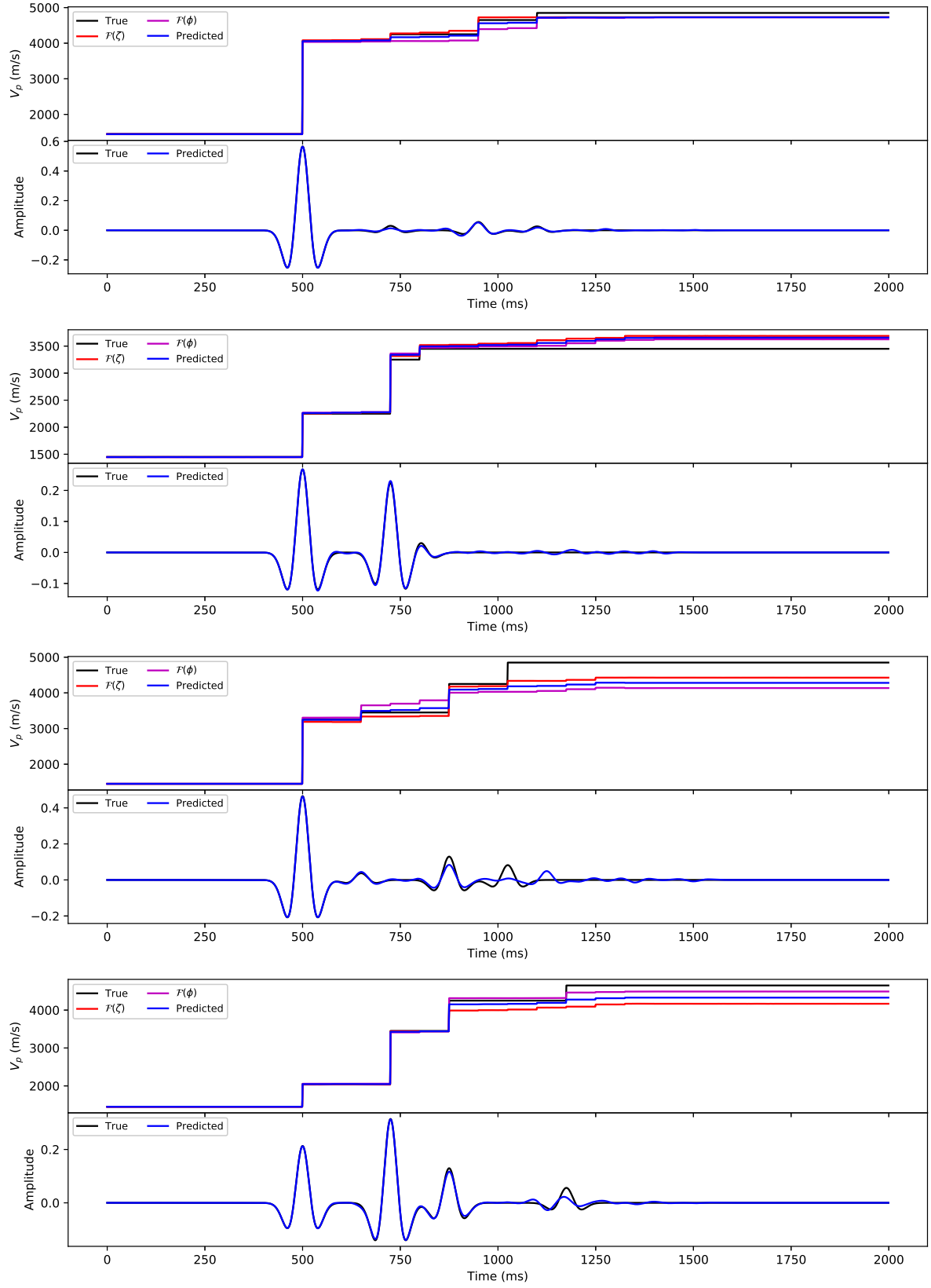
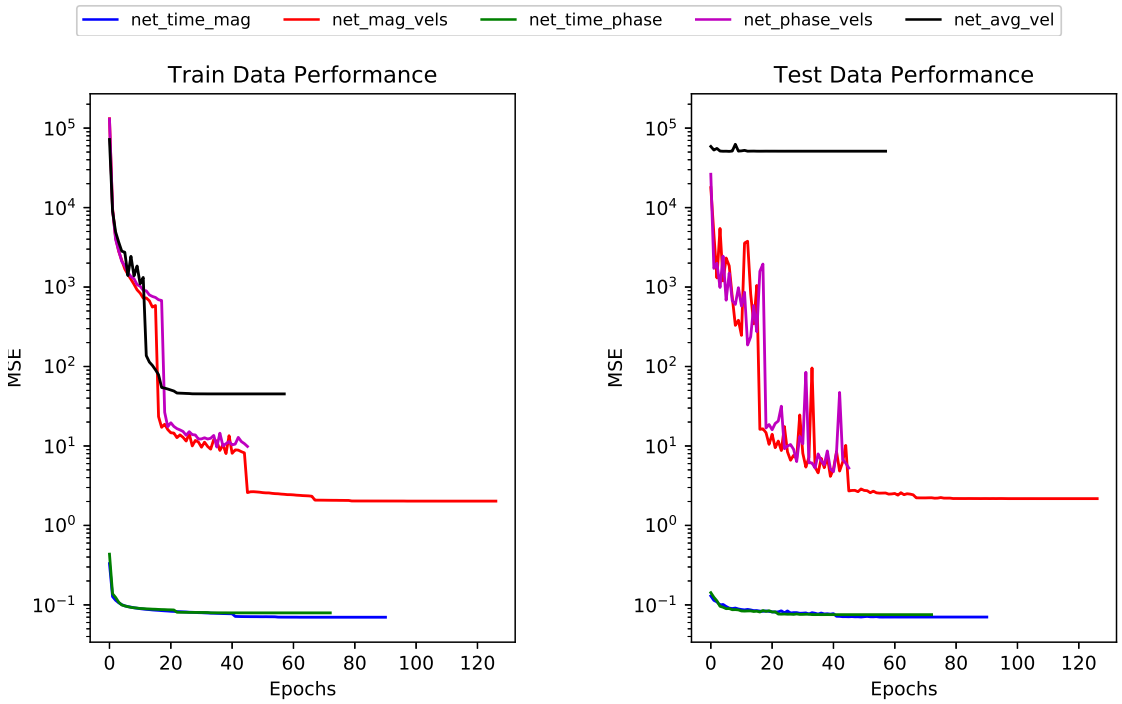
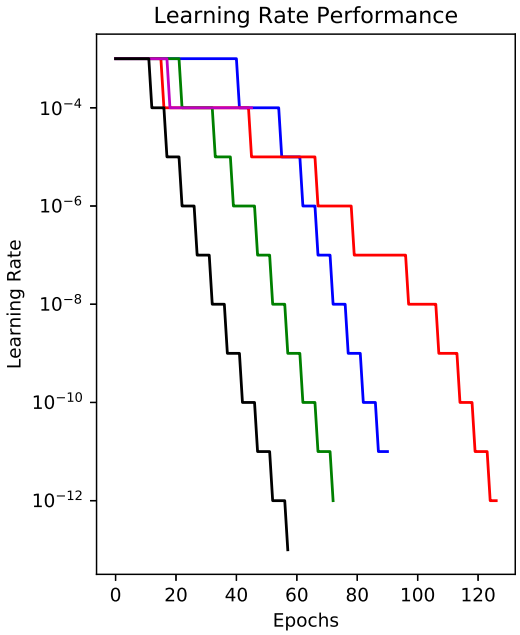


Figure 10: Four different predictions obtained from learned weights of the DNN on unseen data. The top panels are the velocity profile reconstructions from the two NN architecture branches ($\mathcal{F}(\zeta)$ and $\mathcal{F}(\phi)$) and the combined result. Bottom panels are the observed and inverted waveforms.



(a) Training dataset MSE over the different epochs per DNN component. Overall performance is decreasing per epoch, indicating that the DNN is learning to invert.

(b) Test dataset MSE over the different epochs per DNN component.



(c) Learning Rate performance over the different epochs per DNN component.

Figure 11: DNN performance metrics.

stracted component of the data at a different scale. The network employed a sum of squared errors loss function, data batching, early stopping, L_2-norm regularization updates and executed for 200 epochs. A Rectified linear unit or ReLU function given by $f(x) = \max(0, x)$ was used as an activation function. This is a non-linear function which allows for back-propagation of errors. When employed on a network of neurons, the negative component of the function is converted to zero and the neuron is deactivated, thus introducing sparsity within the network and making it efficient and easy for computation. The output from each parallel thread in the flow is fed into another neural network which learns the optimal way of combining the outputs. In total, the DNN had 25 hidden layers. The learning or back-propagation for each network was optimized via an ADAM optimizer (Kingma & Ba, 2014), which is a stochastic gradient descent-based algorithm for first order gradient-based optimisation, which employs on adaptive estimates of lower-order moments. The DNN was implemented in Python 3.7, using Keras 2.2.4 (Chollet, 2015) and TensorFlow 1.13.1 (Abadi et al., 2016) backend.

3.4 Numerical Results

Fig. 10 illustrates the application of DNN architecture in Section 3.3 for a sample of unseen data and the respective reconstruction. Inspection of the first 750 ms indicates that the DNN approach is able to reconstruct both the velocity and the waveform profile with minimal error, irrespective of the number of layers and the magnitude of the acoustic difference in this time range. Beyond 750 ms, reconstructions start suffering from slight degradation. As illustrated in the velocity reconstruction of the middle figure, the inaccuracy is minimal and ranges $\pm 100\,\mathrm{ms}^{-1}$. However, this leads to perturbations in the reconstruction and does not allow for perfect matching. Further inspection suggests that the main source of error is due to the magnitude component of the network (red). To improve this error component, the network inverting for the magnitude component of the FFT would need to be trained and generalised further.

Fig. 11 shows the DNN metric performance over the different epochs per DNN component. Fig. 11(a) and 11(b) illustrate the MSE performance for the training and testing dataset respectively. Considering the former, the plots indicate that the network is indeed learning, since MSE is decreasing at each epoch. Comparing respective DNN components between the training and the testing dataset metrics, there is evidence of no under-fitting or over-fitting with the pseudo-spectral learning components of the DNN architecture (`net_time_mag`, `net_mag_vels`, `net_time_phase`, `net_phase_vel`). Furthermore, there is indeed good-fit, since training and testing MSE both decrease to a point of stability with a minimal difference between the two final MSE values. On the other hand, `net_avg_vel` component, which is learning to average out the velocity from Fourier components, indicates symptoms of an under-presented training dataset. Moreover, these MSE performance plots indicate that the technique might suffer from a *compounding error* issue. The two best performing components are the first layer of learning for the inversion, namely Time-to-FFT-Magnitude (`net_time_mag`) and Time-to-FFT-Phase (`net_time_phase`), as their MSE performance plateaus at 10^{-1}. In the second phase of the inversion, which converts respective FFT components to velocities (FFT-Magnitute-to-Velocity (`net_mag_vels`) and FFT-Phase-to-Velocity (`net_phase_vels`)), the error plateaus are at 10^1, which is two orders of magnitude greater. The final network component sits even higher on the scale at 10^2. Both the train and the test dataset show drastic decreases in the MSE at different epoch levels. These can be attributed to the step-wise reductions in learning rate shown in Fig. 11(c). This varying learning rate allows the network to move to a deeper optimisation level and approach a more global minima for the optimisation problem.

4 Conclusions

In this manuscript we presented the investigation of direct modelling for seismic waveforms using a DNN, which first converts data to a pseudo-spectral domain and subsequently inverts for velocity profiles. Experimental results demonstrated that the use of synthetically generated data to train a DNN proves to be a viable technique in order to learn how to invert via pseudo-spectral data. Although inversion was successfully achieved in the numerical examples presented, one branch of the DNN architecture was lacking in inversion performance and was resulting in a compounding error effect. To improve the overall performance of the technique, data augmentation will be considered, as it is probable that 500,000 random traces are not sufficient to train the magnitude component of the Fourier transform for the network, in order to achieve a desirable performance. In addition, fine-tuning of the NN architecture in the form of in-between layer regularization, neuron drop-out during epoch training and convolutional layers have yet to be investigated. Moreover, in the next stage, this technique will be used for the inversion of more interesting subsurface structures which have a geological relevance, to evaluate image resolution when compared to standard FWI, and to also consider the case of a sequential input in the form of a Recurrent Neural Network, similar to the work of Sun, Niu, Innanen, Li and Trad (2019), but via a pseudo-spectral approach.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... Devin, M. et al. (2016). Tensorflow: large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Adler, J. & Öktem, O. (2017). Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, *33*(12), 1–24.

Adler, J., Ringh, A., Öktem, O. & Karlsson, J. (2017). Learning to solve inverse problems using Wasserstein loss. *Iclr 2018*, 1–13.

Biondi, B. L. (2006). *3D seismic imaging*. Investigations in Geophysics No. 14. Society of Exploration Geophysicists.

Born, M. & Wolf, E. (1980). Principles of optics. *Pergamon Press*, *6*, 188–189.

Bruna, J., Sprechmann, P. & LeCun, Y. (2015). Super-Resolution with Deep Convolutional Sufficient Statistics. *arXiv preprint arXiv:1511.05666*.

Bunks, C., Saleck, F. M., Zaleski, S. & Chavent, G. (1995). Multiscale seismic waveform inversion. *Geophysics*, *60*(5), 1457–1473.

Chang, J. H., Li, C.-L., Póczos, B., Kumar, B. V. K. & Sankaranarayanan, A. C. (2017). One Network to Solve Them All—Solving Linear Inverse Problems using Deep Projection Models. *IEEE International Conference on Computer Vision (ICCV)*.

Chollet, F. et al. (2015). Keras. https://keras.io.

Cireşan, D. C., Meier, U., Masci, J. & Schmidhuber, J. (2011). A committee of neural networks for traffic sign classification. In *The 2011 International Joint Conference on Neural Networks* (pp. 1918–1921). San Jose, CA, USA: IEEE.

Cireşan, D. C., Meier, U., Masci, J. & Schmidhuber, J. (2012). Multi-column deep neural network for traffic sign classification. *Neural networks*, *32*, 333–338.

Claerbout, J. (1971). Toward a unified theoru of relector mapping. *Geophysics*, *36*(June).

Clayton, R. W. & Stolt, R. (1980). A Born-WKBJ Inversion Method for Acoustic Reflection Data. In *Sep report* (Vol. 46, *1*, pp. 135–158).

Cochrane, G. R. & Cooper, A. K. (1991). Sonobuoy seismic studies at ODP drill sites in Prydz Bay, Antarctica. *Proceedings of the Ocean Drilling Program, Scientific Results*, *119*, 27–43.

Cooley, J. W. & Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. *Mathematics of computation*, *19*(90), 297–301.

Deng, L. & Yu, D. (2013). Deep Learning: Methods and Applications. *Foundations and Trends® in Signal Processing*, *7*(3–4), 197–387.

Elshafiey, I. M. (1991). *Neural network approach for solving inverse problems* (Master's thesis, Iowa State University, Ames, Iowa).

Falsaperla, S., Graziani, S., Nunnari, G. & Spampinato, S. (1996). Automatic classification of volcanic earthquakes by using multi-layered neural networks. *Natural Hazards*, *13*(3), 205–228.

Galliani, S., Lanaras, C., Marmanis, D., Baltsavias, E. & Schindler, K. (2017). Learned Spectral Super-Resolution. *arXiv preprint arXiv:1703.09470*.

Gardner, G. H. F., Gardner, L. W. & Gregory, A. R. (1974). Formation Velocity and Density - The diagnostic basics for Stratigraphic Traps. *GEOPHYSICS*, *39*(6), 770–780.

Goodfellow, I., Bengio, Y. & Courville, A. (2016). *Deep Learning*. Adaptive Computation and Machine Learning series. MIT Press.

Hastie, T., Friedman, J. & Tibshirani, R. (2001). The Elements of Statistical Learning. *Springer Series in Statistics*.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., ... Sainath, T. N. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, *29*(6), 82–97.

Hornik, K., Stinchcombe, M. & White, H. (1989). Multilayer feedforward networks are universal approximator. *Neural Networks*, *2*, 359–366.

Igel, H. (2016). *Computational Seismology: A Practical Introduction* (1st ed.). Oxford University Press.

Innanen, K. (2014). Quantifying the incompleteness of the physics model in seismic inversion. *CREWES Research Report*.

Kelly, B., Matthews, T. P. & Anastasio, M. A. (2017). Deep Learning-Guided Image Reconstruction from Incomplete Data. *arXiv preprint arXiv:1709.00584*, (Nips).

Kingma, D. P. & Ba, J. (2014). Adam: a method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A., Sutskever, I., Hinton, G. E., Tasci, T. & Kim, K. (2015). ImageNet Classification with Deep Convolutional Neural Networks. *Stanford cs231b*.

Kumar, J., Ramrez, A. & Butt, S. (2012). Preparing Data for Full Waveform Inversion: A Workflow for Free-surface Multiple Attenuation. In *74th EAGE Conference and Exhibition - Workshops*. Copenhagen, Denmark: EAGE.

Lailly, P. (1983). The seismic inverse problem as a sequence of before stack migrations. In J. Bednar, R. Redner, E. Robinson & A. Weglein (Eds.), *Conference on Inverse Scattering, Theory and Application* (pp. 206–220). Tulsa, Oklahoma: Philadelphia: Society of Industrial and Applied Mathematics.

Larsson, G., Maire, M. & Shakhnarovich, G. (2016). Learning representations for automatic colorization. In *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)* (Vol. 9908 LNCS, pp. 577–593).

Lee, M., Hutchinson, D., Collett, T. & Dillon, W. P. (1996). Seismic velocities for hydrate-bearing sediments using weighted equation. *Journal of Geophysical Research: Solid Earth*, *101*(B9), 20347–20358.

Lewis, W. & Vigh, D. (2017). Deep learning prior models from seismic images for full-waveform inversion. In *Seg technical program expanded abstracts 2017* (pp. 1512–1517). Society of Exploration Geophysicists.

Lines, L. (2014). FWI and the "Noise" Quandary. *CREWES Research Report*.

Lucas, A., Iliadis, M., Molina, R. & Katsaggelos, A. K. (2018). Using Deep Neural Networks for Inverse Problems in Imaging: Beyond Analytical Methods. *IEEE Signal Processing Magazine*, *35*(1), 20–36.

Mispel, J., Furre, A., Sollid, A. & Maaø, F. A. (2019). High Frequency 3D FWI at Sleipner: A Closer Look at the CO2 Plume. In *81st EAGE Conference and Exhibition 2019*. London, UK: EAGE.

Morgan, J., Warner, M., Arnoux, G., Hooft, E., Toomey, D., VanderBeek, B. & Wilcock, W. (2016). Next-generation seismic experiments - II: Wide-angle, multi-azimuth, 3-D, full-waveform inversion of sparse field data. *Geophysical Journal International*, *204*(2), 1342–1363.

Morgan, J., Warner, M., Bell, R., Ashley, J., Barnes, D., Little, R., ... Jones, C. (2013). Next-generation seismic experiments: wide-angle, multi-azimuth, three-dimensional, full-waveform inversion. *Geophysical Journal International*, *195*(3), 1657–1678.

Mosser, L., Kimman, W., Dramsch, J., Purves, S., De la Fuente Briceño, A. & Ganssle, G. (2018). Rapid seismic domain transfer: Seismic velocity inversion and modeling using deep generative neural networks. In *80th EAGE Conference and Exhibition 2018*. Copenhagen, Denmark: EAGE.

Mothi, S., Schwarz, K. & Zhu, H. (2013). Impact of full-azimuth and long-offset acquisition on Full Waveform Inversion in deep water Gulf of Mexico. *SEG Houston 2013 Annual Meeting*, (June 2013), 924–928.

Peng, C., Wang, M., Chazalnoel, N. & Gomes, A. (2018). Subsalt imaging improvement possibilities through a combination of FWI and reflection FWI. *The Leading Edge*, *37*(1), 52–57.

Petersen, P. C., Bölcskei, H., Grohs, P. & Kutyniok, G. (2017). Optimal approximation with sparse deep neural networks. *arXiv preprint arXiv:1705.01714*.

Raschka, S. & Mirjalili, V. (2017). *Python machine learning* (2nd ed.). Packt Publishing Ltd.

Ricker, N. (1943). Further developments in the wavelet theory of seismogram structure. *Bulletin of the Seismological Society of America*, *33*(3), 197–228.

Ronneberger, O., Fischer, P. & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *18th International Conference on Medical image computing and computer-assisted intervention* (pp. 234–241). Springer. Munich, Germany.

Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1985). *Learning internal representations by error propagation*. California Univ San Diego La Jolla Inst for Cognitive Science.

Shin, C., Koo, N.-H., Cha, Y. H. & Park, K.-P. (2010). Sequentially ordered single-frequency 2-D acoustic waveform inversion in the Laplace-Fourier domain. *Geophysical Journal International*, *181*(2), 935–950.

Šíma, J. & Orponen, P. (2003). General-Purpose Computation with Neural Networks: A Survey of Complexity Theoretic Results. *Neural Comput. 15*(12), 2727–2778.

Sun, J., Niu, Z., Innanen, K. A., Li, J. & Trad, D. O. (2019). A theory-guided deep learning formulation of seismic waveform inversion. In *Seg technical program expanded abstracts 2019* (pp. 2343–2347). Society of Exploration Geophysicists.

Tarantola, A. (1984). Inversion of seismic reflection data in the acoustic approximation. *Geophysics*, *49*(8), 1259–1266.

Tarantola, A. (1987). *Inverse Problems Theory, Methods for Data Fitting and Model Parameter Estimation*. Amsterdam, The Netherlands: Elsevier.

Tarantola, A. (2005). *Inverse problem theory and methods for model parameter estimation*. SIAM.

Tikhonov, A. N. (1963). On the Solution of Incorrectly Stated Problems and a Method of Regularization. *Dokl. Acad. Nauk SSSR*, *151*, 501–504.

Tikhonov, A. N. & Arsenin, V. Y. (1977). *Methods for solving ill-posed problems*. John Wiley and Sons, Inc.

Virieux, J. & Operto, S. (2009). An overview of full-waveform inversion in exploration geophysics. *Geophysics*, *74*(6), WCC1–WCC26.

Wang, W., Yang, F. & Ma, J. (2018). Automatic salt detection with machine learning. In *80th EAGE Conference and Exhibition 2018*. Copenhagen, Denmark: EAGE.

Wang, Y. (2015). Frequencies of the ricker wavelet. *Geophysics*, *80*(2), A31–A37.

Warner, M., Ratcliffe, A., Nangoo, T., Morgan, J., Umpleby, A., Shah, N., ... Bertrand, A. (2013). Anisotropic 3D full-waveform inversion. *GEOPHYSICS*, *78*(2), R59–R80.

Wei, Q., Fai, K. & Carin, L. (2017). An Inner-loop Free Solution to Inverse Problems using Deep Neural Networks. *Advances in Neural Information Processing Systems*, 2370–2380.

Yang, F. & Ma, J. (2019). Deep-learning inversion: a next generation seismic velocity-model building method. *Geophysics*, *84*(4), 1–133.

Zhang, C., Frogner, C., Araya-Polo, M. & Hohl, D. (2014). Machine-learning based automated fault detection in seismic traces. In *76th EAGE Conference and Exhibition 2014*. Amsterdam, The Netherlands: EAGE.