Integrating Deep Neural Networks with Full-waveform Inversion: Reparametrization, Regularization, and Uncertainty Quantification

Weiqiang Zhu[1*], Kailai Xu[2*], Eric Darve[2,3], Biondo Biondi[1], and Gregory C. Beroza[1]

[1]Department of Geophysics, Stanford University, Stanford, CA, 94305
[2]Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA, 94305
[3]Mechanical Engineering, Stanford University, Stanford, CA, 94305
{zhuwq, kailaix, darve, biondo, beroza}@stanford.edu

Abstract

Full-waveform inversion (FWI) is an imaging approach for modeling velocity structure by minimizing the misfit between recorded and predicted seismic waveforms. The strong nonlinearity of FWI resulting from fitting oscillatory waveforms can trap the optimization in local minima. We propose a neural-network-based full waveform inversion method (NNFWI) that integrates deep neural networks with FWI by representing the velocity model with a generative neural network. Neural networks can naturally introduce spatial correlations as regularization to the generated velocity model, which suppresses noise in the gradients and mitigates local minima. The velocity model generated by neural networks is input to the same partial differential equation (PDE) solvers used in conventional FWI. The gradients of both the neural networks and PDEs are calculated using automatic differentiation, which back-propagates gradients through the acoustic/elastic PDEs and neural network layers to update the weights and biases of the generative neural network. Experiments on 1D velocity models, the Marmousi model, and the 2004 BP model demonstrate that NNFWI can mitigate local minima, especially for imaging high contrast features like salt bodies, and significantly improves the inversion in the presence of noise. Adding dropout layers to the neural network model also allows analyzing the uncertainty of the inversion results through Monte Carlo dropout. NNFWI opens a new pathway to combine deep learning and FWI for exploiting both the characteristics of deep neural networks and the high accuracy of PDE solvers. Because NNFWI does not require extra training data and optimization loops, it provides an attractive and straightforward alternative to conventional FWI.

*The two authors contributed equally to this paper.

1 Introduction

Full-waveform inversion (FWI) is a high resolution inversion method in exploration seismology (Tarantola, 1984, 2005; Virieux & Operto, 2009), commonly used for estimating subsurface velocity structure based on seismic waves recorded at the surface. FWI falls within the class of PDE-constrained optimization problems. It solves the wave equation, in the acoustic or elastic approximation, to predict seismic waves based on the velocity model. Through the PDE solver, FWI determines the optimal velocity model by minimizing the misfit between predicted and observed seismic waveforms. The gradient of the misfit function can be efficiently calculated by the adjoint method (Plessix, 2006). Although FWI can achieve high accuracy when the full seismic waveform is matched, the non-linearity of the objective function poses a challenge to the optimization process. The inversion result of FWI suffers from local minima due to cycle skipping in the wave oscillations used to form the objective function, i.e. a $\mathcal{L}_2$-norm loss. This is particularly challenging when either a good initial model is lacking or low frequency content of the waveform data is missing. Sources of noise in real seismic recording will also affect the inversion result. Because artifacts originating from local minima interfere with imaging and can lead to misinterpretation of geological structure, a great deal of research has focused on improving the stability of FWI. One solution is to recover or predict the missing low frequency content, such as through envelope inversion (R.-S. Wu et al., 2014), sparse blind deconvolution (P. Zhang et al., 2017), and phase-tracking methods (Y. E. Li & Demanet, 2016). Another solution is to add regularization and preconditioning, such as Laplacian smoothing (Burstedde & Ghattas, 2009), l_2 penalty (Hu et al., 2009), l_1 penalty (total variation) (Guitton, 2012; Lin & Huang, 2015), and prior information as constraints (Asnaashari et al., 2013). Many other solutions have been proposed and tested, such as, multiscale inversion (Bunks et al., 1995), wave-equation traveltime inversion (Luo & Schuster, 1991), tomographic full-waveform inversion (Biondi & Almomin, 2014), dictionary learning (L. Zhu et al., 2017; D. Li & Harris, 2018), model extension (Barnier et al., 2018a), and model reduction (Barnier et al., 2019). In addition to the strong non-linearity of FWI, uncertainty analysis of FWI results is challenging due both to the high dimensionality of the model space and to the demanding computational cost of solving the wave equation (Gebraad et al., 2020).

In recent studies, the success of deep learning has drawn attention to its potential application in FWI. One direction is to directly build an inverse mapping from data to model by training neural networks on paired seismic recordings and velocity models (Y. Wu et al., 2018; Yang & Ma, 2019). This approach does not rely on solving the wave equation but instead treats FWI as a data-driven machine learning problem similar to that in image recognition. The accuracy and generalization of this approach cannot be guaranteed without the PDE constraints. Another research direction is to apply deep learning as an effective signal processing tool to improve the optimization process of conventional FWI. Sun & Demanet (2020) used neural networks to extrapolate the missing low frequencies and help mitigate the cycle skipping problem. A promising direction is to combine neural networks and PDEs to formulate FWI as a physics-constrained machine learning problem. Richardson (2018) and W. Zhu et al. (2020) have implemented FWI using deep learning frameworks so that reverse-mode automatic differentiation and the various effective optimization tools in deep learning frameworks can be used for FWI. These studies demonstrate the similarity between

FWI and neural networks. Y. Wu & McMechan (2019) proposed a convolutional-neural-network-domain FWI, which reparameterizes the velocity model by a convolutional neural network and minimizes the loss by updating the neural network weights.

In this paper, we propose a method, NNFWI, to integrate deep neural networks with full-waveform inversion. Similar to Y. Wu & McMechan (2019)'s idea, we use deep neural networks to generate a physical velocity model, which is then fed to a PDE solver to simulate seismic waveforms. NNFWI does not need pre-training of the neural networks on the initial velocity model. The training process of NNFWI is similar to that of conventional FWI, but with gradients calculated by automatic differentiation instead of by the adjoint-state method. Thus we can easily optimize the two systems of neural networks and PDEs together. In contrast to conventional FWI, which directly estimates the velocity model, NNFWI reparametrizes the velocity model with a generative neural network model and optimizes the neural network's weights and biases. As demonstrated by Ulyanov et al. (2018)'s work on the deep image prior, the inductive bias captured by deep convolutional networks is an effective image prior and can be used as regularization for tasks such as denoising and super-resolution. Due to the regularization effect of neural networks, NNFWI mitigates the effects of local minima and is robust with respect to noise in the data. Furthermore, NNFWI can model uncertainty by adding dropout layers in the neural network. Dropout not only prevents over-fitting during training deep neural networks, but can also approximate Bayesian inference to capture model uncertainty without much extra computational cost (Gal & Ghahramani, 2016). NNFWI has the potential to provide uncertainty quantification for FWI, which otherwise remains a challenging problem. In this paper we seek to exploit the unique advantages of deep neural networks and the high accuracy and generalization of PDEs to improve inversion and uncertainty analysis in FWI

2 Method

FWI aims to minimize the discrepancy between the observed seismic data $u(x,t)$ and the numerical prediction $\hat{u}$, which is the solution to a wave equation (denoted as $F(u,m)=0$); this equation typically takes one of the two following forms (f is the source term)

1. Acoustic wave equation
$$\frac{\partial^2 u}{\partial t^2} = \nabla \cdot (m^2 \nabla u) + f \tag{1}$$

2. Elastic wave equation
$$\begin{aligned} \rho \frac{\partial v_i}{\partial t} &= \sigma_{ij,j} + \rho f_i \\ \frac{\partial \sigma_{ij}}{\partial t} &= \lambda v_{k,k} + \mu (v_{i,j} + v_{j,i}) \end{aligned} \tag{2}$$

In the second form, $m = (\rho, \lambda, \mu)$. Thus the PDE-constrained optimization problem for estimating an unknown field m is given by

$$\begin{aligned} \min_{m} \quad & D(\hat{u}(x,t,m),u) = \sum_{i=1}^{n} \int_0^T ||\hat{u}(x_i,t,m) - u(x_i,t)||_2^2 dt \\ \text{s.t.} \quad & F(\hat{u},m) = 0 \end{aligned}$$

where D is the misfit function, m is the velocity model, and x and t are spatial and temporal coordinates. $\{x_i\}_{i=1}^n$ designates the discrete locations where observations are available.

The standard FWI derives the gradient $\frac{\partial D}{\partial m}$ using the adjoint method and updates $m \in \mathbb{R}^p$ using a gradient-based optimization method, such as L-BFGS (Figure 1a). There are two challenges when this method is used:

1. This inverse problem is usually ill-conditioned. One approach is to use a misfit function with a regularization term, such as Tikhonov and total variation regularization.

2. Quantifying uncertainties of m is very challenging because the forward simulation, i.e., computing u by solving $F(u,m) = 0$, is very expensive. Conventional methods, such as Markov Chain Monte Carlo (MCMC), usually require a large number of forward simulations, and therefore are very computationally demanding.

To address these two issues, we propose Neural-Network-based Full Waveform Inversion (NNFWI). In NNFWI, we use a generative neural network to parametrize the velocity model

$$m = \mathcal{N}(z,w) \tag{3}$$

where $\mathcal{N}$ is a generative neural network, z is the latent variable, and w includes the weights and biases of the neural network. Equation (3) introduces regularization to the velocity field by representing m with a neural network. In this work we show the regularization effect of two types of neural networks consisting of fully connected layers or convolutional layers. For neural networks of fully connected layers, the latent variable z is set to be the coordinate x, so that $\mathcal{N}(x,w)$ imposes a spatial correlation on m, i.e. $\mathcal{N}(x_1,w)$ and $\mathcal{N}(x_2,w)$ are similar if the spatial coordinates x_1 and x_2 are close. For neural networks of convolutional layers, the local convolutional kernels applied across the entire domain imposes self-similarity and spatial correlation. Ulyanov et al. (2018) demonstrated that this deep image prior of convolutional neural network architectures achieved competitive results as handcrafted self-similarity-based and dictionary-based priors, such as total variation norm, in a variety of image processing problems, such as denoising, super-resolution, and inpainting. Thus, reparametrizing the velocity m with convolutional neural networks naturally introduces regularization to FWI. We set the latent variable z as a fixed random vector and use a similar architecture for $\mathcal{N}(z,w)$ as for the generator in DCGANs (Radford et al., 2016), which applies a series of convolutional layers and upsampling operations to convert the latent vector into a 2D image (Table 1).

The optimization problem of NNFWI can be written as:

$$\begin{aligned} \min_{w} \ & D(\hat{u}(x_i,t,\mathcal{N}(z,w)),u(x_i,t)) = \sum_{i=1}^{n} \int_0^T ||\hat{u}(x_i,t,\mathcal{N}(z,w)) - u(x_i,t)||_2^2 dt \\ \text{s.t.} \ & F(\hat{u},\mathcal{N}(z,w)) = 0 \end{aligned}$$

Table 1: The architectures of the generative neural networks in NNFWI. The fully connected neural network (FC) model is used for the 1D FWI case and the convolutional neural network (CNN) model is used for the Marmousi model and the 2004 BP models. We apply scaling factors to the output of neural networks depending on the physical parameters and units. Note that the parameters and layers in these architectures can be modified for different applications.

Model	NN layer	Architecture
FC model	Input	Spatial coordinate x
	Layer 1	Fully-connected layer (channels = 30) + Tanh
	Layer 2	Fully-connected layer (30) + Tanh
	Layer 3	Fully-connected layer (30) + Tanh
	Layer 4	Fully-connected layer (1) + Tanh
CNN model	Input	Random latent vector
	Layer 1	Fully-connected layer (8) + Relu + Reshape + 2×2 Upsampling + Dropout (0.1)
	Layer 2	4×4 Convolutional layer (32) + Relu + 2×2 Upsampling + Dropout (0.1)
	Layer 3	4×4 Convolutional layer (64) + Relu + 2×2 Upsampling + Dropout (0.1)
	Layer 4	4×4 Convolutional layer (32) + Relu + 2×2 Upsampling + Dropout (0.1)
	Layer 5	4×4 Convolutional layer (1) + Tanh

A diagram of the NNFWI approach is shown in Figure 1b. The data setting of NNFWI, i.e. the initial velocity model and the waveform data is same as for conventional FWI, which makes NNFWI applicable to all conventional FWI problems. In NNFWI, we directly combine the output from the generative neural networks with the initial velocity model as the input for the PDE, so the generative neural networks are trained to predict the updates over the initial model. We calculate the gradients of both the generative neural network and the PDEs by automatic differentiation, which simplifies the optimization with both neural networks and PDEs. Because adjoint-state methods and reverse-mode automatic differentiation are mathematically equivalent (W. Zhu et al., 2020), we can also use adjoint-state methods for the optimization of NNFWI. NNFWI is implemented based on ADCME[1] and ADSeismic[2]. We use the Adam algorithm (Kingma & Ba, 2017) for the optimization of NNFWI. For comparison, we also conduct conventional FWI with the L-BFGS algorithm (C. Zhu et al., 1997).

3 Results

In this section, we evaluate the performance of NNFWI by comparing the inversion results between conventional FWI and NNFWI on three cases.

3.1 1D model

To demonstrate the regularization effect of neural networks, we design two simple 1D velocity models: one with a linear velocity profile (Figure 2) and another with a step-change profile (Figure 3). We implement a 1D acoustic wave equation to carry out forward simulation

[1] `https://github.com/kailaix/ADCME.jl`

[2] `https://github.com/kailaix/ADSeismic.jl`

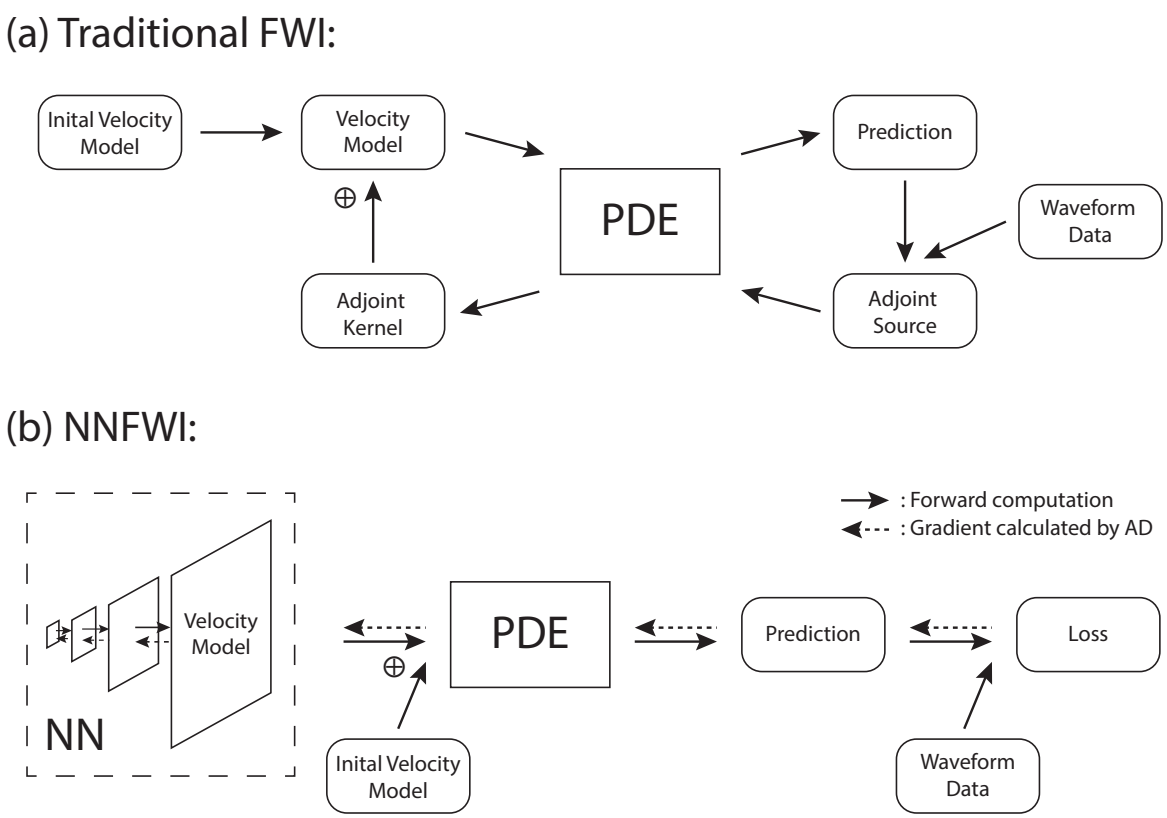

Figure 1: (a) Workflow of conventional FWI based on the adjoint-state method. (b) Diagram of NNFWI, which combines neural networks and PDEs of seismic waves. The input velocity to the PDEs is a direct summation of the initial velocity model and the velocity model generated by the neural network. In this way, NNFWI follows the same data format as conventional FWI and can be directly applied to conventional FWI applications. The gradients of both the neural network and PDE are calculated using automatic differentiation in NNFWI using the ADSeismic package.

and inversion for these examples. We inject a Ricker wavelet at $x = 0$ km and record the received waveforms (e.g. Figure 2b) at both sides of the domain ($x = 0$ km and $x = 1.0$ km). The whole wavefield is also plotted in Figures 2 and 3, but only the waveforms at the two receivers are used for FWI. Due to the limited receivers, the velocity model can not be well constrained by conventional FWI (Figures 2 and 3 (a)), even though the two received waveforms can be matched (Figures 2 and 3 (b)). Adding regularization is an effective approach to constrain the inversion results. We add Tikhonov regularization to the loss function, which produces a much smoother velocity profile (Figures 2 and 3 (d)). Note that the weight of regularization is ad-hoc. The inversion results could be improved by searching for a better weight or by adopting a complex weighting strategy, such as imposing strong regularization and relaxing it with further iterations. Instead of adding regularization to the loss function, we use a fully connected neural network (FC model in Table 1) to reparametrize the velocity model. The neural network learns a continuous function $\mathcal{N}(x, w)$ to represent

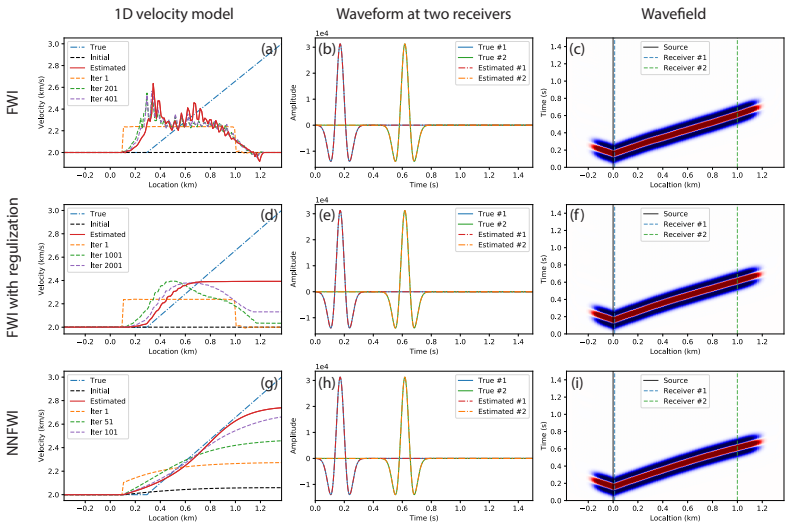

Figure 2: Inversion results of an 1D linear velocity profile. The upper panel (a-c) shows results of conventional FWI. The middle panel (d-f) shows results of FWI with Tikhonov regularization. The lower panel (g-i) shows results of NNFWI. The true and inverted velocity profiles are plotted in the left panel (a, d, g). The estimated profiles at three selected iterations are also plotted to show the convergence. Note that we only invert the velocity at $x \geq 0.1$ km to avoid extreme gradient values at the source. The fitted waveforms are plotted in the middle panel (b, e, h). The whole wavefields are plotted in the right panel (c, f, i)

either the linear or step-change profiles $m(x)$, which implicitly imposes a spatial correlation as regularization. The results demonstrate that the neural network first learns a smooth profile due to the regularization effect and then gradually adapts to complex structures such as the step-change.

3.2 Marmousi model

The Marmousi model (Versteeg, 1994) is a benchmark velocity model commonly used for evaluating FWI algorithms. The true velocity model and the 1D initial model used in this case are shown in Figure 4. We generated synthetic seismic waveform data using 8 active sources and a sequence of receivers on the surface marked by red stars and white triangles respectively in Figure 4b. The source time function is a Ricker wavelet with a peak frequency of 2.5 Hz. We conducted three experiments by adding, respectively: no random noise, random noise with $\sigma = 0.5\sigma_0$, and random noise with $\sigma = \sigma_0$ to the synthetic data, where σ is the standard deviation of Gaussian white noise and σ_0 is the standard deviation of the synthetic seismic recording. The same initial model and synthetic data were used for all experiments for both conventional FWI and NNFWI. The number of discretized grids of the velocity model in conventional FWI is 6,360, while the number of parameters of the generative neural network in NNFWI is 72,448. This means that we have over-parametrized the velocity model with the neural network; however, the inversion result of NNFWI is still robust.

Figure 5 shows the inversion results at different noise levels. We observe a significant

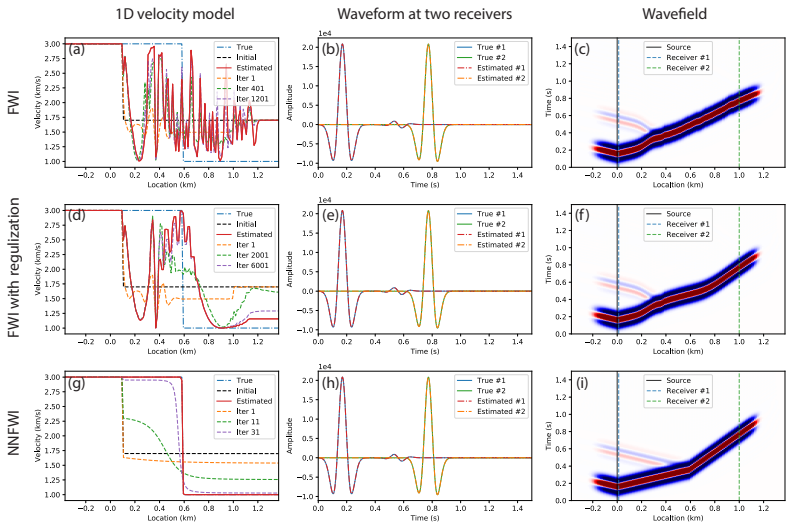

Figure 3: Inversion results of an 1D step-change velocity profile. The panels are plotted in a same way as Figure 2. Note that the initial step-change at $x = 0.1$ km occurs because we only estimate the velocity at $x \geq 0.1$ km and fix the true velocity value at $x < 0.1$ km to avoid source effects. The waveforms at two receivers are well fit for all three methods, but we observe significant differences in the wavefields (c, f, i) between the two receivers due to the incorrect velocity models.

deterioration of the results from conventional FWI with the increase of noise resulting in strong spurious fluctuations in the model. In contrast, the results of NNFWI show much less deterioration. This comparison demonstrates the regularization effect from the generative neural network with a deep image prior (Ulyanov et al., 2018). The inversion results of NNFWI become more smooth and robust compared with those of conventional FWI in the presence of noise. Figure 8a and b show the change of loss functions of the two methods. Because of the regularization effect of NNFWI, its loss (labeled as "NNFWI + Adam") is higher than that of conventional FWI (labeled as "FWI + BFGS") (Figure 8a) and the resolutions is also a bit lower than conventional FWI (Figure 5a and b) for data with no random noise. However, for data with $0.5\sigma_0$ random noise (Figure 8b), both methods converge to a similar loss, but NNFWI gives a much better recovery of the true model (Figure 11c and d). In addition to inversions using Adam for NNFWI and BFGS for conventional FWI, we have also tested inversions using BFGS for NNFWI and Adam for conventional FWI. The loss functions are also plotted in Figure 8a. Both BFGS and Adam methods work well for conventional FWI. Adam proves to be more effective for NNFWI due to the optimization of the generative neural network. Thus, we choose the BFGS method for conventional FWI and the Adam method for NNFWI as the default setting in this paper.

The inversion result of FWI directly depends on the signal frequency band (Bunks et al., 1995; Virieux & Operto, 2009). The lack of low frequency content will accentuate the cycle-skipping issue, i.e., the optimization becomes trapped at a local minimum when the predicted waveform is mismatched by more than a half cycle from the observed waveform. The regularization effect of NNFWI may help mitigate the cycle-skipping issue. To explore

this effect, we increase the peak frequency of the Ricker wavelet to 6 Hz and apply a 6 Hz high-pass filter to suppress the low frequency content of the waveforms. Figure 6 shows the source wavelet and its frequency spectrum after filtering. Here we directly inject this source wavelet for inversions instead of using a multi-scale approach (Bunks et al., 1995), which filters the source wavelet from low frequency to high frequency to mitigate cycle-skipping. The discretization is refined to ensure numerical stability (Figure 7a and b). With this source wavelet, conventional FWI fails to invert the true velocity model (Figure 7c); however, NNFWI still recovers most of the velocity model with some distortions at the edges (Figure 7d). The loss function of NNFWI continuous to decrease, while the loss of conventional FWI plateaus (Figure 8c). We also extract the intermediate inversion results when the loss decreases to 10% to compare the optimization processes (Figure 9a and b). The result of NNFWI contains more smooth and sparse patterns due to the spatial correlation imposed by the convolutional neural networks in NNFWI. This regularization effect helps to mitigate the effect of local minima and improves the convergence.

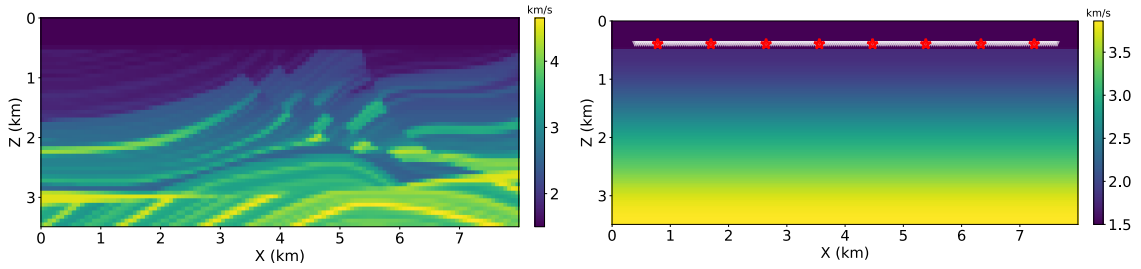

Figure 4: The Marmousi velocity model: (a) the ground-truth model for generating synthetic data; (b) the 1D smoothed initial model for inversion. The source locations (red stars) and the receiver locations (white triangles) are plotted in (b).

3.3 2004 BP model

The 2004 BP benchmark model (Billette & Brandsberg-Dahl, 2005) contains complex salt bodies, which are difficult imaging targets for FWI because of cycle-skipping and amplitude discrepancy cased by sharp contrasts and large-scale salt bodies (Z. Zhang et al., 2018). We extract the left part of the original 2004 BP benchmark model (Figure 10), which is based on a geological cross section through the Western Gulf of Mexico. We use the default Ricker wavelet with a peak frequency of 1.2 Hz for this case. Conventional FWI without regularization fails to recover the shape of the salt body (Figure 11a). In contrast, NNFWI correctly images the salt body, especially the left U-shaped target (Figure 11b). The inversion results with random noise are also plotted in Figure 11. Because conventional FWI is trapped in a local minimum, its loss function is much higher than that of NNFWI (Figure 8d). To gain insight into the model optimization trajectories, we plot the intermediate inversion results when the loss function is reduced by 10% of its initial value in Figure 9c and d. The result of conventional FWI shows that most of the updates focus on shallow layers near the surface. NNFWI, in contrast, updates over a much broader depth range and recovers a

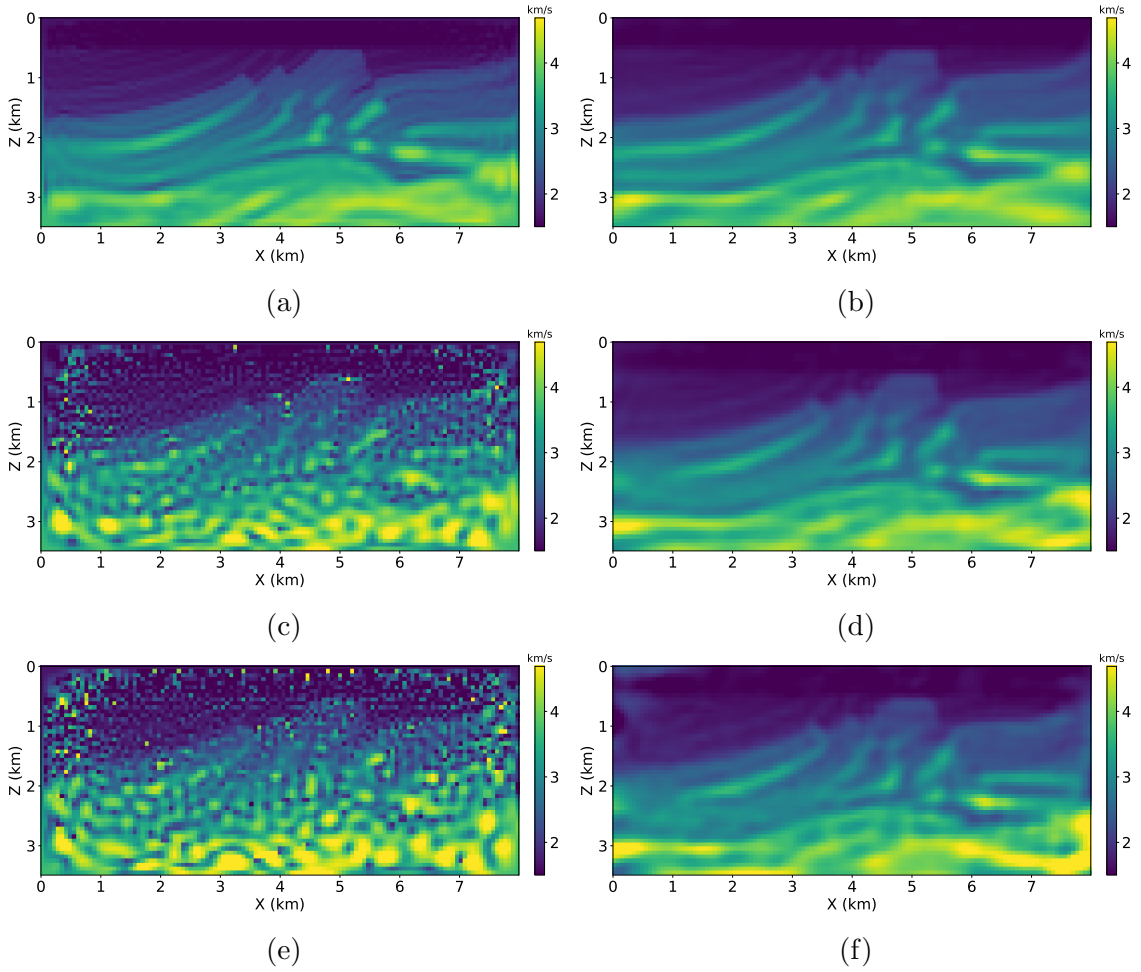

Figure 5: Inversion results of conventional FWI and NNFWI on the Marmousi benchmark model. The left panel: conventional FWI results based on data with (a) no random noise, (c) random noise with $\sigma = 0.5\sigma_0$, and (e) random noise with $\sigma = \sigma_0$. The right panel: NNFWI results based on data with (b) no random noise, (d) random noise with $\sigma = 0.5\sigma_0$, and (f) random noise with $\sigma = \sigma_0$.

smooth and approximate shape of the salt body. This difference demonstrates the spatial regularization from convolutional neural networks, which applies convolutional filters across the whole domain, so that the updates of convolutional kernels affect the entire generated velocity model.

3.4 Uncertainty and Computational Analysis

To analyze the uncertainty in the inversion results, which is a challenging task for conventional FWI, we conducted another experiment by adding dropout layers to the generative neural network (Table 1). Dropout was initially proposed to prevent over-fitting in neural networks by randomly setting a proportion of neurons and their connections to zero (Srivastava et al., 2014). Gal & Ghahramani (2016) demonstrated that dropout in neural networks

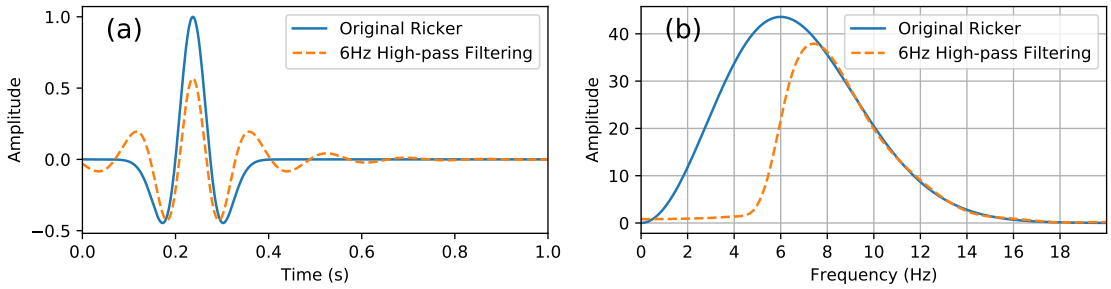

Figure 6: Modified Ricker wavelet: (a) the waveform of original Ricker and after 6 Hz high-pass filtering; (b) the frequency spectrum before and after high-pass filtering.

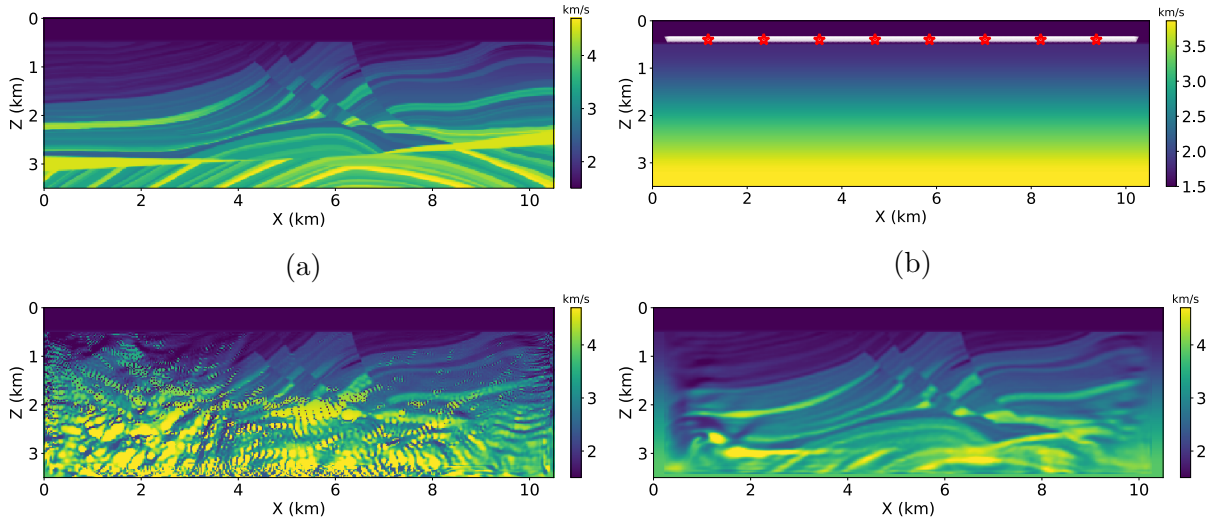

Figure 7: Inversion results with a high-frequency Ricker wavelet. (a, b) The refined Marmousi model and initial model for inversion. (c) The inversion results of conventional FWI. (d) The inversion results of NNFWI.

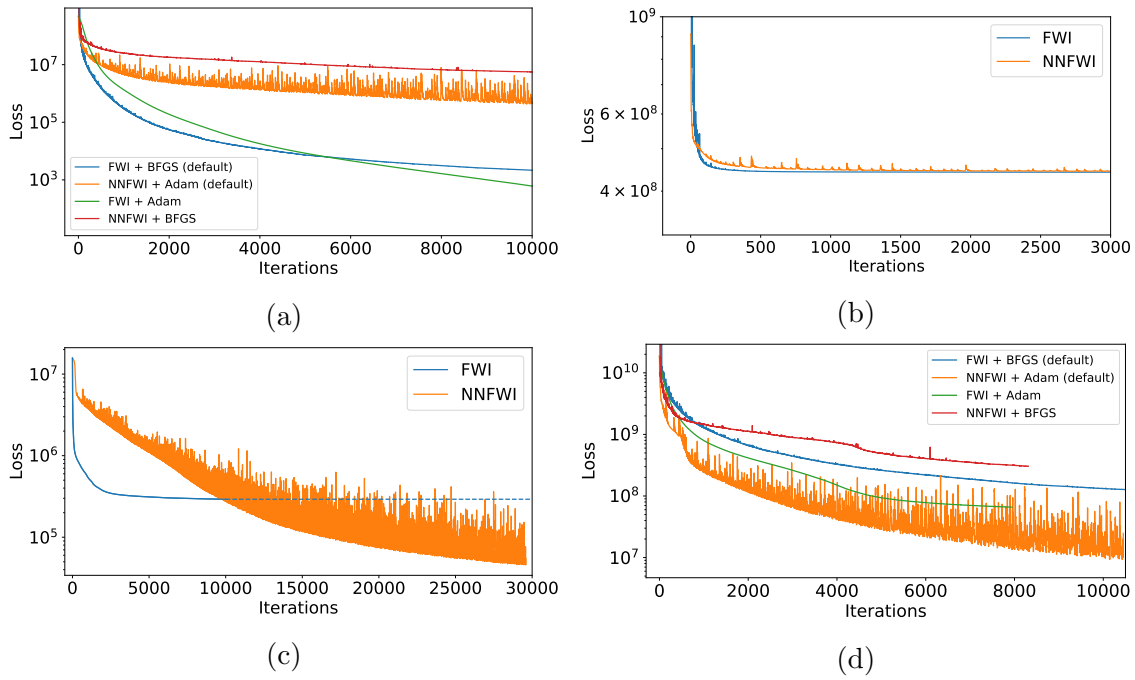

Figure 8: Loss functions: (a) The inversion of Marmousi model with no random noise. (b) The inversion of Marmousi model with random noise ($\sigma = 0.5\sigma_0$). (c) The inversion of Marmousi model with a high-frequency Ricker source wavelet and no random noise. The dashed part of FWI loss are from extrapolation since the loss has already plateaued. (d) The inversion of 2004 BP model with no random noise (explained in Section 3.3). We have tested both BFGS and Adam optimization methods for conventional FWI and NNFWI. Adam proves to be more effective for NNFWI because of the optimization of the generative neural network. We use the BFGS method for conventional FWI and the Adam method for NNFWI as the default settings in this paper.

can also be interpreted in a Bayesian framework to estimate model uncertainty. We ran 100 Monte Carlo samplings based on the optimized neural network model to calculate the standard deviation of the sampled velocity models. Figure 12a and c show the estimated velocity models, which are similar to the inversion results in Figure 5b and Figure 11b. Figure 12b and d show the estimated uncertainty of the inverted models. Because the sampling process only requires inference using the generative neural network, uncertainty estimation is orders of magnitude faster than other sampling-based methods that require solving the PDEs. We note that the estimated uncertainty by dropout in our experiments correlates with the value of the inverted velocity model. Due to the lack of uncertainty benchmarks in FWI, more research is needed to verify the accuracy of uncertainty estimation by dropout.

Finally, we address the computational demands of NNFWI. Based on the compute times per integration on both CPUs and GPUs in Table 2, NNFWI introduces negligible additional computation compared with conventional FWI. This is because generating the velocity model using neural networks is much less expensive than solving the PDEs. Moreover, the optimization of the neural network and PDEs are done simultaneously with automatic differentiation without extra optimization loops as used in dictionary learning (L. Zhu et al.,

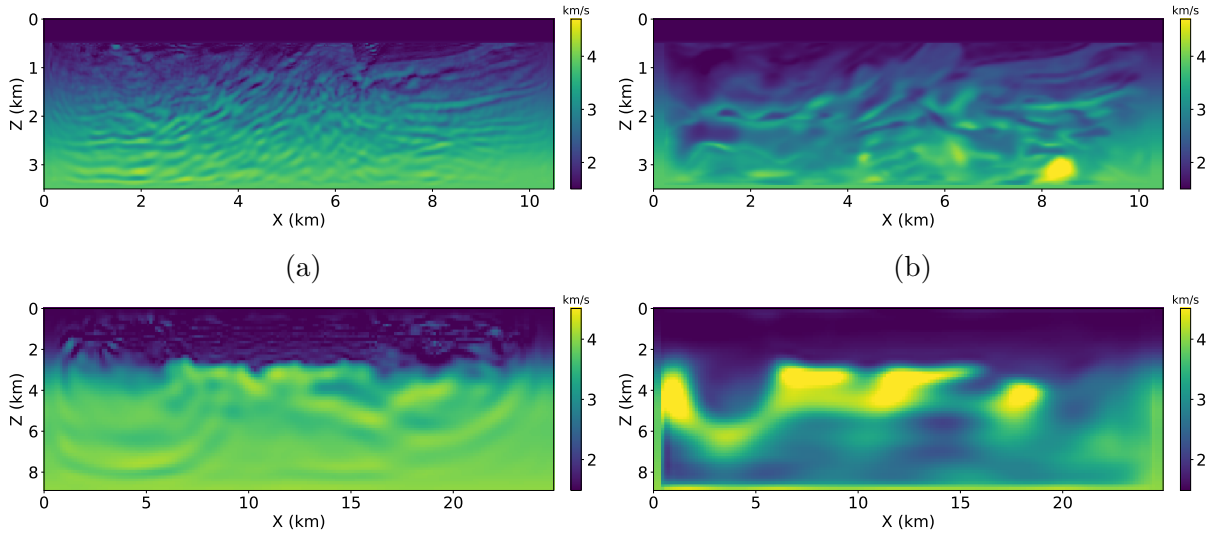

Figure 9: Intermediate inversion results at 10% initial loss. The upper panel shows the inversion results of (a) conventional FWI and (b) NNFWI on the Marmousi model (explained in Section 3.2). The lower panel shows the inversion results of (c) conventional FWI and (d) NNFWI on the 2004 BP benchmark model (explained in Section 3.3).

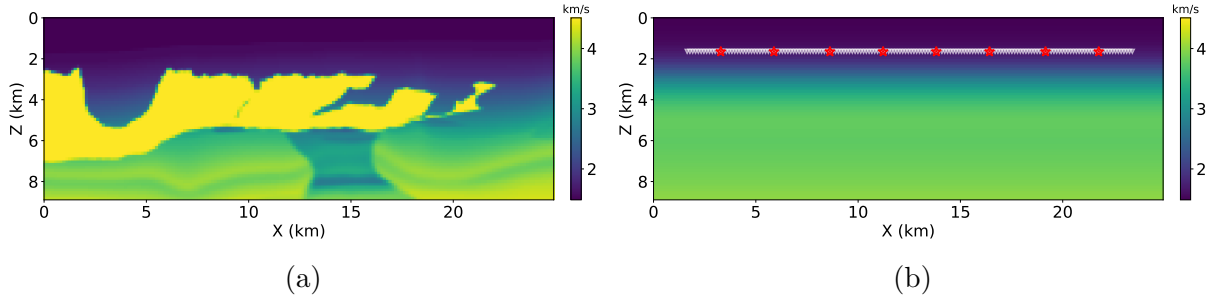

Figure 10: The 2004 BP benchmark model: (a) the ground-truth model for generating synthetic data; (b) the 1D smoothed initial model for inversion. The source locations (red stars) and the receiver locations (while triangles) are plotted in (b).

2017). The computational efficiency of neural networks allows the exploration of deeper and more complex neural network architectures in future research.

4 Discussion

NNFWI combines deep neural networks and PDEs for FWI. Compared with methods relying on only neural networks for direct inverse prediction, NNFWI can use the physical information represented by PDEs such that its accuracy is similar to that of conventional FWI. Moreover, direct inverse prediction relies on training on a large number of data pairs of velocity model and corresponding seismic data, which is not always available, so it may

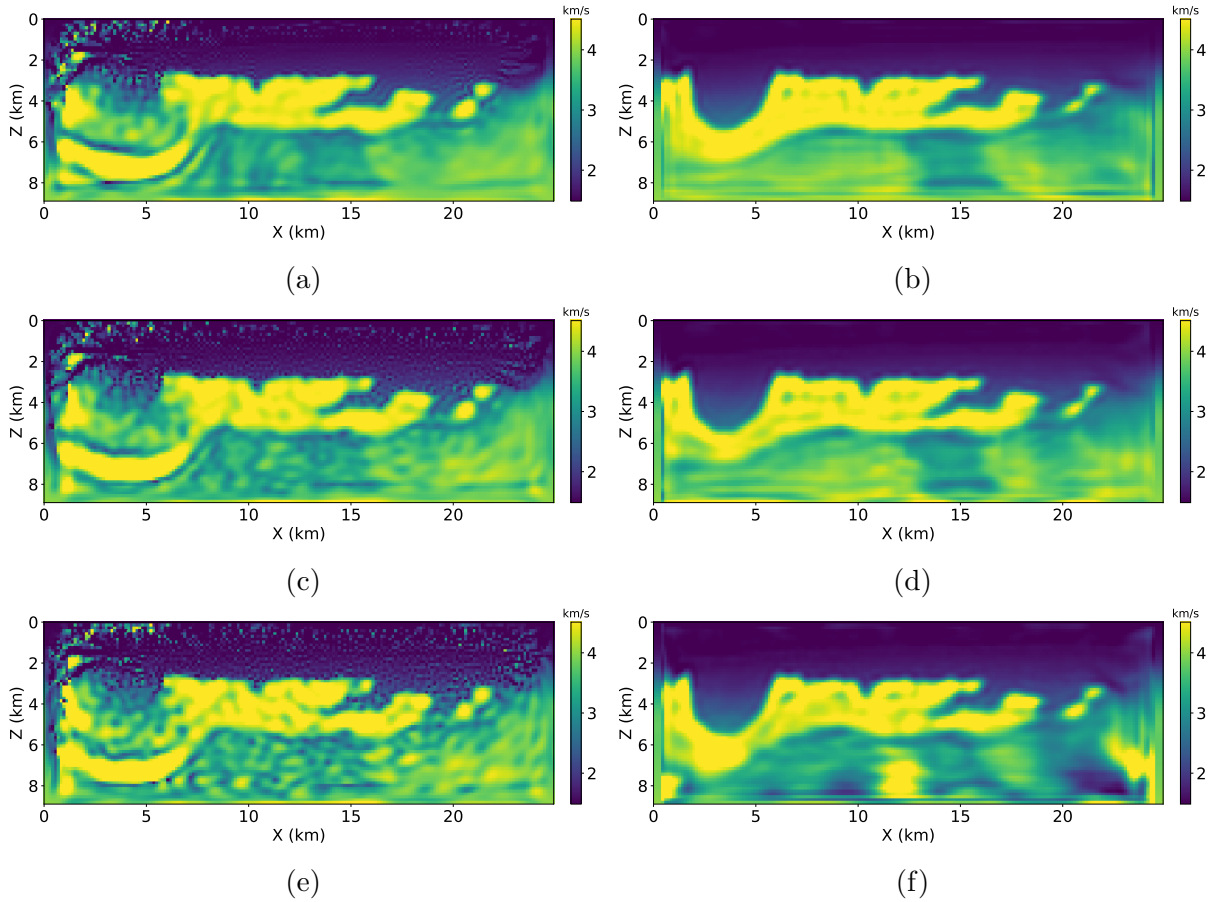

Figure 11: Inversion results of conventional FWI and NNFWI on a subset of the 2004 BP benchmark model. The left panel: conventional FWI results based on data with (a) no random noise, (c) random noise with $\sigma = 0.5\sigma_0$, and (e) random noise with $\sigma = \sigma_0$. The right panel: NNFWI results based on data with (b) no random noise, (d) random noise with $\sigma = 0.5\sigma_0$, and (f) random noise with $\sigma = \sigma_0$.

be susceptible to poor generalization when data does not follow a similar distribution to the training data. NNFWI does not need extra datasets for training the neural network. It applies to the same data settings as conventional FWI. Because the computational cost of deep neural networks is much less than PDEs, NNFWI adds little extra computational cost compared with conventional FWI. The built-in GPU acceleration of deep learning frameworks further speeds up the simulation and optimization processes of NNFWI (W. Zhu et al., 2020).

In conventional FWI, adding additional degrees of freedom by extending velocity into non-physical dimensions shows promise in overcoming local minima and improving the inversion results (Symes, 2008; Biondi & Almomin, 2014; Barnier et al., 2018b). The generative neural network in NNFWI over-parametrizes physical velocity models. Although theories of deep learning are still developing, over-parametrization is believed to be a key factor in the effective optimization and generalization of deep learning methods (Arora et al., 2018; Allen-Zhu et al., 2019; Cao & Gu, 2019). In this paper, we have shown that parameterizing

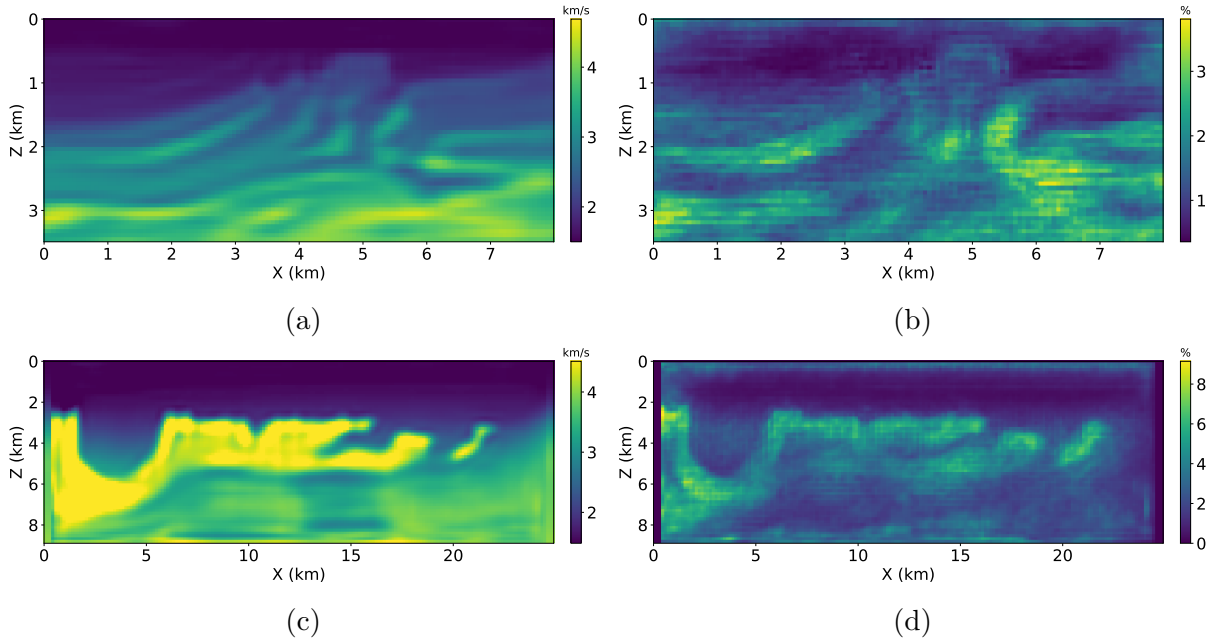

Figure 12: Uncertainty quantification of NNFWI by adding dropout layers during training: (a) the inversion result for the Marmousi model; (b) the estimated relative standard deviation by 100 bootstrap samplings with dropout; (c) the inversion result for the 2004 BP model; (b) the estimated relative standard deviation by 100 Monte Carlo samplings with dropout.

Table 2: The compute time per iteration. The CPU time is averaged over 100 iterations running on a 40-core CPU server with Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20GHz. The GPU time is averaged over 100 iterations running on a 8 GPU server with Nvidia Tesla V100-SXM2-32GB-LS.

Computing time	CPU (s)	GPU (s)
Conventional FWI	6.25	1.91
NNFWI	6.33	1.92

the velocity model by a generative neural network can be used for both regularization and uncertainty quantification for FWI. The regularization effect of NNFWI is similar to dictionary learning in FWI, which adds an extra update loop to learn sparse representations of complex features in the velocity model based on many small training patches from previous iterations. In NNFWI, the generative neural network serves as a rich feature bank to represent the complex velocity model. Compared with the complex and computationally expensive dictionary learning step, the update of the generative neural network in NNFWI directly uses the gradients calculated by automatic differentiation. In NNFWI, the gradients directly back-propagates from the loss though the PDEs and neural network layers to the weights and bias of the generative neural network. Moreover, no extra training loops of the neural networks are needed, making the training workflow of NNFWI as simple as conventional FWI.

In addition to the architecture of the generative neural network used in this paper, other

neural network architectures can also be used as the generator in NNFWI. A variety of components of NNFWI can be tuned and added, such as the number of neural network layers, the choice of activation function, batch normalization layer, recurrent neural network layer, attention layer, and a group of optimization algorithms in deep learning. We can also replace the acoustic wave equation used in this work with the elastic wave equation as the PDE solver in NNFWI for elastic FWI applications. In other words, NNFWI is not limited to the specific configuration used in this paper. Our work provides a general framework for incorporating deep neural network with PDEs for FWI applications. NNFWI also enables more advanced deep learning modeling in FWI. For example, we can use Bayesian neural networks (Kendall & Gal, 2017), which model the epistemic uncertainty in the model, as the generator in NNFWI. We can replace the input latent variable z, which is a randomized vector in this work, with the waveform observation y. In this way, NNFWI becomes similar to an autoencoder (Goodfellow et al., 2016) with the neural network as the encoder and the PDE solver as the decoder. The neural network after training learns a mapping from observations to the velocity model. These extensions are natural directions for future research.

5 Conclusions

We have introduced NNFWI, a way to integrate deep generative neural networks to the PDE-constrained optimization of FWI. NNFWI represents the velocity model of interest using a generative neural network and optimizes the weights and bias of the neural network instead. The gradients of both the neural network and PDEs are calculated using automatic differentiation, which simplifies the optimization of NNFWI without the need of pretraining or extra optimization loops. Our results demonstrate that NNFWI achieves similar accuracy as conventional FWI. More importantly, the features learned by generative neural network automatically filter out noise and introduce a regularization effect, which mitigates cycle-skipping and significantly improves the inversion results for noisy seismic recordings. NNFWI has the further advantage of providing uncertainty quantification using the dropout technique, which does not require additional computation during training, making NNFWI a much more efficient way to estimate uncertainty in the high dimensional model space of velocity estimation. NNFWI can be directly applied to the same datasets as conventional FWI to improve the inversion performance. Extensions to the generative neural work in NNFWI are a promising direction for future work.

Acknowledgments

This work is supported by the Department of Energy Basic Energy Sciences (DE-SC0020445) and the Applied Mathematics Program within the Department of Energy Office of Advanced Scientific Computing Research (ASCR) through the Collaboratory on Mathematics and Physics-Informed Learning Machines for Multiscale and Multiphysics Problems Research Center (DE-SC0019453).

References

Allen-Zhu, Z., Li, Y., & Song, Z. (2019). A Convergence Theory for Deep Learning via Over-Parameterization. *arXiv:1811.03962 [cs, math, stat]*.

Arora, S., Cohen, N., & Hazan, E. (2018). On the Optimization of Deep Networks: Implicit Acceleration by Overparameterization. *arXiv:1802.06509 [cs]*.

Asnaashari, A., Brossier, R., Garambois, S., Audebert, F., Thore, P., & Virieux, J. (2013). Regularized seismic full waveform inversion with prior model information. *GEOPHYSICS*, *78*(2), R25-R36.

Barnier, G., Biondi, E., & Biondi, B. (2018a). Full waveform inversion by model extension. In *Seg technical program expanded abstracts 2018* (pp. 1183–1187). Society of Exploration Geophysicists.

Barnier, G., Biondi, E., & Biondi, B. (2018b). Full-waveform inversion by model extension. In *SEG Technical Program Expanded Abstracts 2018* (pp. 1183–1187). Society of Exploration Geophysicists.

Barnier, G., Biondi, E., & Clapp, R. (2019). Waveform inversion by model reduction using spline interpolation. In *Seg technical program expanded abstracts 2019* (pp. 1400–1404). Society of Exploration Geophysicists.

Billette, F. J., & Brandsberg-Dahl, S. (2005). The 2004 BP Velocity Benchmark. In *67th EAGE Conference & Exhibition* (p. cp). European Association of Geoscientists & Engineers.

Biondi, B., & Almomin, A. (2014). Simultaneous inversion of full data bandwidth by tomographic full-waveform inversion. *GEOPHYSICS*, *79*(3), WA129-WA140.

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

Burstedde, C., & Ghattas, O. (2009). Algorithmic strategies for full waveform inversion: 1D experiments. *GEOPHYSICS*, *74*(6), WCC37-WCC46.

Cao, Y., & Gu, Q. (2019). Generalization Error Bounds of Gradient Descent for Learning Over-parameterized Deep ReLU Networks. *arXiv:1902.01384 [cs, math, stat]*.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *arXiv:1506.02142 [cs, stat]*.

Gebraad, L., Boehm, C., & Fichtner, A. (2020). Bayesian Elastic Full-Waveform Inversion Using Hamiltonian Monte Carlo. *Journal of Geophysical Research: Solid Earth*, *125*(3), e2019JB018428.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Guitton, A. (2012). Blocky regularization schemes for Full-Waveform Inversion. *Geophysical Prospecting*, *60*(5), 870–884.

Hu, W., Abubakar, A., & Habashy, T. M. (2009). Simultaneous multifrequency inversion of full-waveform seismic data. *GEOPHYSICS*, *74*(2), R1-R14.

Kendall, A., & Gal, Y. (2017). What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In I. Guyon et al. (Eds.), *Advances in Neural Information Processing Systems 30* (pp. 5574–5584). Curran Associates, Inc.

Kingma, D. P., & Ba, J. (2017). Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*.

Li, D., & Harris, J. M. (2018). Full waveform inversion with nonlocal similarity and model-derivative domain adaptive sparsity-promoting regularization. *Geophysical Journal International*, *215*(3), 1841–1864.

Li, Y. E., & Demanet, L. (2016). Full-waveform inversion with extrapolated low-frequency data. *GEOPHYSICS*, *81*(6), R339-R348.

Lin, Y., & Huang, L. (2015). Acoustic- and elastic-waveform inversion using a modified total-variation regularization scheme. *Geophysical Journal International*, *200*(1), 489–502.

Luo, Y., & Schuster, G. T. (1991). Wave-equation traveltime inversion. *GEOPHYSICS*, *56*(5), 645–653.

Plessix, R.-E. (2006). A review of the adjoint-state method for computing the gradient of a functional with geophysical applications. *Geophysical Journal International*, *167*(2), 495–503.

Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv:1511.06434 [cs]*.

Richardson, A. (2018). Seismic Full-Waveform Inversion Using Deep Learning Tools and Techniques. *arXiv:1801.07232 [physics]*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929–1958.

Sun, H., & Demanet, L. (2020). Extrapolated full-waveform inversion with deep learning. *GEOPHYSICS*, *85*(3), R275-R288.

Symes, W. W. (2008). Migration velocity analysis and waveform inversion. *Geophysical Prospecting*, *56*(6), 765–790.

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

Tarantola, A. (2005). *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial and Applied Mathematics.

Ulyanov, D., Vedaldi, A., & Lempitsky, V. (2018). Deep image prior. *arXiv:1711.10925 [cs, stat]*.

Versteeg, R. (1994). The Marmousi experience: Velocity model determination on a synthetic complex data set. *The Leading Edge*, *13*(9), 927–936.

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

Wu, R.-S., Luo, J., & Wu, B. (2014). Seismic envelope inversion and modulation signal model. *GEOPHYSICS*, *79*(3), WA13-WA24.

Wu, Y., Lin, Y., & Zhou, Z. (2018). Inversionet: Accurate and efficient seismic-waveform inversion with convolutional neural networks. In *SEG Technical Program Expanded Abstracts 2018* (pp. 2096–2100). Anaheim, California: Society of Exploration Geophysicists.

Wu, Y., & McMechan, G. A. (2019). Parametric convolutional neural network-domain full-waveform inversion. *GEOPHYSICS*, *84*(6), R881-R896.

Yang, F., & Ma, J. (2019). Deep-learning inversion: A next-generation seismic velocity model building method. *GEOPHYSICS*, *84*(4), R583-R599.

Zhang, P., Han, L., Xu, Z., Zhang, F., & Wei, Y. (2017). Sparse blind deconvolution based low-frequency seismic data reconstruction for multiscale full waveform inversion. *Journal of Applied Geophysics*, *139*, 91–108.

Zhang, Z., Mei, J., Lin, F., Huang, R., & Wang, P. (2018). Correcting for salt misinterpretation with full-waveform inversion. In *Seg technical program expanded abstracts 2018* (pp. 1143–1147). Society of Exploration Geophysicists.

Zhu, C., Byrd, R. H., Lu, P., & Nocedal, J. (1997). Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, *23*(4), 550–560.

Zhu, L., Liu, E., & McClellan, J. H. (2017). Sparse-promoting full-waveform inversion based on online orthonormal dictionary learning. *GEOPHYSICS*, *82*(2), R87-R107.

Zhu, W., Xu, K., Darve, E., & Beroza, G. C. (2020). A General Approach to Seismic Inversion with Automatic Differentiation. *arXiv:2003.06027 [physics]*.