CSIAM Trans. Appl. Math.
doi: 10.4208/csiam-am.SO-2024-0027

Vol. **x**, No. x, pp. 1-33
xx 2025

A Deep Learning Approach for Solving the Inverse Problem of the Wave Equation

Xiong-Bin Yan[1], Keke Wu[2], Zhi-Qin John Xu[3,4,5]
and Zheng Ma[3,4,5,6,*]

[1] *School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000, China.*
[2] *School of Mathematical Sciences and Suzhou Institute for Advanced Research, University of Science and Technology of China, Jiangsu 215217, China.*
[3] *School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai 200240, China.*
[4] *Qing Yuan Research Institute, Shanghai Jiao Tong University, Shanghai 200240, China.*
[5] *Institute of Natural Sciences, MOE-LSC, Shanghai Jiao Tong University, Shanghai 200240, China.*
[6] *CMA-Shanghai, Shanghai Jiao Tong University, Shanghai 200240, China.*

Received 24 June 2024; Accepted 13 April 2025

Abstract. Full-waveform inversion is a powerful geophysical imaging technique that infers high-resolution subsurface physical parameters by solving a non-convex optimization problem. However, due to limitations in observation, e.g. limited shots or receivers, and random noise, conventional inversion methods are confronted with numerous challenges, such as the local-minimum problem. In recent years, a substantial body of work has demonstrated that the integration of deep neural networks and partial differential equations for solving full-waveform inversion problems has shown promising performance. In this work, drawing inspiration from the expressive capacity of neural networks, we provide a new deep learning approach aimed at accurately reconstructing subsurface physical velocity parameters. This method is founded on a re-parametrization technique for Bayesian inference, achieved through a deep neural network with random weights. Notably, our proposed approach does not hinge upon the requirement of the labeled training dataset, rendering it exceedingly versatile and adaptable to diverse subsurface models. Furthermore, uncertainty analysis is effectively addressed through approximate Bayesian inference. Extensive experiments show that the proposed approach performs noticeably better than existing conventional inversion methods.

AMS subject classifications: 34K28, 35R30, 62M45

Key words: Full-waveform inversion, Bayesian inference, re-parametrization technique, deep neural network.

---

[*] Corresponding author. *Email addresses:* `yanxb@lzu.edu.cn` (X. Yan), `wukekever@ustc.edu.cn` (K. Wu), `xuzhiqin@sjtu.edu.cn` (Z. Xu), `zhengma@sjtu.edu.cn` (Z. Ma)

1 Introduction

In geophysics, seismic waveform inversion is frequently employed to obtain quantitative estimates of subsurface properties that can accurately predict the observed seismic data. The reconstruction of subsurface properties is a profoundly non-linear inverse problem. There are various techniques available to address the seismic inverse problem, such as velocity analysis of stacked seismic traces [2], migration-based travel-time approaches [6,46], Born approximation [16,26], and full-waveform inversion (FWI) [18,33,38]. Distinguished from other approaches, full-waveform inversion excels in the ability to deduce high-resolution subsurface structures. This is achieved through an iterative process that involves aligning observed and simulated seismograms while harnessing comprehensive wavefield information. This superiority stems from its capacity to fully exploit the informational richness of recorded seismic data, encompassing both amplitude and travel-time components.

In the context of full-waveform inversion, the typical approach for minimizing the objective function involves the use of gradient descent methods. These methods require the explicit computation of gradients pertaining to the cost function with respect to the velocity model. Nevertheless, the application of gradient descent methods to the full-waveform inversion objective function poses considerable challenges. This is primarily due to the presence of numerous local minima, a consequence of the inherent high nonlinearity and ill-posed nature of the problem [36]. Over the past few decades, researchers have developed numerous approaches aimed at enhancing the effectiveness of FWI. These approaches encompass the integration of multiple data components [4], the introduction of regularization terms [1,44], the regularization by discretization [11,14], and the adoption of novel objective functions [25,28,45].

Deep learning, primarily in the form of deep neural networks (DNNs), has gained significant attention in the fields of science and engineering. Its versatile applications span a wide range, including image classification/recognition [49], shape representation [42], and natural language processing [7]. Similarly, this method has also found extensive application in the solution of inverse problems and has demonstrated performance surpassing that of traditional methods. Examples include CT reconstruction [8,37], image processing [5,34], MRI reconstruction [17], and more.

A strong effort has been made in recent years to facilitate deep learning application in geophysics, including studies focused on fault detection [39], random noise attenuation [31], and so on. Some researchers have also attempted to design a DNN architecture that directly maps seismic data to subsurface models in a fully data-driven manner. Numerical experiments [9,19,40,43,48] have demonstrated that the inverse operator for full-waveform inversion can be acquired by training convolutional neural networks (CNNs) using datasets composed of wavefield and velocity model pairs. Nonetheless, it is important to note that the ill-posed and complex nature of seismic inversion poses challenges for training CNNs that are both sufficiently generalized and robust for inversion tasks.

This challenge is exacerbated by the need for extensive datasets containing a diverse range of subsurface model/seismogram pairs. In practical scenarios, obtaining such data can be a formidable task.

Compared to purely data-driven methods, FWI, as a substantial optimization problem, exhibits enduring predictive capabilities and robust generalization. The key to its prowess lies in its foundation on fundamental physical principles, particularly those governing wave propagation. Therefore, it becomes imperative to incorporate equation-based insights when harnessing deep learning techniques to tackle the challenges posed by FWI. Rasht-Behesht *et al.* [29] propose a new approach to solve full-waveform inversion by physics-informed neural networks (PINNs), it parameterizes the velocity model as a neural network and further jointly optimizes the solutions of equations and the velocity model. Sun *et al.* [32] develop a theory-guided system and choose an RNN as the most promising framework. When the forward propagation of information through the network mirrors the forward propagation of a wavefield through a heterogeneous medium, the associated RNN training process can be viewed as a form of waveform inversion. [15,41] parameterize the velocity parameters by a convolutional neural network (CNN) such that the inversion amounts to reconstructing the weights of CNN. While these equation-informed unsupervised deep learning methods have made substantial advancements in the field of geophysical imaging, enhancing the numerical results of inversions, they still face challenges in accurately recovering the physical parameters of deeper subsurface layers, especially in complex geological structures. Therefore, the objective of this paper is to tackle the challenge and achieve accurate geophysical imaging by combining deep neural networks and FWI.

In this study, we introduce a new deep-learning approach designed to tackle the complex problem of 2D full-waveform inversion. This inverse problem is distinguished by its intrinsic nonlinearity and ill-posed characteristics, making it a formidable challenge when employing traditional inversion methodologies. Our approach is grounded in a reparametrization technique for Bayesian inference, leveraging a deep neural network with random weights. By integrating likelihood functions contingent on physical equations with prior velocity model information, we formulate the loss function used for optimizing the neural network. To incorporate prior information into the neural network, we introduce a pretraining method that enables the neural network to learn an initial range of velocity parameter values, thus avoiding optimization from scratch during the inversion process.

However, it is noteworthy that our proposed deep learning inversion method does not require a large amount of training data. Instead, it relies solely on an initial model and observed data. The uncertainty in inversion results using our proposed method can be easily obtained by employing Bayesian inference with the incorporation of dropout neurons. To showcase the inversion capabilities of our algorithm, we conduct an extensive series of numerical experiments, providing compelling evidence that highlights the feasibility and robustness of our proposed inversion methodology. These experiments convincingly demonstrate our approach's ability to produce accurate inversions for the inherently ill-posed inverse problem investigated in this study, effectively mitigating the challenges posed by local minima during the full-waveform inversion process.

Main contributions are summarized as follows:

- We propose a new deep learning approach that transforms the original grid-based reconstruction problem into an optimization problem for neural network parameters based on a physics equation and measurement data. Notably, the optimization of these network parameters requires no paired data whatsoever.
- By combining statistical inversion methods, neural network re-parameterization techniques, and variational Bayesian inference, we deduce an objective function for optimizing neural network parameters.
- We introduce a pretraining method to avoid initializing neural network parameters from scratch during the inversion process.
- We carry out an extensive series of numerical experiments, and the results from these experiments demonstrate that our approach surpasses the state-of-the-art inversion methods. In particular, our inversion method excels in accurately imaging complex subsurface structures and showcases robustness in the presence of measurement noise.

The paper is organized as follows. In Section 2, we introduce the problem setup of full-waveform inversion. Section 3 describes the proposed method and algorithm, and the neural network architecture and implementation details of our approach. In Section 4, we experimentally compare our method against traditional FWI and a deep learning method by different benchmark models. Finally, we conclude our study in Section 5.

2 Full-waveform inversion

In this paper, we consider the acoustic wave equation, which characterizes the propagation of pressure or sound waves in either fluid or solid media. Specifically, we focus on the governing equation for acoustic waves in an isotropic medium with uniform density, given by

$$\begin{cases} \dfrac{1}{v(x)^2}\dfrac{\partial^2 u(x,t)}{\partial t^2}=\nabla^2 u(x,t)+s(x,t;\xi), \\ u(x,0)=0, \\ u_t(x,0)=0, \end{cases} \tag{2.1}$$

where t is time, x is the spatial location, s is source function, and v is the velocity map for the subsurface medium. Here, we use a point source generated by a Ricker wavelet, represented as follows:

$$s(x,t;\xi)=s_0(t)\delta(x-\xi),$$

where $s_0(t)=(1-2\pi^2 f_0^2 t^2)e^{-\pi^2 f_0^2 t^2}$ is the amplitude of the Ricker wavelet with frequency f_0, δ denotes the Dirac delta function, and the ξ is the predetermined horizontal location of the source. To simulate a realistic application, we applied the reflection boundary condition to the top surface of the physical domain. For the remaining boundaries, we implement absorbing boundary layers to effectively simulate wave propagation in an unbounded medium. To achieve this, we employ a perfectly matched layer (PML) approach [22], which provides absorbing boundary conditions along the left, right, and bottom sides. To show the dependence of the solution u on the parameters v, ξ, we will use $u(x,t;v,\xi)$ to represent the solution to Eq. (2.1).

Full-waveform inversion is primarily concerned with the recovery of the subsurface velocity, denoted as v, from the observed wavefield, represented by u. In practical applications, we face limitations on both the number of available receivers and their placement, typically constrained to the surface. To achieve an accurate reconstruction of the subsurface velocity, it is common practice to employ multiple sources to generate the surface wavefield. Specifically, we consider a scenario with n_s sources, denoted as $s(x,t;\xi_s)$, where $s=1,\dots,n_s$. For each source function $s(x,t;\xi_s)$, we denote the corresponding wavefield as $u(x,t;v,\xi_s)$. The observed wavefield is collected only at the surface, with a total of n_r receivers positioned at locations denoted as x_r, where $r=1,\dots,n_r$. A conceptual illustration of full-waveform inversion is provided in Fig. 1.

In this paper, we define our computational domain as $\Omega=[0,L_x]\times[0,L_z]\subset\mathbb{R}^2$. We discretize this domain and represent the discrete coordinates as (x_i,z_j) $(i=1,\dots,m_x$, $j=1,\dots,m_z)$. We denote $v=\{v(x_i,z_j)\}_{i=1,\dots,m_x,\ j=1,\dots,m_z}$, i.e. $v\in\mathbb{R}^{m_x\times m_z}$. Given a velocity model v and a source function $s(x,t;\xi_s)$, we solve Eq. (2.1) to compute the observed wavefield $u_s(x_r,t;v,\xi_s)=Ru(x,t;v,\xi_s)$, where the operator R calculates the wavefield $u(x,t;v,\xi_s)$ at a receiver location x_r. Consequently, we obtain the observed data

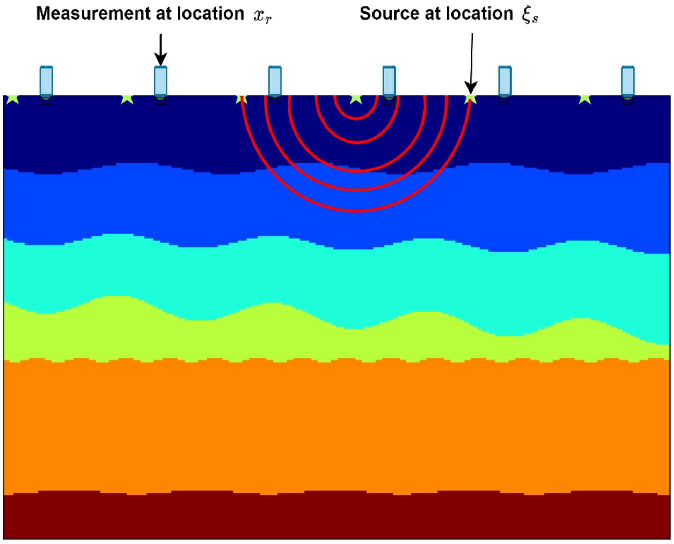


Figure 1: A conceptual illustration of full-waveform inversion. The signal emitted by the sources (Star) is recorded at the sensors (Cylinder).

$d_s(x_r,t)=u_s(x_r,t;v,\xi_s)+\eta_{s,r}$, with $\eta_{s,r}\sim N(0,\Sigma)$. Thus, the goal of l_2 norm full-waveform inversion is to recover the velocity v by minimizing the following cost function:

$$\min_v L(v)=\min_v\left\{\sum_{s=1}^{n_s}\sum_{r=1}^{n_r}\|u_s(x_r,t;v,\xi_s)-d_s(x_r,t)\|^2\right\}.$$

Additionally, we introduce the forward operator $F(v)$, defined as

$$F(v)=u, \tag{2.2}$$

where $v\in\mathbb{R}^{m_x\times m_z}, u\in\mathbb{R}^{n_s\times n_r\times n_T}$. We can then express the general formulation of the observed data as

$$d=F(v)+\eta, \tag{2.3}$$

where $d,\eta\in\mathbb{R}^{n_s\times n_r\times n_T}, \eta\sim N(0,\Sigma)$.

In general, full-waveform inversion based on the l_2-norm relies heavily on the accuracy of the initial velocity model, primarily due to the profound nonlinearity of FWI. The optimization process of l_2-norm FWI is susceptible to getting stuck in local minima caused by the cycle skipping problem. To address this issue, numerous researchers have explored alternative loss functions, such as l_1 norm, Huber and hybrid l_1-l_2 norm [3]. Additionally, the use of the Wasserstein metric, based on optimal transport, has shown promise in FWI and yielded competitive results. Nevertheless, despite these advancements, obtaining accurate inversion results for complex subsurface structures remains an ongoing challenge. Therefore, there is a compelling need to investigate more accurate numerical inversion methods aimed at addressing the challenge of solving the full-waveform inversion problem in complex underground structures. Moreover, addressing the quantification of uncertainty in the solution to the inverse problem, particularly affected by noise during data measurement, is imperative. In this paper, we will systematically tackle this issue.

3 Our approach

In this section, we will provide detailed explanations of our approach for FWI with noisy measurements. First, by leveraging the Bayesian inversion framework and neural network re-parameterization techniques, we introduce the rationale and motivation behind this study. Secondly, by variational Bayesian inference, we formally deduce a loss function that used to optimize the neural network parameters. Thirdly, to encode the prior information of the velocity parameters into the neural network, we propose a pretraining method for the neural network. Subsequently, using the trained neural network, we propose a conditional mean estimator for estimating velocity parameters v. Finally, we provide details regarding the structure of the neural network used to parameterize the velocity parameter v, along with experimental specifics.

3.1 Motivations

In this paper, we propose a new deep learning approach to tackle the intricate nonlinear inverse problem. Our method is grounded in a re-parametrization technique using deep neural networks for Bayesian inference. Within the Bayesian framework, we encapsulate our prior beliefs about the unknown velocity, denoted as v in the prior probability distribution, which is represented as $p(v)$. In addition, we assume that the unknown parameter v is statistically independent of the noise η. Then, the distribution of the v conditioned on the measurement data d, i.e. the posterior distribution $p(v|d)$ follows the Bayes' rule

$$p(v|d)=\frac{1}{p(d)}p(d|v)p(v), \tag{3.1}$$

where $p(d|v)$ denotes the likelihood. Next, we define a potential function as

$$\Phi\big(F(v),d\big)=-\log p(d|v),$$

which represents the negative log-likelihood and quantifies the misfit between the observed data d and the simulated data $F(v)$, as given by Eq. (2.2), for the velocity model v (The specific form of the potential function $\Phi(F(v),d)$ used in this paper will be detailed in Eqs. (3.10), (3.11)). Consequently, the likelihood can be expressed as

$$p(d|v)\propto\exp\big(-\Phi\big(F(v),d\big)\big).$$

In conventional Bayesian inversion methods, the need for computational feasibility often leads to modeling the prior distribution $p(v)$ by a simple distribution. The oversimplified prior distribution has prompted us to explore alternative ways of representing either the prior distribution $p(v)$ or the posterior distribution $p(v|d)$. Drawing inspiration from the potent representational capabilities of deep neural networks, we propose employing a re-parametrization technique for the Bayesian inverse problem. This technique involves re-expressing the variable v using a DNN with random weights

$$v=m(z_0;\boldsymbol{\theta}), \tag{3.2}$$

where z_0 represents a fixed random tensor with the size $[1,1,m_z,m_x]$, and $\boldsymbol{\theta}$ comprises random variable. It is noteworthy that, through re-parametrization, the prior distribution $p(v)$ can become highly intricate, even when the random variable $\boldsymbol{\theta}$ is assumed by simple distributions.

Following the re-parametrization, the variable for Bayesian inference become the random weight denoted as $\boldsymbol{\theta}$ associated with the neural network. Instead of inferring the value of v from the posterior distribution (3.1), our objective now shifts to inferring the variable $\boldsymbol{\theta}$ from the posterior distribution $p(\boldsymbol{\theta}|d)$, which is defined as

$$p(\boldsymbol{\theta}|d)\propto p(d|\boldsymbol{\theta})p(\boldsymbol{\theta}), \tag{3.3}$$

where

$$p(d|\boldsymbol{\theta})\propto\exp\Big(-\Phi\big(F\big(m(z_0;\boldsymbol{\theta})\big),d\big)\Big), \tag{3.4}$$

and $p(\boldsymbol{\theta})$ represents the prior distribution for the random weight $\boldsymbol{\theta}$.

Due to the infeasibility of high-dimensional computations in solving the posterior distribution $p(\boldsymbol{\theta}|d)$ as outlined in (3.3), approximate inference methods have been developed. The central challenge lies in creating an expressive approximation to the true posterior while maintaining computational efficiency and scalability, especially within modern deep-learning architectures. Variational inference stands out as a popular deterministic approximation approach for addressing this challenge. In variational inference, the approximate posterior is assumed to be fully factorized distributions, often referred to as mean-field variational inference [13,20]. In general, mean-field variational approximation promotes computational tractability and effective optimization. However, it does have limitations in capturing the intricate structure of the true posterior [27].

In recent years, there have been numerous studies demonstrating the use of dropout regularization as an approximation in Bayesian inference models [12,21,27]. The literature highlights that inference methods based on Bayesian dropout have delivered competitive performance in predictive accuracy across various tasks, offering several promising avenues for enhancing approximate inference in Bayesian models.

In this paper, we employ the variational dropout approximation method [12], which approximates the posterior distribution $p(\boldsymbol{\theta}|d)$ using a set of approximation distributions $q(\boldsymbol{\theta}|\boldsymbol{\mu})$ parametrized by $\boldsymbol{\mu}$, i.e. we introduce a set of distributions characterized by the following relationship:

$$\boldsymbol{\theta}=\boldsymbol{\mu}\odot\boldsymbol{b}:\theta_i=\mu_i*b_i,\quad i=1,\ldots,N, \tag{3.5}$$

where μ_i represents the distribution parameter associated with θ_i, and $b_i\sim\boldsymbol{B}(p_i)$ follows a Bernoulli distribution with a probability p_i. In other words, the probability density function of b_i is defined as

$$p(b_i)=p_i^{b_i}(1-p_i)^{1-b_i},\quad b_i\in\{0,1\}.$$

The optimal parameters $\boldsymbol{\mu}$ of the distribution $q(\boldsymbol{\theta}|\boldsymbol{\mu})$ are chosen based on a metric that quantifies the difference between the approximation and the actual posterior distribution. A commonly used metric in variational inference is the Kullback-Leibler (KL) divergence. In general, the KL divergence can be expressed as

$$D_{KL}\big(q(x)\|p(x)\big)=\int q(x)\log\frac{q(x)}{p(x)}dx.$$

Thus, in this paper, we choose the optimal parameters $\boldsymbol{\mu}$ by solving the following minimization problem:

$$\min_{\boldsymbol{\mu}}D_{KL}\big(q(\boldsymbol{\theta}|\boldsymbol{\mu})\|p(\boldsymbol{\theta}|d)\big).$$

Through some detailed calculations (specific computation details will be deferred to the subsequent context), we can deduce the following optimization problem:

$$\begin{aligned}\min_{\boldsymbol{\mu}}L_{W_1}(\boldsymbol{\mu})=\min_{\boldsymbol{\mu}}E_{\boldsymbol{b}\sim\boldsymbol{B}(p)}\Bigg\{&\sum_{s=1}^{n_s}\sum_{r=1}^{n_r}W_1\Big(\mathcal{P}\big(u_s(x_r,t;m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b}),\xi_s)\big),\mathcal{P}\big(d_s(x_r,t)\big)\Big)\\&+\alpha TV\big(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\big)\Bigg\},\end{aligned} \tag{3.6}$$

where $W_1(\cdot,\cdot)$ denotes W_1 distance [47], $\mathcal{P}$ represents a scaling operator, TV represents the total variation regularization and α is a regularization parameter. Then, we denote the reconstructed velocity model by

$$v_{CM} \approx \frac{1}{M}\sum_{k=1}^{M} m(z_0;\boldsymbol{\mu}^* \odot \boldsymbol{b}_k), \quad \boldsymbol{b}_k \sim \boldsymbol{B}(p), \tag{3.7}$$

where $\boldsymbol{\mu}^*$ is a minimizer of problem (3.6). In summary, a schematic diagram of our approach is presented in Fig. 2.

Next, we will provide a detailed explanation of how to obtain the optimization problem (3.6) and how to derive the estimation formula for the reconstructed velocity model (3.7).

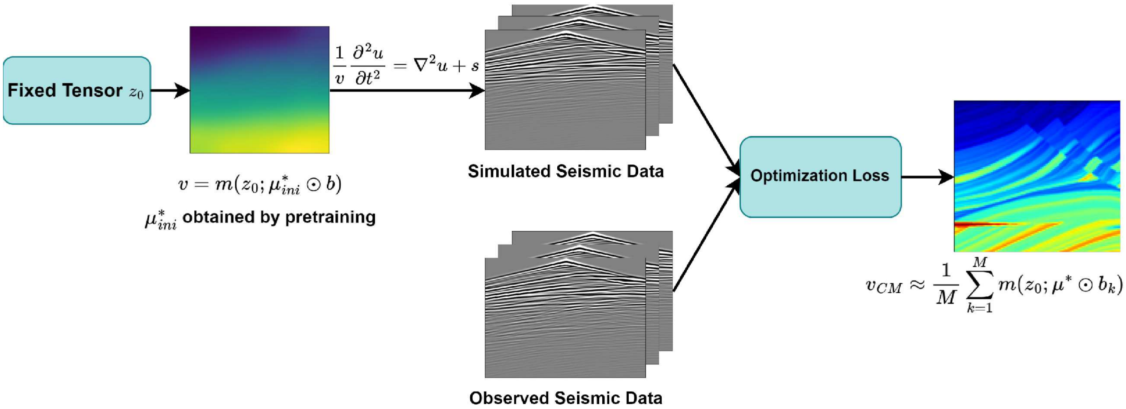


Figure 2: Schematic workflow of our proposed method for full-waveform inversion. $\boldsymbol{\mu}^*_{ini}$ is obtained through the pretraining detailed in the following sections.

3.2 Constructing the loss

As we employ $q(\boldsymbol{\theta}|\boldsymbol{\mu})$ to approximate the posterior distribution $p(\boldsymbol{\theta}|d)$, we determine the optimal approximation by minimizing the following KL-divergence:

$$\begin{aligned}
&\min_{\mu} D_{KL}\big(q(\boldsymbol{\theta}|\boldsymbol{\mu})\|p(\boldsymbol{\theta}|d)\big)\\
=&\min_{\mu}\int q(\boldsymbol{\theta}|\boldsymbol{\mu})\log\frac{q(\boldsymbol{\theta}|\boldsymbol{\mu})}{p(\boldsymbol{\theta}|d)}d\boldsymbol{\theta}\\
\propto&\min_{\mu}\left[\int q(\boldsymbol{\theta}|\boldsymbol{\mu})\log\frac{q(\boldsymbol{\theta}|\boldsymbol{\mu})}{p(\boldsymbol{\theta})}d\boldsymbol{\theta}-\int q(\boldsymbol{\theta}|\boldsymbol{\mu})\log p(d|\boldsymbol{\theta})d\boldsymbol{\theta}\right]\\
=&\min_{\mu}\big[D_{KL}\big(q(\boldsymbol{\theta}|\boldsymbol{\mu})\|p(\boldsymbol{\theta})\big)-E_{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\mu})}\log p(d|\boldsymbol{\theta})\big].
\end{aligned} \tag{3.8}$$

For the first term in formula (3.8), assuming that $p(\boldsymbol{\theta})$ follows a uniform distribution over a sufficiently large region, we can represent $p(\theta_i)$ as $1/s_i$, where s_i corresponds to the length of the domain associated with the variable θ_i. By (3.5), we have

$$q(\theta_i|\mu_i)=p_i^{\frac{\theta_i}{\mu_i}}(1-p_i)^{1-\frac{\theta_i}{\mu_i}}, \quad \theta_i\in\{0,\mu_i\},$$

where we abuse the notion $0/0=1$. Consequently, we obtain

$$\begin{aligned}D_{KL}\big(q(\boldsymbol{\theta}|\boldsymbol{\mu})\|p(\boldsymbol{\theta})\big)&=\sum_{i=1}^{N}D_{KL}\big(q(\theta_i|\mu_i)\|p(\theta_i)\big)\\&=\sum_{i=1}^{N}\int q(\theta_i|\mu_i)\log\frac{q(\theta_i|\mu_i)}{p(\theta_i)}d\theta_i\\&=\sum_{i=1}^{N}\left(\int q(\theta_i|\mu_i)\log q(\theta_i|\mu_i)d\theta_i+\int q(\theta_i|\mu_i)\log s_i d\theta_i\right)\\&=\sum_{i=1}^{N}\big((1-p_i)\log(1-p_i)+p_i\log p_i+\log s_i\big)=C_0,\end{aligned}$$

where the constant C_0 independent of the parameters $\boldsymbol{\mu}$.

By (3.4), we have

$$\log p(d|\boldsymbol{\theta})\propto-\Phi\big(F\big(m(z_0;\boldsymbol{\theta})\big),d\big),$$

then

$$\min_{\boldsymbol{\mu}}D_{KL}\big(q(\boldsymbol{\theta}|\boldsymbol{\mu})\|p(\boldsymbol{\theta}|d)\big)\propto\min_{\boldsymbol{\mu}}E_{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\mu})}\Phi\big(F\big(m(z_0;\boldsymbol{\theta})\big),d\big).$$

Using (3.5), we can deduce that

$$\begin{aligned}&\min_{\boldsymbol{\mu}}E_{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\mu})}\Phi\big(F\big(m(z_0;\boldsymbol{\theta})\big),d\big)\\=&\min_{\boldsymbol{\mu}}\int\Phi\big(F\big(m(z_0;\boldsymbol{\theta})\big),d\big)q(\boldsymbol{\theta}|\boldsymbol{\mu})d\boldsymbol{\theta}\\=&\min_{\boldsymbol{\mu}}\int\Phi\big(F\big(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\big),d\big)\boldsymbol{B}(p)d\boldsymbol{b}\\=&\min_{\boldsymbol{\mu}}E_{\boldsymbol{b}\sim\boldsymbol{B}(p)}\Phi\big(F\big(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\big),d\big).\end{aligned}$$

To mitigate the risk of potential overfitting, we introduce an additional regularization term to the estimation process. We adopt the commonly employed total variation (TV) regularization and incorporate it into the loss function. Therefore, the loss function used for training the neural network becomes

$$\min_{\boldsymbol{\mu}}L(\boldsymbol{\mu})=\min_{\boldsymbol{\mu}}E_{\boldsymbol{b}\sim\boldsymbol{B}(p)}\big\{\Phi\big(F\big(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\big),d\big)+\alpha TV\big(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\big)\big\}. \tag{3.9}$$

Next, we consider different potential functions $\Phi(F(v),d)$ and the corresponding data models. In general, l_2 norm is commonly used in full-waveform inversion as a misfit function to quantify the discrepancy between simulated and observed data. Specifically, it is defined as

$$\Phi_{L_2}\left(F\left(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\right),d\right)=\frac{1}{2}\left\|F\left(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\right)-d\right\|_{\Sigma}^{2}, \tag{3.10}$$

where $\|\cdot\|_{\Sigma}=\|\Sigma^{-1/2}\cdot\|$.

Full-waveform inversion methods based on the l_2 norm frequently encounter the issue of cycle-skipping, which can cause the optimization process to become trapped in local minima. This problem is particularly significant when an adequate initial model is unavailable or when the observed wavefield data lacks low-frequency information. To address these challenges, recent research has proposed objective loss functions based on optimal transport theory for FWI [10, 25, 28, 44, 45]. The Wasserstein distance, derived from optimal transport, has been shown to improve the landscape of the loss function, leading to more effective and robust optimization in FWI. In the following section, we provide a detailed formulation of the misfit function based on the Wasserstein distance. Let

$$\Phi_{W_p}\left(F\left(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\right),d\right)=\frac{1}{2}\sum_{s=1}^{n_s}\sum_{r=1}^{n_r}W_p^p\Big(\mathcal{P}\left(u_s\left(x_r,t;m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b}),\xi_s\right)\right),\mathcal{P}\left(d_s(x_r,t)\right)\Big), \tag{3.11}$$

where $W_p(\mathcal{P}(u_s(x_r,t;m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b}),\xi_s)),\mathcal{P}(d_s(x_r,t)))$ denotes p-Wasserstein distance. The operator $\mathcal{P}$ transforms the seismic signals into probability density functions. For instance, for $c>0$, the affine scaling operator is defined as $\mathcal{P}(d)=(d+c)/\langle d+c\rangle$, and the exponential scaling operator is given by $\mathcal{P}(d)=\exp(cd)/\langle\exp(cd)\rangle$ for a seismic signal d, where $\langle\cdot\rangle$ represents the integral operator.

For two positive functions $f(x),g(x)$ defined on a space $\Omega=[a,b]^n\subset\mathbb{R}^n$ satisfying the mass conservation assumption

$$\int_\Omega f(x)dx=\int_\Omega g(x)dx.$$

The p-Wasserstein distance defined by

$$W_p(f,g)=\left\{\min_{\gamma(x,y)\in\Pi_{f,g}}\int_\Omega\int_\Omega\gamma(x,y)\|x-y\|^p dxdy\right\}^{\frac{1}{p}},$$

where

$$\Pi_{f,g}=\left\{\gamma(x,y),\int_\Omega\gamma(x,y)dx=g(y),\int_\Omega\gamma(x,y)dy=f(x)\right\}.$$

In one dimensional case, $n=1$, the p-Wasserstein distance can be given analytically by

$$W_p^p(f,g)=\int_0^1\left|F_{cdf}^{-1}(x)-G_{cdf}^{-1}(x)\right|^p dx.$$

For $p=1,2$, it can be further simplified that

$$W_2^2(f,g)=\int_0^1\left|x-G_{cdf}^{-1}\left(F_{cdf}(x)\right)\right|^2 f(x)dx,$$

$$W_1(f,g)=\int_0^1 |G_{cdf}(x)-F_{cdf}(x)|dx,$$

where F_{cdf},G_{cdf} are the cumulative distribution functions (CDF) of f,g respectively.

In Fig. 3, we display a comparison of different misfit functions for two shift Ricker wavelets $f(t)$ and $f(t-s)$. The W_1 norm with affine scaling operator global minimum vally outperforms the convexity of l_2 norm and the W_2 norm with affine scaling and exponential scaling. This explains the advantage of Wasserstein distance as a misfit function to overcome cycle-skipping problem in full-waveform inversion. Therefore, we consider W_1 distance [47] as the measure to quantify the discrepancy between the observed and simulated data. Consequently, the corresponding loss function (3.9) is formulated as

$$\begin{aligned}\min_{\boldsymbol{\mu}} L_{W_1}(\boldsymbol{\mu}) &= \min_{\boldsymbol{\mu}} E_{\boldsymbol{b}\sim \boldsymbol{B}(p)}\left\{\Phi_{W_1}\left(F\left(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\right),d\right)+\alpha TV\left(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\right)\right\}\\ &= \min_{\boldsymbol{\mu}} E_{\boldsymbol{b}\sim \boldsymbol{B}(p)}\left\{\frac{1}{2}\sum_{s=1}^{n_s}\sum_{r=1}^{n_r} W_1\left(\mathcal{P}\left(u_s\left(x_r,t;m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b}),\xi_s\right)\right),\mathcal{P}\left(d_s(x_r,t)\right)\right)\right.\\ &\qquad\qquad \left.+\alpha TV\left(m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})\right)\right\}. \end{aligned}\tag{3.12}$$

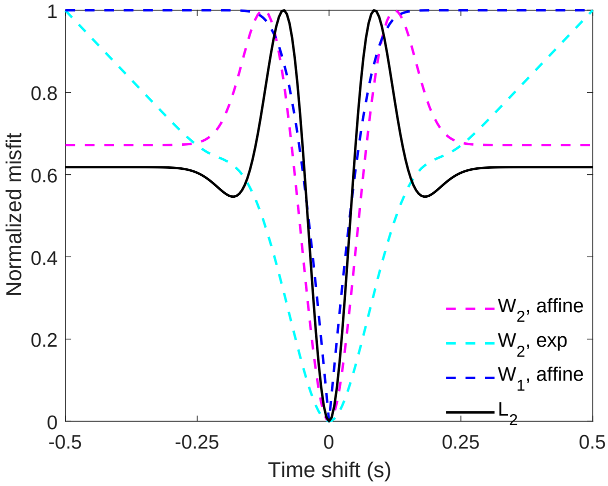


Figure 3: The different misfit functions between Ricker wavelets $f(t)$ and $f(t-s)$.

3.3 Encoding the prior

In the field of inverse problem research, the accuracy of the reconstructed solution is heavily reliant on the amount of prior information embedded during the inversion process. This prior information may encompass the smoothness, sparsity, or the choice of initial values for the inverse problem solution. Generally, the more known prior information is available, the more accurate the resulting inverse problem solution will be. Conversely, when there is limited prior knowledge about the inversion parameters, obtaining a high-precision solution for the inverse problem becomes more challenging.

In our research, full-waveform inversion is a well-known nonlinear inverse problem, which leads to a minimization problem with multiple local minima. Therefore, in order to achieve a more precise solution to this inverse problem, it is necessary for the neural network to learn some prior information. In our implementation, we empower the DNN, denoted as $m(z_0;\boldsymbol{\theta})=m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})$, to learn some prior information regarding the initial value v_{ini}. To accomplish this, we minimize the following problem:

$$\boldsymbol{\mu}_{ini}^{*}=\operatorname{argmin} J(\boldsymbol{\mu})=\operatorname{argmin} E_{\boldsymbol{b}\sim\boldsymbol{B}(p)}\|m(z_0;\boldsymbol{\mu}\odot\boldsymbol{b})-v_{ini}\|_{l_1}, \tag{3.13}$$

where $\|\cdot\|_{l_1}$ denotes the l_1 norm, and v_{ini} represents an initial model of the velocity parameter v.

3.4 Inference

After training the neural network by minimizing the loss function as specified in (3.12), we can obtain an approximation for the posterior distribution $p(\boldsymbol{\theta}|d)$, which is represented as $q(\boldsymbol{\theta}|\boldsymbol{\mu}^*)$. During the testing phase of our method, we estimate the velocity parameter v using a conditional mean estimator. Given the observation data d, the velocity estimator v_{CM} is defined as

$$v_{CM}=\int v p(v|d)dv.$$

Utilizing (3.2), we can deduce that

$$v_{CM}=\int m(z_0;\boldsymbol{\theta})p(\boldsymbol{\theta}|d)d\boldsymbol{\theta}.$$

By employing the variational inference method, where the distribution $q(\boldsymbol{\theta}|\boldsymbol{\mu}^*)$ approximates the posterior distribution $p(\boldsymbol{\theta}|d)$, then we can express

$$v_{CM}\approx\int m(z_0;\boldsymbol{\theta})q(\boldsymbol{\theta}|\boldsymbol{\mu}^*)d\boldsymbol{\theta}=\int m(z_0;\boldsymbol{\mu}^*\odot\boldsymbol{b})\boldsymbol{B}(p)d\boldsymbol{b}.$$

In practice, the integration is computed using Monte Carlo (MC) method. After the neural network is trained, we estimate the velocity parameter v by

$$v_{CM}\approx\frac{1}{M}\sum_{k=1}^{M}m(z_0;\boldsymbol{\theta}_k)=\frac{1}{M}\sum_{k=1}^{M}m(z_0;\boldsymbol{\mu}^*\odot\boldsymbol{b}_{\boldsymbol{k}}), \tag{3.14}$$

where $\boldsymbol{b}_k\sim\boldsymbol{B}(p)$.

Based on the description provided above for our inversion method, we summarize the algorithm in Algorithm 1.

Algorithm 1. Our Algorithm.

```
Inputs: $z_0, v_{ini}, \epsilon, It_{pmax}, It_{Imax}$.
Phase 1:
    Solve the problem
        $\min J(\boldsymbol{\mu}),$
    where $J(\boldsymbol{\mu})$ defined by (3.13). The iterations stops and outputs $\boldsymbol{\mu}^*_{ini}$ when $J(\boldsymbol{\mu}^*_{ini}) < \epsilon$ or reaches the maximum number of iterations $It_{pmax}$.
Phase 2:
    Let $\boldsymbol{\mu}_0 = \boldsymbol{\mu}^*_{ini}$ and solve the minimization problem (3.12), where $\boldsymbol{\mu}_0$ is a initial value of parameter $\boldsymbol{\mu}$ of the minimization problem (3.12). The iterations stop when reach the maximum number of iterations $It_{Imax}$.
Inference:
    Calculate $v_{CM}$ by (3.14).
```

3.5 Implementation details

This paper utilizes an encoder-decoder neural network [5] with skip-connections to parameterize the velocity model v. Table 1 presents a detailed overview of the network

Table 1: The DNN architecture utilized in the proposed method is described, including parameters such as the dropout ratio for Dropout, the receptive field size for max pooling, the scaling factor for upsampling, and the sequence numbers of the two blocks for concatenation (#Ch=#Channels, Par=Parameter(s)).

No	Function	#Ch	Par	No	Function	#Ch	Par
Encoder				17	Dropout+Conv.+LReLU	96	37.5%
1	-	C	-	18	Upsampling	96	2X
2	Dropout+Conv.+LReLU	48	7.5%	19	Concatenation	144	#8, #18
3	Dropout+Conv.+LReLU	48	7.5%	20	Dropout+Conv.+LReLU	96	30%
4	Max Pooling	48	2×2	21	Dropout+Conv.+LReLU	96	30%
5	Dropout+Conv.+LReLU	48	15%	22	Upsampling	96	2X
6	Max Pooling	48	2×2	23	Concatenation	144	#6, #22
7	Dropout+Conv.+LReLU	48	22.5%	24	Dropout+Conv.+LReLU	96	22.5%
8	Max Pooling	48	2×2	25	Dropout+Conv.+LReLU	96	22.5%
9	Dropout+Conv.+LReLU	48	30%	26	Upsampling	96	2X
10	Max Pooling	48	2×2	27	Concatenation	144	#4, #26
11	Dropout+Conv.+LReLU	48	37.5%	28	Dropout+Conv.+LReLU	96	15%
12	Max Polling	48	2×2	29	Dropout+Conv.+LReLU	96	15%
13	Dropout+Conv.+LReLU	48	45%	30	Upsampling	96	2X
Decoder				31	Concatenation	96+C	#1, #30
14	Upsampling	48	2X	32	Dropout+Conv.+LReLU	64	7.5%
15	Concatenation	96	#10, #14	33	Dropout+Conv.+LReLU	32	7.5%
16	Dropout+Conv.+LReLU	96	37.5%	34	Dropout+Conv.+Sigmoid	C	7.5%

architecture. The encoder (decoder) section comprises six (five) blocks, each containing a convolutional layer with dropout applied to its input. Consistent with standard practice, all convolutional layers have a kernel size of 3×3, a stride of 1, and zero padding with a length of 2. Max pooling layers are configured in CEIL mode, and the feature maps are uniformly clipped to the same size before concatenation. Bi-linear interpolation is employed for upsampling. The hyper-parameter for each Leaky ReLU (LReLU) is configured as 0.01.

4 Numerical experiments

In this section, we demonstrate the advantages of our method through three numerical tests, which include the Marmousi model, the overthrust model, and the Marmousi2 model. To evaluate the performance of our proposed method, we will use the relative l_2 error, defined as

$$\text{Relative } l_2 \text{ error} = \frac{\|\tilde{v}-v\|}{\|v\|},$$

where $\tilde{v},v$ represent the inverted velocity model and the ground truth model, respectively. Additionally, we calculate the signal-to-noise ratio (SNR) between the inverted velocity model $\tilde{v}$ and the ground truth model v using the formula

$$\mathrm{SNR}(\tilde{v},v) = 10\log_{10}\frac{\|v\|^2}{\|\tilde{v}-v\|^2}.$$

Furthermore, to further assess the similarity between the inverted velocity $\tilde{v}$ and the ground truth v, we compute the structural similarity index (SSIM) defined as

$$\mathrm{SSIM}(\tilde{v},v) = \frac{(2\mu_v\mu_{\tilde{v}}+c_1)(2\sigma_{v\tilde{v}}+c_2)}{\left(\mu_v^2+\mu_{\tilde{v}}^2+c_1\right)\left(\sigma_v^2+\sigma_{\tilde{v}}^2+c_2\right)},$$

where $\mu_v,\mu_{\tilde{v}},\sigma_v,\sigma_{\tilde{v}}$, and $\sigma_{v\tilde{v}}$ are the local mean, standard deviation, and cross-covariances for v and $\tilde{v}$, respectively. Higher SSIM and SNR values indicate a better quality of the inversion result.

In this study, to demonstrate the effectiveness of our proposed method, we conduct some comparison with state-of-the-art FWI methods. This includes two traditional inversion methods utilizing l_2 loss and the $1D-W_1$ loss [47], as well as a deep learning-based method (DNN-FWI) [15]. Additionally, in the traditional inversion methods, we introduce a total variation (TV) regularization term to improve the inversion performance, where we also use α denotes the regularization parameter. We name these two methods as $\mathrm{FWI(l_2)+TV}$ and $\mathrm{FWI(W_1)+TV}$, respectively.

To simulate wave propagation for Eq. (2.1), we use a uniform grid and a finite difference scheme with second-order accuracy in the time domain and eighth-order accuracy in the spatial domain. We choose a Ricker wavelet with a peak frequency of 5 Hz as the source wavelet. The time interval for forward simulation is set to 3 ms, and the total recording time is configured as 6 s. We position thirty equally spaced sources on the surface at $z=0$ and take the number of receivers to match the model size m_x for different velocity models. To facilitate these simulations, we leverage the Deepwave toolbox [30], which offers wave propagation modules and facilitates automatic updates of variables. To ensure a fair comparison, we fine-tune the hyper-parameters and implement each method multiple times to obtain the best solutions.

Our method is implemented in the PyTorch interface on an NVIDIA 3090 GPU graphics card with 24 G memory. In Algorithm 1, we set $It_{pmax}=10000$, $It_{Imax}=200$, and $\epsilon=1\times10^{-3}$. In phase 1 of Algorithm 1, we apply the Adam optimizer with a learning rate of 0.01 to solve the minimization problem (3.13). In phase 2 of Algorithm 1, we minimize the cost function (3.12) by Adam optimizer with learning rate 5×10^{-4}. We initialize the value of the fixed random tensor z_0 in (3.2) as a uniform distribution with a size of $[1,1,m_z,m_x]$. We take the regularization parameter $\alpha=1\times10^{-8}$ in (3.12) and the number of Monte Carlo samples $M=400$ in (3.14). Additionally, to ensure full optimization, we set the iteration steps for the other three compared methods to 1000. For two traditional inversion methods, we selected optimal regularization parameters to enhance their inversion performance. In the Marmousi model, the regularization parameters for both $\mathrm{FWI(l_2)+TV}$ and $\mathrm{FWI(W_1)+TV}$ are set to 1×10^{-8}. For the overthrust and Marmousi2 models, we set the regularization parameters for both methods to 1×10^{-10}.

Unless otherwise specified, we assume that the variance of the measurement noise is $\Sigma=\sigma^2 I$ across all experiments.

4.1 Comparison of accuracy

This subsection evaluates the accuracy of our inversion algorithm by comparing it with three other methods across two benchmark models. The assessments include evaluating inverted velocity models, examining velocity profiles at specific locations, and quantitatively comparing SNR, SSIM and relative l_2 error values.

4.1.1 Marmousi model

In this part, we validate the effectiveness of our proposed method using the Marmousi model [35]. The Marmousi model is a well-known acoustic velocity model and serves as a standard benchmark model in geophysical exploration. For our experiments, we reduce the resolution of the velocity model to $(z\times x)=(100\times310)$ with a spatial grid increment of 0.03 km. The velocity values in the model range from 1472 m/s to 5772 m/s and the true model is depicted in Fig. 4.

Fig. 4 presents a comparison of subsurface velocity models reconstructed through various inversion methods using noise-free measurements. These methods yield reasonable results overall. However, $\mathrm{FWI(l_2)+TV}$ and DNN-FWI demonstrate limited accuracy in deeper regions, contrasting with our approach and $\mathrm{FWI(W_1)+TV}$, which excel in these areas.

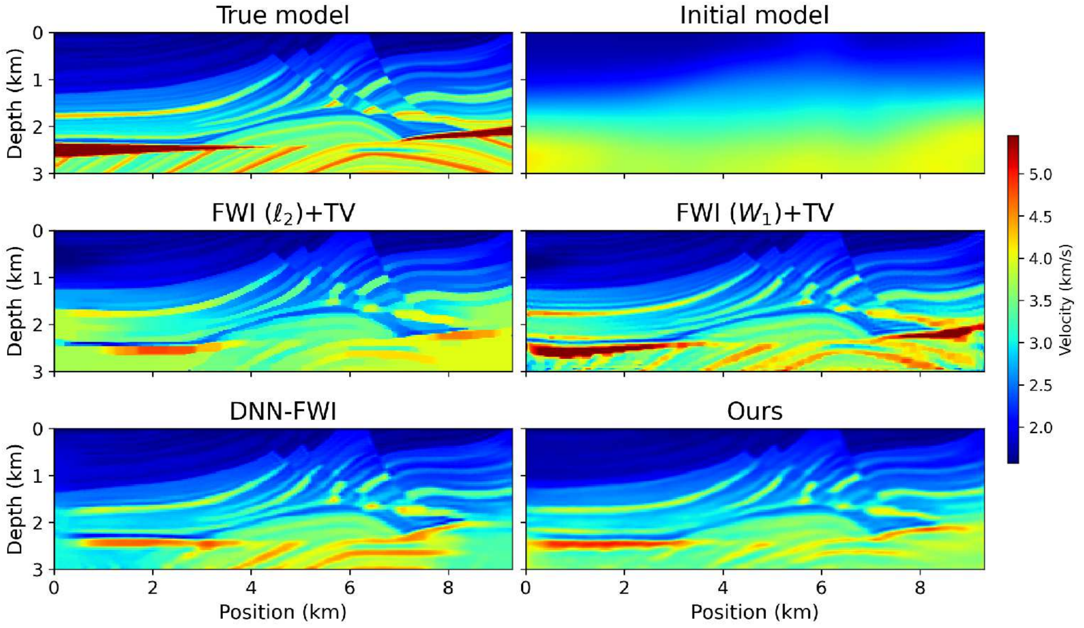


Figure 4: Inversion results for the noise-free observed data of the Marmousi model. The top row displays the true velocity model and the initial guess. The second row shows the inverted results obtained by the traditional inversion methods FWI(l_2)+TV and FWI(W_1)+TV. The third row presents the inverted results obtained through DNN-FWI and our method.

We further extract subsurface velocity model profiles at horizontal positions of $x=3.6$km, $x=5.1$km, and $x=7.2$km, as displayed in Fig. 6. This comparative analysis underscores the superior performance of our method in matching the true model. Quantitative metrics, including SNR, SSIM, and relative l_2 error (as outlined in Table 2), provide empirical support for these findings.

To showcase our method's capability in reconstructing the subsurface velocity model with noisy measurement data, we conduct a comparison of various inversion methods, as depicted in Fig. 5. The results clearly indicate that, in comparison to the other three methods, our approach yields superior numerical results for measurement data contaminated with noise. This conclusion is further supported by the vertical profiles in Fig. 6 and the quantitative metrics presented in Table 2.

Figs. 7 and 8 depict how SNR, SSIM, and relative l_2 error evolve during the optimization process for the different inversion methods, under both noisy and noise-free measurement conditions. The consistent upward trends in SNR and SSIM along with the consistent decrease in relative l_2 error of our inversion method, signify the stability of our optimization process, which is a highly desirable characteristic for practical applications. It is worth noting that the optimization process displayed in Figs. 7 and 8 exhibits notably greater stability, suggesting a potentially more favorable loss landscape for our method.

In Figs. 4 and 5, the inversion results obtained from the methods FWI(l_2)+TV and FWI(W_1)+TV show a closer match to the true velocity model in the shallow regions.

Table 2: The SNR, SSIM, and relative l_2 error (Error) for the inverted Marmousi model obtained by different approaches.

Case	Index	FWI(l_2)+TV	FWI(W_1)+TV	DNN−FWI	Ours
Noise-free	SNR	16.05	18.46	16.86	**19.67**
	SSIM	0.6833	0.8220	0.7900	**0.7987**
	Error	0.1576	0.1193	0.1435	**0.1038**
Noise ($\sigma=0.01$)	SNR	16.01	16.31	15.64	**19.51**
	SSIM	0.6801	0.7280	0.7688	**0.7900**
	Error	0.1584	0.1529	0.1653	**0.1057**

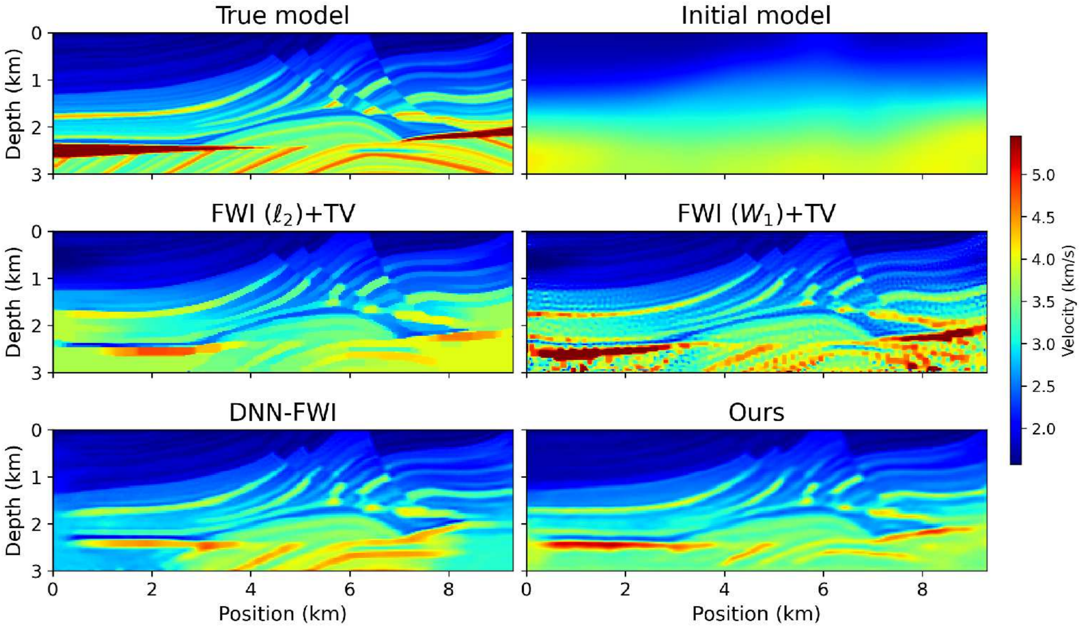


Figure 5: Inversion results for the noisy observed data of the Marmousi model ($\sigma=0.01$). The top row displays the true velocity model and the initial guess. The second row shows the inverted results obtained by the traditional inversion methods FWI(l_2)+TV and FWI(W_1)+TV. The third row presents the inverted results obtained through DNN-FWI and our method.

Additionally, these two traditional inversion methods are able to capture finer details of the true model in the Marmousi model. This finding is further confirmed by the structural similarity index measurements (SSIM) in Table 2. However, considering all the numerical results, our proposed inversion method achieves the most accurate overall inversion results.

4.1.2 Overthrust model

The overthrust model [23] portrays a stratigraphy characterized by complex thrusting, overlaying a previous extensional and rift sequence. The model exhibits varying com-

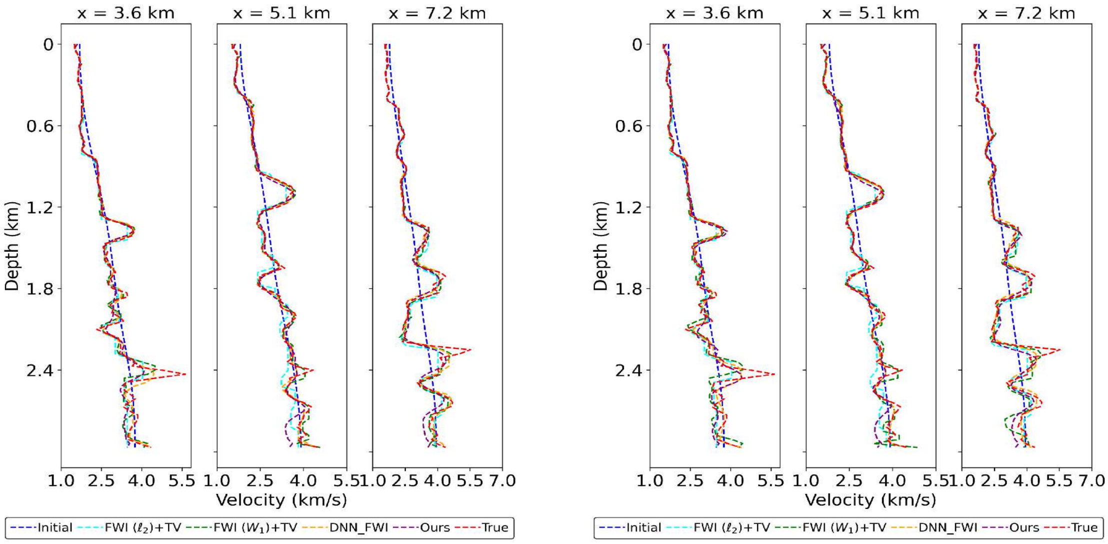


Figure 6: The Marmousi velocity profiles of the true, initial, and inverted models obtained by different approaches at three horizontal locations. Left: noise-free observed data. Right: noisy observed data ($\sigma = 0.01$).

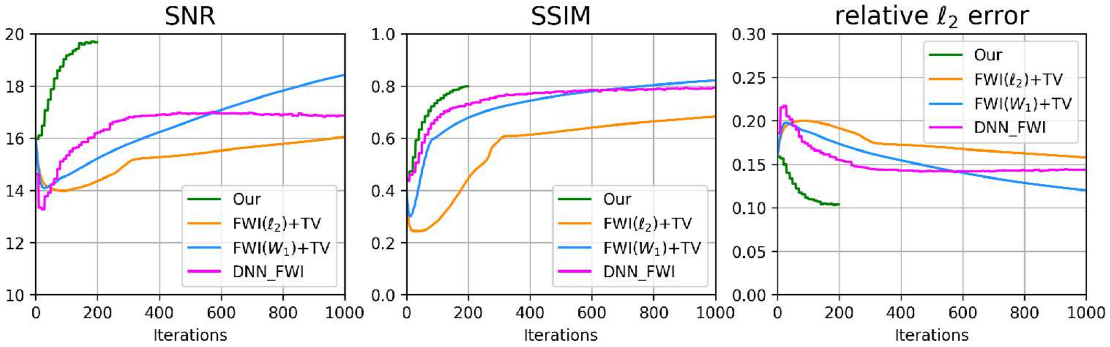


Figure 7: The trend of SNR, SSIM, and relative l_2 error of different inversion methods with respect to the number of iterations for the Marmousi model with noise-free observed data.

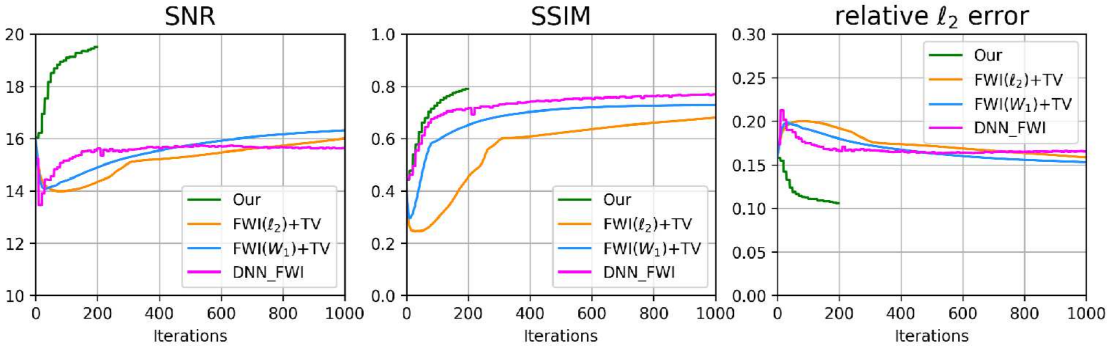


Figure 8: The trend of SNR, SSIM, and relative l_2 error of different inversion methods with respect to the number of iterations for the Marmousi model with noisy data ($\sigma = 0.01$).

plexities, including a central thrust faulted anticline, an external monocline, and a flat zone. The uppermost layer of the overthrust has undergone erosion and is covered by a surface layer, symbolizing recent sediments.

In Fig. 9, we present the inversion results obtained using noise-free measurement data. It is evident that the FWI(l_2)+TV method faces challenges in accurately reconstructing the subsurface velocity model. For the DNN-FWI method, only the upper region is effectively inverted. In contrast, both the FWI(W_1)+TV method and our proposed approach yield significantly improved solutions, accurately recovering the velocity structure across all regions. Furthermore, Fig. 11 illustrates the overthrust velocity profiles obtained using various approaches. These results reveal that the FWI(W_1)+TV regularization method delivers the best inversion results with noise-free measurement data, as corroborated by the results in Table 3.

Similarly, Fig. 10 provides a comparison of inversion results using measurement data affected by random noise. It is evident that our method outperforms all other compared approaches in terms of the inversion results. The velocity profiles, as depicted in Fig. 11, and the quantitative metrics listed in Table 3, once again emphasize the superiority of our method.

Furthermore, we have presented convergence curves in Figs. 12 and 13 that track the progress of SNR, SSIM, and relative l_2 error throughout the optimization process

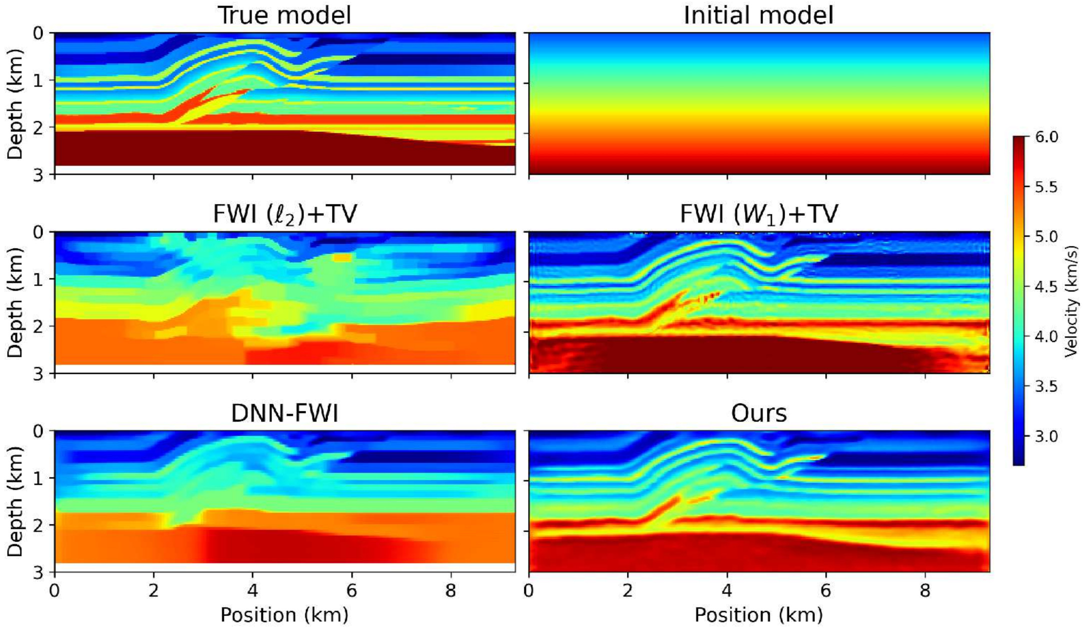


Figure 9: Inversion results for the noise-free observed data of the overthrust model. The top row displays the true velocity model and the initial guess. The second row shows the inverted results obtained by the traditional inversion methods FWI(l_2)+TV and FWI(W_1)+TV. The third row presents the inverted results obtained through DNN-FWI and our method.

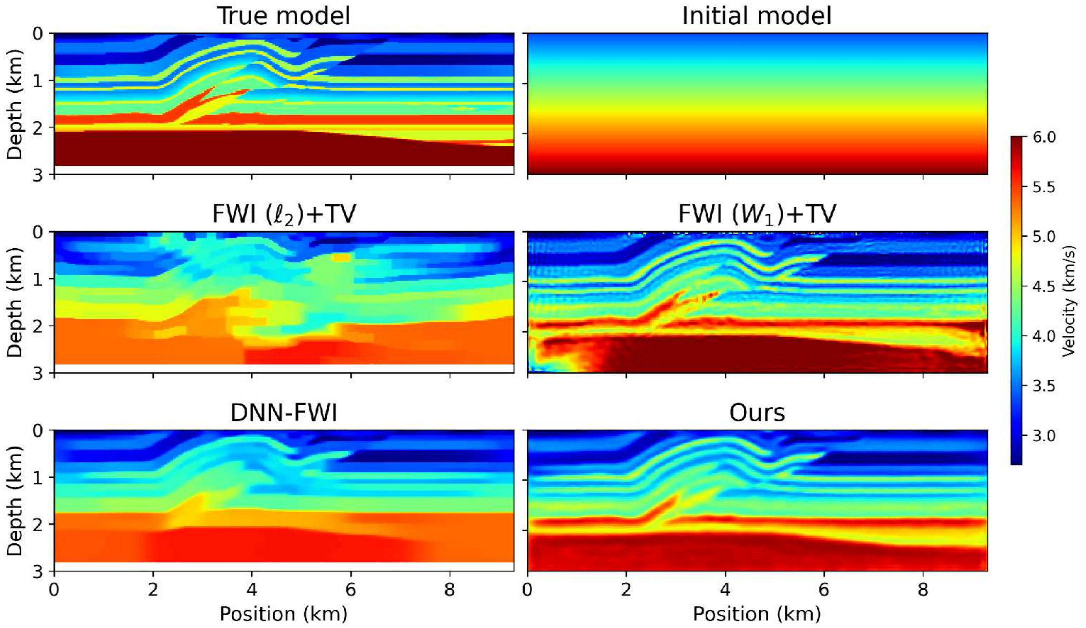


Figure 10: Inversion results for the noisy observed data of the overthrust model ($\sigma=0.01$). The top row displays the true velocity model and the initial guess. The second row shows the inverted results obtained by the traditional inversion methods $\mathrm{FWI}(\mathrm{l}_2)+\mathrm{TV}$ and $\mathrm{FWI}(\mathrm{W}_1)+\mathrm{TV}$. The third row presents the inverted results obtained through DNN-FWI and our method.

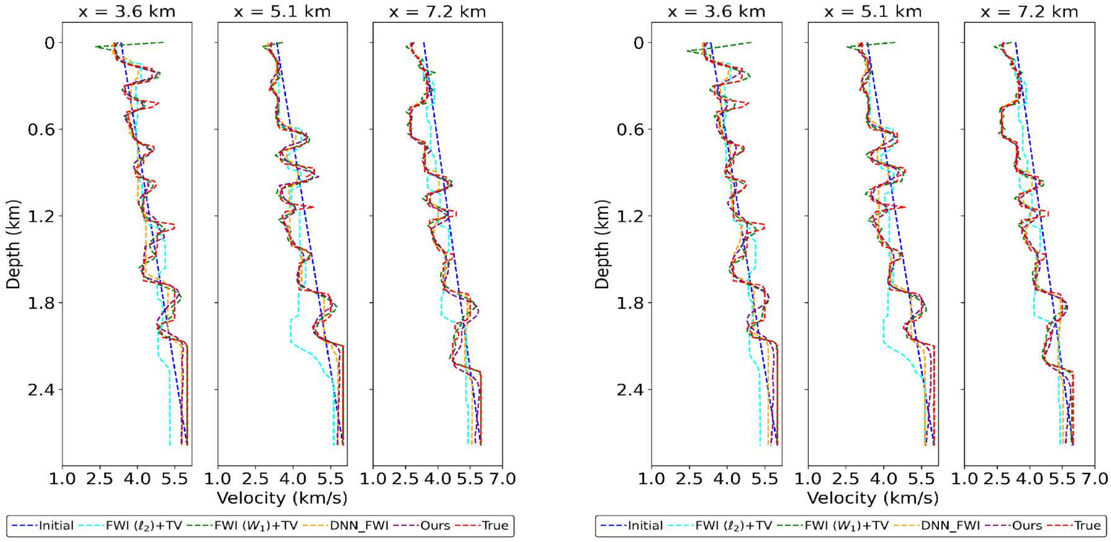


Figure 11: The overthrust velocity profiles of the true, initial, and inverted models obtained by different approaches at three horizontal locations. Left: noise-free observed data. Right: noisy observed data ($\sigma=0.01$).

facilitated by different inversion methods. These curves vividly illustrate the rapid convergence achieved by our inversion approach. Moreover, the elevated values of SNR and

Table 3: The SNR, SSIM, and relative l_2 error (Error) for the inverted overthrust model obtained by different approaches.

Case	Index	FWI(l_2)+TV	FWI(W_1)+TV	DNN−FWI	Ours
Noise-free	SNR	17.75	27.51	22.24	**25.19**
	SSIM	0.3098	0.8416	0.6266	**0.7486**
	Error	0.1300	0.0421	0.0773	**0.0550**
Noise($\sigma=0.01$)	SNR	0.1776	23.69	21.77	**25.87**
	SSIM	0.3117	0.8121	0.6023	**0.7735**
	Error	0.1295	0.0654	0.0815	**0.0509**

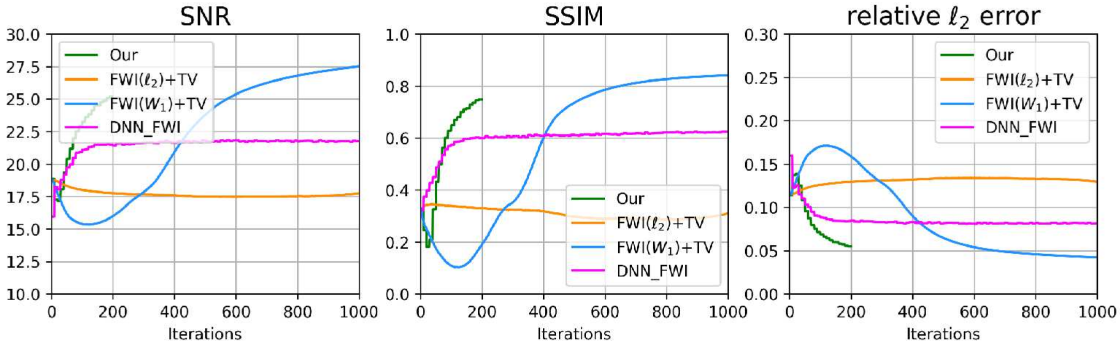


Figure 12: The trend of SNR, SSIM, and relative l_2 error of different inversion methods with respect to the number of iterations for the overthrust model with noise-free observed data.

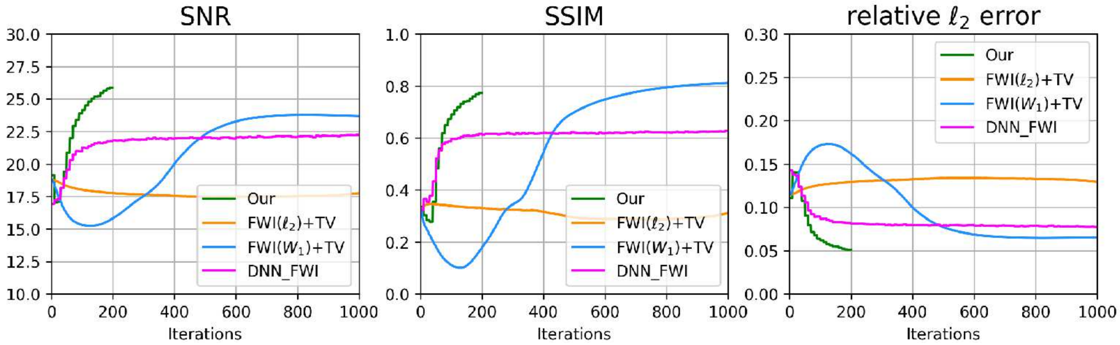


Figure 13: The trend of SNR, SSIM, and relative l_2 error of different inversion methods with respect to the number of iterations for the overthrust model with noisy observed data ($\sigma=0.01$).

SSIM, coupled with the low relative l_2 error, underscore the high precision of the inverted velocity model obtained by our method.

Similarly, in this example, we observe that the FWI(W_1)+TV regularization method reconstructs the finer details of the overthrust velocity model more accurately compared to our approach. However, our method achieves more precise results when inverting the velocity model with noisy measurement data. Furthermore, as shown in Fig. 10, the FWI(W_1)+TV regularization method struggles to accurately recover the lower-left region of the overthrust velocity model, whereas our proposed method demonstrates superior performance in this area.

4.2 Uncertainty

In this subsection, we assess predictive uncertainty using a benchmark model. To achieve this, we conduct 400 realizations using the trained network with activated dropouts, and subsequently calculate the mean and standard deviation of these realizations.

4.2.1 Marmousi2 model

Lastly, we further assess the effectiveness of our inversion method using the Marmousi2 model, an extension of the original Marmousi model. The Marmousi2 model encompasses a broader region, with the original Marmousi model positioned near its center. This extended model incorporates a range of structurally simple but stratigraphically complex features, surpassing the complexity of the original version. The downsampled version of the Marmousi2 used in our experiments has dimensions of $(z\times x)=(100\times 300)$. The velocity in the model ranges from $1140\,\mathrm{m/s}$ to $4700\,\mathrm{m/s}$, and the true model is depicted in Fig. 14.

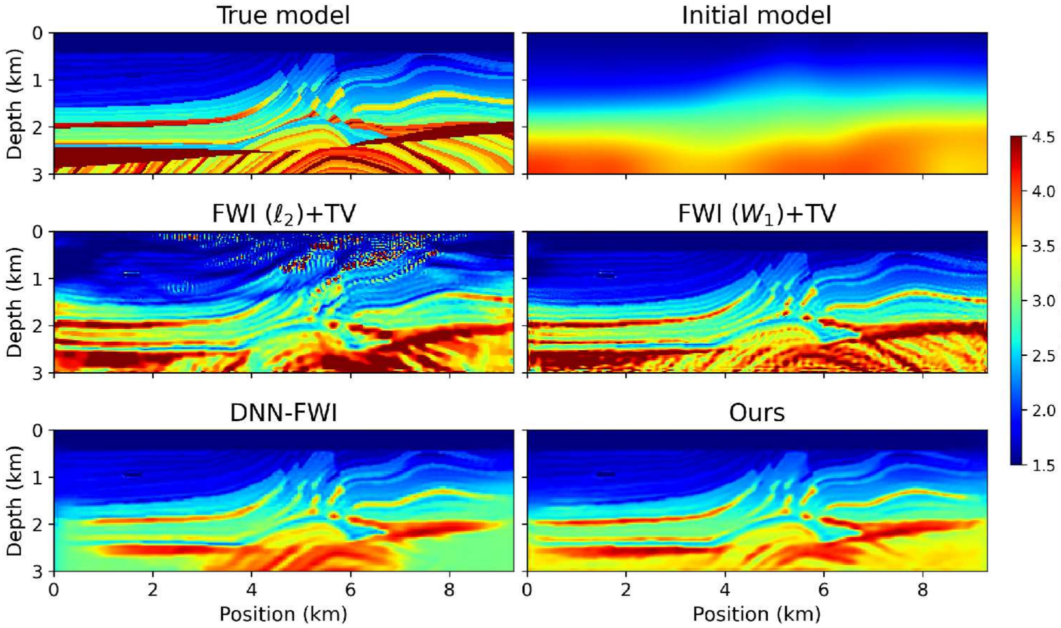


Figure 14: Inversion results for the noise-free observed data of the Marmousi2 model. The top row displays the true velocity model and the initial guess. The second row shows the inverted results obtained by the traditional inversion methods FWI(l_2)+TV and FWI(W_1)+TV. The third row presents the inverted results obtained through DNN-FWI and our method.

Fig. 14 displays the velocity results obtained by different inversion methods. The $\mathrm{FWI}(l_2)+\mathrm{TV}$ inversion method offer only approximate velocity estimates and struggle to accurately reconstruct the detail of the true model. DNN-FWI improves inversion performance, accurately reconstructing the upper part of the velocity model, but faces challenges in delivering accurate results for deeper regions. Our method and the $\mathrm{FWI}(W_1)+\mathrm{TV}$ inversion method consistently provide more precise reconstructions for the Marmousi2 model, especially in the deeper layers. These findings hold true even when working with noisy measurement data, as illustrated in Fig. 15. Additionally, in Fig. 15, we observe that our proposed inversion method outperforms the $\mathrm{FWI}(W_1)+\mathrm{TV}$ regularization method in recovering deeper layers of the model when using measurement data with noise.

Fig. 16 illustrates the sample standard deviations for the Marmousi2 model. Small variations appear in the shallow regions, particularly near the source-receivers level, indicating accurate inversion in this areas. Conversely, pronounced standard deviations are observed in the deeper regions and corners. This aligns with our intuition: when data is only available for the upper surface, accurate inversion of the velocity model is possible only in the shallow area. Those regions are poorly illuminated, and the inferred velocities are spread out over a wide range of values. In regions of the inverted velocity model exhibiting high uncertainty, the reliability of the inversion results is diminished.

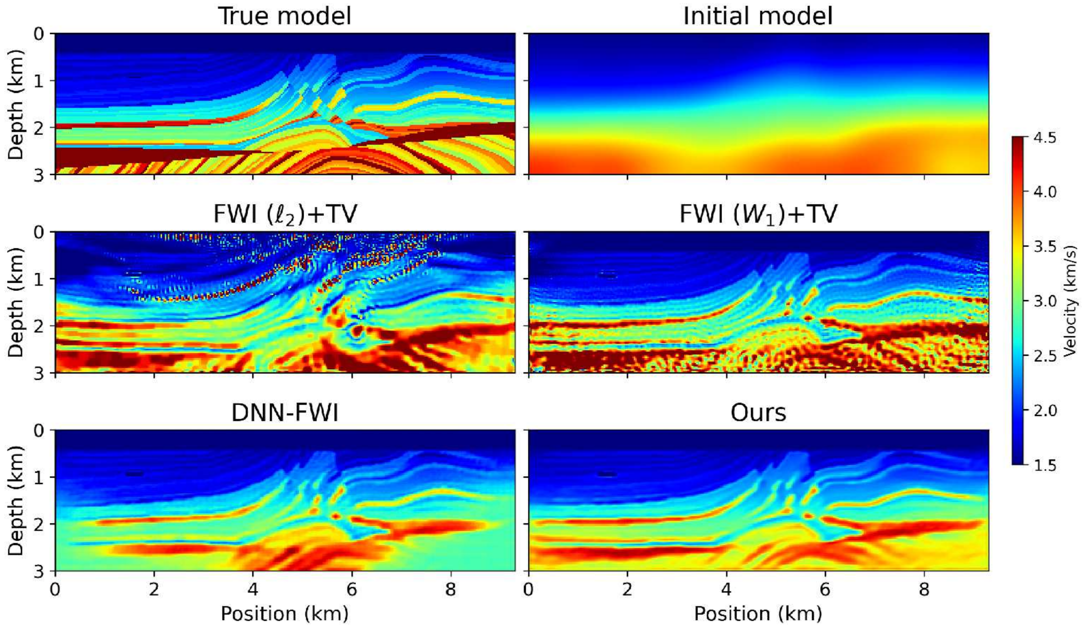
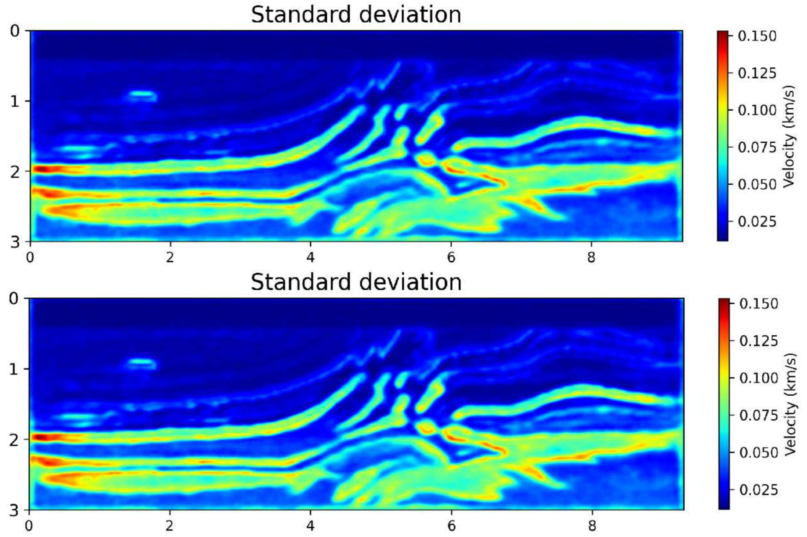


Figure 15: Inversion results for the noisy observed data of the Marmousi2 model ($\sigma=0.01$). The top row displays the true velocity model and the initial guess. The second row shows the inverted results obtained by the traditional inversion methods $\mathrm{FWI}(l_2)+\mathrm{TV}$ and $\mathrm{FWI}(W_1)+\mathrm{TV}$. The third row presents the inverted results obtained through DNN-FWI and our method.


Standard deviation

Standard deviation

Figure 16: The standard deviation of our method for Marmousi2 model. Top: noise-free observed data. Bottom: noisy observed data ($\sigma = 0.01$).

To enhance this reliability, it is necessary to adjust the algorithm's parameters or increase the observational data.

In our experiments, the uncertainty in the inversion results was investigated using measurement data that were contaminated with noise. In typical deterministic inversion methods, higher levels of measurement noise tend to make the inversion process more susceptible to overfitting, which often results in less accurate and less reliable inversion outcomes. This issue is clearly reflected in the findings presented in Table 4. However, the inversion method we proposed offers a significant advantage by not only providing inversion results but also quantifying the confidence in these results. As shown in Fig. 16, our method allows for a more nuanced interpretation of the inverted velocity model. Specifically, at deeper depths, the inverted model shows regions with lower confidence levels, indicating areas where the inversion results are less reliable due to the noise. This capability enables a more comprehensive assessment of the inversion's accuracy, making our method more robust and reliable compared to traditional deterministic approaches.

Fig. 17 illustrates a comparison of vertical velocity profiles obtained by different inversion methods. Quantitative metrics, including SNR, SSIM, and relative l_2 error, are summarized in Table 4. These observations demonstrate the consistent and stable performance of our method in addressing the full-waveform inversion problem. Importantly, we observe that our proposed approach delivers consistent accurate inversion results in scenarios with and without noise, affirming its robustness to noise.

Similarly, we have also plotted the convergence curves for SNR, SSIM, and relative l_2 error of the iterations progress of different inversion methods, as shown in Figs. 18, 19. These results further demonstrate the reliability of the inversion method provided by us.

Table 4: The SNR, SSIM, and relative l_2 error (Error) for the inverted Marmousi2 model obtained by different approaches.

Case	Index	$\mathrm{FWI}(l_2)+\mathrm{TV}$	$\mathrm{FWI}(W_1)+\mathrm{TV}$	DNN−FWI	Ours
Noise-free	SNR	13.45	19.78	17.14	**19.41**
	SSIM	0.2785	0.7579	0.7261	**0.7452**
	Error	0.2125	0.1025	0.1390	**0.1070**
Noise($\sigma=0.01$)	SNR	12.32	17.55	16.1317	**19.53**
	SSIM	0.1755	0.6840	0.7069	**0.7377**
	Error	0.2422	0.1325	0.1561	**0.1056**

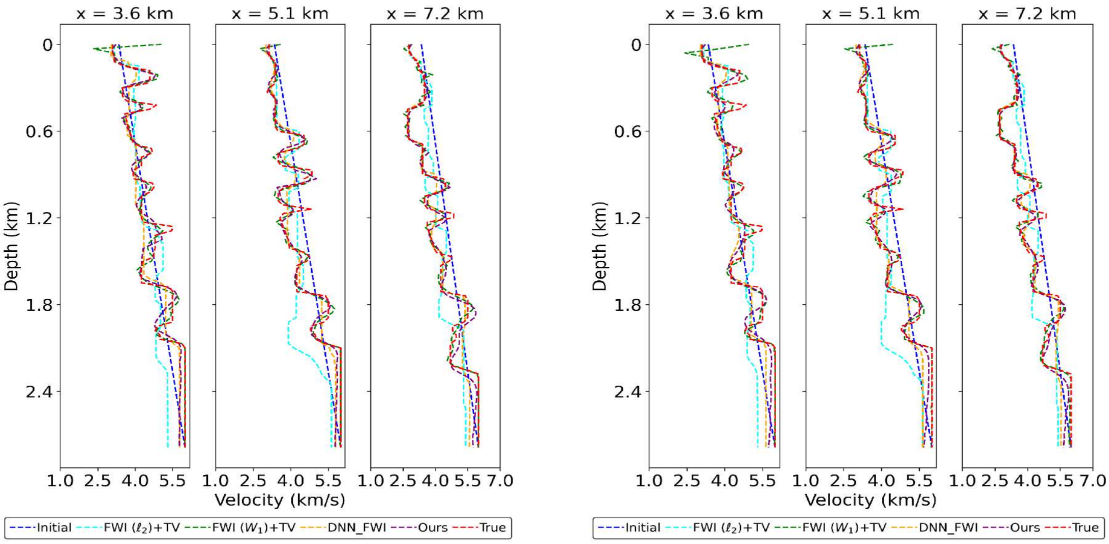


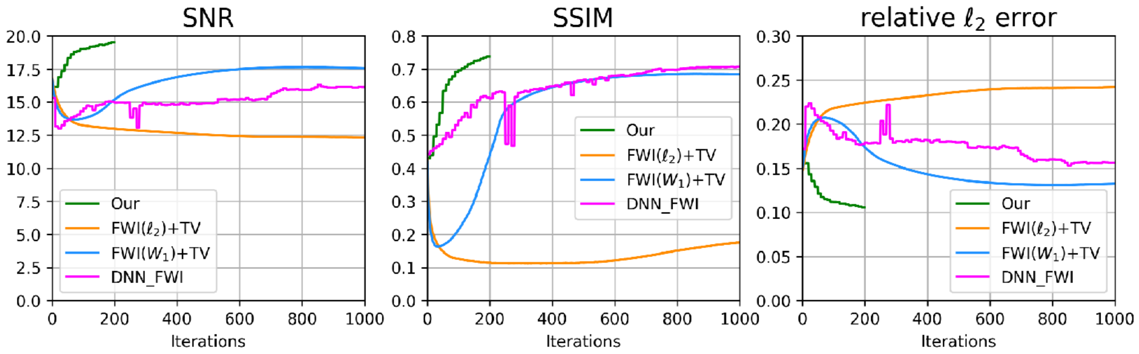
Figure 17: The Marmousi2 velocity profiles of the true, initial, and inverted models obtained by different approaches at three horizontal locations. Left: noise-free observed data. Right: noisy observed data ($\sigma=0.01$).

In Figs. 14 and 15, we observe that the $\mathrm{FWI}(W_1)+\mathrm{TV}$ regularization method reconstructs the detailed information of the model more effectively, particularly in the shallow regions of the Marmousi2 model, the finding supported by the structural similarity metrics in the Table 4. However, when the measurement data includes noise, the $\mathrm{FWI}(W_1)+\mathrm{TV}$ regularization struggles to accurately reconstruct the deeper layers of the model. In these regions, our inversion method demonstrates superior performance.

4.3 Algorithm investigations

4.3.1 Robustness to noise

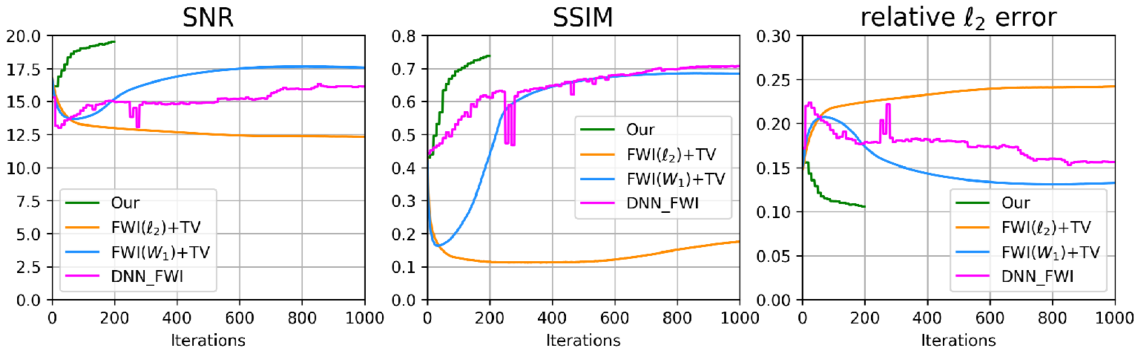
In this section, we evaluate the performance of the proposed inversion method across varying levels of noise, using the Marmousi model as a reference. The noise standard


SNR

SSIM

relative ℓ_2 error

Figure 18: The trend of SNR, SSIM, and relative l_2 error of different inversion methods with respect to the number of iterations for the Marmousi2 model with noise-free observed data.


SNR

SSIM

relative ℓ_2 error

Figure 19: The trend of SNR, SSIM, and relative l_2 error of different inversion methods with respect to the number of iterations for the Marmousi2 model with noisy observed data ($\sigma = 0.01$).

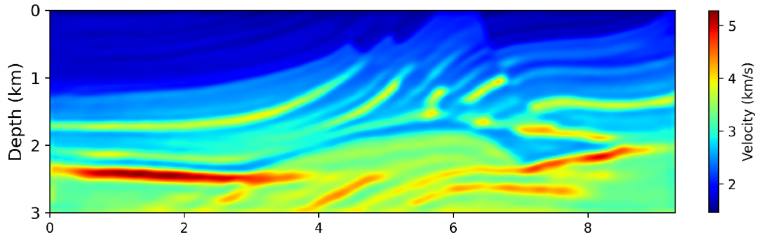
deviation σ is set to $[0, 0.001, 0.005, 0.01, 0.05]$ as outlined in Table 5. For our method, the number of iterations is fixed at 200, while the two traditional inversion methods are allowed 1000 iterations each. The relative l_2 error for different noise levels is recorded and presented in Table 5. Our observe that the proposed method consistently outperforms the two traditional inversion techniques across all noise levels examined. The data in Table 5 further emphasizes the exceptional robustness of our method in the presence of noise.

To further demonstrate the effectiveness of our proposed inversion method, we consider a more complex scenario. Specifically, for wave sources at different locations, the variance of the noise in the measured data from various receivers differs, represented as $\eta_{s,r} \sim N(0, \sigma_{sr}^2 I)$, where σ_{sr} varies for $s = 1, \ldots, n_s, r = 1, \ldots, n_r$. We assume $\sigma_{sr} \sim U(0, \sigma_{\max})$ for $s = 1, \ldots, n_s, r = 1, \ldots, n_r$, with $\sigma_{\max} = 0.05$ or $\sigma_{\max} = 0.1$. We utilize the Marmousi model as a case study.

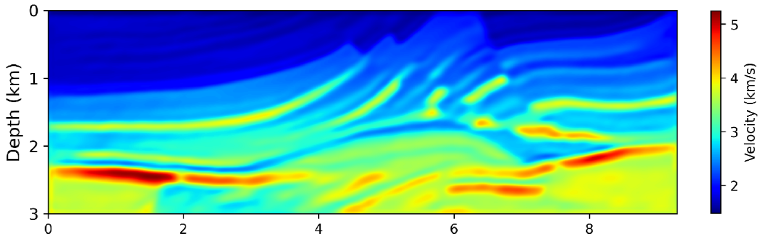
Fig. 20 presents the inversion results for different values of $\sigma_{\max}$. Specifically, when $\sigma_{\max} = 0.05$, the relative l_2 error of the inversion is 0.1204, while for $\sigma = 0.1$, the relative l_2 error is 0.1248. These results demonstrate that, even in this complex noise modeling scenario, our proposed inversion method effectively reconstructs the subsurface velocity model v.

Table 5: The relative l_2 error of the inversion v for the Marmousi model is evaluated under different noise levels. We fix the number of iterations for our method at 200, while the two traditional inversion methods are set to 1000 iterations.

σ / Methods	0	0.001	0.005	0.01	0.05
FWI(l_2)+TV	0.1576	0.1601	0.1608	0.1584	0.1582
FWI(W_1)+TV	0.1193	0.1225	0.1381	0.1529	0.1536
Our method	0.1038	0.1077	0.1063	0.1057	0.1050



(a) Inverse velocity v for Marmousi model under the $\sigma_{max}=0.05$

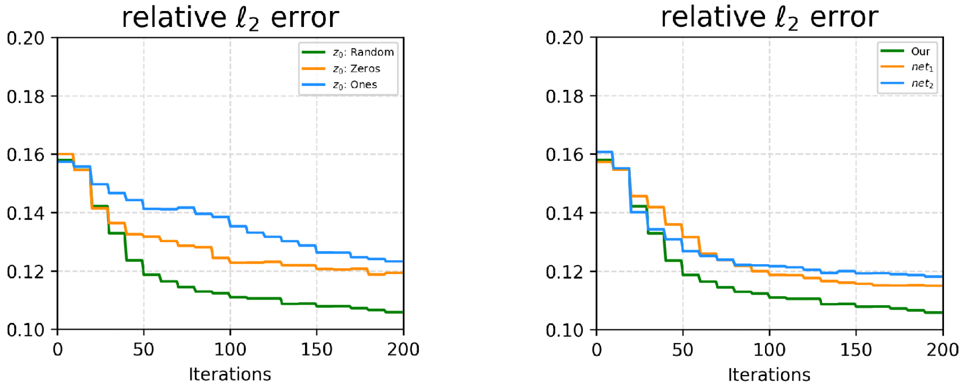


(b) Inverse velocity v for Marmousi model under the $\sigma_{max}=0.1$

Figure 20: Inversion under complex noise modeling scenarios. (a) Inverse velocity v for Marmousi model under the $\sigma_{max}=0.05$. (b) Inverse velocity v for Marmousi model under the $\sigma_{max}=0.1$. We fix the number of iterations to 200 and the regularization parameter to $\alpha=1\times10^{-8}$ in (3.12).

4.3.2 Different parameterizations

In this section, we analyze the impact of different parameterization forms of $m(z_0,\boldsymbol{\theta})$ on inversion performance. First, we fix the neural network structure and select three distinct initializations for z_0: a uniformly distributed random tensor, a zero tensor, and a tensor of ones. Using the Marmousi model as a case study, we evaluate how these different initializations of z_0 affect inversion accuracy. In this experiment, we set the standard deviation of noise for the measurement data to $\sigma=0.01$. As shown in Fig. 21(a), initializing z_0 as a uniformly distributed random tensor yields the most accurate inversion results. Furthermore, compared to the other methods listed in Table 2, this methods achieve the best



(a) Different inputs z_0 of $m(z_0,\boldsymbol{\theta})$ (b) Different networks of $m(z_0,\boldsymbol{\theta})$

Figure 21: (a) With a fixed neural network structure, we examine the relative l_2 error of the inverted velocity v (Marmousi model) in $m(z_0,\boldsymbol{\theta})$ for three different initializations of z_0: a uniformly distributed random tensor, a tensor of all zeros, and a tensor of all ones. (b) With z_0 fixed as a uniformly distributed random tensor in $m(z_0,\boldsymbol{\theta})$, we plot the relative l_2 error of the inverted velocity v (Marmousi model) for different neural network structures.

accuracy of 0.1529. Even when z_0 is initialized as a zero tensor or a tensor of ones, our method still produces precise inversion results.

Additionally, we also test the impact of different neural network structures m on inversion accuracy, while fixing z_0 as a uniformly distributed random tensor. To evaluate this, we selected the following two neural network architectures as references:

net_1: We construct a new neural network by reducing the number of channels in each layer of our proposed deep neural network structure (Table 1) by half.

net_2: The neural network structure from reference [24], which has been proven effective in various image processing tasks, is also considered.

Similarly, we use the Marmousi model as an example, setting the noise standard deviation of the measurement data to $\sigma = 0.01$. The experimental results, shown in Fig. 21(b), indicate that all three different neural network structures can yield accurate inversion results.

4.3.3 Computational efficiency

In this section, we evaluate the computational cost of our proposed inversion method using the Marmousi model and compare it with two traditional inversion methods. In this experiment, the standard deviation of the measurement data noise is set to $\sigma = 0$. We fix the maximum number of iterations for our method at 200, while the two traditional methods are set to 1000 iterations. The time taken for each method to complete the inversion, along with the GPU memory usage, are recorded and presented in Table 6. The results show that our method requires less overall computation time than the two traditional methods, as it converges with fewer iterations. However, compared to the traditional methods, our method demands a larger amount of memory.

Table 6: The computational cost of the different inversion methods for the Marmousi model. We fix the number of iterations for our method at 200, while the two traditional inversion methods are set to 1000 iterations.

Methods / Total cost	FWI(l_2)+TV	FWI(W_1)+TV	Our method
Time (s)	4737	4913	1239
Memory usage (MB)	822	833	2028

5 Conclusion

In this paper, to solve FWI, we propose a new deep learning inversion method that utilizes re-parametrization through a neural network with random weights to approximate the posterior distribution of the Bayesian inverse problem. This approach does not necessitate external training samples, offering flexibility and practical ease of use.

We validate our approach using the established Marmousi, Marmousi2, and overthrust models, both with noise-free and noisy observed data. Our experiments demonstrate superior performance compared to two representative non-learning-based methods and one recent deep-learning method. Significantly, we have observed that the provided inversion method exhibits robustness to measurement noise. In other words, even when the measurement data contains noise, our inversion method consistently produces accurate results.

Several promising avenues for further research are available. Firstly, our inversion method relies on an initial guess. We can explore techniques like model-agnostic meta-learning and adversarial pretraining to improve initialization. Secondly, our method has so far been tested on benchmark models. Our future plans involve applying it to practical model inversion problems.

Acknowledgments

This work is sponsored by the National Key R&D Program of China (Grant Nos. 2022YFA1008200 (Z. Q. Xu) and 2020YFA0712000 (Z. Ma)), by the Shanghai Sailing Program (Z. Q. Xu), by the Natural Science Foundation of Shanghai (Grant No. 20ZR1429000) (Z. Q. Xu), by the National Natural Science Foundation of China (Grant Nos. 62002221 (Z. Q. Xu), 12101401 and 12031013 (Z. Ma), 12401563 (X. B. Yan)), by the Shanghai Municipal of Science and Technology Major Project No. 2021SHZDZX0102, and by the HPC of School of Mathematical Sciences and the Student Innovation Center, and by the Siyuan-1 cluster supported by the Center for High Performance Computing at Shanghai Jiao Tong University.

References

[1] A. Asnaashari, R. Brossier, S. Garambois, F. Audebert, P. Thore, and J. Virieux, *Regularized seismic full waveform inversion with prior model information*, Geophysics, 78(2):R25–R36, 2013.

[2] A. J. Berkhout, *Pushing the limits of seismic imaging, Part II: Integration of prestack migration, velocity estimation, and AVO analysis*, Geophysics, 62(3):954–969, 1997.

[3] R. Brossier, S. Operto, and J. Virieux, *Which data residual norm for robust elastic frequency-domain full waveform inversion*, Geophysics, 75(3):R37–R46, 2010.

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

[5] M. Chen, Y. Quan, T. Pang, and H. Ji, *Nonblind image deconvolution via leveraging model uncertainty in an untrained deep neural network.*, Int. J. Comput. Vis., 130(7):1770–1789, 2022.

[6] F. Clement, G. Chavent, and S. Gómez, *Migration-based traveltime waveform inversion of 2-D simple structures: A synthetic example*, Geophysics, 66(3):845–860, 2001.

[7] L. Deng and Y. Liu, *Deep Learning in Natural Language Processing*, Springer, 2018.

[8] Q. Ding, H. Ji, Y. Quan, and X. Zhang, *A dataset-free deep learning method for low-dose CT image reconstruction*, Inverse Problems, 38(10):104003, 2022.

[9] W. Ding, K. Ren, and L. Zhang, *Coupling deep learning with full waveform inversion*, arXiv: 2203.01799, 2022.

[10] B. Engquist and Y. Yang, *Optimal transport based seismic inversion: Beyond cycle skipping*, Comm. Pure Appl. Math., 75(10):2201–2244, 2022.

[11] F. Faucher, O. Scherzer, and H. Barucq, *Eigenvector models for solving the seismic inverse problem for the Helmholtz equation*, Geophys. J. Int., 221(1):394–414, 2020.

[12] Y. Gal and Z. Ghahramani, *Dropout as a Bayesian approximation: Representing model uncertainty in deep learning*, in: International Conference on Machine Learning, PMLR, 1050–1059, 2016.

[13] A. Graves, *Practical variational inference for neural networks*, in: Advances in Neural Information Processing Systems, Vol. 24, Curran Associates, Inc., 2348–2356, 2011.

[14] M. J. Grote and U. Nahum, *Adaptive eigenspace for multi-parameter inverse scattering problems*, Comput. Math. Appl., 77(12):3264–3280, 2019.

[15] Q. He and Y. Wang, *Reparameterized full-waveform inversion using deep neural networks*, Geophysics, 86(1):V1–V13, 2021.

[16] J. Hudson and J. Heritage, *The use of the Born approximation in seismic scattering problems*, Geophys. J. Int., 66(1):221–240, 1981.

[17] C. M. Hyun, H. P. Kim, S. M. Lee, S. Lee, and J. K. Seo, *Deep learning for undersampled MRI reconstruction*, Phys. Med. Biol., 63(13):135007, 2018.

[18] M. Jakobsen and B. Ursin, *Full waveform inversion in the frequency domain using direct iterative T-matrix methods*, J. Geophys. Eng., 12(3):400–418, 2015.

[19] V. Kazei, O. Ovcharenko, P. Plotnitskii, D. Peter, X. Zhang, and T. Alkhalifah, *Deep learning tomography by mapping full seismic waveforms to vertical velocity profiles*, in: EAGE 2020 Annual Conference and Exhibition Online, European Association of Geoscientists & Engineers, 2020.

[20] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, *Improved variational inference with inverse autoregressive flow*, in: Advances in Neural Information Processing Systems, Vol. 29, Curran Associates, Inc., 4743–4751, 2016.

[21] D. P. Kingma, T. Salimans, and M. Welling, *Variational dropout and the local reparameterization trick*, in: Advances in Neural Information Processing Systems, Vol. 28, Curran Associates, Inc., 2575–2583, 2015.

[22] D. Komatitsch and J. Tromp, *A perfectly matched layer absorbing boundary condition for the*

second-order seismic wave equation, Geophys. J. Int., 154(1):146–153, 2003.

[23] J. Lecomte, E. Campbell, and J. Letouzey, *Building the SEG/EAEG overthrust velocity macro model*, in: EAEG/SEG Summer Workshop-Construction of 3-D Macro Velocity-Depth Models, cp-96-00024, 1994.

[24] V. Lempitsky, A. Vedaldi, and D. Ulyanov, *Deep image prior*, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Curran Associates, Inc., 9446–9454, 2018.

[25] Z. Li, Y. Tang, J. Chen, and H. Wu, *The quadratic Wasserstein metric with squaring scaling for seismic velocity inversion*, arXiv:2201.11305, 2022.

[26] K. Muhumuza, M. Jakobsen, T. Luostari, and T. Lähivaara, *Seismic monitoring of CO2 injection using a distorted Born T-matrix approach in acoustic approximation*, J. Seism. Explor., 27:403–431, 2018.

[27] S. Nguyen, D. Nguyen, K. Nguyen, K. Than, H. Bui, and N. Ho, *Structured dropout variational inference for Bayesian neural networks*, in: Advances in Neural Information Processing Systems, Vol. 34, Curran Associates, Inc., 15188–15202, 2021.

[28] L. Qiu, J. Ramos-Martìnez, A. Valenciano, Y. Yang, and B. Engquist, *Full-waveform inversion with an exponentially encoded optimal-transport norm*, SEG Technical Program Expanded Abstracts 2017, Society of Exploration Geophysicists, 1286–1290, 2017.

[29] M. Rasht-Behesht, C. Huber, K. Shukla, and G. E. Karniadakis, *Physics-informed neural networks (PINNs) for wave propagation and full waveform inversions*, J. Geophys. Res. Solid Earth, 127(5):e2021JB023120, 2022.

[30] A. Richardson, *Deepwave*, `https://ausargeo.com/deepwave/`, 2023.

[31] O. M. Saad and Y. Chen, *Deep denoising autoencoder for seismic random noise attenuation*, Geophysics, 85(4):V367–V376, 2020.

[32] J. Sun, Z. Niu, K. A. Innanen, J. Li, and D. O. Trad, *A theory-guided deep-learning formulation and optimization of seismic waveform inversion*, Geophysics, 85(2):R87–R99, 2020.

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

[34] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo, and C. W. Lin, *Deep learning on image denoising: An overview*, Neural Netw., 131:251–275, 2020.

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

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

[37] G. Wang, J. C. Ye, and B. De Man, *Deep learning for tomographic image reconstruction*, Nat. Mach. Intell., 2(12):737–748, 2020.

[38] M. Warner, A. Ratcliffe, T. Nangoo, J. Morgan, A. Umpleby, N. Shah, V. Vinje, I. Štekl, L. Guasch, and C. Win, *Anisotropic 3D full-waveform inversion*, Geophysics, 78(2):R59–R80, 2013.

[39] X. Wu, L. Liang, Y. Shi, and S. Fomel, *FaultSeg3D: Using synthetic data sets to train an end-to-end convolutional neural network for 3D seismic fault segmentation*, Geophysics, 84(3):IM35–IM45, 2019.

[40] Y. Wu and Y. Lin, *InversionNet: An efficient and accurate data-driven full waveform inversion*, IEEE Trans. Comput. Imaging, 6:419–433, 2019.

[41] Y. Wu and G. A. McMechan, *Parametric convolutional neural network-domain full-waveform inversion*, Geophysics, 84(6):R881–R896, 2019.

[42] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, *3D shapenets: A deep represen-*

tation for volumetric shapes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 1912–1920, 2015.

[43] F. Yang and J. Ma, *Deep-learning inversion: A next-generation seismic velocity model building method*, Geophysics, 84(4):R583–R599, 2019.

[44] F. Yang and J. Ma, *Wasserstein distance-based full-waveform inversion with a regularizer powered by learned gradient*, IEEE Trans. Geosci. Remote Sens., 61:1–13, 2023.

[45] Y. Yang, B. Engquist, J. Sun, and B. F. Hamfeldt, *Application of optimal transport and the quadratic Wasserstein metric to full-waveform inversion*, Geophysics, 83(1):R43–R62, 2018.

[46] C. Zelt and R. Smith, *Seismic traveltime inversion for 2-D crustal velocity structure*, Geophys. J. Int., 108(1):16–34, 1992.

[47] H. Zhang and J. Ma, *Optimal transport with a new preprocessing for deep-learning full waveform inversion*, in: 2022 IEEE International Conference on Image Processing (ICIP), IEEE, 1446–1450, 2022.

[48] M. Zhu, S. Feng, Y. Lin, and L. Lu, *Fourier-DeepONet: Fourier-enhanced deep operator networks for full waveform inversion with improved accuracy, generalizability, and robustness*, arXiv:2305.17289, 2023.

[49] Q. Zou, L. Ni, T. Zhang, and Q. Wang, *Deep learning based feature selection for remote sensing sceneclassification*, IEEE Geosci. Remote Sens. Lett., 12(11):2321–2325, 2015.