Bayesian Physics-Informed Neural Networks for the Subsurface Tomography Based on the Eikonal Equation

Rongxi Gou, Yijie Zhang®, Xueyu Zhu®, and Jinghuai Gao®, Member, IEEE

Abstract—The high cost of acquiring a sufficient amount of seismic data for training has limited the use of machine learning in seismic tomography. In addition, the inversion uncertainty due to the noisy data and data scarcity is less discussed in the conventional seismic tomography literature. To mitigate the uncertainty effects and quantify their impacts in the prediction, the so-called Bayesian physics-informed neural networks (BPINNs) based on the eikonal equation are adopted to infer the velocity field and reconstruct the travel-time field. In BPINNs, two inference algorithms, including Stein variational gradient descent (SVGD) and Gaussian variational inference (VI), are investigated for the inference task. The numerical results of several benchmark problems demonstrate that the velocity field can be estimated accurately and the travel time can be well approximated with reasonable uncertainty estimates by BPINNs. This suggests that the inferred velocity model provided by BPINNs may serve as a valid initial model for seismic inversion and migration.

Index Terms—Bayesian physics-informed neural networks (BPINNs), eikonal equation, Stein variational gradient descent (SVGD), tomography, variational inference (VI).

I. INTRODUCTION

Seismic tomography is one of the most popular methods for studying the structure of the underground. In particular, travel-time tomography is an effective and mature technique to invert subsurface structure based on the ray theory [1]. The widely used partial differential equation (PDE) in travel-time tomography is the eikonal equation, which is the high-frequency approximation for the wave equation [2]. The eikonal equation can be solved through several numerical methods, including the finite-difference method [3], the fast marching method [4], and the fast sweeping method [5]. Travel-time tomography [6], [7], [8] can be typically solved by minimizing the misfit of the observed travel time and simulated travel time based on the eikonal equation with a synthetic velocity model, which has been applied to image the source area of the earthquake [9] and guide the full waveform inversion (FWI) [10]. In addition, the Bayesian tomography of the first arrival time is addressed [11], which introduces sampling Monte Carlo-type algorithm to generate the posterior distribution for the velocity models. Nonetheless, traditional methods for travel-time tomography face several challenges: they often require a good initial model, which can affect the approximation quality significantly. In addition, since the measured data always contain noises, it is essential to quantify its impacts on the inversion results, which is useful to interpret the estimated velocity model. However, this is a lack of study in the traditional travel-time tomography literature [6], [7], [8], [9]. Therefore, it is desirable to develop flexible algorithms that are not only less sensitive to the initial velocity model and noises but also provide uncertainty estimates on the estimated velocity model.

Recently, there is a growing interest to leverage machine learning power to solve the inverse problems based on the PDEs in order to overcome the shortcomings of the traditional methods [12], [13]. However, the performance of purely data-driven machine learning approaches is heavily dependent on the quality of training data, which could lead to poor performance for sparse/noisy datasets [14], [15]. Many recent efforts in the scientific machine learning community [16], [17], [18], [19] have been focused on these challenges. Notably, physics-informed neural networks (PINNs) have been demonstrated to solve a variety of problems with small datasets [17], where the underlying governing equations are introduced as a regularization term into the loss function. For seismic applications, PINNs have been used to construct the travel-time field from sparse observed travel-time data [20], [21] and solve the corresponding inverse problems to infer the velocity field [22] based on the eikonal equation. In addition, PINNs are also used for the FWI based on acoustic wave equation [23]. Besides, PINNs have also been used in electrical properties tomography [24] and cardiac electrophysiology [25].

Recently, Bayesian approaches have been integrated into PINNs framework to better account for the uncertainty and provide the uncertainty estimations of predicted parameters by the network. Notably, the so-called Bayesian PINNs (BPINNs) use physics equations as prior knowledge to compensate for the lack of training data, while Bayesian inference is
used to predict the uncertainty of output results. BPINNs are applied for the fluid flow reconstruction based on the Navier–Stokes equation, where the Stein variational gradient descent (SVGD) algorithm is used to enable the efficient Bayesian inference [26]. For the cardiac electrophysiology, the BPINNs based on the eikonal equation are used to estimate velocity fields [27]. In the context of geophysics, the hypocenter inversion is investigated by BPINNs with Stein variational inference (VI) to handle highly multimodal posterior distributions efficiently [28]. BPINNs with Laplacian approximation (Laplace HypoPINN) are used for hypocenter estimation, which shows the promising results in estimating the locations of the hypocenter and providing uncertainty quantification [29], [30]. Despite its inference efficiency, Laplace approximation is, in general, not suitable if the posterior is not symmetric and unimodal [31], [32].

To address the high dimensional issues in Bayesian inference, VI and Markov chain Monte Carlo (MCMC) method have been proposed. In contrast with the traditional MCMC method, VI is a deterministic method approximating the target distribution by minimizing Kullback–Leibler (KL) divergence, which is computationally efficient for large datasets in general. In particular, Blundell et al. [33] proposed a VI algorithm called Gaussian VI for neural networks. However, the accuracy of VI depends on the set of predefined distributions to approximate the target distribution. To address this problem, SVGD, a general VI algorithm that uses a set of particles rather than distributions to approximate the target distribution [34], has been demonstrated to be efficient in different applications [26], [27], [28].

As previously stated, the traditional approaches for travel-time tomography often require an initial model and do not offer uncertainty estimates about the inferred velocity model. Motivated by the above developments in the scientific machine-learning community, we present a Bayesian physics-informed machine-learning framework for travel-time tomography with a limited number of observed data. Specifically, we use neural networks to approximate the travel-time fields and velocity field models. In addition to the data misfit, we also incorporate the eikonal equation into the loss to acknowledge the underlying governing physics. To cope with uncertainty, we formulate the problem in the Bayesian framework and investigate the inference performance of the BPINNs using two popular methods from the literature—Gaussian VI and SVGD. The contributions of this work include the following.

1) We adopt BPINNs for travel-time tomography, given a limited amount of the observed travel time from the surface and the wells. To further improve the prediction performance, the velocity data from the specific locations are provided.

2) We introduced depth-dependent velocity uncertainty to better account for the data uncertainty of subsurface tomography.

3) We demonstrated that BPINNs can provide a reasonably accurate velocity field and meaningful uncertainty estimate for both the velocity field and travel-time field.

4) We demonstrated that the velocity model parameterized by a randomly initialized neural network and does not require a good initial velocity model in contrast to the traditional methods.

This article is organized as follows. In Section II, we introduce the eikonal equation and travel-time tomography setup. Then, we briefly discuss PINNs, BPINNs with Gaussian VI, and SVGD. The depth-dependent uncertainty is introduced to better account for the data uncertainty under the context of travel-time tomography in Section III. Several numerical benchmark problems are provided to demonstrate the effectiveness of the method in Section IV. The validity and weaknesses of our method are discussed in Section V. Finally, we conclude in Section VI.

II. PROBLEM SETUP

The propagation of seismic waves through the underground obeys Fermat’s principle [35]. Fermat’s principle states that the path by a ray between two given points prefers the one with the shortest travel time. For example, the direct wave would travel in a straight line from the source to the receiver if the underground is isotropic and homogeneous. Due to the compositional layering and tectonic structure, the seismic waves would be reflected and refracted, where the wave shall travel along the path with the shortest travel time. In the literature, the eikonal equation (1) is used to model the relationship between travel time and velocity field as follows:

\[
\begin{align*}
|\nabla T(x, s)|^2 - \frac{1}{v^2(x)} &= 0 \\
T(x, s) &= 0 \\
T(x_0, s_0) &= \tau(x_0, s_0)
\end{align*}
\]

where \(T(x, s)\) represents the travel time from the location of the point source \(x_0\) to any point \(x\) in the domain \(\Omega\) and \(v(x)\) is the velocity defined in \(\Omega\).

Since the singular point exits at the point source in (1), we factorize \(T(x, s)\) into two factors [21], [22] as follows:

\[
T(x, s) = T_0(x, s)\tau(x, s)
\]

where

\[
T_0(x, s) = \frac{|x - x_s|}{v(x)}.
\]

Substituting (2) into (1), we get the residual \(R(x, s)\) of the factorized eikonal equation

\[
\begin{align*}
R(x, s) &= \left\{ |\nabla (T_0(x, s)\tau(x, s))|^2 - \frac{1}{v^2(x)} = 0 \right\} \forall x \in \Omega \\
\tau(x, s) - 1 &= 0
\end{align*}
\]

In this work, our goal is to infer velocity field \(v(x)\) and reconstruct travel time \(T(x, s)\) with a limited number of observations. The measured velocity data \(v(x)\) could be acquired from the well logs [36], which is rescaled as follows:

\[
v(x) = \frac{v(x) - v_{\min}}{v_{\max} - v_{\min}}
\]

where \(v_{\max}\) and \(v_{\min}\) represent the chosen scaling factors. \(v_{\max}\) should be chosen larger than the maximum value of
the velocity model, while $v_{\min}$ should be smaller than the minimum value to constrain the predicted output as 0 to 1. $v_{\max}$ and $v_{\min}$ can be chosen appropriately based on the well-logs data or the prior knowledge.

III. METHODS

In this section, we shall briefly introduce PINNs and their application for eikonal equation and then discuss Bayesian neural networks (BNNs) and BPINNs for eikonal equation.

A. Physics-Informed Neural Networks

Standard PINNs approximate the unknown solution $y(x)$ of the underlying PDEs by a neural network $N(x; \theta)$ parameterized by $\theta$. In contrast with purely data-driven machine learning algorithms, PINNs incorporate the residual of the underlying governing equation into the loss function in order to provide additional knowledge. The loss function for PINNs can be written as follows:

$$\mathcal{L}(\theta) = \frac{\lambda_1}{|\mathcal{T}_f|} \sum_{x \in \mathcal{T}_f} ||\mathcal{F}(N(x; \theta), x)|| + \frac{\lambda_2}{|\mathcal{T}_d|} \sum_{x \in \mathcal{T}_d} ||N(x; \theta) - y(x)||$$

(6)

where $\lambda_1$ and $\lambda_2$ are the weights for each term. $\lambda_1$ and $\lambda_2$ can be chosen by trial and error or adaptive weights algorithm [37]. $\mathcal{F}$ is the residual of the governing physics equation. $\mathcal{T}_f$ denotes the set to the locations to evaluate the residual $\mathcal{F}$, and $|\mathcal{T}_f|$ is the number of data points. $\mathcal{T}_d$ represents the locations of the observed data set, while $|\mathcal{T}_d|$ is the number of observed data.

For seismic tomography in this work, two independent neural networks $N_t$ and $N_v$ are adopted to approximate the travel-time factor and velocity field

$$\hat{\tau}(x, s) = N_t(x, s; \theta_t), \quad \hat{v}(x) = N_v(x; \theta_v)$$

(7)

where $\theta_t$ and $\theta_v$ are the weights and bias of the neural networks $N_t$ and $N_v$, respectively. Here, the sigmoid function is used in the last layer of $N_v$ to restrict the output of $N_v$ between 0 and 1. Subject to the eikonal equation (4), the corresponding loss function can be formulated as follows:

$$\mathcal{L}(\theta) = \frac{\lambda_1}{|\mathcal{T}_e|} \sum_{(x, s) \in \mathcal{T}_e} ||\mathcal{R}(x, s, \theta) - \tau(x, s)||$$

$$+ \frac{\lambda_2}{|\mathcal{T}_v|} \sum_{x \in \mathcal{T}_v} ||\hat{\tau}(x, s) - \tau(x, s)||$$

$$+ \frac{\lambda_3}{|\mathcal{T}_d|} \sum_{(x, s) \in \mathcal{T}_d} ||\mathcal{R}(x, s; \theta) - \tau(x, s)||$$

(8)

where $\theta = \{\theta_t, \theta_v\}$. $\lambda_1$-$\lambda_3$ are the weights for each term. $\mathcal{T}_e$ denotes the set of the locations of the source-receiver pairs to evaluate the residual $\mathcal{R}(x, s)$ in (4). $\mathcal{T}_v$ and $\mathcal{T}_d$ are the sets of the locations of the observed velocity and travel time, respectively. The parameters $\theta$ is typically optimized by Adam [38]. Once the parameters are optimized, the approximated travel-time factor and velocity field are obtained.

B. BPINNs for the Eikonal Equation

Nonetheless, vanilla PINNs [17] have limited capability to account for the uncertainty from the model, unknown parameters, and noisy data. To better quantify their impacts on the neural network outputs, we shall first introduce the BNN, followed by its adaption in the context of PINNs. After that, we shall discuss two variants of BPINNs for inference.

1) Bayesian Neural Network: For traditional neural networks, the network weights and bias $\theta$ are assumed to be deterministic values [39]. In contrast, BNNs consider $\theta$ as random variables with specific distributions that can be learned based on the Bayes’ theorem

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$$

(9)

where $\theta$ and $D$ represent the network parameters and measurements, respectively. By using (9), the posterior distribution of the parameters $p(\theta|D)$ can be computed by the prior distribution $p(\theta)$ and the likelihood $p(D|\theta)$. Since the dataset is independent of $\theta$, $p(D)$ can be treated as a normalized constant during the training process. Once the network is trained for the specific dataset $D$, the predicted results and the associated uncertainty can be approximated by the posterior sample mean value and standard deviation

$$\mathbb{E}_{p(\theta|D)}[y|x, D] \approx \frac{1}{M} \sum_{i=1}^{M} \mathcal{N}(x; \theta_i)$$

$$\text{Var}_{p(\theta|D)}[y|x, D] \approx \frac{1}{M} \sum_{i=1}^{M} (\mathcal{N}(x; \theta_i) - \mathbb{E}_{p(\theta|D)}[y|x, D])^2$$

(10)

where the values of $\mathcal{N}(x; \theta_i)$ represent the BNN with the corresponding parameter of the $i$th sample of $\theta$, respectively. $M$ is the number of samples drawn from the posterior distribution of neural network parameters $\theta$. $x$ and $y$ are the input and output of the neural network.

2) Bayesian Physics-Informed Neural Networks: Similar to traditional neural networks, BNN may suffer from a lack of data. Incorporating the underlying governing equation to BNN can provide additional prior knowledge to improve the generalization, which is referred to as BPINNs [26, 27, 28, 29]. With the physical constraints, Bayes’ theorem under the context of BPINNs can be formulated as follows:

$$p(\theta|\mathcal{R}) = \frac{p(\mathcal{R}|\theta)p(\theta)}{p(\mathcal{R})}$$

(11)

where $\mathcal{R}$ represents the residual of the eikonal equation in (4). The likelihood $p(\mathcal{R}|\theta)$ is

$$\log p(\mathcal{R}|\theta) = \log p(\mathcal{D}|\theta) + \log p(\mathcal{R}|\theta)$$

(12)

where $p(\mathcal{D}|\theta)$ is the likelihood of the observed data and $p(\mathcal{R}|\theta)$ represents the likelihood about the physics equation. The whole structure of BPINNs for seismic tomography is illustrated in Fig. 1. Two independent fully connected neural networks $\mathcal{N}_t$ and $\mathcal{N}_v$ are adopted to approximate the velocity field and the travel-time field.
To approximate the target distribution by a class of reparameterized density functions, SVGD [34] is proposed to approximate the posterior distribution with a set of particles. In this work, we shall leverage and compare these two algorithms to perform the inference task for BPINNs.

\textit{a) BPINNs with Gaussian VI:} Computing the posterior distribution \((\theta; \xi)\) can be intractable for high dimensional problems, which is common for neural networks. To address this issue, VI [40], [41] has been adopted to approximate \(p(\theta|D, R)\) with a predefined family of distributions \(Q(\theta; \xi)\), where the values of \(\xi = (\xi_0, \xi_1, \ldots, \xi_n)\) parameterize the distribution of \(Q(\theta; \xi)\). In this work, the distribution family is chosen to be normal distributions, where \(\xi = (\xi_\mu, \xi_\sigma)\), \(\xi_\mu\) and \(\xi_\sigma\) represent the corresponding mean and standard deviation, respectively. Under the assumption that the parameters of the neural network are independent, \(Q(\theta; \xi)\) can be expressed as follows:

\[
Q(\theta; \xi) = \prod_{i=1}^{d_\theta} q(\theta_i; \xi_{\mu,i}, \xi_{\sigma,i}) \tag{13}
\]

where \(d_\theta\) is the number of neural network’s parameters, \(\theta_i\) represents \(i\)th parameter of the neural network, obeying 1-D Gaussian distribution \(\mathcal{N}(\xi_{\mu,i}, 2 \log(1 + e^{\xi_{\sigma,i}}))\), and the standard deviation is represented by \(\log(1 + e^{\xi_{\sigma,i}})\) instead of \(\xi_{\sigma,i}\) to avoid standard deviation less than 0. Instead of using the sampling approach, such as MCMC, VI reformulates it as a deterministic optimization problem by minimizing the KL divergence between the posterior distribution \(p(\theta|D, R)\) and the proposed distribution \(Q(\theta; \xi)\)

\[
D_{\text{KL}}(Q(\theta; \xi)||p(\theta|D, R)) = \int Q(\theta; \xi) \log \frac{Q(\theta; \xi)}{p(\theta|D, R)} \, d\theta
\]

Algorithm 1: BPINNs With Gaussian VI for Seismic Tomography

\textbf{Input:} A set of sparse, noisy travel-time and velocity data.

\textbf{Output:} Trained parameters \(\xi\) through sampling the networks parameters \(\theta\).

\textbf{Initialization:} initialize \(\xi = (\xi_\mu, \xi_\sigma) = \mathcal{N}(0_{d_\theta}, I_{d_\theta})\), \(d_\theta\) is the number of neural network’s parameters.

\textbf{for} \(i=1:N_c\) \textbf{do}

1) Sample \(\epsilon_j\) from \(\mathcal{N}(0_{d_\theta}, I_{d_\theta})\) independently, where \(j = 0, 1, \ldots, n\), \(n\) is the number of sample of networks parameters;

2) \(\theta_j = \xi_\mu + \log(1 + \exp(\xi_\sigma)) \odot \epsilon_j\);

3) \(L(\xi) = \frac{1}{n} \sum_{j=1}^{n} \left[ \log Q(\theta_j; \xi) - \log p(\theta_j) - \log p(D(\theta_j)) - \log p(R(\theta_j)) \right]\);

4) Use Adam optimizer to upgrade \(\xi\) based on the gradient of \(L(\xi)\).

\textbf{end}

\(\propto \int Q(\theta; \xi) \log \frac{Q(\theta; \xi)}{p(\theta)p(D, R)} \, d\theta = E_\theta \sim Q[\log Q(\theta; \xi) - \log p(\theta) - \log p(D(\theta)) - \log p(R(\theta))]. \tag{14}\)

It can be shown that KL divergence has its minimum value when \(Q(\theta; \xi)\) and \(p(\theta|D, R)\) are equal, which enables us to approximate target distribution [41].

To simplify computations, we assume that prior distribution \(p(\theta)\) is a Gaussian distribution \(\mathcal{N}(\mu_\theta, I_{d_\theta})\). The likelihoods of the data and the model are assumed to be Gaussian distributions

\[
\log p(D(\theta)) \propto -\frac{1}{2\sigma_D^2} \sum_{x \in T_c} (\hat{v}(x) - v(x))^2
\]
\[
+ \left[ -\frac{1}{2\sigma_D^2} \sum_{(x, s) \in T_c} (\hat{\tau}(x, s) - \tau(x, s))^2 \right]
\]

\[
\log p(R(\theta)) \propto -\frac{1}{2\sigma_R^2} \sum_{(x, s) \in T_R} (R(x, s) - 0)^2 \tag{15}
\]

where \(\sigma_D\) and \(\sigma_R\) represent the standard deviation for data likelihood \(p(D(\theta))\) of predicted velocity and travel-time factor. \(\sigma_R\) is the standard deviation for model likelihood \(p(R(\theta))\).

The minimization of KL divergence as (14) is performed through gradient backpropagation by Adam optimizer. The algorithm of VI is shown in Algorithm 1.

\textbf{b) BPINNs with SVGD:} Alternatively, SVGD is a general VI algorithm based on Stein’s identity and kernelized Stein discrepancy. SVGD employs a set of particles to approximate the target posterior distribution by adjusting the distribution of the particles. To update these particles, kernelized Stein discrepancy is adopted to find the steepest descent for the KL divergence [34].

Suppose \(p(\theta)\) and \(q(\theta)\) are the target distribution and approximated distribution, respectively. \(f(\theta)\) is a smooth function and satisfied \(\int_{-\infty}^{\infty} \nabla_{\theta} f(\theta)p(\theta) \, d\theta = 0\). When Stein
operation \( A_\theta \) acts on \( f(\theta) \), we have

\[
A_\theta f(\theta) = \nabla \theta \log p(\theta) f(\theta)^T + \nabla \theta f(\theta).
\]  

(16)

Stein’s identity \( \mathbb{E}_{\theta \sim q}[A_\theta f(\theta)] \) states when \( f(\theta) \) satisfies the above conditions, Stein’s identity equals zero. If we consider the expectation of \( A_\theta f(\theta) \) under \( q(\theta) \), we get Stein discrepancy \( \mathbb{E}_{\theta \sim q}[A_\theta f(\theta)] \). The magnitude of Stein discrepancy measures how different \( p(\theta) \) and \( q(\theta) \) values are. When \( p(\theta) \) and \( q(\theta) \) are the same distribution, Stein discrepancy equals zero. During the iteration, the parameters \( \theta \) will be updated as follows: \( \theta = \theta + \epsilon \phi(\theta) \), where \( \epsilon \) is the step size and \( \phi(\theta) \) represents the updating direction of parameters. According to Liu and Wang’s [34] work, KL divergence is proved to decay fastest when in Algorithm 1

\[
\nabla \theta \text{ KL}(q||p) = -\mathbb{E}_{\theta \sim q}[\text{tr}(A_\theta f(\theta))].
\]  

(17)

Unlike Gaussian VI introduced in the previous subsection, SVGD employs a set of deterministic particles \( \{\theta_j\}_{j=1}^n \) to minimize KL divergence instead of sampling from the variational distribution family. With the gradient direction, we can use the SVGD algorithm to update particles as follows:

\[
\phi(\theta) = \frac{1}{n} \sum_{j=1}^n \left[ k(\theta_j, \theta) \nabla \theta_j^T \log p(\theta_j^T) + \log(\text{likelihood}) \right]
\]

\[
+ \nabla \theta_j^T k(\theta_j, \theta)
\]

\[
\theta_{j+1} = \theta_j + \epsilon_t \phi(\theta_j)
\]  

(18)

where \( \epsilon_t \) represents the step size in the \( t \)th iteration, \( k(\cdot, \cdot) \) represents a positive kernel function. \( \epsilon_t \) is the small step size in the \( t \)th iteration. In this work, radial basis function (RBF) is used

\[
k(\theta_j^T, \theta) = \exp \left( -\frac{|\theta - \theta_j|^2}{2l^2} \right)
\]  

(19)

where \( l \) is the median distance between the particles \( \{\theta_j\}_{j=1}^n \) to control the length scale of the kernel.

In Gaussian VI we discussed above, we consider that the standard deviations \( (\sigma_D, \sigma_R) \) for likelihood are the same for each \( \theta_j \). The reason is that we sample networks parameters from \( Q(\theta; \xi) \), but the standard deviations are independent of \( \xi \). However, for SVGD, we take the particle approach, which means for each particle \( \theta_j \), independent standard deviations should be selected corresponding to the likelihood for each \( \theta_j \). We assume that the diagonal matrices \( \Sigma_D = [\text{diag}(\sigma_D^2), \text{diag}(\sigma_D^2), \text{ diag}(\sigma_R^2), \text{diag}(\sigma_R^2)] \) and \( \Sigma_R = \text{diag}(\sigma_R^2) \) are the trainable parameters. \( \Sigma_D \) is the covariance matrix of the distributions \( p(D|\theta, \Sigma_D), p(R|\theta, \Sigma_R) \) denote the sum of data and model likelihood for each particle, respectively. If data likelihood \( p(D|\theta, \Sigma_D) \) and model likelihood \( p(R|\theta, \Sigma_R) \) are normal distributions, the logarithmic likelihood function can be written as follows:

\[
\text{log(likelihood)} = \log p(D|\theta, \Sigma_D) + \log p(R|\theta, \Sigma_R)
\]

\[
= \sum_{j=1}^n \left[ \log p(D|\theta_j, \sigma_{D,j}) + \log p(R|\theta_j, \sigma_{R,j}) \right]
\]  

(20)

where \( \theta_j \) is the network parameters of the \( j \)th particle and \( \theta \) represents the network parameters of all \( n \) particles. Here, the likelihood for one particle can be represented as follows:

\[
\log p(D|\theta_j, \sigma_{D,j})
\]

\[
\propto -\frac{1}{2\sigma_{D,j}} \sum_{x \in T_c} (\hat{v}(x) - v(x))^2 + |T_r| \log \left( \frac{1}{\sigma_{D,j}} \right)
\]

\[
+ \frac{1}{2\sigma_{R,j}} \sum_{(x, x') \in T_r} (\hat{\tau}(x, x') - \tau(x, x'))^2 + |T_r| \log \left( \frac{1}{\sigma_{R,j}} \right)
\]  

(21)

where \( \sigma_{D,j} = [\sigma_{D,v,j}, \sigma_{D,\tau,j}] \), \( \sigma_{R,j} \) represents the standard deviation of data uncertainty and model uncertainty for particle \( \theta_j \). \( \hat{v} \) and \( \hat{\tau} \) in (21) are the outputs from particle \( \theta_j \), while \( v \) and \( \tau \) are the corresponding training data. The corresponding posterior variance is defined as follows:

\[
\text{Var}_{p(\theta|D)}[\hat{y}|x, D] \approx \frac{1}{n} \sum_{j=1}^n \left( N(x; \theta_j) - \mathbb{E}_{p(\theta|D)}[y|x, D] \right)^2 + \frac{\text{variance of particles}}{\text{predicted data uncertainty}}
\]  

(22)

where \( \hat{y} \) represents the predicted velocity and travel-time fields. The variances from \( n \) particles provide the uncertainty estimate contributed by the posterior distribution of network parameters, while the predicted data uncertainty represents the uncertainty due to measurement noise or lack of observed data. According to (20), the posterior distribution could be estimated through a set of particles \( \{\theta_j\}_{j=1}^n \). In summary, we list the SVGD algorithm in Algorithm 2.

c) Depth-dependent uncertainty: It is worth noting that the data uncertainty \( \sigma_{D,z} \) in (20) is assumed to be the learnable constant across the entire domain. Nevertheless, in surface tomography, the receivers are not evenly distributed, and the number of receivers near the surface is typically more than those in the layer far from the surface. As a result, higher uncertainty about the velocity is expected, as the depth from the surface increases. Besides, in most situations, as the depth increases, strata pressure rises, which leads to higher wave propagation velocity. This may cause higher uncertainty at the deep layer due to the noise included in measurement data. Motivated by this observation, we assume that the uncertainty of predicted velocity \( \Sigma_{D_z} \) follows the linear relationship with the depth \( z \) as follows:

\[
\Sigma_{D_z}(z) = \Sigma_{D_z}(z_{\min}) + \frac{z - z_{\min}}{z_{\max} - z_{\min}} \left[ \Sigma_{D_z}(z_{\max}) - \Sigma_{D_z}(z_{\min}) \right]
\]  

(23)
Algorithm 2 BPINNs Based on SVGD for Seismic Tomography

**Input:** A set of sparse, noisy travel-time and velocity data.

**Output:** Trained networks parameters \( \{ \theta_j \}_{j=1}^D \) in order to reconstruct travel-time and estimate velocity field in the computational domain. \( \Sigma_D \) and \( \Sigma_R \) are the data variance and model uncertainty, respectively. \( n \) is the number of particles.

**Initialization:**
Sample initial values for \( \theta \) from Gaussian distribution \( \mathcal{N}(0, \sigma^2) \). Sample each parameter in \( \Sigma_D \) from Gamma distribution \( \text{Gamma}(\Sigma|\alpha_1, \beta_1) \). \( \alpha_1 \) and \( \beta_1 \) are the shape and rate parameters, respectively. Sample each parameter \( \Sigma_R \) from \( \text{Gamma}(\Sigma|\alpha_2, \beta_2) \). Sample each parameter \( \theta_j \) from \( \mathcal{N}(0, \sigma^2) \) independently; here, \( \sigma_v \) and \( \sigma_r \) represent the level of noise for velocity and travel-time factor.

**for** \( t=0 : t \leq \tau \) **do**

1. Calculate \( \log p(\theta^t) \), \( \log p(\Sigma_D) \) and \( \log \text{likelihood} \).
2. Update networks parameters \( \theta_j^{t+1} = \theta_j^t + \epsilon_v \phi(\theta_j^t) \) and \( \Sigma_D \) by stochastic gradient descent.

where \( z_{\min} \) and \( z_{\max} \) are the minimum and maximum of the depth of the velocity model, respectively. \( \Sigma_D, (z_{\min}) \) and \( \Sigma_D, (z_{\max}) \) are the trainable parameters that represent the corresponding uncertainty of the velocity at \( z_{\min} \) and \( z_{\max} \), respectively.

IV. NUMERICAL EXAMPLES

In this section, we shall assess the feasibility and performance of the proposed algorithms via several benchmark problems. We first study a 1-D homogeneous model with an analytic solution to verify the inference capability of the proposed methods. Then, a 2-D model with an ellipsoidal inclusion is used to further explore the inference capability of VI and SVGD. Finally, depth-dependent uncertainty is employed to test the effectiveness of SVGD with additional prior knowledge of uncertainty distribution.

In the numerical examples, we use the CPU toolkit named scikit-fmm [42] to generate the travel-time data for the synthetic velocity models. The velocity data (from the ground-truth model) at certain locations are provided to mimic the detailed records from well logs in the practical setup. Besides, we corrupted the observed travel time and velocity with Gaussian noises as follows:

\[
\begin{align*}
\nu_d &= v(1 + \epsilon_v), \quad \epsilon_v \sim \mathcal{N}(0, \sigma_v^2) \\
\tau_d &= \tau(1 + \epsilon_r), \quad \epsilon_r \sim \mathcal{N}(0, \sigma_r^2)
\end{align*}
\]

where \( \tau_d \) and \( \nu_d \) are the measured travel-time factor and velocity, respectively. \( \epsilon_v \) and \( \epsilon_r \) are sampled from the distribution \( \mathcal{N}(0, \sigma_v^2) \) and \( \mathcal{N}(0, \sigma_r^2) \) independently; here, \( \sigma_v \) and \( \sigma_r \) represent the level of noise for velocity and travel-time factor.

To quantify the performance of the proposed method, we compute the correlation coefficient \( \gamma \) and the absolute relative error (ARE):

\[
\gamma = \frac{\sum_{i=1}^n (\nu_i - \bar{\nu}) (\tau_i - \bar{\tau})}{\sqrt{\sum_{i=1}^n (\nu_i - \bar{\nu})^2 \sum_{i=1}^n (\tau_i - \bar{\tau})^2}}
\]

\[
\text{ARE} = \frac{1}{n} \sum_{i=1}^n \frac{|\nu_i - \tau_i|}{|\tau_i|}
\]

where \( n \) is the total number of test points. \( \nu \) is the output results of BPINNs, and \( \tau \) is the ground truth corresponding to \( \bar{\nu} \). \( \bar{\nu} \) and \( \bar{\tau} \) are the mean values of predicted \( \nu \) and the ground truth \( \tau \), respectively. All neural network is trained by the PyTorch with NVIDIA GeForce RTX 2080 Ti GPU.

A. 1-D Eikonal Equation of Homogeneous Model

To test the effectiveness of BPINNs in quantifying uncertainty, we first consider the 1-D eikonal equation of a homogeneous model with a constant velocity \( v \) as follows, motivated by Ceccarelli [27]:

\[
\begin{align*}
\left| \frac{\partial T(x)}{\partial x} \right| &= \frac{1}{v^2} \quad \forall x \in [0, 2] \\
T(x_0) &= 0
\end{align*}
\]

where \( x \) is the location of the receiver. The source location \( x_s \) is 0 km. The ground-truth velocity is set to be \( v = 2 \text{ km/s}. \) In this case, the analytical travel time is \( T(x) = (x/v) \).

We generate a synthetic dataset of size \( n_v \) noisy measurements \( D = \{x_i, T_i(x_i)\}_{i=1}^{n_v} \) by corrupting the noise-free data generated by the analytical travel time with 5% Gaussian \( \sigma_{TV} = 0.05 \) noise in (24). The likelihood for dataset \( D \) conditioned on \( v \) can be written as follows:

\[
p(D|v) \propto \prod_{i=0}^{n_v} \frac{1}{\sqrt{2\pi \sigma_v^2}} \exp \left( -\frac{(T_i(x_i) - \frac{x_i}{v})^2}{2\sigma_v^2 T^2(x_i)} \right)
\]

where \( T \) and \( T_i \) are the ground truth and noisy measures of the travel time. To simplify the derivation, we consider finding \( p(v^{-1}|D) \) instead of \( p(v|D) \). We assume that the prior distribution \( p(v^{-1}) \) of \( v^{-1} \) obeys Gaussian distribution \( \mathcal{N}(0, \sigma_0^2) \). As a result, the posterior distribution could be calculated through

\[
p(v^{-1}|D) = \frac{p(D|v^{-1})p(v^{-1})}{p(D)} \propto \prod_{i=0}^{n_v} \frac{1}{\sqrt{2\pi \sigma_v T(x_i)}} \exp \left( -\frac{(T_i(x_i) - \frac{x_i}{v^{-1}})^2}{2\sigma_v^2 T^2(x_i)} \right) \times \frac{1}{\sqrt{2\pi \sigma_0^2}} \exp \left( -\frac{(v^{-1})^2}{2\sigma_0^2} \right).
\]

As \( p(v^{-1}|D) \) is the product of a series of Gaussian distributions, the analytical solution of \( p(v^{-1}|D) \) could be derived based on (29), which also obeys Gaussian distribution. The
The true value of \( v \) is written as follows:

\[
\mu_{\text{post}} = \frac{a\sigma_0^2}{b\sigma_0^2 + a}
\]

\[
\sigma_{\text{post}} = \frac{a\sigma_0^2 T_2^2(x_1) T_2^2(x_2)}{b\sigma_0^2 + a^2} \quad (30)
\]

where

\[
a = x_1 T_2(x_1) T_2^2(x_2) + x_2 T_2(x_2) T_2^2(x_1)
\]

\[
b = x_1^2 T_2^2(x_1) + x_2^2 T_2^2(x_2). \quad (31)
\]

In our case, the standard deviation of the prior distribution is \( \sigma_0 = 1 \), and the level of relative Gaussian noise \( \sigma_\tau \) is 0.05. Two measurement data are located at \( x_1 = 1 \) km and \( x_2 = 2 \) km. The true value of \( v^{-1} \) is 0.5.

To test the performance of BPINNs-VI and BPINNs-SVGD for quantifying uncertainty in BPINNs, a trainable parameter is used to approximate \( v^{-1} \). The travel time \( T(x) \) is approximated by a fully connected network with two hidden layers and 20 neurons for each layer with the Swish activation function after each layer. For VI, we update the parameters for 5000 epochs, and 100 samples of posterior distribution are collected. As for SVGD, 100 particles are chosen and trained for 5000 epochs; 100 evenly spaced points \( x \in [0, 2] \) km are used as the residual points for both methods.

We compare the results from VI and SVGD with the analytic solution of \( v^{-1} \) based on (30). From Table I, we can see that the approximated mean and standard deviation of BPINNs-SVGD are closer to the true posterior mean and standard deviation than BPINNs-VI. Furthermore, Fig. 2 shows that BPINNs-SVGD with 100 particles fits better with the analytic distribution of \( u^{-1}(D) \) than BPINNs-VI. It appears that BPINNs-SVGD is more accurate in estimating the posterior distribution of \( v^{-1} \) in this example.

### B. Cross-Hole Tomography

Next, we consider a synthetic velocity model with ellipsoidal anomaly to investigate the performance of the BPINNs with VI and SVGD, shown in Fig. 3(a). In this case, \( 2 \times 5 \) equally spaced sources are placed on both left boundary (\( x = 0 \) km) and the right boundary (\( x = 2 \) km). The \( 2 \times 51 \) receivers are evenly spaced on the left boundary and right boundary. Two well logs at \( x = 0 \) and 2 km provide \( 2 \times 51 \) measured velocity data at the same locations as the receivers. The 5% Gaussian noise is added to the measurement data with \( \sigma_x = \sigma_\tau = 0.05 \) in (24).

For this problem, two independent fully connected neural networks are used to represent \( \tau \) and \( v \), respectively. For the travel-time factor, the neural network consists of five hidden layers with 20 neurons per layer with Swish activation function. For the velocity, we chose the network that includes five hidden layers with ten neurons for each layer with exponential linear unit (ELU) activation function. A sigmoid function is used to normalize the network output, and scaling factors \( \tau_{\text{max}} \) and \( v_{\text{min}} \) introduced in (5) are used to approximate predicted velocity. For SVGD, the prior distributions of \( \Sigma_{\mathcal{D}_\tau} \), \( \Sigma_{\mathcal{D}_v} \), and \( \Sigma_{\mathcal{T}} \) are Gamma distribution Gamma(2, \( 10^{-6} \)). In this example, the particle number of SVGD has been set to 5.

### Table I

| Method | \( \mu \) | \( \sigma \) |
|--------|--------|--------|
| True   | 0.5041 | 0.0178 |
| VI     | 0.5132 | 0.0390 |
| SVGD   | 0.5059 | 0.0143 |

Fig. 2. (a) Posterior distribution for \( v^{-1} \) with 5% relative Gaussian noise by BPINNs-VI compared with the true posterior distribution for the 1-D homogeneous model. (b) Posterior distribution by BPINNs-SVGD with the true posterior distribution.
Fig. 4. Uncertainty estimation, posterior mean, and posterior realizations for inclusion model when $Z = 1$ km. The black solid line represents the mean value of predicted velocity. The red and green solid lines represent the realizations from the posterior distribution of predicted velocity model. The blue shaded areas are uncertainty covered by $\pm 1$ std. (a) BPINNs-SVGD. (b) BPINNs-VI.

TABLE II

| Method | ARE $(v)$ | $\gamma$ $(v)$ | ARE $(T)$ | $\gamma$ $(T)$ |
|--------|-----------|---------------|-----------|---------------|
| VI     | 0.0830    | 0.7691        | 0.0443    | 0.9932        |
| SVGD   | 0.0078    | 0.8594        | 0.0260    | 0.9978        |

with VI and SVGD are trained for 1000 epochs by Adam optimizer. These configurations will be used in subsequent experiments unless otherwise specified.

Figs. 3 and 4 show the results of BPINNs with VI and SVGD, respectively. Despite the fact that no direct measurements have been given in the anomaly area between 0.4 and 1.6 km, BPINNs with SVGD are capable of reconstructing the anomaly with the help of physics constraints. Due to the lack of information about the anomaly area, the velocity value of the anomaly is underestimated. Nonetheless, BPINNs indeed suggest higher uncertainty around the anomaly. Furthermore, as the seismometers are located on the left and right boundaries, the associated uncertainty is lower than we expected. While SVGD provides a more accurate inversion of the velocity field as well as reasonable uncertainty estimation, BPINNs with VI provide a less accurate estimation of the velocity field and appear to be overconfident in the predicted results.

To further demonstrate the performance of BPINNs, we compared the ground truth and predicted travel-time field by BPINNs with SVGD and VI in Fig. 5. Compared with results by BPINNs-VI, BPINNs-SVGD provided a more accurate travel-time field. Nonetheless, the predicted travel-time fields appear to be smoother than the ground-truth one, and BPINNs cannot capture the detailed effects induced by the inclusion. Besides, Table II lists the quantitative comparison of ARE and correlation coefficient $\gamma$ of BPINNs with VI and SVGD, indicating that the BPINNs-SVGD performs better for the cross-hole tomography in both performance metrics.

C. Surface Tomography

In this section, we shall investigate the influence of different levels of noise on surface tomography by BPINNs-SVGD. For surface tomography, the travel-time data are often collected on the surface; therefore, less information about the deeper layer can be provided, resulting in the uncertainty increase with depth. With this prior knowledge, depth-dependent uncertainty is introduced to better characterize the uncertainty.

Fig. 6 shows the ground-truth velocity for the surface tomography. At the surface of the model ($z = 0$ km), 11 evenly spaced sources with a distance of 0.5 km between each source location and 51 evenly spaced receivers with a distance of 100 m between each sensor location are placed. In addition, 50 evenly spaced receivers along $x = 2.5$ km with a distance of 20 m between each sensor location are placed. A well log with 50 evenly spaced receivers across $x = 2.5$ km is used to measure the velocity. The measured travel time and velocity are perturbed by 5%, 15%, and 25% three different levels of relative Gaussian noise. The architecture and hyperparameters of BPINNs-SVGD are the same as in Section IV-B.

The predicted velocity and the uncertainty estimates by BPINNs-SVGD with 5% Gaussian noise are shown in Fig. 7(a). BPINNs-SVGD appears to overestimate the uncertainty at the shallow depth while underestimating it as the depth increases. The uncertainty shown in Fig. 7(a) tends to be more uniform over the domain due to the depth-independent data uncertainty introduced in (22). To address this issue, we introduced the depth-dependent

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Fig. 6. Ground-truth velocity model. Eleven sources are equispaced at \( z = 0 \) km (red star); 51 receivers are uniformly spaced at \( z = 0 \) km, and 50 equispaced receivers with velocity sample points are located at \( x = 2.5 \) km (cyan star).

Fig. 7. Predicted mean and standard deviation based on BPINNs-SVGD with (a) depth-independent uncertainty and (b) depth-dependent uncertainty under 5\% Gaussian noise.

Fig. 8. Predicted velocity realizations, posterior mean, and corresponding uncertainty under 5\% noise with/without assumption of depth-dependent uncertainty. The black solid line represents the mean value of predicted velocity. The red and green solid lines represent the posterior realizations. The shaded blue areas are uncertainty covered by \( \pm 1 \) std. (a) Depth-independent uncertainty. (b) Depth-dependent uncertainty.

Fig. 9. (a) Ground-truth travel-time field computed by the fast marching method with scikit-fmm toolkit. (b) Predicted travel-time field by BPINNs-SVGD with 5\% Gaussian noise. Here, the source is located at \((2.5, 0)\) km. (c) Contour of predicted and ground-truth travel time.

uncertainty with depth discussed in Section III-B. The prior distribution of \( \Sigma D_v(z_{\text{min}}) \) and \( \Sigma D_v(z_{\text{max}}) \) is represented by Gamma distributions: Gamma\( (\Sigma D_v(z_{\text{min}})|2, 10^{-6}) \) and Gamma\( (\Sigma D_v(z_{\text{max}})|1.5, 10^{-6}) \), respectively. According to our experiments, the results are insensitive to the parameters of the distribution when they are restricted in a reasonable range. Fig. 7(b) shows the results by BPINNs-SVGD with the uncertainty with respect to the depth. After adopting depth-dependent uncertainty, the predicted uncertainty provides a much more reasonable uncertainty in the spatial domain, especially with respect to the depth \( z \).

Fig. 8 compares the predicted velocity and the uncertainty based on SVGD with/without depth-dependent uncertainty at depth \( z = 0, 0.25, \) and \( 0.5 \) km, respectively. For instance, 51 receivers are placed at \( z = 0 \), and the information is rich enough for BPINNs-SVGD to precisely predict the velocity with lower uncertainty. As the depth increases, the data provides less information, resulting in a larger deviation from the ground-truth velocity, and higher uncertainty can be expected. By introducing the depth-dependent uncertainty, the overestimation of uncertainty at shallow depth shown in SVGD with depth-independent uncertainty [Fig. 8(a)] can be significantly reduced, as in Fig. 8(b). In addition, the predicted travel-time field by BPINNs-SVGD with depth-dependent uncertainty is displayed in Fig. 9. It can be seen that the predicted travel time matches well the ground truth, which demonstrates the quality of the forward approximation of BPINNs.

To further study the noise influence on the performance of BPINNs, we test the proposed method over the observed data with 15\% and 25\% Gaussian noise based on the depth-dependent velocity uncertainty \( \Sigma D_v(z) \) in (23) by
BPINNs-SVGD. The predicted velocity model and uncertainty are shown in Fig. 10. Even with 25% Gaussian noise, BPINNs can still accurately recover the velocity model, demonstrating the robustness to noise of BPINNs. The corresponding error metrics are shown in Table III. As expected, the ARE and estimated uncertainty increase with the noise level. With 25% noise included in the measured data, the ARE is less than 2% and correlation reaches 98%, demonstrating the robustness of BPINNs. In addition, as we expected, the predicted uncertainty by BPINNs increased as the depth increases, as shown in Fig. 10.

D. Overthrust Model

In Sections IV-B and IV-C, we tested BPINNs on two relatively simple benchmarks. Here, we shall discuss the performance of the proposed scheme on a more complex and realistic model. The overthrust model is a 3-D geological model proposed as a result of collaboration by over 50 organizations. It is built with erosional truncation and sediment that cover the basement blocks. The overthrust model includes complex structures, such as reverse faults and converging thrusts. Because of its complicated geological characteristics, the overthrust model is used to test different imaging and inversion algorithms [43].

We have intercepted a 2-D slice from the 3-D overthrust model, as shown in Fig. 11(a). To acquire the observed travel time for training, 11 × 2 evenly spaced sources located on the left boundary (x = 0 km) and the right boundary (x = 3 km) are chosen, while two series of receivers located on the left and right boundaries with an interval of 0.02 km provide 2 × 91 travel-time data for each source. Four well logs located at x = 0–3 km provide velocity data with an interval of 0.02 km along the depth. Given its complex structure, the network for inferring the velocity has been set to ten hidden layers with 40 neurons, and noise-free training data are used. Other settings are the same as in Section IV-B.

The estimated velocity model by BPINNs-SVGD is shown in Fig. 11(b). Even though the details, such as the thrusts, are not accurately captured by the proposed method, it does capture well the relatively simple structure on the left half-domain of the overthrust model. Furthermore, the location of the high-speed region in the deeper layer is well captured. We also show the estimated uncertainty about the predicted velocity by BPINNs-SVGD for the overthrust model in Fig. 11(c). Since we can access the velocity data at x = 0–3 km, the estimated uncertainty is relatively low at these locations. Because of its simple structure, the prediction in the homogeneous region at depth ranging from 1.4 to 1.9 km presents a lower uncertainty. For the complex arch and fault structure in the overthrust model from x = 1.5 to 3 km, higher uncertainty is predicted as expected, indicating the effectiveness of the uncertainty estimated by BPINNs.

Fig. 12 shows the ground truth and predicted travel time by BPINNs-SVGD for the overthrust model. Even though
some details are missing, the BPINNs-SVGD predictions can capture the rough travel-time field.

V. DISCUSSION

In Section IV, three different velocity model has been tested, which illustrates the effectiveness of BPINNs for travel-time tomography. Even with limited noisy measurements, the BPINNs predicted results shows in Sections IV-B and IV-C have strong similarity with target models, which prove that BPINNs can perform travel-time tomography fairly well for relative simple models. Besides, reasonable uncertainty quantification is given to help researchers assess the reliability of the predicted velocity model. However, as for the complex overthrust model, BPINNs can only restore general structure of velocity model. Despite this, the predicted uncertainty is also credible as we analyze in Section IV-D.

The eikonal equation describes the relationship between the first-arrival time and the velocity, where the first-arrival time contains significantly less information than the seisograms observed in the setup of FWI. The seisograms contain direct waves, reflected waves from different layers, and so on, so each receiver could record a time series from the specific source. Therefore, based on the seisograms, it can easily describe the main layers of subsurface media. However, for the first-arrival time tomography, a receiver can only record a value of the arrival time from the specific source, so it is challenging for travel-time tomography to perfectly invert such a relatively complex model, such as overthrust. Nonetheless, the predicted velocity model by the proposed method could offer an effective initial model for FWI or migration imaging. In addition, the uncertainty quantification by the proposed method can provide a meaningful assessment of the reliability of the predictions and the basis for decision-making for further exploration.

VI. CONCLUSION

We introduced the BPINNs for seismic travel-time tomography based on the eikonal equation. The Gaussian VI and SVGD algorithms are adopted to infer the velocity and the corresponding uncertainty quantification. To better account for the higher uncertainty that exists in the deep layer for surface tomography, we further introduced depth-dependent uncertainty as the prior knowledge. After demonstrating the effectiveness of BPINNs-SVGD in the 1-D homogeneous model and ellipsoidal model, we adopt it to study the influence of different levels of noise on predictive uncertainty for surface tomography. We also employ BPINNs-SVGD for a more realistic overthrust model. Despite its complex structure and insufficient information, the proposed method provides a reasonably accurate velocity model and meaningful uncertainty estimation, demonstrating the feasibility and potential of the proposed method for realistic applications. If a better velocity model is needed, the suggested method may serve as an effective initial model for FWI.

REFERENCES

[1] R. P. Bording, A. Gerstenkorn, L. R. Lines, J. A. Scales, and S. Treitel, “Applications of seismic travel-time tomography,” Geophys. J. Int., vol. 90, no. 2, pp. 285–303, Aug. 1987.
[2] K. Aki and P. G. Richards, Quantitative Seismology. San Francisco, CA, USA: Univ. Science Books, 2002.
[3] J. E. Vidale, “Finite-difference calculation of traveltimes in three dimensions,” Geophysics, vol. 55, no. 5, pp. 521–526, May 1990.
[4] N. Rawlinson and M. Sambridge, “Wave front evolution in strongly heterogeneous layered media using the fast marching method,” Geophys. J. Int., vol. 156, no. 3, pp. 631–647, Mar. 2004.
[5] H. Zhao, “A fast sweeping method for Eikonal equations,” Math. Comput., vol. 74, no. 250, pp. 663–672, Apr. 2005.
[6] J. Hole, “Nonlinear high-resolution three-dimensional seismic travel time tomography,” J. Geophys. Res., Solid Earth, vol. 97, no. B5, pp. 6553–6562, 1992.
[7] S. Leung and J. Qian, “An anode jet method for three-dimensional transmission traveltime tomography using first-arrivals,” Commut. Math. Sci., vol. 4, no. 1, pp. 249–266, 2006.
[8] B. Shin and D. Shutin, “Distributed traveltime tomography using kernal-based regression in seismic networks,” IEEE Geosci. Remote Sens. Lett., vol. 19, 2022, Art. no. 8029905.
[9] J. Li, H. Li, H. Chen, J. Su, Y. Liu, and P. Tong, “Eikonal equation-based seismic tomography of the source areas of the 2008 mw 7.9 Wenchuan earthquake and the 2013 mw 6.6 Lushan earthquake,” Bull. Seismol. Soc. Amer., vol. 110, no. 2, pp. 886–897, Apr. 2020.
[10] E. Treister and E. Haber, “Full waveform inversion guided by travel time tomography,” SIAM J. Sci. Comput., vol. 39, no. 5, pp. S587–S609, Jan. 2017.
[11] A. Gesret, N. Desassis, M. Noble, T. Romary, and C. Maisons, “Propagation of the velocity model uncertainties to the seismic event location,” Geophys. J. Int., vol. 200, no. 1, pp. 52–66, Nov. 2014.
[12] M. Araya-Polo, J. Jennings, A. Adler, and T. Dahlin, “Deep-learning tomography,” Lead. Edge, vol. 37, no. 1, pp. 58–66, Jan. 2018.
[13] F. Yang and J. Ma, “Deep-learning inversion: A next-generation seismic velocity model building method,” Geophysics, vol. 84, no. 4, pp. R583–R599, Jul. 2019.
[14] M. J. Bianco, P. Gerstoft, K. B. Olsen, and F.-C. Lin, “High-resolution seismic tomography of long beach, CA using machine learning,” Sci. Rep., vol. 9, no. 1, pp. 1–11, Oct. 2019.
[15] I. E. Yildirim, T. Alkhalfiah, and E. U. Yildirim, “Machine learning-enabled traveltime inversion based on the horizontal source-location perturbation,” Geophysics, vol. 87, no. 1, pp. U1–U8, Jan. 2022.
