SEISMIC FIRST ARRIVAL TRAVELTIME INVERSION HARNESSING PHYSICS INFORMED NEURAL NETWORKS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF APPLIED MATHEMATICS
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

İSA EREN YILDIRIM

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
SCIENTIFIC COMPUTING

FEBRUARY 2022

Approval of the thesis:

SEISMIC FIRST ARRIVAL TRAVELTIME INVERSION HARNESSING PHYSICS INFORMED NEURAL NETWORKS

submitted by **İSA EREN YILDIRIM** in partial fulfillment of the requirements for the degree of **Master of Science in Scientific Computing Department, Middle East Technical University** by,

Prof. Dr. A. Sevtap Kestel _____
Dean, Graduate School of **Applied Mathematics**

Prof. Dr. A. Sevtap Kestel _____
Head of Department, **Scientific Computing**

Prof. Dr. Ömür Uğur _____
Supervisor, **Scientific Computing, IAM, METU**

Assist. Prof. Dr. Umair bin Waheed _____
Co-supervisor, **Department of Geosciences, KFUPM**

Examining Committee Members:

Prof. Dr. Ayhan Aydın _____
Mathematics Department, Atılım University

Prof. Dr. Ömür Uğur _____
Scientific Computing, IAM, METU

Assist. Prof. Dr. Önder Türk _____
Scientific Computing, IAM, METU

Date: _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: İSA EREN YILDIRIM

Signature :

ABSTRACT

SEISMIC FIRST ARRIVAL TRAVELTIME INVERSION HARNESSING PHYSICS INFORMED NEURAL NETWORKS

Yıldırım, İsa Eren
M.S., Department of Scientific Computing
Supervisor : Prof. Dr. Ömür Uğur
Co-Supervisor : Assist. Prof. Dr. Umair bin Waheed

February 2022, 36 pages

In Seismic prospecting, huge amounts of data are collected and processed to infer the structural and lithological composition of the subsurface. The key step in this procedure is velocity model building. First arrival traveltime inversion is one of the velocity model building tools commonly used for predicting near-surface velocity structures in seismic exploration. Conventionally, the inversion is carried out using ray-based methods or gradient-based algorithms. Though the gradient-based algorithms find the gradient that is needed to update the model parameters without requiring ray tracing, it can be computationally demanding. On the other hand, despite its robustness and efficiency ray-based methods suffer from complex regions as the ray theory relies on the high-frequency approximation. Instead of using these approaches for a traveltime inversion problem, we propose a machine learning based approach, specifically harnessing the physics informed neural networks exploiting the mathematical model represented by the eikonal equation to estimate the near-surface subsurface velocities. Through synthetic tests and the application of real data, we show the reliability of the physics informed machine learning based traveltime inversion which can be a potential alternative tool to the traditional tomography frameworks.

Keywords: inverse problems, machine learning, physics informed neural networks

ÖZ

FİZİK BİLGİLİ SİNİR AĞLARI KULLANARAK SİSMİK İLK VARIŞ SEYAHAT SÜRESİ TERS ÇÖZÜMÜ

Yıldırım, İsa Eren
Yüksek Lisans, Bilimsel Hesaplama Bölümü
Tez Yöneticisi : Prof. Dr. Ömür Uğur
Ortak Tez Yöneticisi : Yrd. Doç. Dr. Umair bin Waheed

Şubat 2022, 36 sayfa

Sismik araştırmada, yeraltının yapısal ve litolojik bileşimini ortaya çıkarmak için büyük miktarda veri toplanır ve işlenir. Bu prosedürdeki kilit adım, hız modeli oluşturmadır. İlk varış seyahat zamanının tersine çevrilmesi, sismik keşifte yüzeye yakın hız yapılarını tahmin etmek için yaygın olarak kullanılan hız modeli oluşturma araçlarından biridir. Geleneksel olarak, tersine çevirme, ışın tabanlı yöntemler veya gradyan tabanlı algoritmalar kullanılarak gerçekleştirilir. Gradyan tabanlı algoritmalar, model parametrelerini ışın izleme gerektirmeden güncellemek için gereken gradyanı bulsa da, hesaplama açısından zorlayıcı olabilir. Öte yandan, sağlamlığına ve verimliliğine rağmen ışın tabanlı yöntemler, ışın teorisi yüksek frekanslı yaklaşıma dayandığından karmaşık bölgelerden muzdariptir. Bu yaklaşımları bir seyahat zamanı tersine çevirme problemi için kullanmak yerine, yüzeye yakın yeraltı hızlarını tahmin etmek için eikonal denklem tarafından temsil edilen matematiksel modelden yararlanan fizik bilgili sinir ağlarından özellikle yararlanan makine öğrenimi tabanlı bir yaklaşım öneriyoruz. Sentetik testler ve gerçek verilerin uygulanması yoluyla, geleneksel tomografi çerçevelerine potansiyel bir alternatif araç olabilecek fizik bilgili makine öğrenimi tabanlı seyahat zamanı tersine çevirmenin güvenilirliğini gösteriyoruz.

Anahtar Kelimeler: ters problemler, makine öğrenmesi, fizik bilgili sinir ağları

To my family

ACKNOWLEDGMENTS

I would like to give my warmest thanks to my thesis supervisor Prof. Dr. Ömür Uğur for his assistance and valuable comments throughout the completion of this thesis. I would also like to express my sincere gratitude to my co-supervisor Assist. Prof. Dr. Umair bin Waheed for his precious contributions that helped me to produce this work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

PDE	Partial Differential Equation
FMM	Fast Marching Method
FSM	Fast Sweeping Method
PINN	Physics informed neural network
DL	Deep Learning
NN	Neural Network
DNN	Deep Neural Network
CD	Conjugate Direction
MLP	Multilayer Perceptron
AD	Automatic Differentiation
FWI	Full Waveform Inversion
TV	Total Variation Regularizer
ADAM	Adaptive Moment Estimation
L-BFGS	Limited Memory Broyden Fletcher Goldfarb Shanno Algorithm
GPUs	Graphics Processing Units

CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition

Seismology is a scientific discipline dealing with seismic waves propagating through (inside and around) the Earth. One of the main objectives of investigating seismic signals is retrieving useful information about the Earth from regional to global scales. For exploration purposes, for example, seismic waves are processed and interpreted for prospecting subsurface resources, such as fossil fuels, geothermal, and minerals. While for the global scale it enables us to understand the inner structures of the Earth and provides a crucial understanding of the formation of the planet, tectonic mechanisms as well as earthquakes and volcanos.

In geophysics, the main purpose is mostly to infer some physical properties of the subsurface from observations. The link between these observations and unknown parameters is embedded in a mathematical expression under certain assumptions and approximations. The eikonal equation, which originally appeared on wave and ray optics is a mathematical representation providing a connection between traveltimes and the wavefront propagation velocities. It is a non-linear, first-order partial differential equation (PDE) and belongs to the family of Hamilton-Jacobi equations [19].

Apart from geophysics, the equation has wide applications in various science and engineering disciplines. For example, in medical applications, it is used for tumor growth modeling [18] and activation sequences identification from activation maps measured at the outermost protective layer of the heart [11]. In robotics, the equation find its place for navigation and path planning [9]. The solution of the equation is

also employed in a wide range of computer vision tasks such as an optimal algorithm for shaping from shading [14], surface segmentation of thin structures [8] and shape analysis [15]. In addition to shape recovery, Malladi and Sethian [23] used the PDE for noise removal and image enhancement.

The computation of the traveltimes from the velocity function is needed in many modeling tasks and can be effectively performed by solving the eikonal equation using the Fast Marching Method (FMM) or the Fast Sweeping Method (FSM). The former uses a finite-difference operator to track the minimum traveltime with Dijkstra-like sorting and updating [30]; the latter, on the other hand, is a Gauss-Seidel-based iterative method using upwind difference for discretization [37].

Oppositely, predicting the velocity field from the traveltimes is an inverse problem and can be solved by a gradient-based optimization with forward and backward propogation of the traveltime fields of the eikonal equation. It means the inversion task amounts to solving forward and inverse problems consecutively and iteratively. In the forward problem stage, the eikonal equation is solved for traveltimes on rectangular grids with the algorithms mentioned earlier. While, in the inversion stage, one may try to minimize the objective function, which is the misfit between the observed first arrival times on the receivers and the modeled ones by linearizing the tomography operator. The linearized system is then updated iteratively by employing any gradient-based approaches. This method, however, demands the explicit computation of the Jacobian matrix, which results in high computational costs. To mitigate this issue, the traveltime tomography problem can be solved directly as a non-linear optimization setting utilizing the adjoint state method [20, 26]. Though this non-linear optimization approach avoids the explicit computation of the Jacobian matrix, its gradient may ignore some information available along the shot dimension, which can be resulted in undesired velocity estimations [21].

Physics informed neural networks (PINNs) [28], which combine the underlying physics of a tackled problem with deep learning (DL) have recently emerged as a powerful tool for solving both forward and inverse problems. This neural network (NN) aided approximators capable of considering physics are effective and efficient for ill-posed inverse problems [13]. Therefore, in this thesis, we approach the traveltime tomogra-

phy problem in a novel way leveraging PINNs.

1.2 Contributions and Outline of the Thesis

The main contributions of this work can be summarized as follows:

- For PINNs, there is no need to have a huge amount of data that is required in other deep neural networks (DNNs) based inversion approaches conducted in a supervised manner. Therefore, using PINNs is computationally cheap as one has not to generate lots of feature-label pairs (velocity models and the travel-times).
- Successful implementations of traditional gradient-based inversion frameworks strongly depend on initial models. Starting velocities far away from the solution may cause an algorithm to trap to local minima or even diverge. PINN inversion does not depend on an initial model.
- Contrary to the conventional approaches (finite difference, finite elements), PINN methodology is mesh-free, which can be useful when handling complex computational boundaries, such as models having irregular topography.
- In a non-linear optimization setting, traveltime tomography may suffer from unwanted velocity estimates because of the contradicting information in its gradient for different shots. As PINN learning is based on training DNNs, this can be avoided.

This thesis is organized as follows. In Chapter 2, a brief summary of traveltime tomography problem and background for PINNs, are given. Implementation of the proposed method to synthetic examples and corresponding results are provided in Chapter 3. Then, in Chapter 4, we apply the proposed method to real field data. Finally, Chapter 5, summarizes the study and touches on possible extensions of the thesis.

CHAPTER 2

PRELIMINARIES

2.1 Eikonal Equation Based Traveltime Tomography

2.1.1 Eikonal Equation

In an acoustic constant density media, seismic waves with a wave speed of v propagate according to the following equation:

$$\nabla^2\psi = \frac{1}{v^2}\frac{\partial^2\psi}{\partial t^2}, \tag{2.1}$$

where ∇^2 is the Laplace operator applied to the scalar acoustic pressure ψ. For high frequencies, WKB (Wentzel–Kramers–Brillouin) expansion reduces to the following asymptotic approximation, which satisfies the (2.1).

$$\psi = A(x)e^{-i\omega T(x)}, \tag{2.2}$$

where ω is the angular frequency; $T(x)$ is a phase function (traveltime), and $A(x)$ is the amplitude coefficient of the oscillatory function. Applying the divergence of the gradient to the geometric optics approximation of the wave equation (applying the Laplace operator on (2.2)) yields:

$$\begin{aligned}\nabla^2\psi = \nabla^2 Ae^{-i\omega T} &- i\omega\nabla A\cdot\nabla Te^{-i\omega T}\\ &- i\omega\nabla A\cdot\nabla Te^{-i\omega T}\\ &- i\omega A\nabla^2 Te^{-i\omega T}\\ &- \omega^2 A\nabla T\cdot\nabla Te^{-i\omega T}.\end{aligned} \tag{2.3}$$

On the other hand, the second derivative of the solution of the wave equation ψ with respect to time t is given

$$\frac{\partial^2 \psi}{\partial t^2} = -\omega^2 A e^{-i\omega T}. \tag{2.4}$$

Plugging the (2.3) and (2.4) into the wave equation produces

$$\nabla^2 A - \omega^2 A|\nabla T|^2 - i\omega(2\nabla A \cdot \nabla T + A\nabla^2 T) = \frac{-A\omega^2}{v^2}. \tag{2.5}$$

This equation has real and imaginary components. Propagation information comes from the real part

$$\nabla^2 A - \omega^2 A|\nabla T|^2 = \frac{-A\omega^2}{v^2}. \tag{2.6}$$

Dividing both sides by $A\omega^2$ and applying high frequency approximation by taking the limit as $w \to \infty$ yields the eikonal equation:

$$|\nabla T(x)|^2 = \frac{1}{v^2}. \tag{2.7}$$

Additionally, amplitude of the propagating waves comes from the imaginary part

$$2\omega \nabla A \cdot \nabla T + \omega A \nabla^2 T = 0. \tag{2.8}$$

Frequency terms vanish and this gives the transport equation:

$$2\nabla A \cdot \nabla T + A\nabla^2 T = 0. \tag{2.9}$$

In this thesis, we ignore the amplitude part and consider only the propagation of wavefronts described in (2.7). Physically, the eikonal equation states that the gradient of a wavefront (traveltime) at position x is given by the square of the reciprocal of the velocity at that position. Also, the direction of the wave propagation is described by the direction of the gradient of traveltime.

2.1.2 Tomographic Reconstruction

Equation (2.7) presents a relation between traveltime and velocity field. Finding a solution in the model space (velocity) from data (traveltime) is called traveltime tomography. More formally, it is a data fitting procedure that can be written as:

$$\min_{\mathbf{v}} \frac{1}{2} \|\mathbf{t}^{\text{pre}} - \mathbf{t}^{\text{obs}}\|_2^2. \tag{2.10}$$

Here the notation $\|\cdot\|$ denotes the L_2 norm taken in Euclidean Space; $\mathbf{t}^{\text{pre}}$ represents the computed traveltimes from a forward modeling; $\mathbf{t}^{\text{obs}}$ represents the observed traveltimes in the domain of interest. The objective in (2.10) can be rewritten in terms of the dot product of the residual vector $\mathbf{r} = \mathbf{t}^{\text{pre}} - \mathbf{t}^{\text{obs}}$ such that:

$$\min_{\mathbf{v}} \tfrac{1}{2}\|\mathbf{r}\|^2 = \tfrac{1}{2}(\mathbf{r}\cdot\mathbf{r}). \tag{2.11}$$

If we take the derivative of (2.11) with respect to the model parameters $\mathbf{v}$, we have

$$\tfrac{1}{2}\frac{\partial}{\partial \mathbf{v}}(\mathbf{r}\cdot\mathbf{r}) = \mathbf{G}^T\cdot\mathbf{r}, \tag{2.12}$$

where $\mathbf{G}$ is the Jacobian matrix and defines the sensitivity of the data residuals with respect to velocity perturbations. This expression gives us the gradient that is needed to update the model parameters.

The solution of the eikonal equation (2.7) depends on the source location x_s, and has a boundary condition of $T(x_s) = 0$. Hence, if we consider multiple shot case, and take the total number of shot as N_s, the total Jacobian matrix will be the concatenation of individual ones

$$\mathbf{G} = \begin{bmatrix}\mathbf{G}_1 & \mathbf{G}_2 & \dots & \mathbf{G}_{N_s}\end{bmatrix}^T. \tag{2.13}$$

Therefore, the total gradient term will be

$$\mathbf{g} = \sum_{i=1}^{N_s} \mathbf{G}_i^T\cdot\mathbf{r}_i. \tag{2.14}$$

where i stands for the ith shot.

More accurate update in the solution space can be accomplished by considering the second derivatives $\mathbf{G}^T\mathbf{G}$, the Hessian matrix. In this case, the model perturbation is obtained by premultiplying the inverse Hessian with the gradient:

$$\delta_{\mathbf{v}} = \left[\mathbf{G}^T\mathbf{G}\right]^{-1}\mathbf{g}. \tag{2.15}$$

Because large problems tend to suffer from computing Hessian matrix, we will rely on conjugate search directions for all the conventional tomography implementations. More specifically, the conjugate-direction (CD) [6] method is chosen as a traditional tool for tomography in the context of this thesis. Algorithm 1 summarizes the tomographic reconstruction.

Algorithm 1: Traveltime Tomography Algorithm

Input: t^{obs} (Observed traveltimes)

$v_0 \leftarrow 0$ (Model Initialization)

$k \leftarrow 0$ (Step Initialization)

while *data residual not converged* **do**

  $t_k^{\text{pre}} \leftarrow v_k$ (FMM)

  $r_k \leftarrow t^{\text{obs}} - t_k^{\text{pre}}$

  $J_k \leftarrow \frac{\partial t_k^{\text{pre}}}{\partial v_k}$

  $r \leftarrow r_k$

  $m \leftarrow v_k$

  while *predetermined iteration number is not reached* **do**

    $\Delta m \leftarrow J_k^* r$

    $\Delta r \leftarrow J_k \Delta m$

    $(m, r) \leftarrow \text{cgstep}(m, r, \Delta m, \Delta r)$

  end

  $g_k \leftarrow m$

  $v_{k+1} \leftarrow v_k + \alpha g_k$

  $k \leftarrow k + 1$

end

The inversion process starts with an initial velocity model $\mathbf{v}_0$. Forward modeling is performed by the Fast Marching Method [30], which solves the eikonal equation on spaced grids by finding the fastest arrivals at grid points. From the derivatives of the predicted times with respect to the background velocity model, forward and adjoint operators are constructed. The operators go into the inner loop that performs the conjugate-direction method updates via cgstep() subroutine written in Fortran 90 [6]. The subroutine takes the background model, background data residual, gradient in the model space, and the conjugate gradient, and outputs model update and data residual. Then the model update is used to update the background model. Therefore, the outer part of the framework performs linearization updates, while the inner part does the conjugate gradient iterations.

2.2 Physics Informed Neural Networks (PINNs)

2.2.1 Deep Feedforward Neural Network

Deep feedforward neural network, also known as a multilayer perceptron (MLP), is a simple structure of function approximator commonly used in machine learning. It maps a given input $\mathbf{x}$ to an output through successive affine-linear mappings and non-linear transformations, which can be described in a composite function for K-layer neural network such that:

$$\mathcal{N}_{\boldsymbol{\theta}}(\mathbf{x}) = L_K \circ \sigma \circ L_{K-1} \cdots \circ \sigma \circ L_1, \tag{2.16}$$

where $\circ$ denotes function composition; L is an affine mapping and is given more explicitly $L_k\mathbf{x}_k = \mathbf{W}_k\mathbf{x}_k + \mathbf{b}_k$ in which $\mathbf{W}_k \in \mathbb{R}^{n_{k+1}\times n_k}$ are the weights of the network, $\mathbf{b}_k \in \mathbb{R}^{n_{k+1}}$ represents the bias vector, $\mathbf{x}_k \in \mathbb{R}^{n_k}$ is an input vector for k representing the k-th hidden layer including n_k neurons; σ is a nonlinear activation function, and $\boldsymbol{\theta} = \left\{\mathbf{W}_k, \mathbf{b}_k\right\}_{1\leq k\leq K}$. An illustration of a deep feedforward neural network is given in Figure 2.1.

2.2.2 PINNs as a PDE Solver and Parameter Estimator

Let $\Omega \subset \mathbb{R}^n$ be the domain of interest with a boundary defined as $\partial\Omega$ for some $n \geq 1$. An abstract representation of a PDE and relevant boundary conditions can be written as:

$$\begin{aligned} \mathcal{D}(\xi(\mathbf{x})) &= 0, \quad \text{in } \Omega \\ \mathcal{B}(\xi, \mathbf{x}) &= 0, \quad \text{in } \partial\Omega \end{aligned} \tag{2.17}$$

where $\mathcal{D}$ is a spatial-temporal differential operator operating on the PDE solution ξ; $\mathcal{B}(\xi, \mathbf{x})$ can be any type of boundary conditions (Dirichlet, Neumann, Robin or mixed). Moreover, we assume that $\mathbf{x}$ includes a temporal parameter t for time-dependent PDEs.

PINNs solve a PDE by satisfying the physics and boundary conditions subject to a neural network $\mathcal{N}_{\boldsymbol{\theta}}(\mathbf{x})$ that approximates to the solution of a PDE $\xi(\mathbf{x})$. Any derivatives of the network outputs $\tilde{\xi}(\mathbf{x})$ with respect to the inputs defining a PDE as well

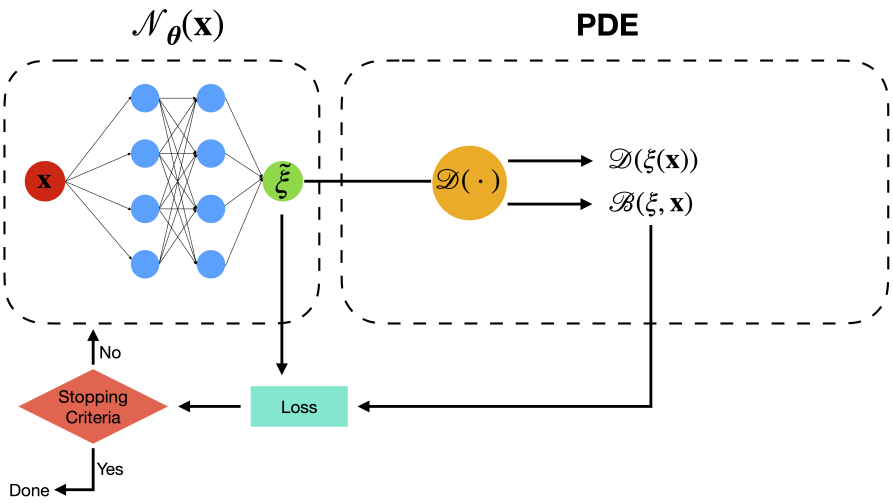


Figure 2.1: An illustration of a physics informed neural network procedure.

as needed derivatives for boundary conditions are computed via automatic differentiation (AD) [3], which applies the chain rule to compute the derivatives of a compositional function. If we define the training data for a PDE as $\Omega_p \subset \Omega$ and for boundary conditions as $\partial\Omega_b \subset \partial\Omega$, the total loss function can be written such that:

$$\mathcal{L}(\boldsymbol{\theta}) = w_p \frac{1}{|\Omega_p|} \sum_{\mathbf{x}\in\Omega_p} ||\mathcal{D}(\tilde{\xi}(\mathbf{x}))|| + w_b \frac{1}{|\partial\Omega_b|} \sum_{\mathbf{x}\in\partial\Omega_b} ||\mathcal{B}(\tilde{\xi}, \mathbf{x})||. \tag{2.18}$$

Here, the L_2 norm $\|\cdot\|$ is applied to the PDE residual in the first term, and to the boundary conditions in the second term of the loss function; w_p and w_b are appropriately chosen weights.

The objective is to find an optimal $\boldsymbol{\theta}$ which minimizes the loss function $\mathcal{L}(\boldsymbol{\theta})$ by a chosen gradient based optimizer. The illustration of a PINN procedure is given in Figure 2.1.

Considering an inverse problem, let's rewrite the definition of an abstract PDE (2.17) parameterized by $\boldsymbol{\gamma}$ such that:

$$\mathcal{D}(\xi(\mathbf{x}); \boldsymbol{\gamma}) = 0 \quad \mathbf{x} \in \Omega. \tag{2.19}$$

Moreover, assume we know the solution $\xi(\mathbf{x})$ on some points $\Omega_d \subset \Omega$. In this case, the unknown parameter $\boldsymbol{\gamma}$ can also be predicted by PINNs just by adding an extra loss

term to the loss defined earlier, in a way that

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\gamma}) = w_p \frac{1}{|\Omega_p|} \sum_{\mathbf{x} \in \Omega_p} ||\mathcal{D}(\tilde{\xi}(\mathbf{x}); \boldsymbol{\gamma})|| \\ + w_b \frac{1}{|\partial\Omega_b|} \sum_{\mathbf{x} \in \partial\Omega_b} ||\mathcal{B}(\tilde{\xi}, \mathbf{x})|| \\ + w_d \frac{1}{|\Omega_d|} \sum_{\mathbf{x} \in \Omega_d} ||\tilde{\xi}(\mathbf{x}) - \xi(x)||,\end{aligned} \tag{2.20}$$

where w_d is the corresponding weight for data fitting term. Hence, the goal of the unconstrained optimization problem becomes

$$\arg\min_{\boldsymbol{\theta}, \boldsymbol{\gamma}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\gamma}). \tag{2.21}$$

CHAPTER 3

PINN ENABLED TRAVELTIME TOMOGRAPHY

Seismic first arrival traveltime picking and then inverting them have been used as a common approach for visualizing near-surface seismic velocities. For example, Zhu at al. [38] illustrated that the traveltime tomography can estimate near surface velocities more accurately than refraction statics methods. Zhang and Toksoz [36] developed a nonlinear refraction traveltime tomography method using ray tracing for the forward calculations. Taillandier at al. [33] introduced the adjoint-state technique for computation of the gradient of the misfit function that avoids ray tracing and estimation of the costly Jacobian matrix in the inverse problem. Accurately estimating long-wavelength velocity structures in near-surface is essentially important for engineering and environmental geophysical applications. Moreover, reliably retrieved macro-feature velocities can serve as an initial for full-waveform inversion (FWI) technique [34, 31, 5, 27, 29, 35] to obtain more accurate near-surface velocity models.

Therefore, in this chapter, we describe the PINN-based traveltime inversion as a new tool for retrieving near-surface velocity models and show its effectiveness through synthetic examples by comparing it with the traditional gradient-based inversion.

3.1 Theory

The residual of the eikonal equation in a 2D domain (Ω) can be written as

$$r(\boldsymbol{x}) := \sqrt{\nabla T(\boldsymbol{x}) \cdot \nabla T(\boldsymbol{x})} - S(\boldsymbol{x}) = 0, \quad \boldsymbol{x} \in \Omega, \tag{3.1}$$

where $S(\boldsymbol{x})$ is the slowness which is the reciprocal of the velocity field $V(\boldsymbol{x})$.

we use two separate artificial neural networks composed of non-linear functions to approximate traveltimes $T(\boldsymbol{x})$ and velocities $V(\boldsymbol{x})$ in the eikonal equation, such that

$$\begin{aligned} \tilde{T}(\boldsymbol{x}) &= \mathcal{N}_{\mathcal{T}}(x_s, \boldsymbol{x}; w_T, b_T), \\ \tilde{V}(\boldsymbol{x}) &= \mathcal{N}_{\mathcal{V}}(\boldsymbol{x}; w_V, b_V), \end{aligned} \tag{3.2}$$

where $\mathcal{N}_{\mathcal{T}}$ and $\mathcal{N}_{\mathcal{V}}$ are the feed-forward neural networks for traveltime and velocity respectively. Here the trainable parameters are weights and biases shown as w_T and b_T for traveltime NN and w_V and b_V for velocity NN. To honor multiple shots on the surface, horizontal coordinates of the source location, denoted as x_s, are used as an additional input to the traveltime network, contrary to the velocity network.

For stability as well as ensuring the solutions remain positive, sigmoid activation function σ is applied for both output of the networks before multiplying them with the maximum expected velocity and traveltime values

$$\begin{aligned} \tilde{T}(\boldsymbol{x}) &= T_{max}\ \sigma(\mathcal{N}_{\mathcal{T}}(x_s, \boldsymbol{x}; w_T, b_T)), \\ \tilde{V}(\boldsymbol{x}) &= V_{max}\ \sigma(\mathcal{N}_{\mathcal{V}}(\boldsymbol{x}; w_V, b_V)), \end{aligned} \tag{3.3}$$

where V_{max} can be decided considering geology of the medium; whereas, T_{max} can easily be obtained from the observed times. The same strategy for constraining the PINN solution as strictly positive was previously used for prediction of activation maps in cardiac electrophysiology [7]. Recovering the possible sharp transitions in the velocity solution, isotropic total variation regularizer (TV) is also added to the problem. Thus, the composite loss function to train both networks concurrently reads:

$$\begin{aligned} \mathcal{L}(w_T, b_T, w_V, b_V) = \alpha \frac{1}{N_S N_T} \sum_{n=1}^{N_S} \sum_{i=1}^{N_T} (\tilde{T}(\boldsymbol{x}_{n,i}) - T_{n,i})^2 \\ + \frac{1}{N_S N_C} \sum_{n=1}^{N_S} \sum_{i=1}^{N_C} (\sqrt{\nabla \tilde{T}(\boldsymbol{x}_{n,i}) \cdot \nabla \tilde{T}(\boldsymbol{x}_{n,i})} - \tilde{S}(\boldsymbol{x}_{n,i}))^2 \\ + \lambda \frac{1}{N_C} \sum_{i=1}^{N_C} \left\| \nabla \tilde{V}(x_i) \right\|. \end{aligned} \tag{3.4}$$

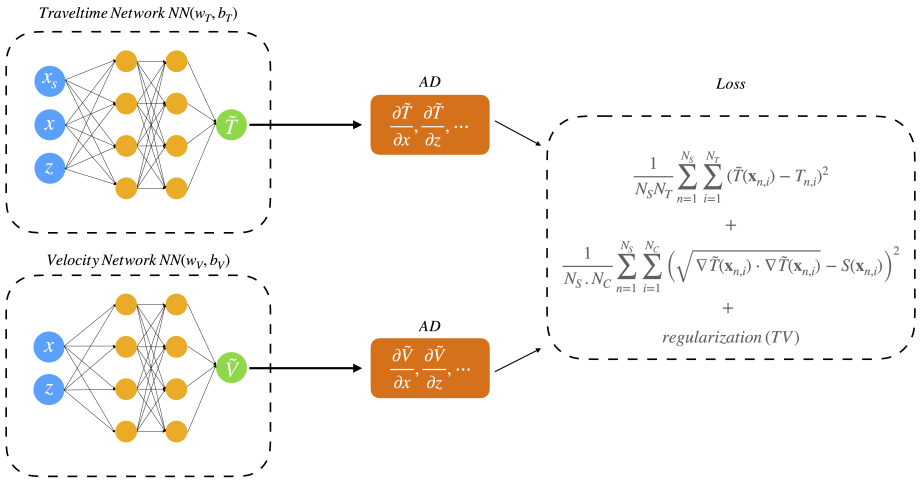
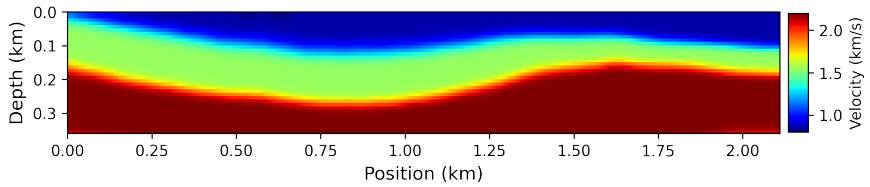


Figure 3.1: Illustration of the PINN framework solving for traveltime tomography problem. Two neural networks are used to approximate the traveltime and the velocity. The automatic differentiation is used to calculate the derivatives used in the loss function. Training is performed with a loss function containing the term that minimizes the residuals of the observed times and the network estimated ones, residual of the eikonal equation, and the regularization term.

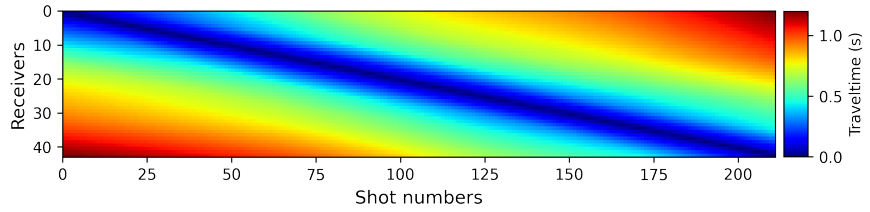
The first term, which corresponds to the third term of the loss function in 2.20 measures the misfit between the observed traveltimes and the network predicted ones on the receiver positions for each source location. α is the weight used as a penalty for this term. N_S denotes the total number of shots and N_T represents the number of receivers for a shot. The second term, which corresponds to the first term of the loss function in 2.20 is the eikonal residual computed at the selected collocation points, shown as N_C, in the computational domain. The third term, on the other hand, introduces TV as a stabilizer to the inverse problem. The impact of it controlled by the regularization parameter λ. Figure 3.1 illustrates the PINN based inversion schematic which is used in this thesis.

Hence, the unconstrained optimization problem seeks to find the optimal network parameters w_T, b_T, w_V, b_V such that

$$\underset{(w_T, b_T, w_V, b_V)}{\arg\min} \quad \mathcal{L}(w_T, b_T, w_V, b_V). \tag{3.5}$$



(a)



(b)

Figure 3.2: (a) An example of a velocity model for testing the proposed PINN enabled inversion and (b) Image representation of the data aquired from the model. There are 43 shots each having 211 observed traveltimes that are placed in rows to form the image.

3.2 Numerical Implementation

First, we test the proposed PINN tomography framework on synthetic examples and then compare the results with the traditional gradient-based traveltime tomography. All PINN enabled tomography implementations are conducted in TensorFlow [1]. For the training, ADAM optimizer [16] with a default learning rate of 0.001 (in our experiments higher learning rates prevent convergence; On the other hand, using lower values than the default one requires more iterations to the solution.) is chosen before switching to the L-BFGS [22] method for the sake of accelerating the convergence. In order to save computational time, all training is performed with a minibatch implementation in which the collocation points are randomly selected from the computational domain using a Latin hypercube sampling [32].

The first model example representing a near-surface is a three-layered model with minor folds (Figure 3.2a). The model size is assumed 36 $\times$ 211 and sampled with a uniform spacing of 10 m both vertically and horizontally ($\Delta z = \Delta x = 10\,m$). To model the traveltimes, we employ the fast marching method, and use 43 shots with a regular interval of 50 m on the surface. There are 211 receivers for each shot, and thus, 43$\times$211 observed surface traveltime data for the PINN implementation (Figure 3.2b).

From the computational domain 300 samples are chosen with the Latin hypercube sampling method to train the model. After examining the outputs of different settings of the hyperparameters, we use the following tunings: 7 hidden layers with 50 neurons for the traveltime network; for the velocity, 5 hidden layers with 50 neurons each are decided sufficient as training the velocity network is easier than training the traveltime network. The weight for the TV term is chosen as $\lambda = 10^{-7}$. Moreover, we use $\alpha = 100$ as the weight for the data mismatch term in the loss. Penalizing this specific term during the training helps to improve the accuracy of the retrieved model. The neural network weights are initialized with Xavier initializer [10].

The estimated velocity model after 20,000 ADAM iterations (Figure 3.3) is followed by the L-BFGS is shown in Figure 3.4. The retrieved model successfully captures the main features of the true model (Figure 3.2a).

To be more specific, we additionally perform traditional traveltime tomography using the conjugate-direction (CD) method [6]. The acquisition geometry is the same as for the PINN tomography and the tomographic inversion is obtained by implementing 10 linearization updates each having 30 conjugate-gradient iterations. The convergence history for the traditional approach is presented in Figure 3.5. Starting velocity model for the inversion is obtained by taking the vertical profile from the true model (Figure 3.2a) and then applying strong smoothing. The inverted model and the initial one is given in Figure 3.6. Even though the inversion is started with a priori knowledge of the vertical profile from the region, the inversion cannot approach to the true velocities in many places as the optimization traps in local minimum points.

The next example for testing the proposed approach is a more complicated near-surface model containing fault systems (Figure 3.7a). The acquisition parameters selected in the first example is again employed for acquiring the data (Figure 3.7b). This time, PINN inversion is performed by using 40,000 Adam iterations (Figure 3.8) with a minibatch size of 500 and followed by the L-BFGS algorithm. The final inversion result is provided in Figure 3.9.

Again for comparison, traditional tomography is implemented. No further improvement in the convergence is observed after the 10th linearization step (Figure 3.10). Although the inverted model (Figure 3.11a) is generally similar to the predicted model

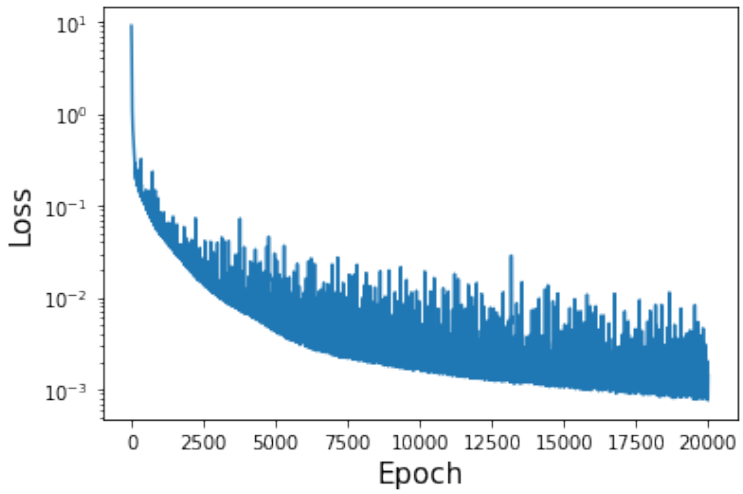


Figure 3.3: Loss curve of the PINN tomography after 20,000 ADAM iterations using minibatch implementation with a batch size of 300.

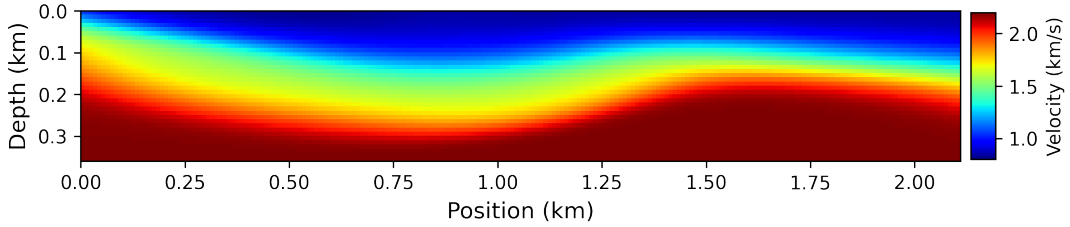


Figure 3.4: PINN predicted velocity model after L-BFGS iterations.

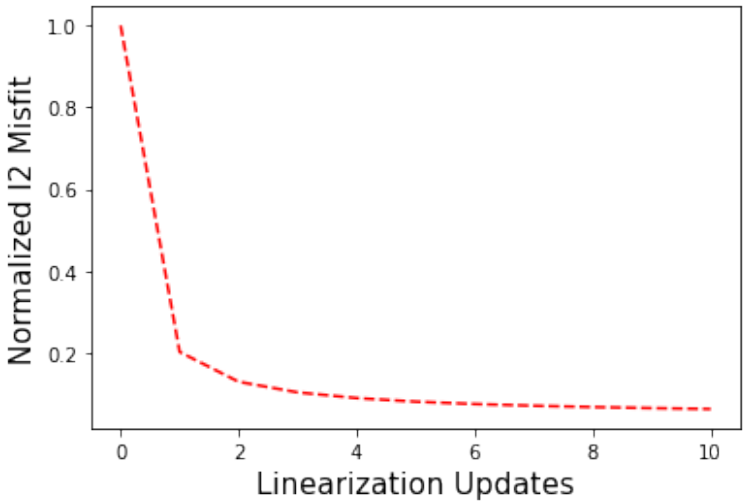
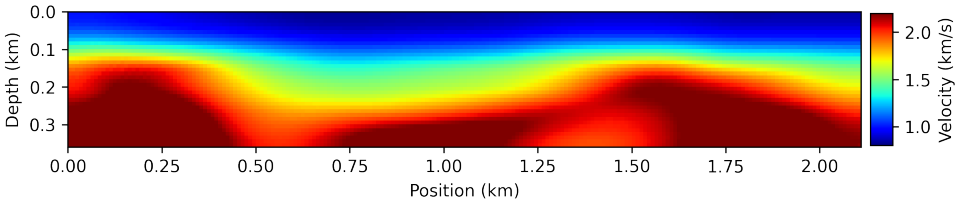
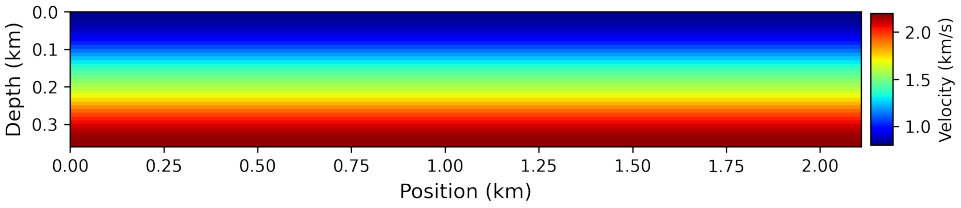


Figure 3.5: Convergence history of the gradient-based tomography after 10 linearization updates. 30 conjugate-gradient iterations are used for each linearization.



(a)



(b)

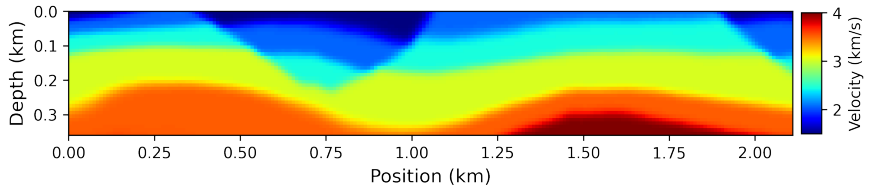
Figure 3.6: (a) Inverted model from the conventional tomography and (b) Starting velocity model for tomography.

by the PINN (Figure 3.9), it overestimates the velocities in some of the areas in the model probably bacause of the contradicting gradient information coming from each shot.

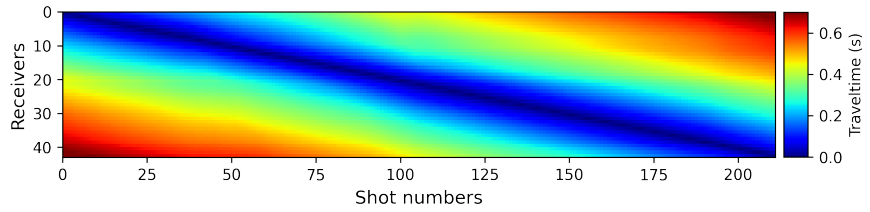
To present a quantitative assessment of the results from both examples, we compare vertical velocity profiles obtained from four different positions (Figure 3.12 and Figure 3.13) as well as the percent error maps (Figure 3.14). As expected from the traveltime tomography, both methods provide a smooth representation of the actual models. However, in deeper parts of the models traditional approach tends to deviate from the true velocities, which is more observable especially for the first example. As expected from the traveltime tomography, both methods provide a smooth representation of the actual models. Also, the percentage velocity errors from the PINN-based approach are generally lower than the errors from the classical method.

3.3 Discussion of the Synthetic Results

PINN-based traveltime tomography clearly shown from the synthetic experiments, can be used as a reliable tool alternative to conventional tomography. It has a significant advantage over the traditional tool in that it does not need to have a good initial guess which requires knowing a priori information on the investigated area. Neverthe-



(a)



(b)

Figure 3.7: (a) An example of a velocity model for testing the proposed PINN enabled inversion and (b) Image representation of the data aquired from the model. There are 43 shots each having 211 observed traveltimes that are placed in rows to form the image.

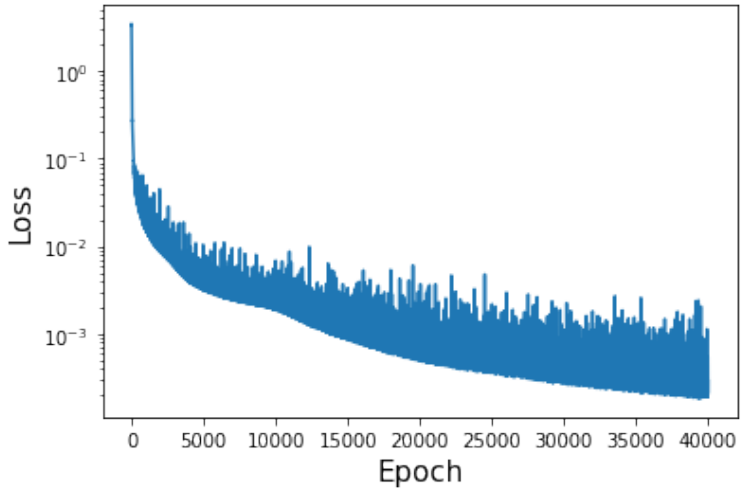


Figure 3.8: Loss curve of the PINN tomography after 40,000 ADAM iterations using minibatch implementation with a batch size of 500.

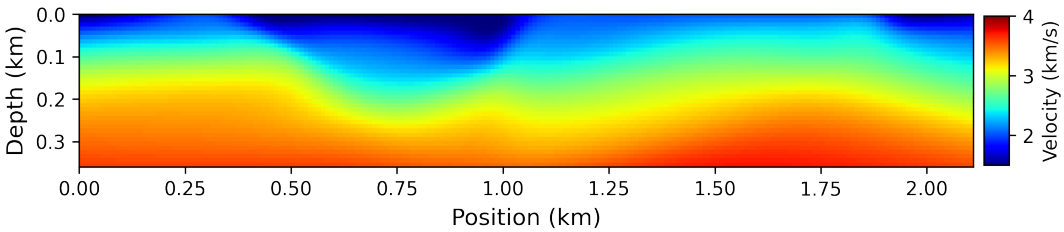


Figure 3.9: PINN predicted velocity model after L-BFGS iterations.

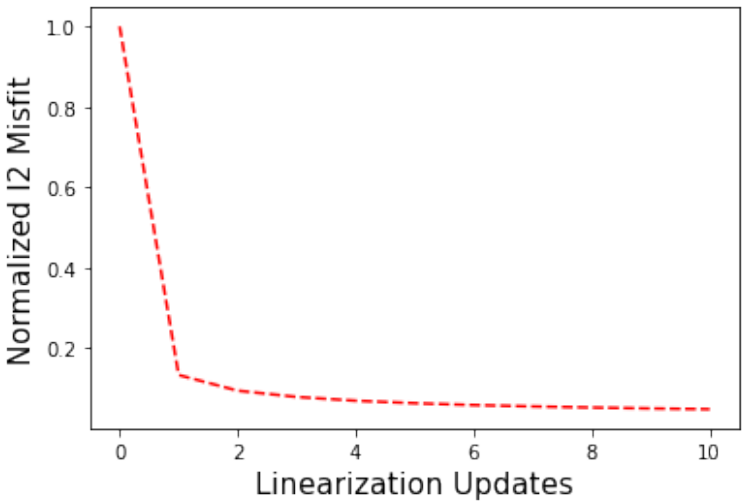
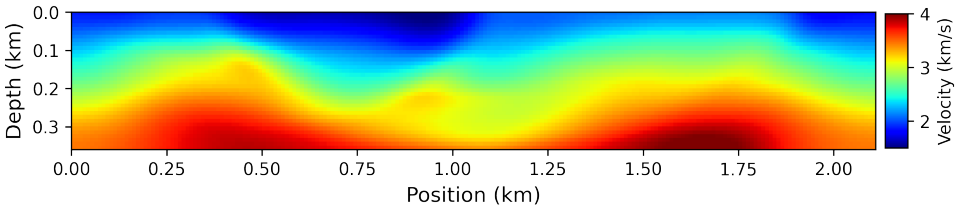
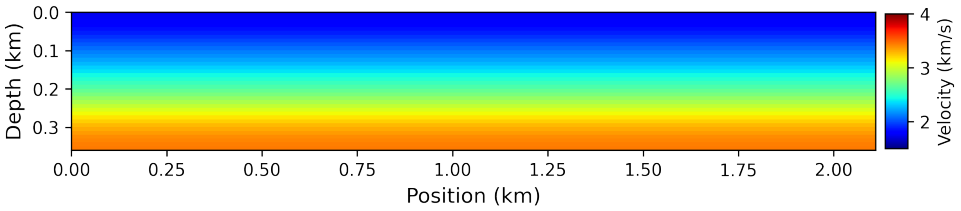


Figure 3.10: Convergence history of the gradient-based tomography after 10 linearization updates. 30 conjugate-gradient iterations are used for each linearization.



(a)



(b)

Figure 3.11: (a) Inverted model from the conventional tomography and (b) Starting velocity model for tomography.

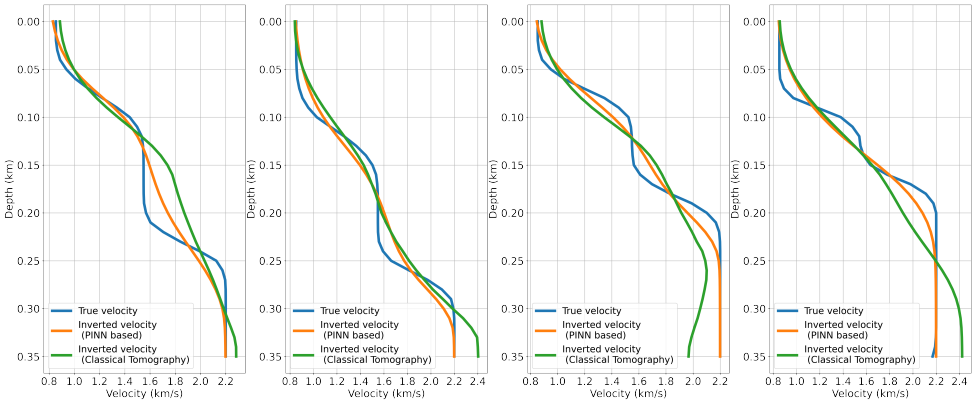


Figure 3.12: Vertical velocity profiles at 0.4 km, 0.9 km, 1.3 km and 1.9 km from left to right respectively for the first example.

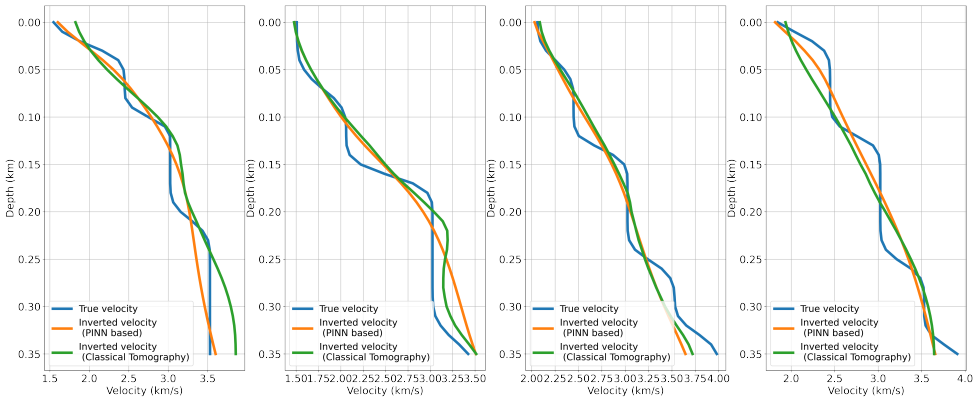
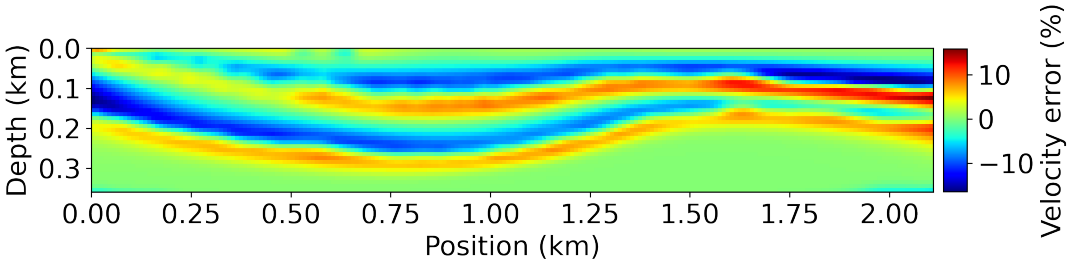
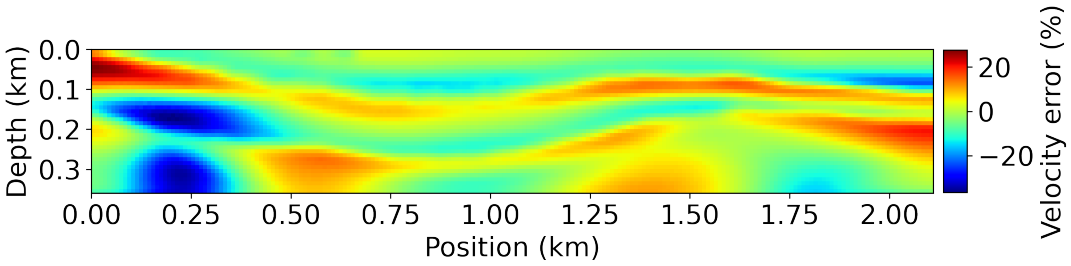


Figure 3.13: Vertical velocity profiles at 0.4 km, 0.9 km, 1.3 km and 1.9 km from left to right respectively for the second example.

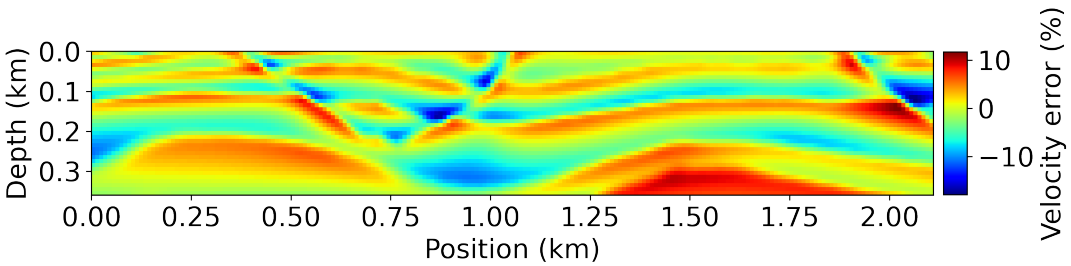
less, the selection of hyperparameters (network architectures, number of optimization iterations, minibatch size, weights of the loss components) plays an important role in the accuracy of the retrieved model. Among them, special attention needs to be paid to balance the weights of the loss components.

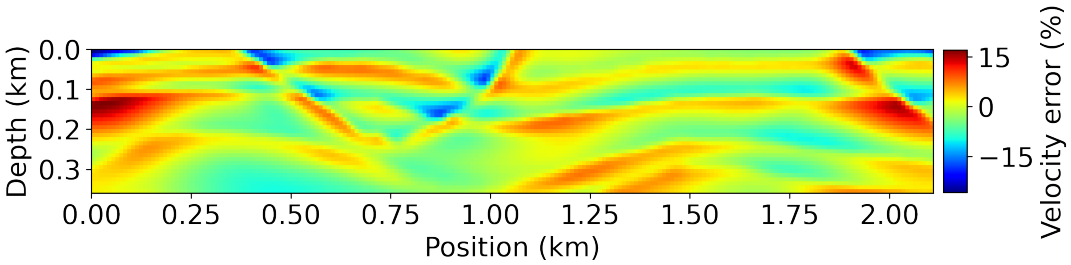
We achieved robust convergence in the examples by taking the weight of the data fitting term 100 of the PDE loss (eikonal) term. The influence of the weight of the regularizer on the problem is another important factor so that careful consideration needs to be given to decide the acceptable value for the weight of this specific term. Therefore, optimizing the weights for each component of the loss along with the network weights could be a solution to this issue; thus removing the need for time-consuming trial-and-error tasks.





(a)





(b)

Figure 3.14: (a) Percentage error maps between the true model and the inversion results from the PINN approach (top) and the conventional approach (bottom) for the first example, and (b) for the second example, error map of PINN approach (top) and error map of conventional approach (bottom).

CHAPTER 4

FIELD DATA EXAMPLE

In this chapter, the proposed PINN tomography to a field data is applied and we compare the result with the previous findings from the region.

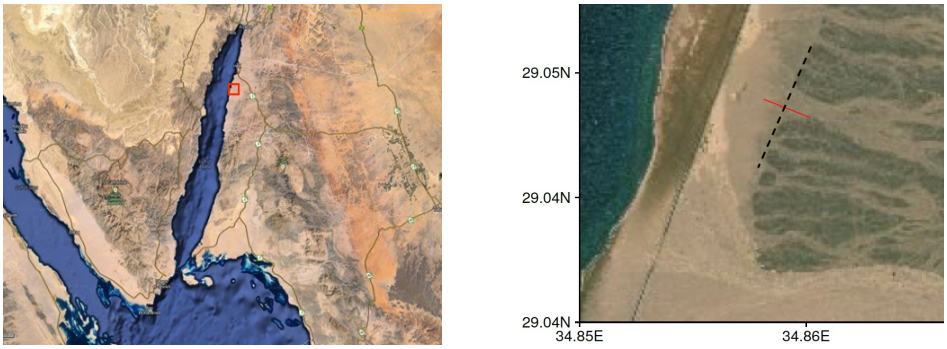
4.1 Study Area and Data Acquisition

The study area is located on the Gulf of Aqaba cost in Saudi Arabia (Figure 4.1a). An earthquake along the Dead Sea strike-slip fault with a magnitude of 7.3, which happened in 1995 [17], was affected the area by causing surface raptures that might be parts of the primary faulting system [2]. Geophysical investigations across one of these surface raptures (Figure 4.1b) to locate and characterize the faults in subsurface were previously conducted [12]. As for the seismic prospecting, 2D refraction data[1] were collected along the survey line. A total of 120 common shot gathers with a regular source shift of 2.5 m was performed. For each shot, there were 120 traces at the receivers which were also deployed in a 2.5 m regular intervals. Seismic data were recorded with a 1 ms sampling interval for a total recording time of 0.5 s. An example of a shot gather and corresponding first arrivals is shown in Figure 4.3.

4.2 PINN Implementation of Traveltime Inversion

We perform PINN tomography using 14,400 picked traveltimes (Figure 4.2). The traveltime network has 5 hidden layers with 50 neurons each, as opposed to the ve-

[1] The data is available online and can be accessed from https://repository.kaust.edu.sa/handle/10754/627400



(a) (b)

Figure 4.1: (a) A Google Earth satellite image showing the study area (courtesy of Sherif M. Hanafy, King Fahd University of Petroleum and Minerals). Red square at the eastern side of Gulf of Aqaba marks the location of the study area. (b) Zoomed in view of the study area. Blacked dashed line indicates the fault raptured at the 1995 earthquake. Red dashed line indicates the seismic profile.

locity network, that has 4 hidden layers of 20 neurons. We constrain the velocity predictions with an upper bound of 2.2 km/s based on the findings from the experiments conducted on the region previously [12]. As in the synthetic examples, we fix the data fitting term 100 times more weight than the eikonal loss. Because the data shows a high signal- to-noise ratio (Figure 4.3), this strategy ensures a robust convergence, we use 50,000 Adam with a minibatch size of 100, with less than 35,000 L-BFGS iterations to train the PINN (Figure 4.4). The optimization is assumed to stop after the loss drops below 5×10^{-4}.

The inverted velocity model is given in Figure 4.5. To evaluate the performance of the trained model, we also present comparisons of the observed times with the predicted ones at the receivers. Surface traveltime comparisons at the 20th, 60th, and the 100th shots are given in Figure 4.6a, Figure 4.6b and Figure 4.6c, respectively. It is observable that the estimated traveltimes are reasonably fit the first-break picks (observed traveltimes).

4.3 Interpretation of the Prediction

The study area were examined by geophysical methods in November 2013 [12]. The resulting refraction tomogram is presented (Figure 4.7a) for comparison with the PINN based tomography result, which is illustrated as contour map (Figure 4.7b).

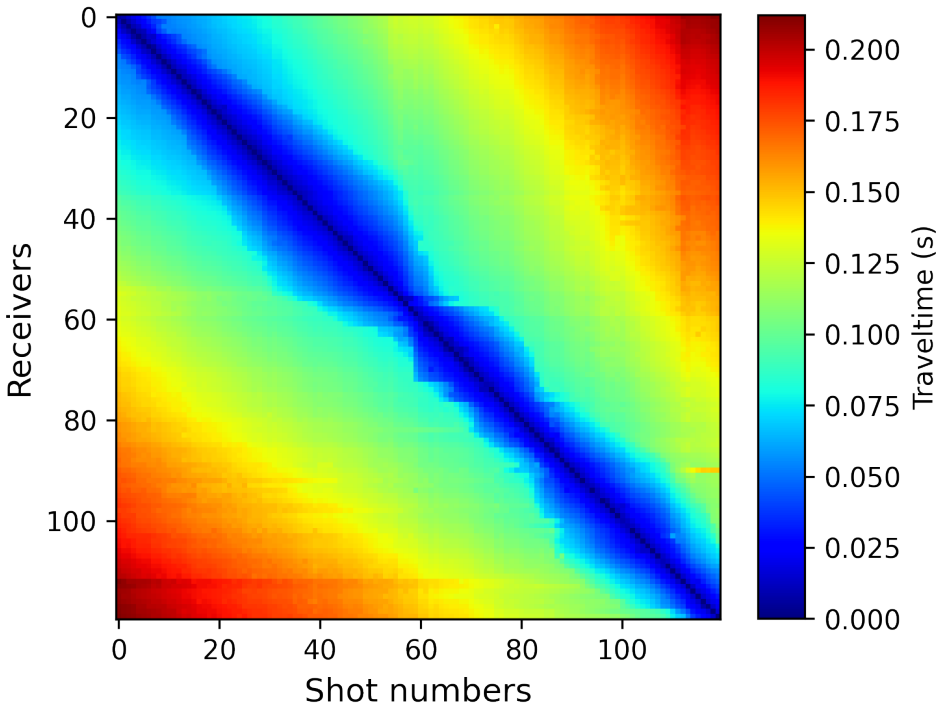


Figure 4.2: Image representation of the field data. There are 120 shots each having 120 observed traveltimes that are placed in rows to form the image.

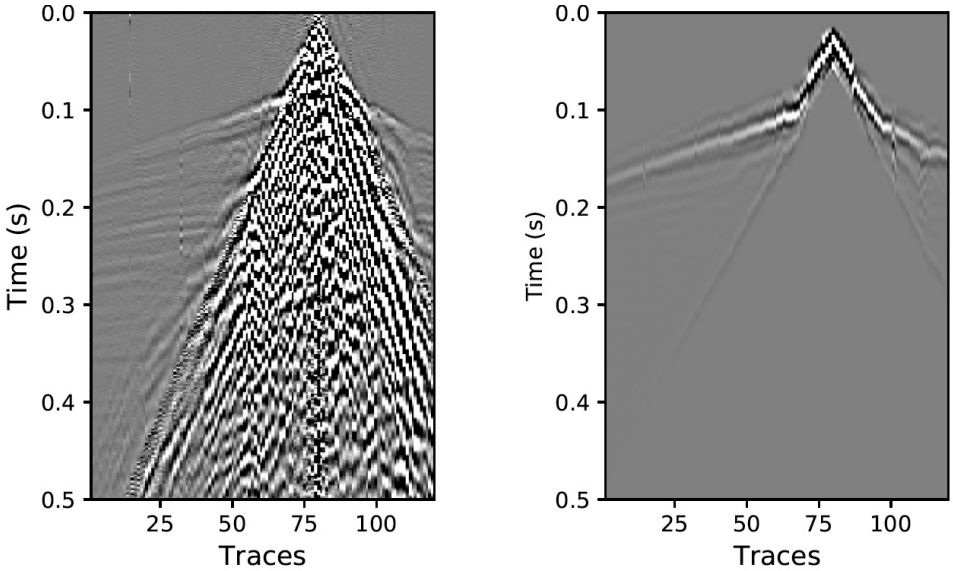
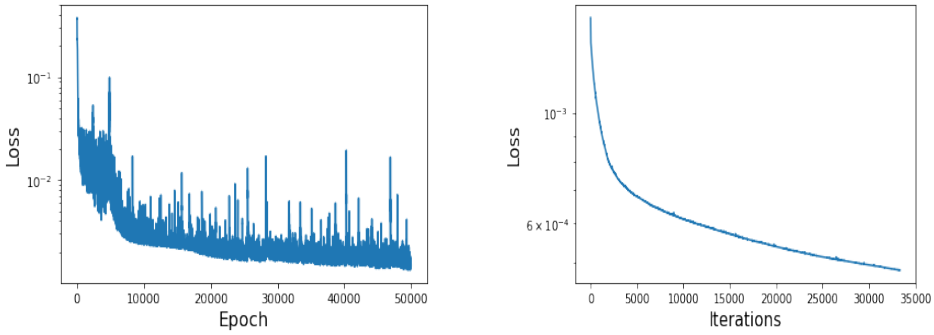


Figure 4.3: An example of a common shot gather (left), and processed of the gather showing the first arrival times (right).



(a) (b)

Figure 4.4: (a) Loss curve of the PINN tomography using ADAM optimizer, and (b) following L-BFGS iterations.

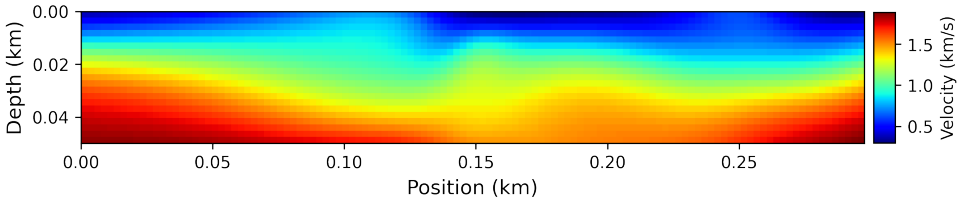
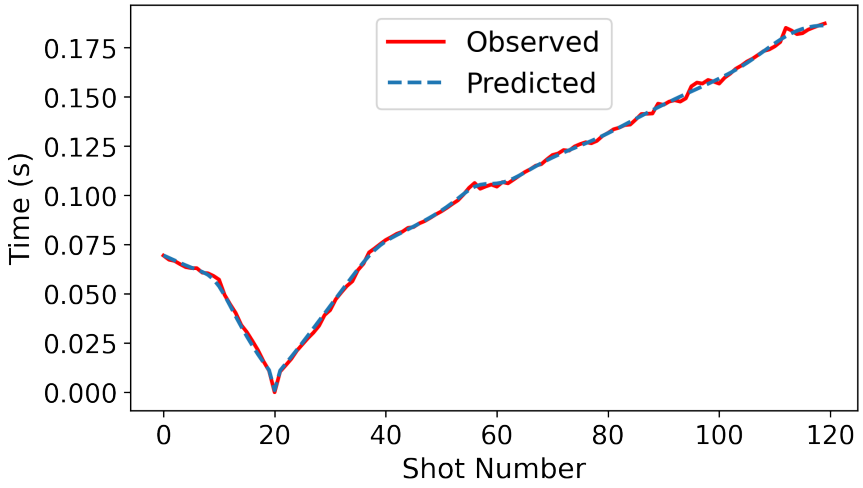


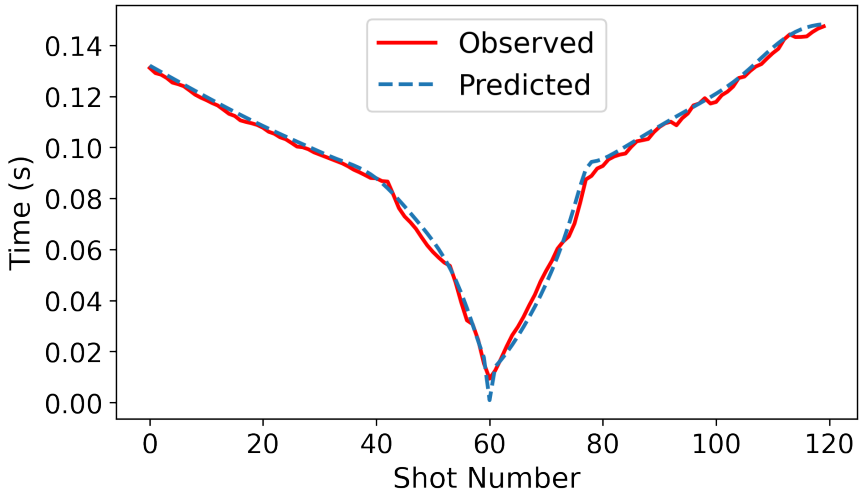
Figure 4.5: PINN predicted velocity model.

In the tomogram created from a conventional tool, a low velocity anomaly, which is called colluvial wedge, was identified between offsets 120 and 145 m. We also observed this low velocity wedge in the PINN based tomogram (shown inside the black circle in Figure 4.7b) nearly at the same offset intervals. This type of low velocity structure can be seen as an indication of the location of a possible existing fault [4, 24, 25]. It is indeed a normal fault cutting the surface approximately at 150 m and noticable at the eastern end of the wedge which coincides with the previous findings.

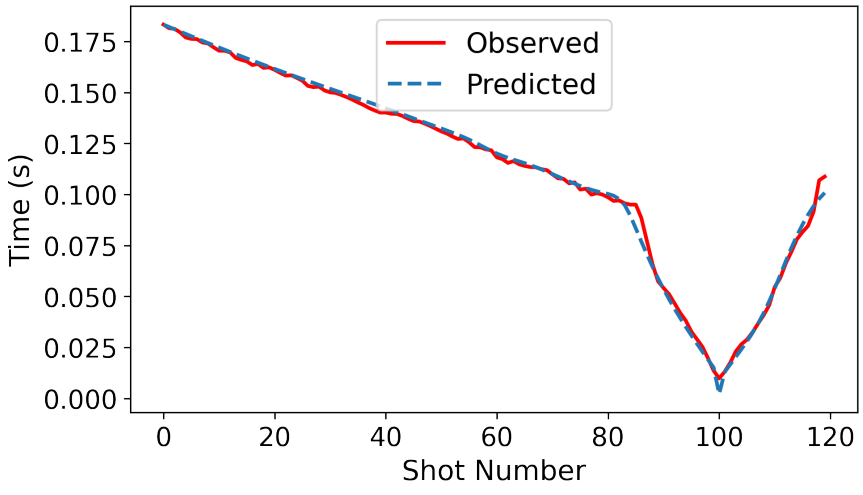
There is another fault, which is not clearly seen in the traditional tomogram, delinated in the eastern side of the PINN estimation between offsets 220 and 240 m (Figure 4.7b). Overall, in lieu of leveraging traditional approaches, it can be easily said that PINN based traveltime inversion of the field data can be used to locate and characterize faults in alluvial sediments.



(a)

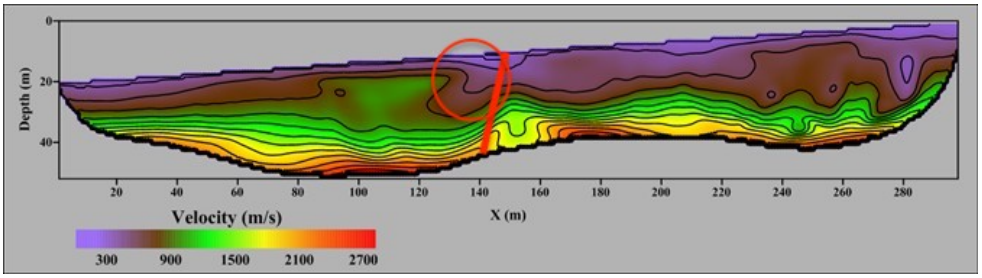


(b)

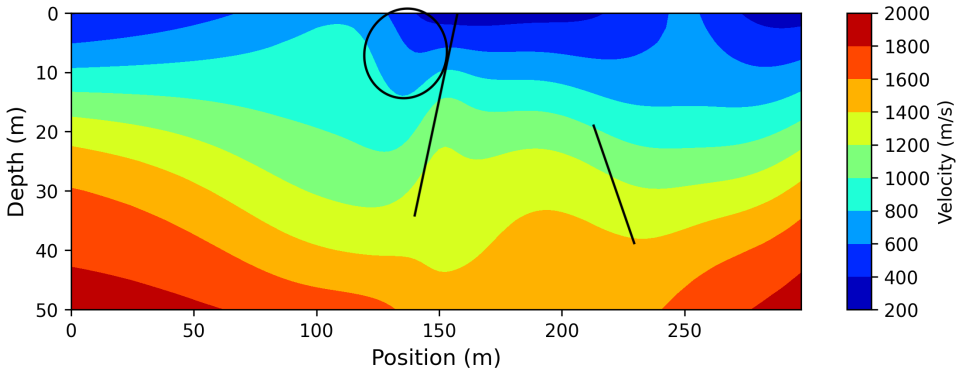


(c)

Figure 4.6: A comparison between the observed traveltimes (solid red) and the PINN estimated ones (dashed blue) at the 20th (a), 60th (b), and the 100th (c) shots.



(a)



(b)

Figure 4.7: (a) Traveltime tomogram obtained from a conventional tool [12]. The red line marks the interpreted fault location, while the red circle is interpreted as a colluvial wedge. (b) Contour map representation of the tomogram predicted by PINN. The blacked lines indicate the interpreted faults, while the black circle is interpreted as a low-velocity wedge.

CHAPTER 5

CONCLUSION AND FUTURE WORK

Traveltime tomography is an important inverse problem used in seismology to retrieve long-wavelength velocity structures of the subsurface. Improving the velocity model results is crucial for obtaining high-quality depth imaging. To this end, we demonstrated a novel approach leveraging training neural networks to invert for velocity from the observed seismic arrival times at the surface stations. Unlike other machine learning models which depend on only data, our proposed approach adheres to the physics inherent in the problem. Utilizing the physics, given in the eikonal equation, as a regularizer of the neural network training helps overcome the ill-posedness of the inverse problem.

This thesis begins with mathematical preliminaries of the traveltime tomography as well as the definition of the physics informed neural networks. In Chapter 2, the idea of how to build physics informed neural networks based traveltime tomography was given and tested on synthetic examples. Moreover, the results obtained from the proposed approach were compared with the traditional tool. Finally, in Chapter 3, the method was applied to field data. Both synthetic tests and field data results showed the reliability of the introduced approach.

Beyond the reliability of the results, It is also important to note the advantages of the proposed approach over conventional algorithms. Probably chief among them is that the PINN-based tomography results do not depend on the starting velocity models, which should be more or less reflect the overall pattern of the true model to achieve a satisfying convergence in traditional methods. Furthermore, neural network training is a mesh-free approach paving the way for flexible use of sources and receivers

without needing them to be on regular grids in the computational domain.

Despite the fact that the computational efficiency of the proposed approach depends on several factors, such as minibatch and network size, the method would be more favorable than the conventional tools when the computational domain is large thanks to the small minibatch implementation during training the networks. In our implementations, we achieved reasonably accurate results using Google's graphics processing units (GPUs) just with a training time of about 30 minutes for the synthetic examples and approximately 1 hour for the field data.

As future work, the suggested method for traveltime tomography in this thesis can easily be extended for the 3D surveys. Moreover, the given algorithm can be an inspiration for solving multi parameters simultaneously when the underlying physics is more complex.

REFERENCES

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: A system for large-scale machine learning., in *OSDI*, volume 16, pp. 265–283, 2016.

[2] J. Angelier, P. Hancock, M. Al-Dail, and N. Sha'at, Etude seismotectonique de failles actives entre Haql et Maqnah, Arabie Saoudite: Trace du séisme du Golfe d'Aqaba, in *16th La Réunion des Sciences de la Terre (RST)*, 1996.

[3] A. G. Baydin, B. A. Pearlmutter, A. Radul, and J. M. Siskind, Automatic differentiation in machine learning: a survey, Journal of Machine Learning Research, 18, pp. 1–43, 2018.

[4] M. L. Buddensiek, J. Sheng, T. Crosby, G. T. Schuster, R. L. Bruhn, and R. He, Colluvial wedge imaging using traveltime and waveform tomography along the Wasatch fault near Mapleton, Utah, Geophysical Journal International, 172, pp. 686–697, 2008.

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

[6] J. Claerbout, *Geophysical image estimation by example*, lulu.com, 2014.

[7] F. S. Costabal, Y. Yang, P. Perdikaris, D. E. Hurtado, and E. Kuhl, Physics-informed neural networks for cardiac activation mapping, Frontiers in Physics, 8(42), 2020.

[8] T. Deschamps and L. D. Cohen, Fast extraction of tubular and tree 3d surfaces with front propagation methods, International Conference on Pattern Recognition, pp. 731–734, 2002.

[9] S. Garrido, D. Álvarez, and L. Moreno, Path planning for mars rovers using the fast marching method, Advances in Intelligent Systems and Computing, 417, pp. 93–105, 2016.

[10] X. Glorot and Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, Sardinia, Italy, 2010.

[11] T. Grandits, K. Gillette, A. Neic, J. Bayer, E. Vigmond, T. Pock, and G. Plank, An inverse eikonal method for identifying ventricular activation sequences from

epicardial activation maps, Journal of Computational Physics, 419, p. 109700, 2020.

[12] S. M. Hanafy, S. Jonsson, and Y. Klinger, Imaging normal faults in alluvial fans using geophysical techniques: Field example from the coast of Gulf of Aqaba, Saudi Arabia, in *SEG Technical Program Expanded Abstracts*, pp. 4670–4674, SEG, 2014.

[13] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, Physics-informed machine learning, Nature Reviews Physics, 3, pp. 422–440, 2021.

[14] R. Kimmel and J. A. Sethian, Optimal algorithm for shape from shading and path planning, Journal of Mathematical Imaging and Vision, 14, pp. 237–244, 2001.

[15] R. Kimmel, D. Shaked, N. Kiryati, and A. Bruckstein, Skeletonization via distance maps and level sets, Computer vision and Image Understanding, 62, pp. 382–391, 1995.

[16] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arxiv [Preprint] arXiv:14126980, 2014.

[17] Y. Klinger, L. Rivera, H. Haessler, and J. C. Maurin, Active faulting in the Gulf of Aqaba: New knowledge from the Mw7.3 earthquake of 22 November 1995, Bulletin of the Seismological Society of America, 89, pp. 1025–1036, 1999.

[18] E. Konukoglu, M. Sermesant, O. Clatz, J. M. Peyrat, H. Delingette, and N. Ayache, A recursive anisotropic fast marching approach to reaction diffusion equation: Application to tumor growth modeling, in *Information Processing in Medical Imaging*, volume 4584, pp. 687–699, 2007.

[19] Y. A. Kravtsov and Y. I. Orlov, *Geometrical Optics of Inhomogeneous Media*, Springer Berlin Heidelberg, 1990.

[20] S. Leung and J. Qian, An adjoint state method for three-dimensional transmission traveltime tomography using first-arrivals, Communications in Mathematical Sciences, 4(1), pp. 249–266, 2006.

[21] S. Li, A. Vladimirsky, and S. Fomel, First-break traveltime tomography with the double-square-root eikonal equation, Geophysics, 78(6), pp. U89–U101, 2013.

[22] D. C. Liu and J. Nocedal, On the limited memory bfgs method for large scale optimization, Mathematical Programming, 45, pp. 503–528, 1989.

[23] R. Malladi and J. A. Sethian, A unified approach to noise removal, image enhancement, and shape recovery, IEEE Transactions on Image Processing, 5, pp. 1554–1568, 1996.

[24] D. Morey and G. T. Schuster, Paleoseismicity of Oquirrh fault, Utah from shallow seismic tomography, Geophysical Journal International, 138(1), pp. 25–35, 1999.

[25] J. Nolan, S. D. Sloan, S. W. Broadfoot, R. McKenna, and O. M. Metheny, Near-surface void identification using MASW and refraction tomography techniques, in *SEG Technical Program Expanded Abstracts*, volume 30, pp. 1401–1405, SEG, 2011.

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

[27] R. G. Pratt, Seismic waveform inversion in the frequency domain, part 1: Theory and verification in a physical scale model, Geophysics, 64(3), pp. 888–901, 1999.

[28] M. Raissi, P. Perdikaris, and G. E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys., 378, pp. 686–707, 2019.

[29] C. Ravaut, S. Operto, L. Improta, J. Virieux, A. Herrero, and P. Dell'Aversana, Multiscale imaging of complex structures from multifold wide-aperture seismic data by frequency-domain full-waveform tomography: application to a thrust belt, Geophysical Journal International, 159(3), pp. 1032–1056, 2004.

[30] J. A. Sethian, A fast marching level set method for monotonically advancing fronts, Proceedings of the National Academy of Sciences, 93(4), pp. 1591–1595, 1996.

[31] P. R. Shaw and J. A. Orcutt, Waveform inversion of seismic refraction data and applications to young pacific crust, Geophysical Journal International, 82(3), pp. 375–414, 1985.

[32] M. Stein, Large sample properties of simulations using latin hypercube sampling, Technometrics, 29, pp. 143–151, 1987.

[33] C. Taillandier, M. Noble, H. Chauris, and H. Calandra, First-arrival traveltime tomography based on the adjoint-state method, Geophysics, 74(6), pp. WCB1–WCB10, 2009.

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

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

[36] J. Zhang and M. N. Toksöz, Nonlinear refraction traveltime tomography, Geophysics, 63(5), pp. 1726–1737, 1998.

[37] H. Zhao, A fast sweeping method for eikonal equations, Math. Comput., 74(250), pp. 603–627, 2005.

[38] X. Zhu, D. P. Sixta, and B. G. Angstman, Tomostatics: Turning-ray tomography + static corrections, The Leading Edge, 11(12), pp. 15–23, 1992.