Anisotropic eikonal solution using physics-informed neural networks

Umair bin Waheed^{*1}, Ehsan Haghighat², and Tariq Alkhalifah³

¹KFUPM, ²MIT, ³KAUST

SUMMARY

Traveltimes are essential for seismic applications ranging from imaging to tomography. Traveltime computations in anisotropic media, which are better representative of the true Earth, require solving the anisotropic eikonal equation. Numerical techniques to solve the anisotropic eikonal equation are known to suffer from instability and increased computational cost compared to the isotropic case. Here, we employ the emerging paradigm of physics-informed neural networks to solve the anisotropic qP-wave eikonal equation. By minimizing a loss function formed by imposing the validity of the eikonal equation, we train a neural network to produce traveltime solutions that are consistent with the underlying partial differential equation. We observe considerably higher accuracy compared to the first-order finite-difference solution using the fast sweeping method. We also show that once the network is trained for a particular source location in a given anisotropic model, the traveltimes for a new source location and/or an updated model can be computed much more efficiently using the pre-trained network. This feature is particularly attractive as it can speed up seismic imaging and inversion applications significantly.

INTRODUCTION

Fast and accurate traveltime computation is essential to the success of many seismic applications including statics and moveout correction (Lawton, 1989), traveltime tomography (Taillandier et al., 2009), microseismic source localization (Grechka et al., 2015), and ray-based migration (Lambaré et al., 2003). Many of the actively explored regions in the world exhibit seismic anisotropy, for example, the North Sea, the Canadian foothills, and offshore West Africa. Therefore, traveltime computation algorithms must honor seismic anisotropy for high-resolution imaging of the subsurface.

The two most popular approaches to compute traveltimes in anisotropic media are ray tracing and the finite-difference solution of the anisotropic eikonal equation. Ray tracing methods compute traveltimes along the characteristics of the eikonal equation by solving a system of ordinary differential equations (Cerveny, 2001). For practical applications, traveltimes computed using ray tracing need to be interpolated on a regular grid. This not only requires additional computational cost but is also quite challenging in complex media in which rays may diverge from each other, resulting in large spatial gaps known as shadow zones. Therefore, numerical solutions of the anisotropic eikonal equation have been a topic of continued research interest.

Fast marching (Sethian, 1999) and fast sweeping (Zhao, 2005) are the two most commonly used algorithms to numerically

solve an eikonal equation. Fast marching methods rely on the fact that the direction of energy propagation of a wavefront, given by the group velocity, is aligned with the wavefront normal direction, given by the phase velocity. While this is true for isotropic media, the phase velocity vector in an anisotropic medium deviates from the group velocity vector.

Fast sweeping, on the other hand, solves an eikonal equation by sweeping the computational domain in alternating directions. The idea is that all characteristic directions can be divided into a finite number of groups and each sweeping iteration covers a group of characteristics. The algorithm converges in a finite number of iterations and is more robust and flexible for general equations than the fast marching method. The complications associated with using fast marching for the anisotropic eikonal equation have resulted in a preference for the fast sweeping method (FSM). However, the higher-order nonlinear terms in the anisotropic eikonal equations require computationally expensive procedures, such as using an iterative fast sweeping algorithm (Waheed et al., 2015) or using the Discontinuous-Galerkin (DG) fast sweeping method (Le Bouteiller et al., 2019).

On a different front, deep learning is fast emerging as a potential disruptive tool to tackle longstanding research problems across science and engineering disciplines. Recent advances in the field of Scientific Machine Learning have demonstrated the largely untapped potential of deep learning for applications in scientific computing. The idea to use an artificial neural network (ANN) for solving a partial differential equation (PDE) originated in the 90s (Lagaris et al., 1998). However, the idea did not receive much traction until recently (Rudy et al., 2019; Raissi et al., 2019). The emerging paradigm of physics-informed neural networks (PINNs) has already been used to solve a wide range of nonlinear PDEs including Burger's equation, Schrodinger's equation, and the Navier-Stokes' equation (Raissi et al., 2019). PINNs leverage the capabilities of deep neural networks as a universal function approximator (Hornik et al., 1989). However, in contrast to the standard deep learning approaches, PINNs restrict the space of admissible solutions by imposing validity of the PDE through a loss function.

Here, we develop a PINN-based algorithm for solving the anisotropic acoustic qP-wave eikonal equation. We first construct an approximation space using a feed-forward neural network and then train the network, by imposing a loss function defined by the anisotropic eikonal equation, to yield traveltime solutions for the corresponding anisotropic medium. By doing so we avoid the need to use computationally costly procedures due to the high-order nonlinear terms in the anisotropic eikonal equation, such as the fixed-point iterative scheme of Waheed et al. (2015) or the DG method of Le Bouteiller et al. (2019). Furthermore, once the network is trained for a particular source location for a given anisotropic model, training it

PINN anisotropic eikonal solver

Figure 1: Schematic representation of a feed-forward neural network with $L - 1$ hidden layers.

for a new source location and a modified model requires significantly less computational effort. This property of PINNs is highly advantageous for seismic inversion applications, which require repeated forward modeling for thousands of source locations with updated model parameters. We demonstrate these assertions through tests on benchmark synthetic models.

THEORY

A feed-forward neural network is a set of neurons organized in layers in which evaluations are performed orderly through the layers. It can be seen as a computational graph having an input layer, an output layer, and an arbitrary number of hidden layers. In a fully connected ANN, neurons in adjacent layers are connected with each other but neurons within a single layer share no connection. Thanks to the universal approximation theorem, an ANN with n neurons in the input layer and m neurons in the output layer can be used to represent a function $u : \mathbb{R}^n \rightarrow \mathbb{R}^m$ (Hornik et al., 1989), as shown in Figure 1. For illustration, we consider a network of $L + 1$ layers starting with input layer 0, the output layer L , and $L - 1$ hidden layers. The number of neurons in each layer is denoted as $k_0 = n, k_1, \dots, k_L = m$. Each connection between the i -th neuron in layer $l - 1$ and j -th neuron in layer l has a weight w_{ji}^l associated with it. Moreover, for each neuron in layer l , we have an associated bias term $b_i, i = 1, \dots, k_l$.

Each neuron represents a mathematical operation, whereby it takes a weighted sum of its inputs plus a bias term and passes it through an activation function (Bishop, 2006),

$$u_k^l = \sigma \left(\sum_{j=1}^{k_{l-1}} w_{kj}^l u_j^{l-1} + b_k^l \right), \quad (1)$$

where $\sigma(\cdot)$ represents the activation function.

In a conventional deep learning application, an ANN is trained by minimizing a loss function, which typically measures the mismatch between the network's predicted values and some known outputs, known as the training set. However, there are several limitations associated with such models that solely rely on a labeled dataset and are oblivious to the scientific principles governing real-world phenomena. For cases when the size

of training and test data are small, such models often learn spurious relationships that are misleading. However, the biggest concern of such data-driven models is the lack of scientific consistency of their predictions to known physical laws that have been the cornerstone of knowledge discovery across scientific disciplines for many years.

On the contrary, here, we leverage the capabilities of ANN as a universal function approximator (Hornik et al., 1989) and define a loss function that minimizes the residuals of the anisotropic eikonal equation at a chosen set of training points. This is achieved with a simple feed-forward neural network leveraging the concept of automatic differentiation. To demonstrate the idea, we consider the 2D qP-wave eikonal equation for transversely isotropic media with tilted axis of symmetry (TTI), under the acoustic assumption (Alkhalifah, 2000):

$$(1 + 2\varepsilon) \left(\cos \theta \frac{\partial T}{\partial x} + \sin \theta \frac{\partial T}{\partial z} \right)^2 + \left(\cos \theta \frac{\partial T}{\partial z} - \sin \theta \frac{\partial T}{\partial x} \right)^2 \times \left(1 - \frac{2\eta v^2(1 + 2\varepsilon)}{1 + 2\eta} \left(\cos \theta \frac{\partial T}{\partial x} + \sin \theta \frac{\partial T}{\partial z} \right)^2 \right) - \frac{1}{v^2} = 0, \quad (2)$$

where $T(x, z)$ is the travelt ime measured from a source to a grid-point (x, z) , v is the P-wave velocity along the symmetry axis, ε is one of the Thomsen's anisotropy parameter, η is the anellipticity parameter, and θ is the tilt angle that the symmetry axis makes with the vertical.

To solve equation 2 using an ANN, we consider a network with two neurons in the input layer for the spatial coordinates (x, z) , an output neuron for the travelt ime $T(x, z)$, and a number of hidden layers. The partial derivatives of the output (T) w.r.t. the inputs (x, z) can be computed using automatic differentiation. The loss function that we seek to minimize is:

$$\mathfrak{J} = \sum_{i,j} \|\mathcal{L}\|^2 + \|T(s_x, s_z) - 0\|^2 + \sum_{i,j} H(-T(x_i, z_j)) \|T(x_i, z_j)\|^2, \quad (3)$$

where $\mathcal{L}$ represents the expression on the left-hand side of equation 2, i.e. the residual of the TTI eikonal equation, and $H(\cdot)$ is the Heaviside function.

In the loss function given by equation 3, the first term on the right side imposes the validity of the TTI eikonal equation 2 for a given number of training points by minimizing its residual at these points; i, j refer to the x - and z -coordinates of a training point. The second term requires the solution to be zero at the point-source (s_x, s_z) . The last term enforces the solution to be positive, where the Heaviside function is used to penalize negative travelt ime solutions.

It is worth emphasizing that the proposed approach is different from traditional (or non-physics constrained) deep learning techniques. The training of the network here refers to the tuning of weights and biases of the network such that the resulting travelt ime field minimizes the loss function $\mathfrak{J}$ on a given set of training points. Also noteworthy is the fact that, contrary to conventional deep learning applications, the network here learns without any labeled set. To understand this concept,

PINN anisotropic eikonal solver

consider a randomly initialized network, which will output a certain traveltime $T_{i,j}$ value for each point (i, j) in the training set which will be used to calculate the residual using equation 3. Based on this residual, the network adjusts its weights and biases allowing it to produce traveltimes that adhere to the underlying TTI eikonal equation (equation 2). Since there is no labeled dataset, the approach can be understood as a class of reinforcement learning where the system is penalized when it does not satisfy the underlying physical laws.

NUMERICAL TESTS

In this section, we test the performance of the proposed approach in solving the TTI eikonal equation. For accuracy comparison, we use a reference solution computed using a third-order FSM (Waheed, 2019) on a fine grid. The PINN model is implemented using SciANN (Haghighat and Juanes, 2019), a high-level Tensorflow wrapper for scientific computations.

First, we consider a $1 \times 1 \text{ km}^2$ homogeneous TTI model with $v = 2 \text{ km/s}$, $\varepsilon = 0.2$, $\eta = 0.083$, and $\theta = 30^\circ$. We consider a point-source in the center at $(0.5 \text{ km}, 0.5 \text{ km})$ and compute traveltimes by minimizing the loss function given in equation 3 on 101×101 regularly sampled training points in the computational domain. We consider a network with 8 hidden layers with 20 neurons each and use the Adam optimizer, which is based on stochastic gradient-descent (Kingma and Ba, 2014). The arctangent activation function is used for the hidden layers, while the final layer has a linear activation function. The network is trained for 3000 epochs.

Figure 2: Traveltime contours for the reference solution (solid black), the PINN-based anisotropic eikonal solution (dashed blue), and the first-order fast sweeping solution (dotted red) for the homogeneous TTI model. The source is located at $(0.5 \text{ km}, 0.5 \text{ km})$.

Figure 2 shows traveltime contours obtained using the PINN-based approach (dashed blue) compared against the reference solution (solid black). We also plot the traveltime solution computed using a first-order iterative fast sweeping TTI eikonal solution (Waheed et al., 2015) on the same 101×101 grid (dotted red). In Figure 3, we plot absolute traveltime errors for PINN-based traveltime solution (Fig. 3(a)) and the first-order fast sweeping TTI eikonal solution (Fig. 3(b)). We observe considerably higher accuracy for the PINN-based trav-

Figure 3: Absolute traveltime errors for solutions computed using the proposed method (a) and the first-order FSM (b) for the homogeneous TTI model. The source is located at $(0.5 \text{ km}, 0.5 \text{ km})$.

Figure 4: Convergence history for the homogeneous TTI model (blue) and the vertically varying TTI model (orange). For the vertically varying model, the pre-trained network from the homogeneous TTI model is used.

elttime solution compared to the first-order fast sweeping solution.

Next, we consider a $1 \times 1 \text{ km}^2$ vertically varying TTI model with v varying smoothly from 2 km/s to 3 km/s, ε from 0.2 to 0.4, η from 0.083 to 0.167, and θ from 30° to 50° . The lower value for each parameter is at zero depth while the higher value is at 1 km depth with a smooth increase in between. For a point-source at $(0.3 \text{ km}, 0.4 \text{ km})$, using the pre-trained network from the homogeneous TTI model case, we run the algorithm to compute the traveltime field for this inhomogeneous model. We observe, despite a significant change in the TTI model parameters and the source location, the algorithm converges in only 10% of the total epochs needed for the homogeneous TTI model case (Figure 4).

This feature of PINN-based eikonal solver is particularly attractive for seismic imaging and inversion applications where repeated forward modeling is needed for multiple source locations and updated velocity models. Figure 5 shows traveltime contours for the PINN-based solution compared with the reference solution and the first-order fast sweeping solution. We again observe considerably higher accuracy for the PINN-based eikonal solution. Figure 6 clearly demonstrates this by plotting absolute traveltime errors for the PINN-based eikonal solution and the first-order fast sweeping solution computed using the reference solution.

Finally, we test the performance of the proposed approach on the complex BP TTI model (Shah, 2007). We consider a portion of the model, as shown in Figure 7, for the tests. For a source located at $(50 \text{ km}, 5 \text{ km})$, similar to previous models, accuracy comparison using traveltime contours and absolute

PINN anisotropic eikonal solver

Figure 5: Traveltime contours for the reference solution (solid black), the PINN-based anisotropic eikonal solution (dashed blue), and the first-order fast sweeping solution (dotted red) for the vertically varying TTI model. The source is located at (0.3 km, 0.4 km).

Figure 6: Absolute traveltime errors for solutions computed using the proposed method (a) and the first-order FSM (b) for the vertically varying TTI model. The source is located at (0.3 km, 0.4 km).

traveltime errors are shown in Figures 8 and 9, respectively. We observe that even for such a complex and heterogeneous model, the PINN-based algorithm is significantly more accurate than the first-order FSM. Both solutions were computed using a 161×161 regular grid while the reference solution is computed on a 1601×1601 using a third-order fast sweeping eikonal solver. The PINN-based approach needed 5000 epochs to converge for this case.

CONCLUSIONS

We proposed a method to solve the TTI eikonal equation using a feed-forward neural network. Through tests on synthetic models, we show the accuracy of the proposed method to be significantly better than the first-order fast sweeping solution. The idea can be easily extended to eikonal equation for media with lower anisotropy symmetry by modifying the loss function accordingly. We also showed that by using a pre-trained network, the algorithm converges rapidly even for a significantly different velocity model and a different source location. This property is quite attractive for seismic imaging and inversion applications as they require repeated forward modeling for multiple source locations and updated models. Although we used uniform grid points for training the network, the approach is completely mesh-free and can also be used by training on randomly scattered points in the computational domain.

Figure 7: BP TTI model parameters for the velocity along the symmetry axis v (a), the Thomsen parameter ϵ (b), the anellipticity parameter η (c), and the tilt angle θ (d).

Figure 8: Traveltime contours for the reference solution (solid black), the PINN-based anisotropic eikonal solution (dashed blue), and the first-order fast sweeping solution (dotted red) for the BP TTI model. The source is located at (50 km, 5 km).

Figure 9: Absolute traveltime errors for solutions computed using the proposed method (a) and the first-order FSM (b) for the BP TTI model. The source is located at (50 km, 5 km).

REFERENCES

- Alkhalifah, T., 2000, An acoustic wave equation for anisotropic media: *Geophysics*, **65**, 1239–1250.
- Bishop, C. M., 2006, *Pattern recognition and machine learning*: Springer.
- Cerveny, V., 2001, *Seismic ray theory*: Cambridge university press.
- Grechka, V., A. De La Pena, E. Schissel-Rebel, E. Auger, and P.-F. Roux, 2015, Relative location of microseismicity: *Geophysics*, **80**, WC1–WC9.
- Haghighat, E., and R. Juanes, 2019, Sciann: A keras wrapper for scientific computations and physics-informed deep learning using artificial neural networks: <https://sciann.com>.
- Hornik, K., M. Stinchcombe, and H. White, 1989, Multilayer feedforward networks are universal approximators: *Neural networks*, **2**, 359–366.
- Kingma, D. P., and J. Ba, 2014, Adam: A method for stochastic optimization: arXiv preprint arXiv:1412.6980.
- Lagaris, I. E., A. Likas, and D. I. Fotiadis, 1998, Artificial neural networks for solving ordinary and partial differential equations: *IEEE transactions on neural networks*, **9**, 987–1000.
- Lambar, G., S. Operto, P. Podvin, and P. Thierry, 2003, 3D ray+ born migration/inversion—part 1: Theory: *Geophysics*, **68**, 1348–1356.
- Lawton, D. C., 1989, Computation of refraction static corrections using first-break traveltimes differences: *Geophysics*, **54**, 1289–1296.
- Le Bouteiller, P., M. Benjema, L. Mtivier, and J. Virieux, 2019, A discontinuous galerkin fast-sweeping eikonal solver for fast and accurate traveltimes computation in 3D tilted anisotropic media: *Geophysics*, **84**, C107–C118.
- Raissi, M., P. Perdikaris, and G. E. Karniadakis, 2019, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations: *Journal of Computational Physics*, **378**, 686–707.
- Rudy, S., A. Alla, S. L. Brunton, and J. N. Kutz, 2019, Data-driven identification of parametric partial differential equations: *SIAM Journal on Applied Dynamical Systems*, **18**, 643–660.
- Sethian, J. A., 1999, Fast marching methods: *SIAM review*, **41**, 199–235.
- Shah, H., 2007, The 2007 BP anisotropic velocity-analysis benchmark: Presented at the 70th Annual EAGE meeting workshop.
- Taillandier, C., M. Noble, H. Chauris, and H. Calandra, 2009, First-arrival traveltimes tomography based on the adjoint-state method: *Geophysics*, **74**, WCB1–WCB10.
- Waheed, U., 2019, High-order accurate method for solving the anisotropic eikonal equation: Presented at the SPE Middle East Oil and Gas Show and Conference, Society of Petroleum Engineers.
- Waheed, U., C. E. Yarman, and G. Flagg, 2015, An iterative, fast-sweeping-based eikonal solver for 3D tilted anisotropic media: *Geophysics*, **80**, C49–C58.
- Zhao, H., 2005, A fast sweeping method for eikonal equations: *Mathematics of computation*, **74**, 603–627.