EikoNet: Solving the Eikonal Equation using Deep Neural Networks, with applications to earthquake location

Manuscript - https://arxiv.org/abs/2004.00361

Jonthan D. Smith[1]; Kamyar Azizzadenesheli[2]; Zachary E. Ross[1]; Jack Muir[1]

[1]Division Office Geological and Planetary Sciences - California Institute of Technology,

[2]Department of Computer Science, Purdue University

Abstract

The recent deep learning revolution has created an enormous opportunity for accelerating compute capabilities in the context of physics-based simulations. Here, we propose EikoNet, a deep learning approach to solving the Eikonal equation, which characterizes the first-arrival-time field in heterogeneous 3D velocity structures. Our grid-free approach allows for rapid determination of the travel time between any two points within a continuous 3D domain. These travel time solutions are allowed to violate the differential equation, which casts the problem as one of optimization, with the goal of finding network parameters that minimize the degree to which the equation is violated. In doing so, the method exploits the differentiability of neural networks to calculate the spatial gradients analytically, meaning the network can be trained on its own without ever needing solutions from a finite difference algorithm. Training and inference are highly parallelized, making the approach well-suited for GPUs. EikoNet has low memory overhead, and further avoids the need for travel-time lookup tables

EikoNet is rigorously tested on several toy velocity models, real-world velocity models to demonstrate robustness and versatility. We outline the application to earthquake hypocenter inversion, ray multi-pathing, and tomographic modelling.

Methods

The Eikonal equation is a nonlinear first-order partial differential equation representing a high frequency approximation to the propagation of waves in heterogenous media.

Our approach to solving the Factored Eikonal equation by training a deep neural network, f_θ, to predict the travel-time field, $T_{s \to r}$, between any input source-receiver coordinates. This is achieved by leveraging the analytical differentiability of a neural network, training the neural network gradients to satisfy the partial differential formulation of the Factored eikonal equation for a user defined velocity model;

$$V(\vec{x_r}) = \left[T_0^2 \| \nabla_r \tau_{s \to r} \|^2 + 2\tau_{s \to r}(\vec{x_p} - \vec{x_s}) \cdot \nabla_r \tau_{s \to r} + \tau_{s \to r}^2 \right]^{-\frac{1}{2}}.$$
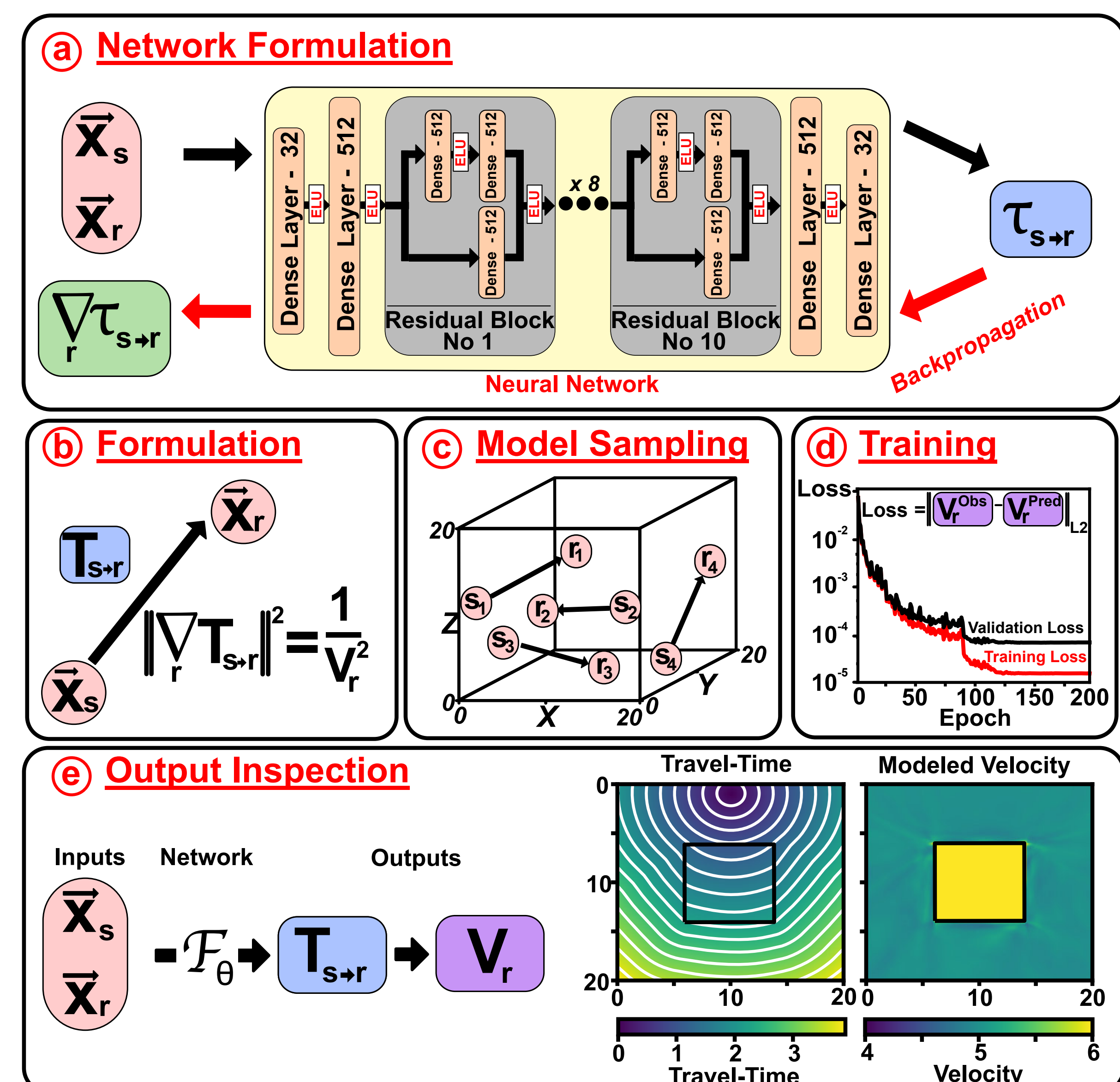

Figure 1: Overview of processing workflow. (a) Neural network architecture composed of fullyconnected layers and residual blocks. Each residual block is composed of 3 fully-connected layers with 512 neurons. ReLu activations are applied on all hidden layers. (b) Summary of Eikonal equation for Ts→r and Vr. (c) Sampling of source-receiver pairs across the 3D volume to build the training dataset. (d) Network training through the minimization of loss function relating predicted and observed velocity values. (e) Inspection of neural network outputs by passing user defined source receiver pairs.

Velocity Model Experiments

Outlined are a series of experiments designed to demonstrate the versatility of our approach for learning travel time fields in complex 3D velocity models, with applications to toy problems shown in Figure 2 and to the Marmousi2 model in Figure 3.
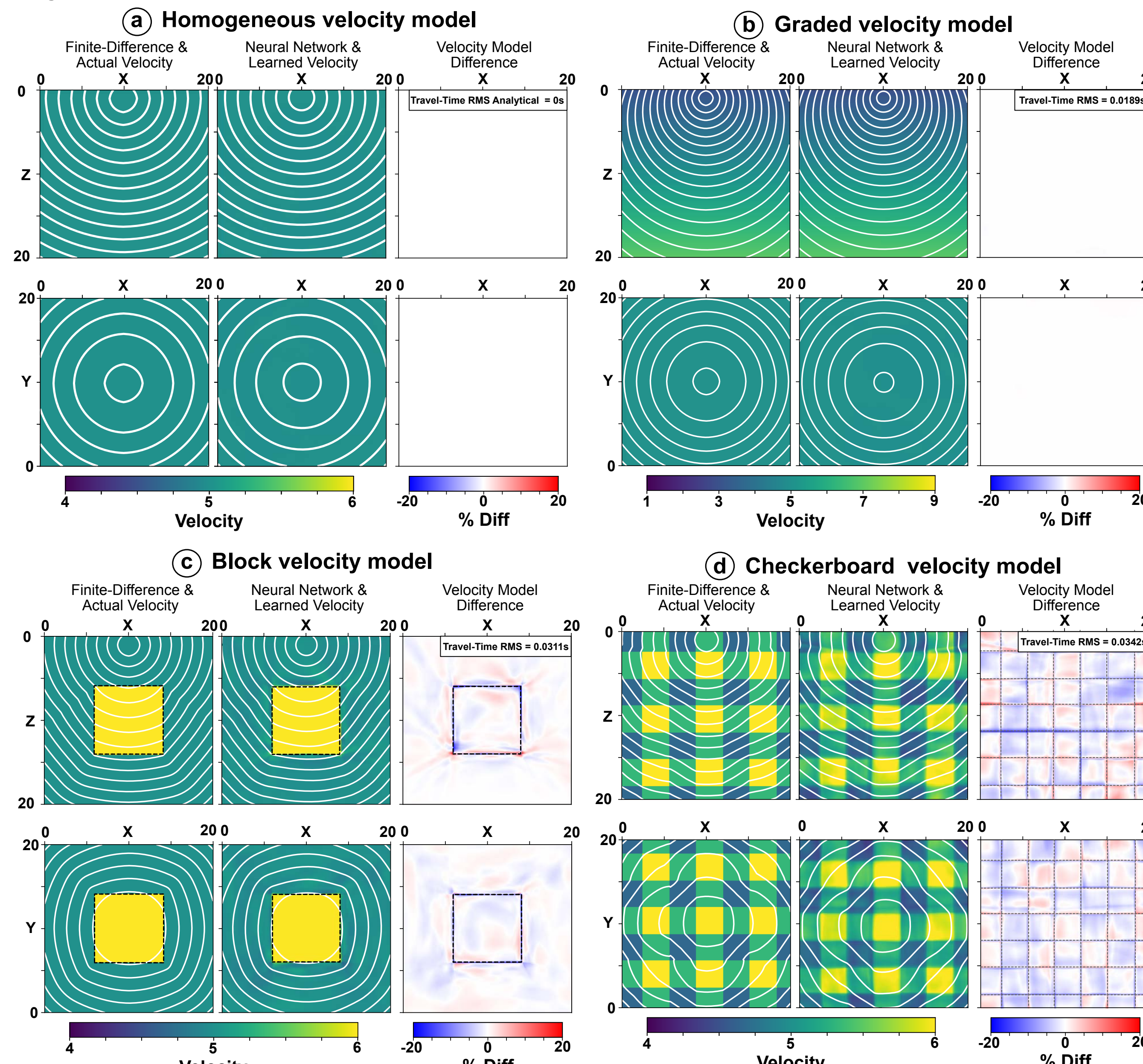
Figure 2 : Velocity model experiments with comparison to finite-difference and imposed velocity models. *Left* panels represent the X-Z and X-Y slice from the imposed velocity model, overlayed by the finite-difference expected travel-time. *Middle* panels represent the X-Z and X-Y slice from the neural network recovered velocity model and neural-network travel-time. *Right* panels represent X-Z and X-Y slice velocity models differences between the imposed and recovered velocity model.
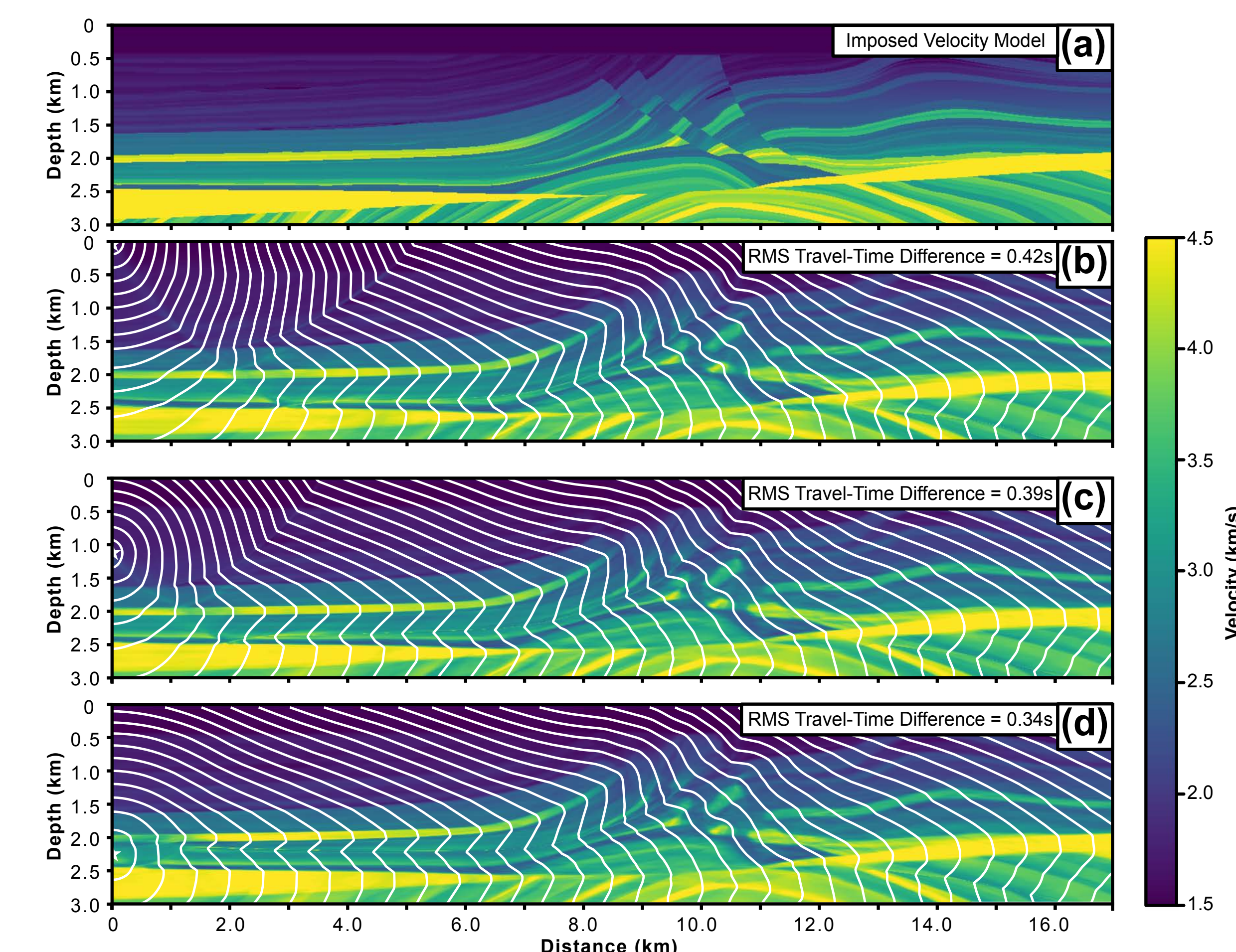

Figure 3 : Marmousi2 two-dimensional travel-time formulations using EikoNet and Finite-Difference Methods. (a) represents a colormap of the imposed 2D velocity model. (b) the recovered travel-time and velocity model fields from a source location at [0, 0] to a point grid at 0.001km spacing. Colormap represents the recovered velocity and white contours represent the travel-time at 0.1s spacing. (c) and (d) represent a similar plot to (b) but with the different source locations of [0km, 1.1km] and [0km, 2.25km] respectively.

HypoSVI - Stein Variation Earthquake Location

Using EikoNet we outline an inversion procedure leveraging a Stein Variational Gradient Descent method. In this procedure we use pre-trained Vp and Vs EikoNets for Southern California, inverting the seismic station phase arrivals for 15k events in 2019 to determine earthquake hypocentral location and location uncertainty.

Earthquake locations are consistent with the prior NonLinLoc location methods (Lomax et al 2010).
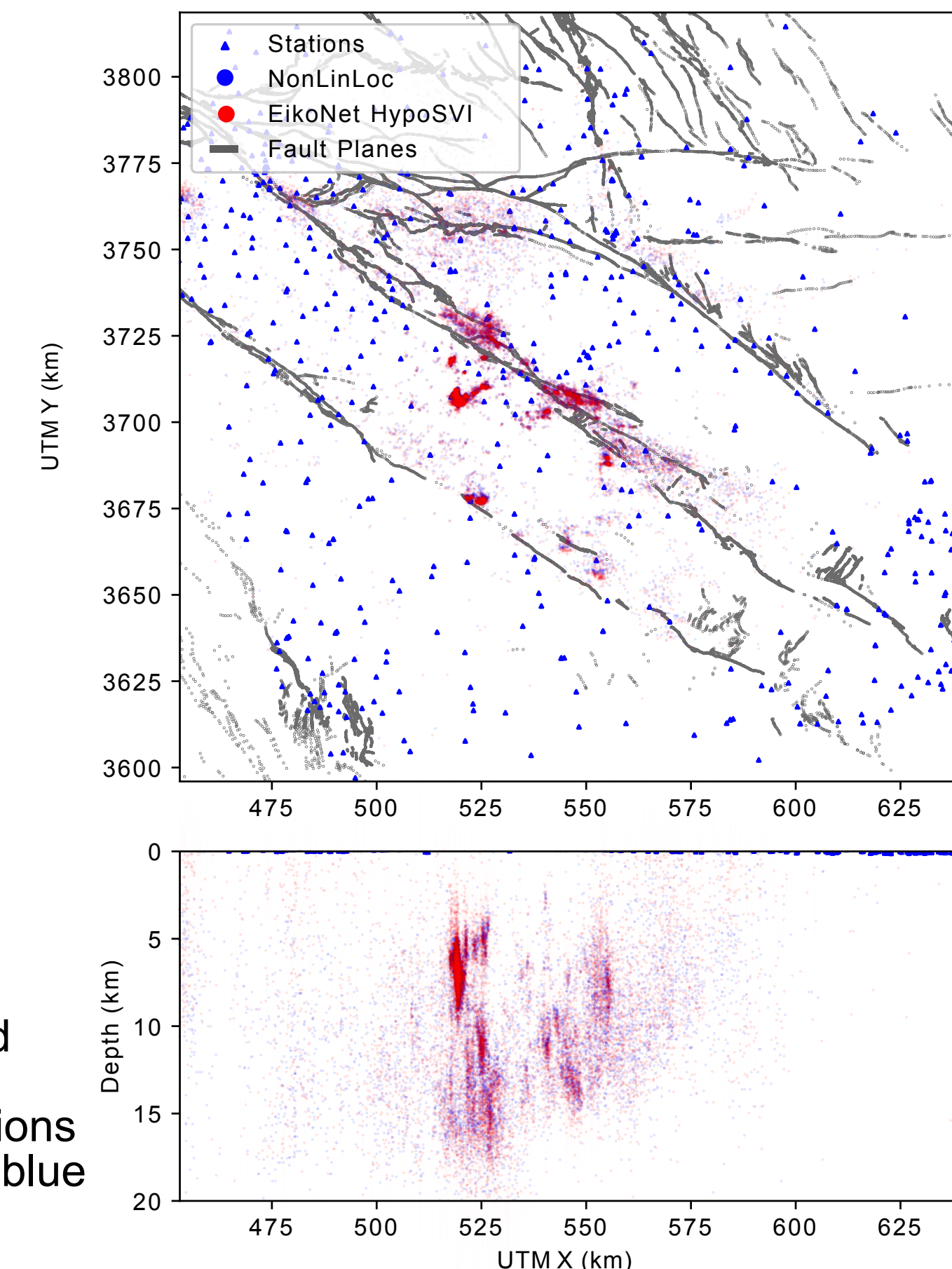
Figure 4: Application of EikoNet earthquake location package HypoSVI for earthquakes in Southern California using P- and S-wave picked catalogue. Comparison made with the prior NonLinLoc hypocentral locations. Station Locations and surface mapped fault planes are shown by blue triangles and grey lines respectively.
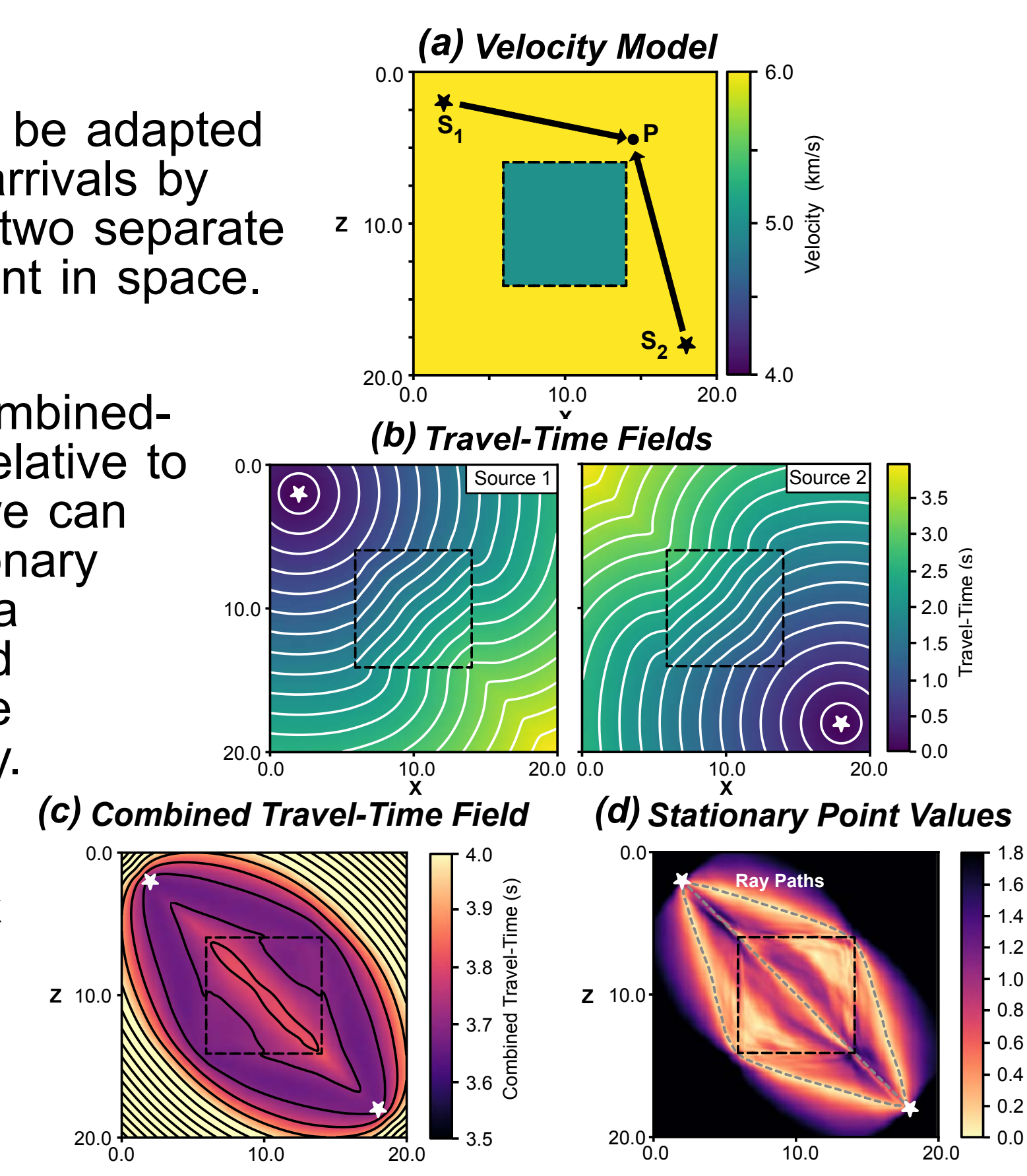
Future Applications

In this section we discuss the application of EikoNet to a series of travel-time required problems, outlining the advantage of EikoNet over conventional Finite-difference methods and how these procedures would implemented.
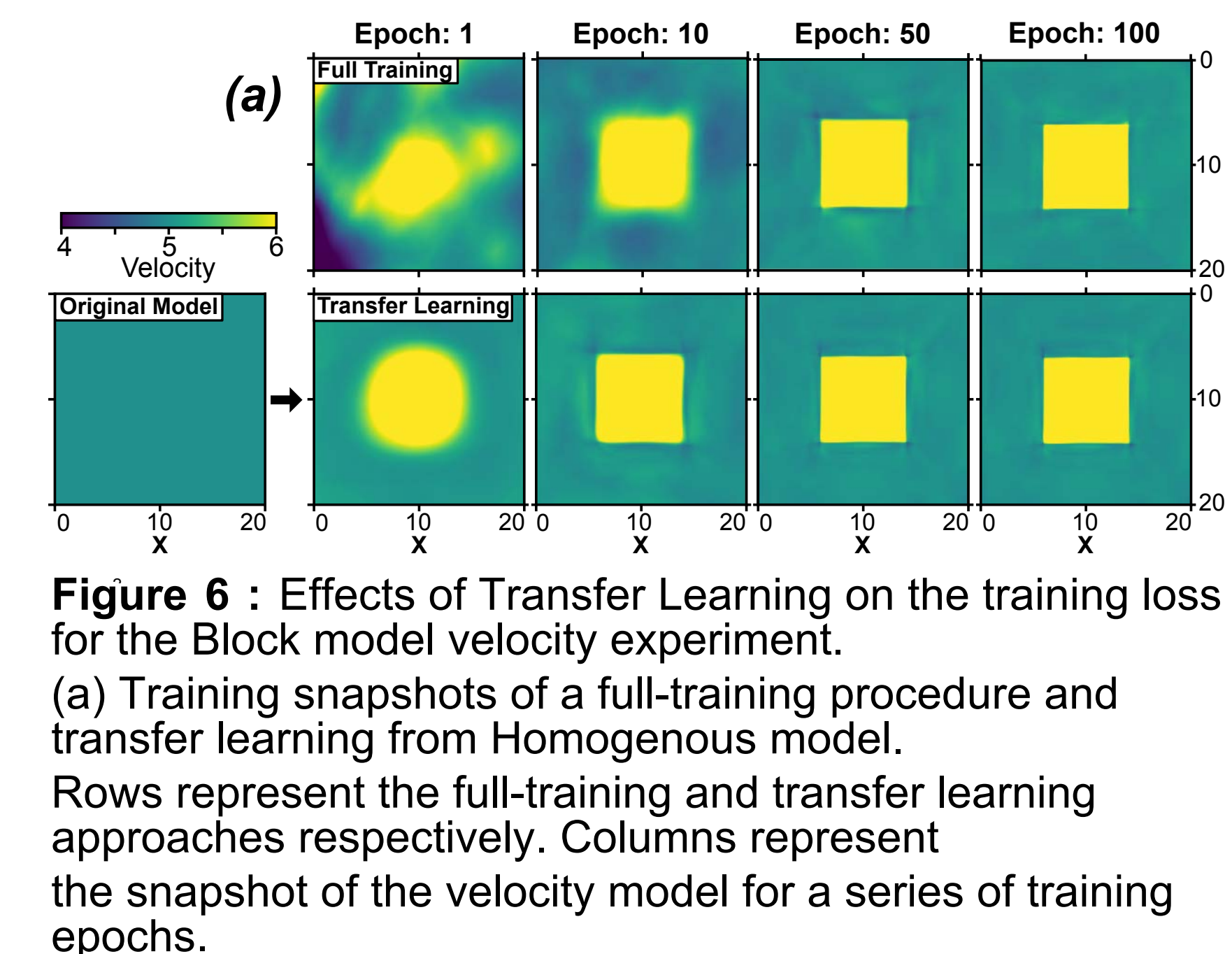
Ray Multipathing

The deep learning approach can be adapted to include additional secondary arrivals by determining the travel-time from two separate source locations to the same point in space. (Figure 5b, 5c)

By taking the gradient of this combined-travel time across the network relative to the point locations (Figure 6d) we can determine if the points are stationary (Rawlinson et al, 2009), having a gradient value close to zero, and therefore representing a possible secondary phase arrival pathway.

Figure 5 : Seismic ray multipathing procedure applied to an adapted block model velocity experiment with low velocity block and high velocity background.

Tomographic Modelling

For tomographic inversions which undergo many iterations successively, new travel-time fields must

be computed from scratch for each iteration. Our approach allows for the neural network model

from the previous iteration to be used as the starting point for the next training procedure, which

could rapidly converge to the new velocity model if the perturbations are relatively small.

Figure 6 : Effects of Transfer Learning on the training loss for the Block model velocity experiment.
(a) Training snapshots of a full-training procedure and transfer learning from Homogenous model.
Rows represent the full-training and transfer learning approaches respectively. Columns represent the snapshot of the velocity model for a series of training epochs.

References

Lomax, A., A. Michelini, A. Curtis (2009), Earthquake Location, Direct, Global-Search Methods, in Encyclopedia of Complexity and System Science, Part 5, Meyers, R. A. (ed.), Springer, New York, pp. 2449-2473, doi:10.1007/978-0-387-30440-3. http://www.springerlink.com/content/m057p6112445368/fulltext.html.

N. Rawlinson, M. Sambridge, J. Hauser, Multipathing, reciprocal traveltime fields and raylets, Geophysical Journal International, Volume 181, Issue 2, May 2010, Pages 1077–1092, https://doi.org/10.1111/j.1365-246X.2010.04558.x.