Table 3: Architecture details of final proposed PETAL model.

Layer	Input Dim	Output Dim	Spectral Norm?
$\boldsymbol{x}$ Encoder	2541	1000	✓
P_x	1000	1000	✓
P_y	800	1000	✓
DotProd Out	1000	1000	✗
$\boldsymbol{y}$ Decoder	1000	800	✗
$\boldsymbol{x}$ Decoder	1000	2541	✓

A Training the Forward Model

All experiments were performed on a GeForce RTX 2080 Super.

A.1 Data Preparation

We normalize our data using the training set "pixel-wise" average and standard deviation for training only.

A.2 PETAL

The proposed PETAL model only uses linear layers throughout. However, it is able to learn a complex non-linear model due to the attention-inspired mechanism. The exact details of each sub-component can be found in Table 3. We only make slight changes to existing attention based layers. Specifically, we merge the P_Q and P_K layers into just a P_x layer, but otherwise keep everything else (including the linear out layer referred to as DotProd Out in the table).

The model was trained using ADAMW with a learning rate of 1e-5 for 500 epochs. The learning rate was dropped by a factor of 0.2 at epoch 300.

The model was trained to minimize the MSE of the arrival time prediction as well as a MSE on the SSP reconstruction. The selected model achieved an (unnormalized) AT RMSE of 4.98e-4 and SSP RMSE of 5.37e-2.

A.3 MLP

We experimented with both encoder-decoder like structures as well as models without the bottleneck layers. The final best performing model had 4 hidden layers of dim 1500 with leaky ReLU non-linearities. It achieved an unnormalized AT validation RMSE of 6.08e-4 (higher than PETAL). The model was trained using Adam for 250 epochs with a learning rate of 1e-5.

B Optimization Framework

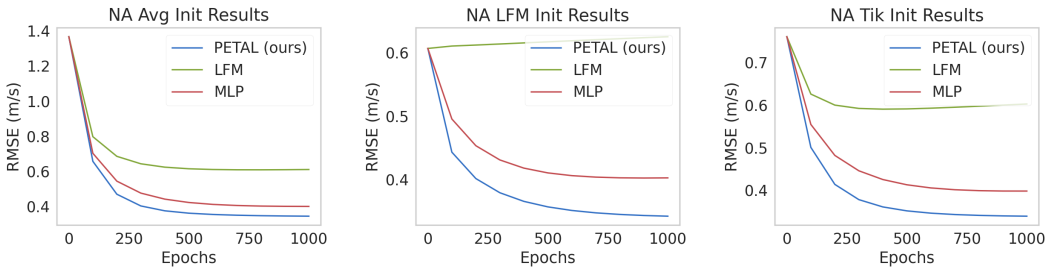
The neural adjoint method is an iterative method to recover an SSP $\boldsymbol{x}$ given some observations $\boldsymbol{y}$. All models are optimized using Pytorch's Stochastic Gradient Descent with a learning rate of 50 for 1000 epochs.

We use two forms of regularization: an ℓ_2 penalty on $\boldsymbol{x}$ with a scale of 1e-7 and a Sobolev penalty (ℓ_2 on the discrete x and y gradient) with a scale of 1e-4.

The optimization is performed in batches. We set an early cutoff rate of 1e-2 such that for any sample, if the forward model observation loss drops below this value, we cut off the gradient to that sample. This value is lower than the final (normalized) mse AT loss of any of the models, so the assumption is that any further optimization beyond this point will just overfit to the model.

B.1 Results

The results of NA given different initializations for each of the forward models can be seen in Figure 5. Although further iterations might yield higher performance, the overall RMSE already begins to



(a) (b) (c)

Figure 5: RMSE of different models vs number of epochs optimized given (a) average, (b) LFM, and (c) Tik initializations.

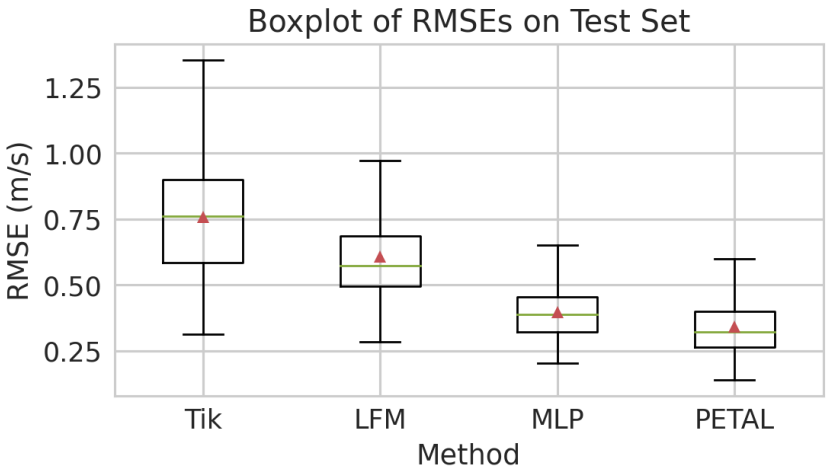


Figure 6: Boxplot of RMSE of different models.

plateau around 1000 epochs. For some models (particularly LFM), the performance already starts to degrade. The distribution of errors after 1000 iterations can be found in Figure 6.

B.2 Robustness to Unseen Slices

In this section we explore the robustness of surrogate models to unseen slices. We perform this experiment by training the surrogate models on only slices 1-9 (with the same train/val/test split) and then evaluating on the entirety of slice 10. The performance can be seen in Figure 7 and Table 4. We refer to the subsets of slice 10 as "Train","Val", and "Test" for convenience, referring to the temporal split of the data, but no samples from slice 10 were available during train time. We select the linearization around the last available SSP in the times corresponding to the train set for LFM.

Both trained surrogate models greatly outperform LFM in the subset of the slice overlapping in time with the trainset, suggesting that there are some shared dynamics across space that can be learned. Notably, most models begin to degrade in the times corresponding to the validation and test set, highlighting the difficulty in capturing dynamic shifts over time. However, the learned models still remained more robust to this shift and the performance only degraded slightly compared to when trained with all slices dropping from 0.33715 (when evaluated only on slice 10) to 0.33736 for our proposed model PETAL.

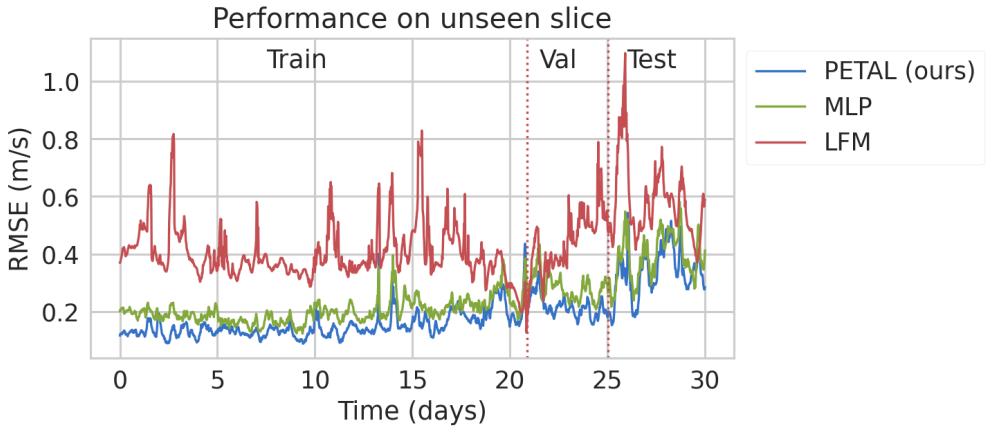


Figure 7: Performance of models on unseen slice. Both trained forward models perform well on the subset of the slice overlapping in time with the trainset, suggesting that dynamics are shared throughout the region.

Table 4: Average RMSE (m/s) of inversion on unseen slice.

Model	Train	Val	Test
LFM	0.405	0.447	0.583
MLP	0.196	0.288	0.402
PETAL (ours)	**0.149**	**0.217**	**0.337**

C Gradient of PETAL

Define a (simplification) of the PETAL model as

$$\begin{aligned}\hat{\boldsymbol{y}} &= \boldsymbol{W}(\sum_i w^i \hat{\boldsymbol{y}}^i) \\ &= \sum_i w^i \boldsymbol{W}(\boldsymbol{A}^i_{\text{ref}}\boldsymbol{x} + \boldsymbol{b}^i),\end{aligned} \tag{12}$$

where $\boldsymbol{W}$ encapsulates all linear layers performed on $\boldsymbol{y}$. Note that by construction, the weights w^i sum up to 1. If we include this in a simple MSE loss we get

$$L = \frac{1}{2}\left\|\hat{\boldsymbol{y}} - \boldsymbol{y}\right\|^2. \tag{13}$$

Computing a gradient w.r.t. $\boldsymbol{x}$ gives

$$\frac{\partial L}{\partial \boldsymbol{x}} = \sum_i \sum_j \frac{\partial w^i}{\partial \boldsymbol{x}} w^j \boldsymbol{W}(\boldsymbol{A}^i\boldsymbol{x}+\boldsymbol{b}^i)(\boldsymbol{W}\boldsymbol{A}^j\boldsymbol{x}+\boldsymbol{W}\boldsymbol{b}^j-\boldsymbol{y}) + w_i w_j \boldsymbol{A}^{i\top}\boldsymbol{W}^\top(\boldsymbol{W}\boldsymbol{A}^j\boldsymbol{x}+\boldsymbol{W}\boldsymbol{b}^j-\boldsymbol{y}), \tag{14}$$

where the right term reduces to a convex combination of the gradient of the linearized physics based forward models, modulated by some matrix $\boldsymbol{W}$, when $i = j$.

D Limitations

Our proposed model was evaluated on noise-less simulations, both with respect to measurements and sensor/receiver placement, which is not true in practice for data collected in the real world. We also did not explore the selection process of the reference points to linearize around, assuming that the chosen subset sufficiently represented the data. However, section B.2 suggests that the selection of reference points is somewhat robust to unseen dynamics.