Predicting seismic wave propagation with a Fourier Neural Operator surrogate model

Fanny Lehmann[1,2]

Filippo Gatti[2], Michaël Bertin[1], Didier Clouteau[2]

1. CEA, DAM, DIF, F-91297, Arpajon, France
2. Université Paris-Saclay, ENS Paris-Saclay, CentraleSupélec, CNRS, LMPS - Laboratoire de Mécanique Paris-Saclay, 91190 Gif-sur-Yvette, France

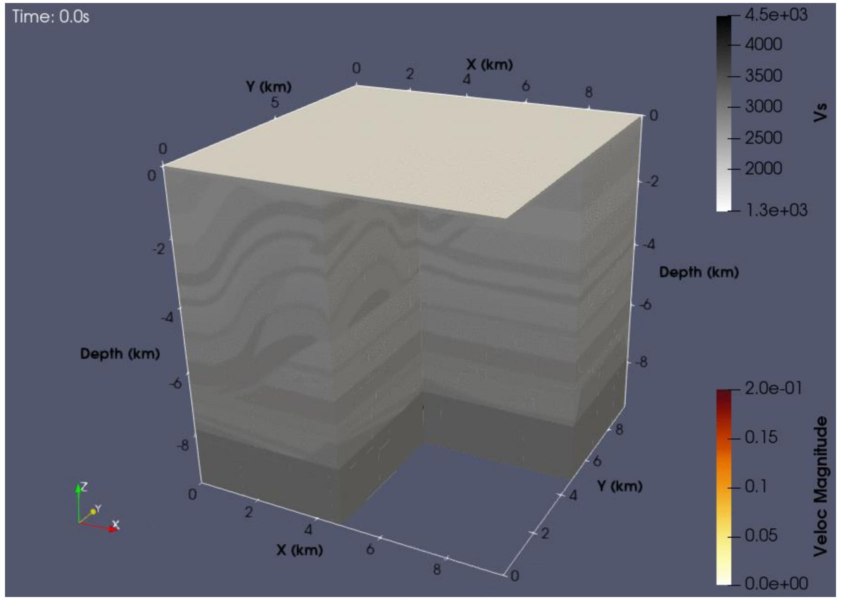
Context

Numerical simulations are essential to assess the impacts of earthquakes, especially to complement recorded data in regions with low to moderate seismicity.

They face several challenges:

- **3D** simulations are required
- properties of the propagation domain are complex
- simulation parameters are **uncertain**: geological properties, source position, source characteristics

→ High computational costs prevent uncertainty quantification analyses



10km x 10km x 10km, f_{max}=5Hz

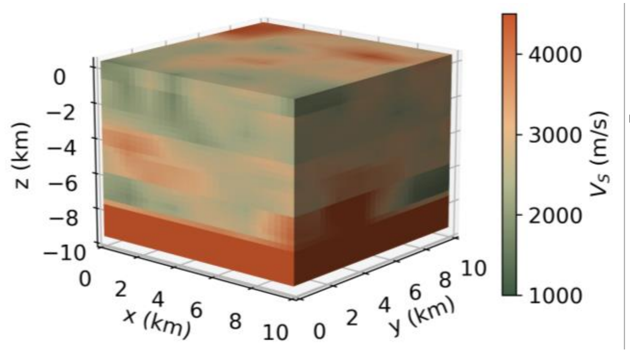
Simulation time = 8s → 22.7h equiv. CPU

Objective:

Design a **surrogate model** that predicts ground motion depending on the geological properties and source characteristics.

Formulation of the problem

geological properties
$\boldsymbol{x} \rightarrow a(\boldsymbol{x})$



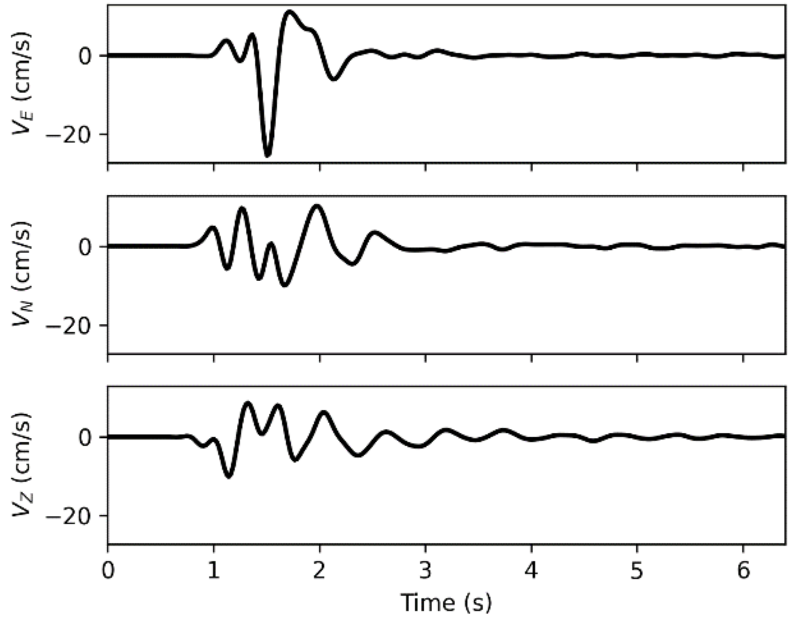
source position ($\boldsymbol{x_s}$)

source characteristics ($\boldsymbol{\theta_s}$)

wave propagation equation

$$\rho \frac{\partial^2 \boldsymbol{u}}{\partial t^2} - \nabla \cdot \sigma(\boldsymbol{u}) = f(\boldsymbol{x_s}, \boldsymbol{\theta_s})$$

ground motion timeseries
$\boldsymbol{x}, t \rightarrow u(\boldsymbol{x}, t)$

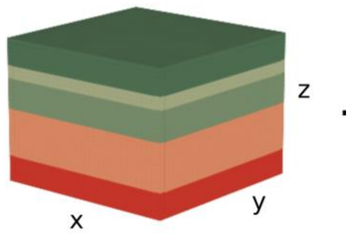


Workflow:

1. Create a training database $(a_i, s_i, u_i)_i$ with SEM3D numerical simulations.
2. Train a deep learning model to predict u_i from (a_i, s_i)

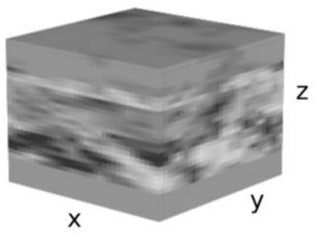
Training data (input): geologies

random homogeneous layers

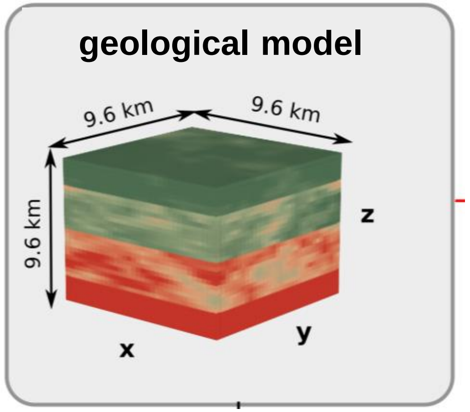


+

random fluctuations



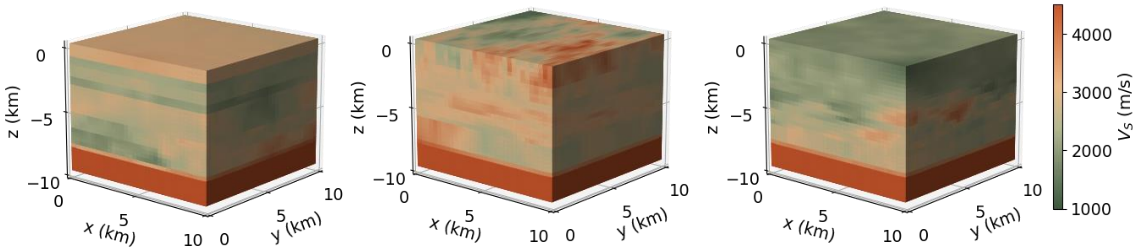
=



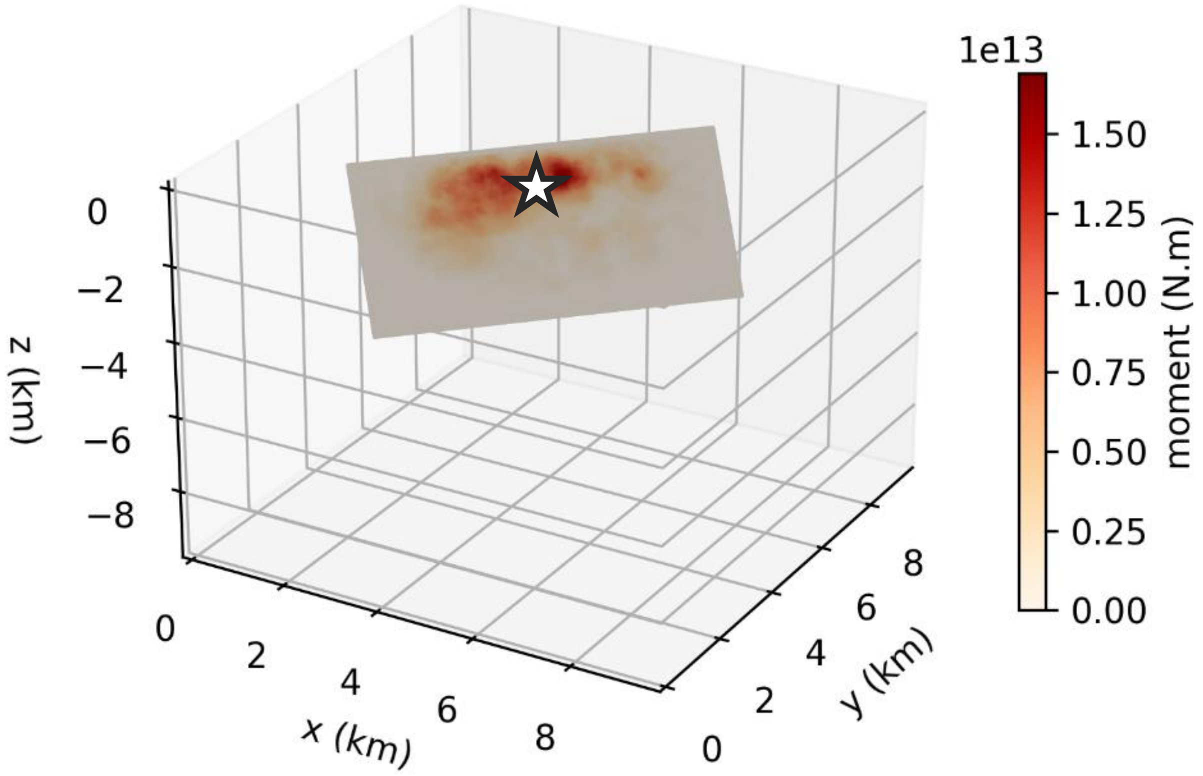
Number of layers: $\mathrm{N}_L \sim \mathcal{U}(\{2, 3, \dots 7\})$
Layer thickness: $h_1, \dots, h_{N_L} \sim \mathcal{U}([0.3, 9.3])$
s.t. $h_1 + \ \dots h_{N_L} = 9.6km$
Layer value: $a_\ell \sim \mathcal{U}([1785; 3214\ m/s])$

Log-normal random field
with a von Karman correlation
Coef. of variation $\sigma_\ell \sim |\mathcal{N}(0.2, 0.1)|$
Correlation length $\ell_\ell^x \sim \mathcal{U}(\{1.5, 3, 4.5, 6\ km\})$
$\ell_\ell^y \sim \mathcal{U}(\{1.5, 3, 4.5, 6\ km\})$
$\ell_\ell^z \sim \mathcal{U}(\{1.5, 3, 4.5, 6\ km\})$

Size: 9.6km x 9.6km x 9.6km
Matrix: 32 x 32 x 32 voxels



Training data (input): sources



Approximate a fault by
- the hypocenter position $\boldsymbol{x_s}$

Latine Hypercube Sampling (LHS):

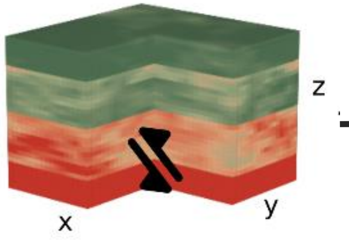
$$x_s \in [1.2\ km, 8.4\ km]$$
$$y_s \in [1.2\ km, 8.4\ km]$$
$$z_s \in [-9.0\ km, -0.6\ km]$$

- 3 angles of the source orientation $\boldsymbol{\theta_s}$

LHS:

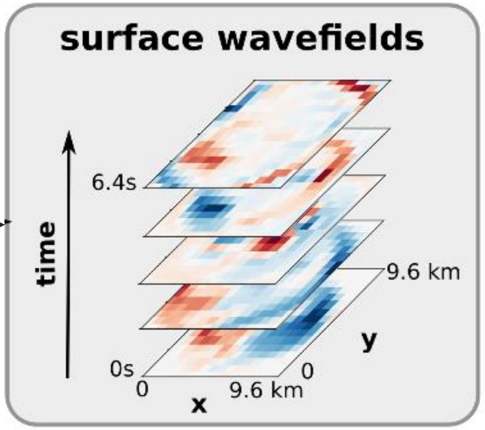
$$\phi \in [0°, 360°]$$
$$\delta \in [0°, 90°]$$
$$\lambda \in [0°, 360°]$$

Training data (output): ground motion

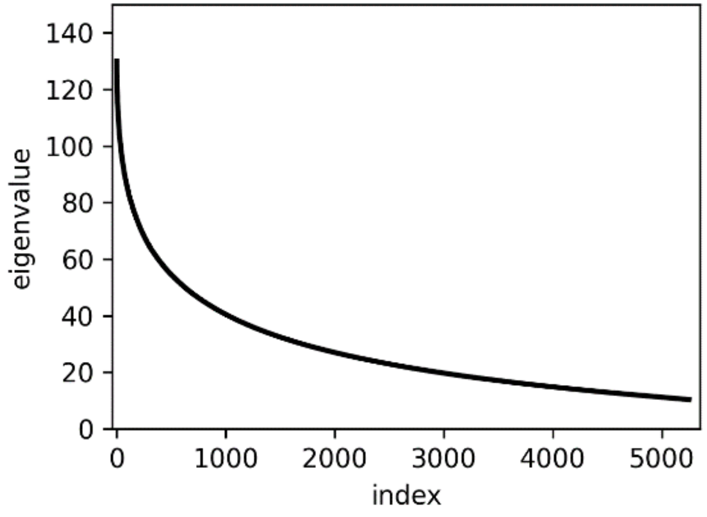


3D elastic wave propagation

Spectral Element code SEM3D
22.7h CPU equiv.



high-dimensional outputs



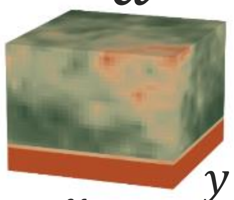
32 x 32 sensors record ground motion at the surface
total time = 6.4s with dt=0.02s
→ 3 outputs 32 x 32 x 320

N_{train} = 30 000 simulations
⇒ $6.8 \cdot 10^5$ h CPU

Lehmann et al., *ESSD (under review)*

Goal

geology



source position

$$\boldsymbol{x_s} = \begin{pmatrix} x_s \\ y_s \\ z_s \end{pmatrix}$$

source orientation

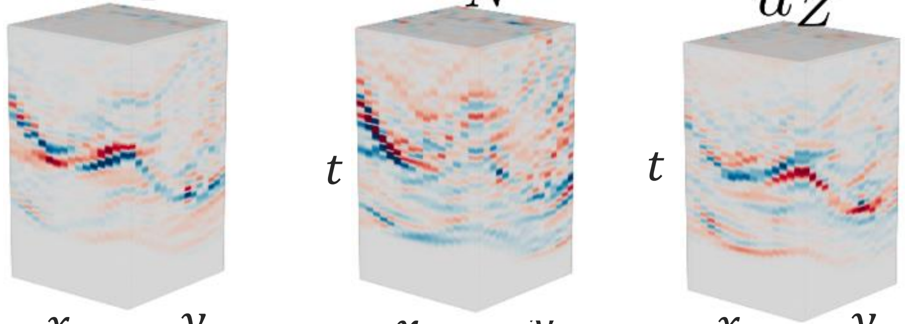
$$\boldsymbol{\theta_s} = \begin{pmatrix} \phi \\ \delta \\ \lambda \end{pmatrix}$$

surrogate model

$$\underbrace{\rho \frac{\partial^2 \boldsymbol{u}}{\partial t^2} - \nabla \cdot \sigma(\boldsymbol{u})} = f(\boldsymbol{x_s}, \boldsymbol{\theta_s})$$

$$L_a \boldsymbol{u} = f(\boldsymbol{x_s}, \boldsymbol{\theta_s})$$

surface ground motion



Neural Operators

[Li et al. 2020]

Parametric PDE: $L_a u(\boldsymbol{x}, t) = f(\boldsymbol{x}, t)$

If G_a is the Green function solution of $L_a G_a(\boldsymbol{x}, \cdot) = \delta_{\boldsymbol{x}}$, then

$$u(\boldsymbol{x}) = \int G_a(\boldsymbol{x}, \boldsymbol{y}) f(\boldsymbol{y}) d\boldsymbol{y}$$

t is ignored in the notation but you can replace $\boldsymbol{x}$ by $(\boldsymbol{x}, t)$

G_a is modelled as a kernel κ_ϕ defined by a neural network with parameters ϕ

$$G_a(\boldsymbol{x}, \boldsymbol{y}) \cong \kappa_\phi\big(\boldsymbol{x}, \boldsymbol{y}, a(\boldsymbol{x}), a(\boldsymbol{y})\big)$$

Introduce hidden variables $v_0, \dots, v_\ell, \dots, v_L$ and the iterative process

$$\boldsymbol{v}_{\ell+1}(\boldsymbol{x}) = \sigma\left(W \boldsymbol{v}_{\boldsymbol{l}}(\boldsymbol{x}) + \underbrace{\int \kappa_\phi\big(\boldsymbol{x}, \boldsymbol{y}, a(\boldsymbol{x}), a(\boldsymbol{y})\big) \boldsymbol{v}_{\boldsymbol{l}}(\boldsymbol{y}) d\boldsymbol{y}}_{K_\phi(a)\boldsymbol{v}_\ell} \right)$$

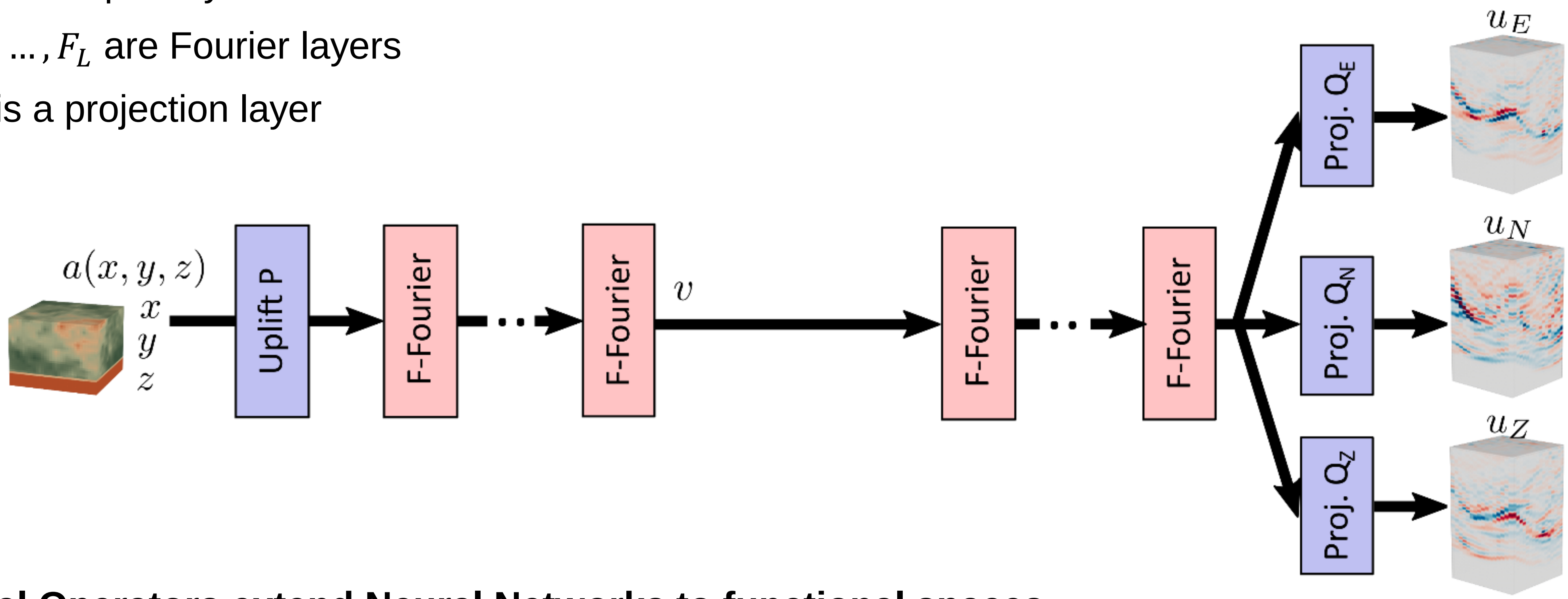
Fourier Neural Operators (FNO)

[Li et al. 2021]

The mapping $a \mapsto u$ is learnt iteratively

$$\begin{pmatrix} a(\boldsymbol{x}) \\ \boldsymbol{x} \end{pmatrix} \xrightarrow{P} \boldsymbol{v_0}(\boldsymbol{x}) \xrightarrow{F_1} \dots \xrightarrow{F_\ell} \boldsymbol{v_l}(\boldsymbol{x}) = \sigma\left(W\boldsymbol{v_{l-1}}(\boldsymbol{x}) + K_\phi(a)\boldsymbol{v_{\ell-1}}\right) \xrightarrow{\boldsymbol{F_{\ell+1}}} \dots \xrightarrow{F_L} \boldsymbol{v_L}(\boldsymbol{x}) \xrightarrow{Q} u(\boldsymbol{x})$$

- P is an uplift layer
- $F_1, \dots, F_L$ are Fourier layers
- Q is a projection layer



Neural Operators extend Neural Networks to functional spaces

Fourier layer

[Li et al. 2021]

For an efficient computation of the integral $K_\phi(a)\boldsymbol{v} = \int \kappa_\phi\big(\boldsymbol{x}, \boldsymbol{y}, a(\boldsymbol{x}), a(\boldsymbol{y})\big)\boldsymbol{v}(\boldsymbol{y})d\boldsymbol{y}$, assume that κ_ϕ is a convolution kernel

$$\kappa_\phi\big(\boldsymbol{x}, \boldsymbol{y}, a(\boldsymbol{x}), a(\boldsymbol{y})\big) = \kappa_\phi(\boldsymbol{x} - \boldsymbol{y})$$

$$\Rightarrow K_\phi(a)\boldsymbol{v} = \boldsymbol{\kappa_\phi} * \boldsymbol{v}$$

From the convolution theorem

$$K_\phi(a)\boldsymbol{v} = FFT^{-1}\Big(FFT\big(\boldsymbol{\kappa_\phi}\big) \cdot FFT(\boldsymbol{v})\Big)$$

The kernel is learnt directly in Fourier space

$$K_\phi(a)\boldsymbol{v} = FFT^{-1}\Big(\boldsymbol{R_\phi} \cdot FFT(\boldsymbol{v})\Big)$$

weights $\in \mathbb{C}$ to learn
inside each layer

Factorized Fourier layer

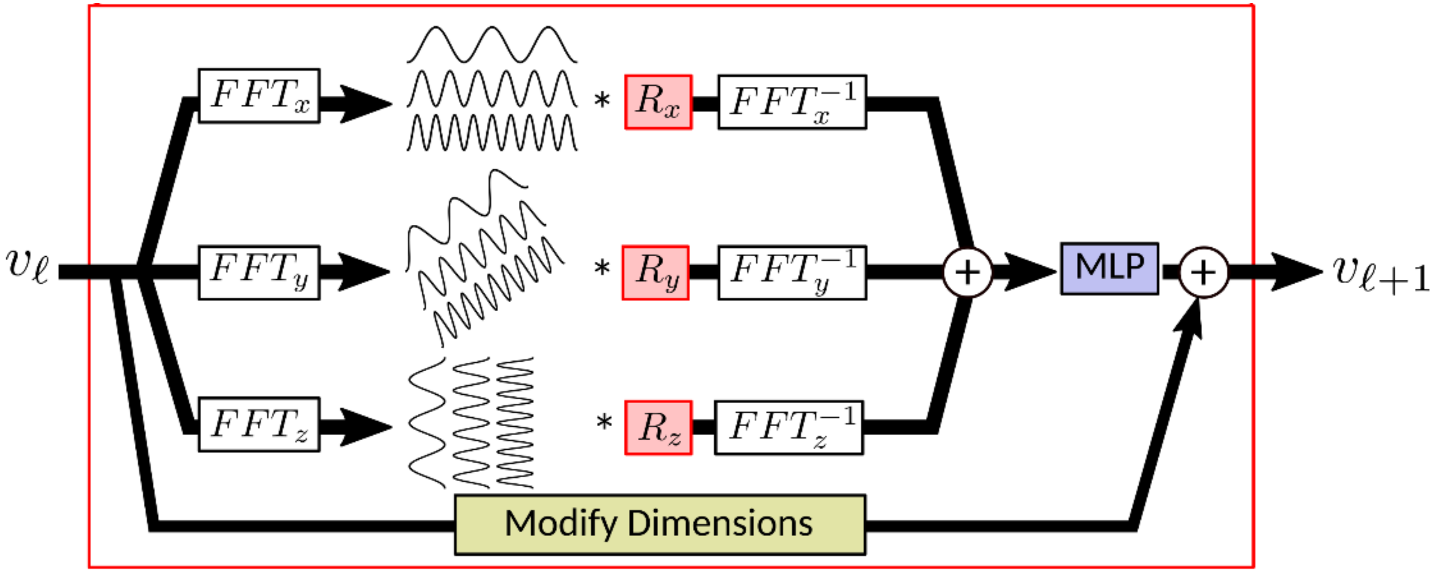
[Tran et al. 2023]

$$K_\phi(a)\boldsymbol{v} = FFT^{-1}\left(\boldsymbol{R}_\phi \cdot FFT(\boldsymbol{v})\right)$$

Dimensions:

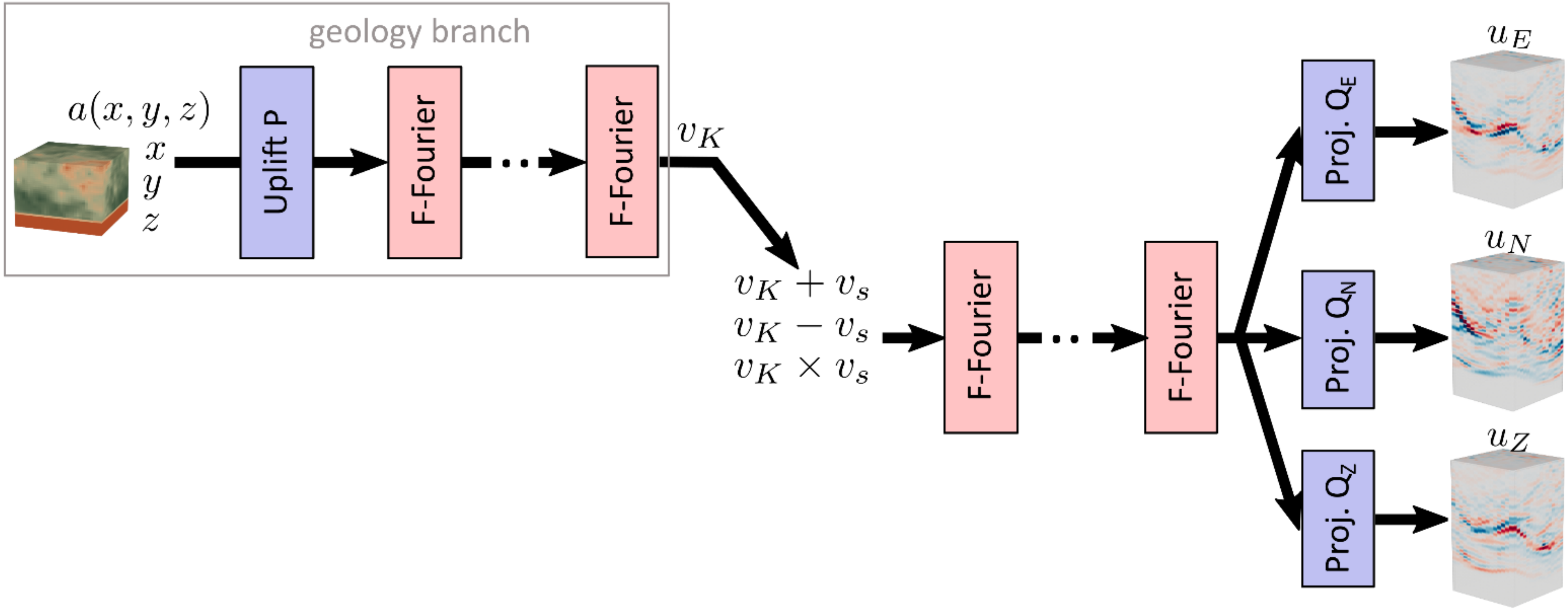
	input a	hidden variable $\boldsymbol{v}$	in Fourier space $\mathbf{R}_\phi := FFT(\boldsymbol{\kappa}_\phi)$
	$S_x \times S_y \times S_z$	$S_x \times S_y \times S_z \times d_v$	$M_x \times M_y \times M_z \times d_v \times d_v$

→ factorize the FFT: each factorized Fourier layer now has $(M_x + M_y + M_z) \times d_v \times d_v$



Multiple Input FNO (MIFNO)

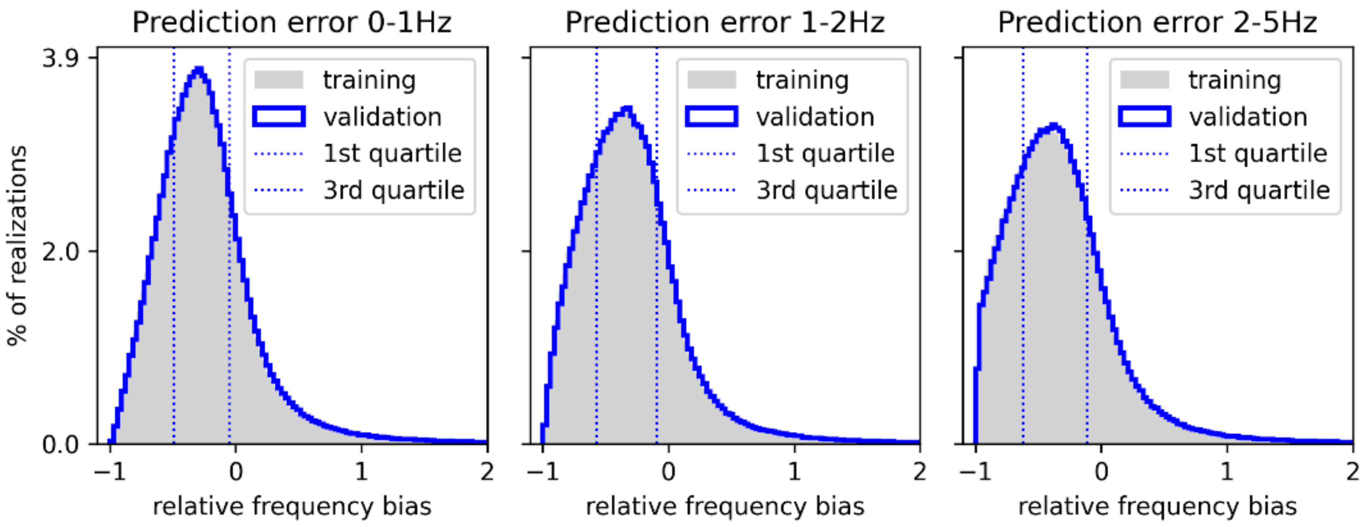
We propose a dedicated architecture for inputs with different representations.



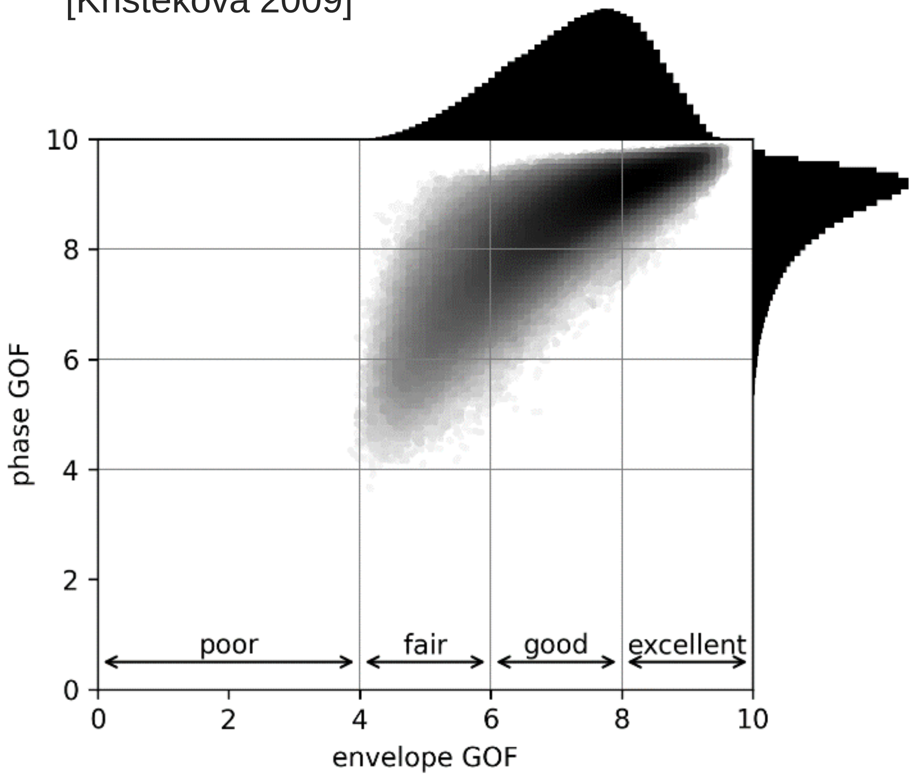
Lehmann et al., *in preparation*

Training results

MIFNO with 16 layers (3.4 million parameters)
27,000 training samples
3,000 validation samples
31h training on 4 GPUs

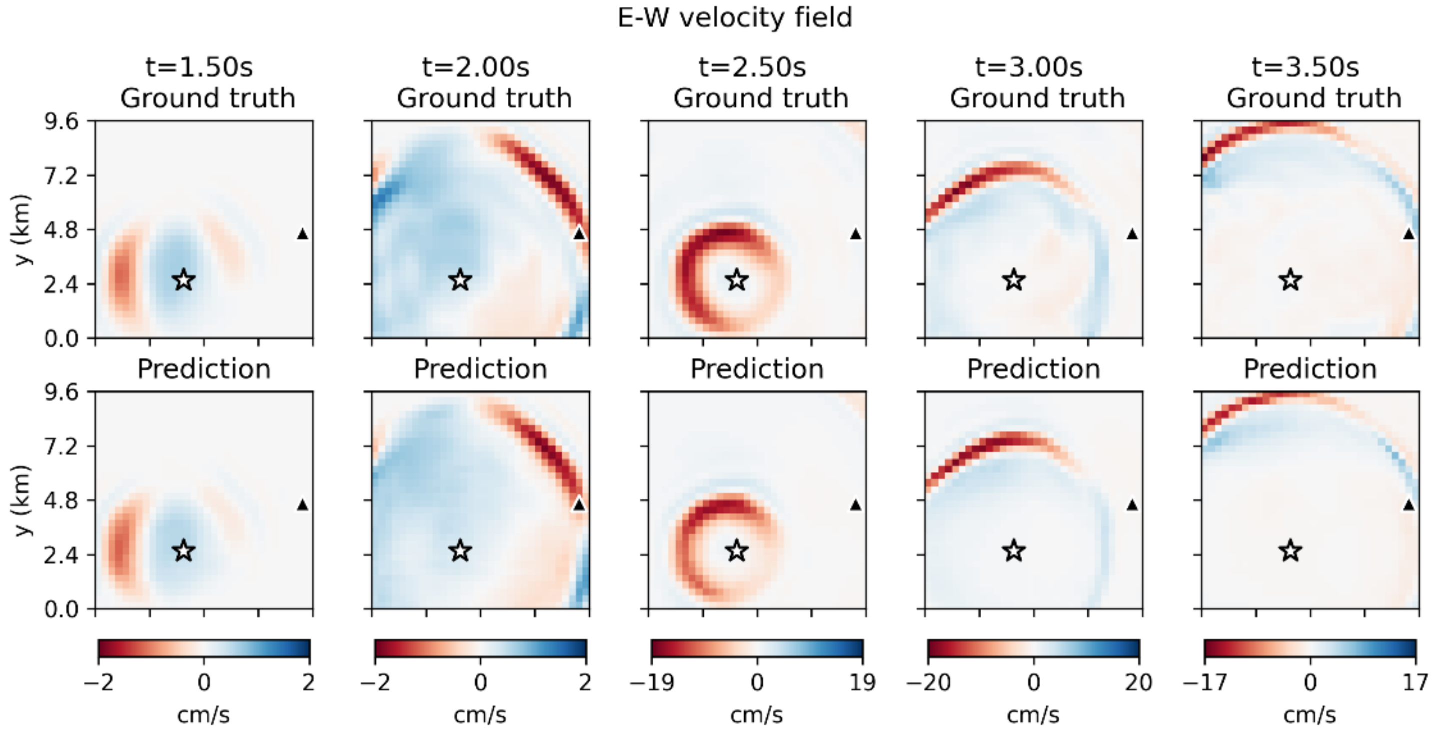


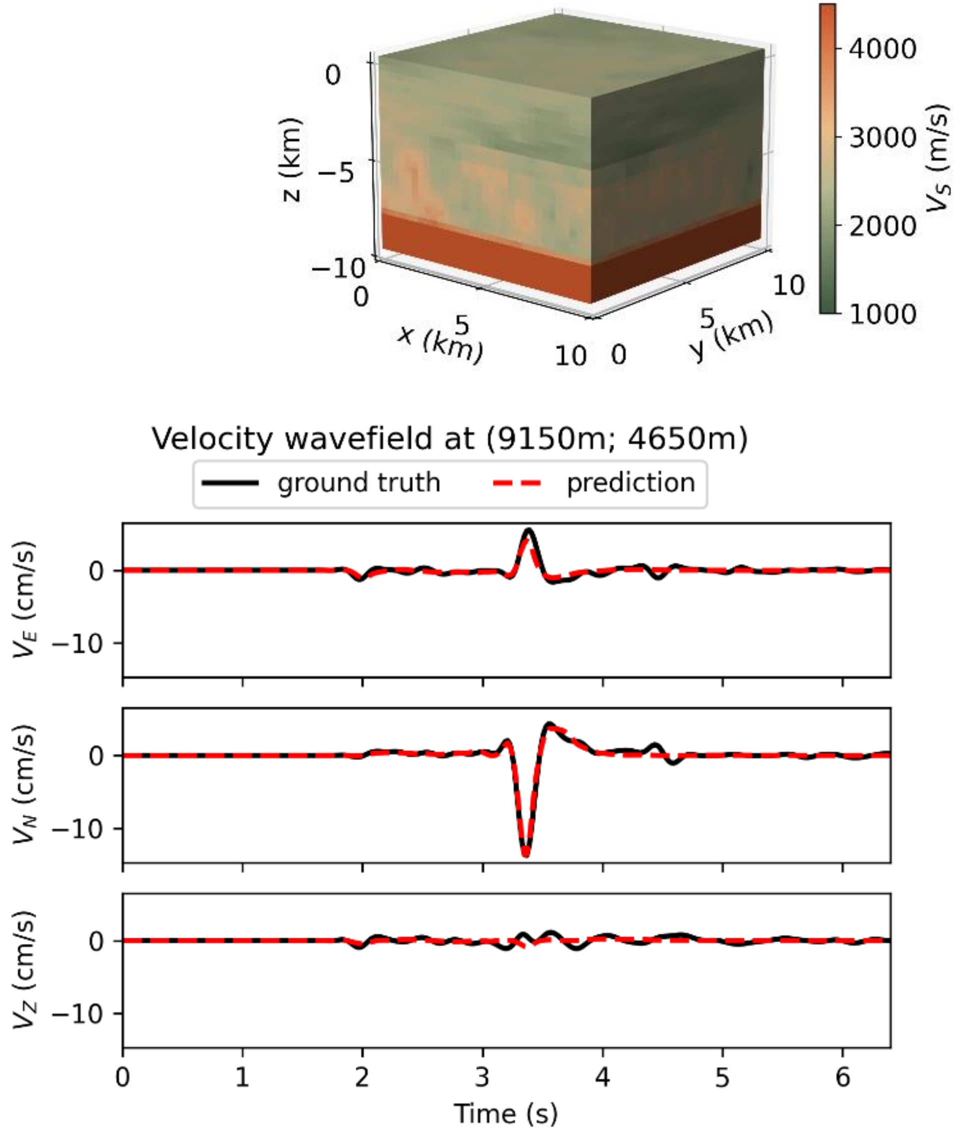
Goodness-Of-Fit (GOF)
[Kristekova 2009]



- ✓ no overfitting
- ✓ same accuracy on the 3 components
- ✓ 80% of predictions have excellent phase GOF
- ✓ 87% of predictions have good envelope GOF

Prediction for a new geology



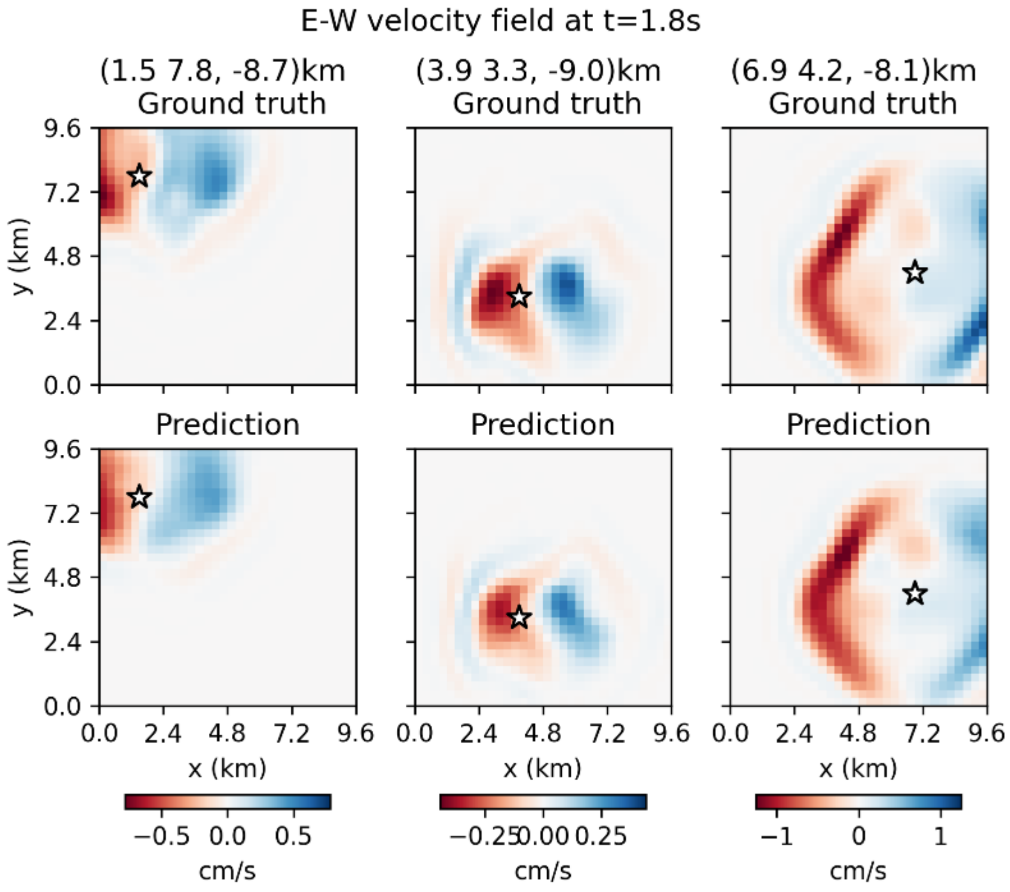


- ✓ Wave arrival times are very accurate
- ✓ Spatial evolution is well reproduced
- ✓ Main fluctuations are correctly captured
- → Small-scale features & small amplitudes more difficult

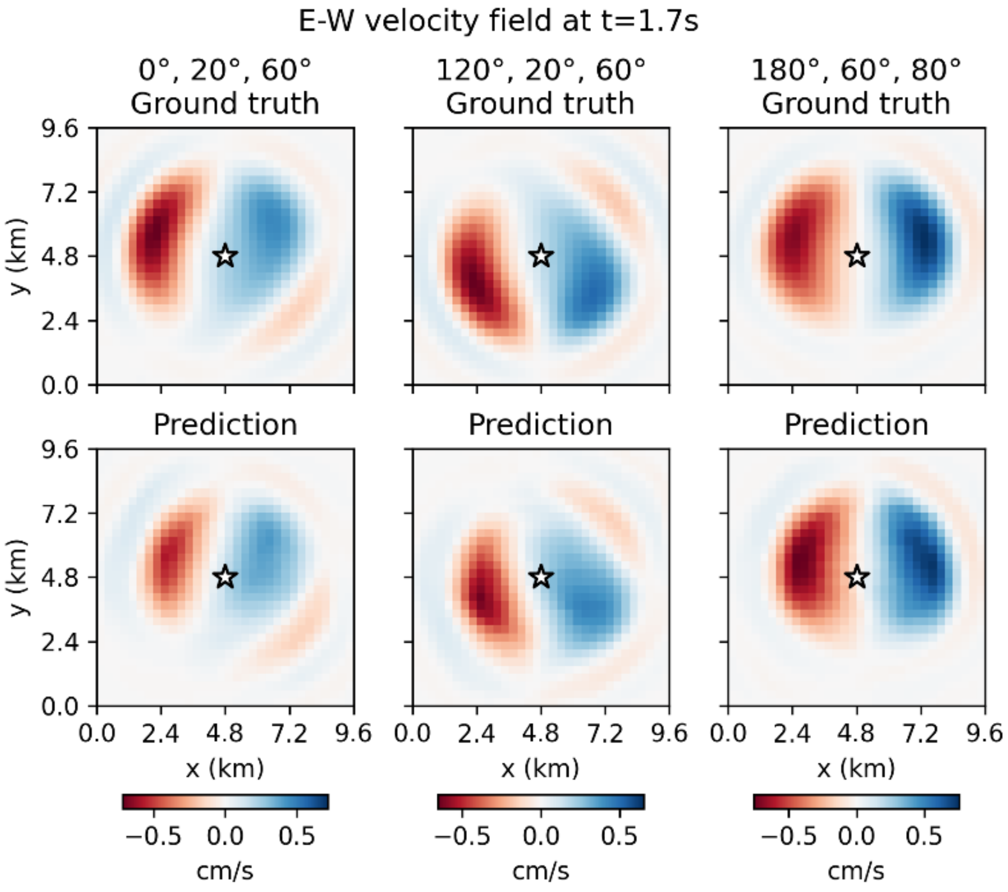
Lehmann et al., *CMAME*, 2024

Prediction for varying sources

For a given geology, move the source

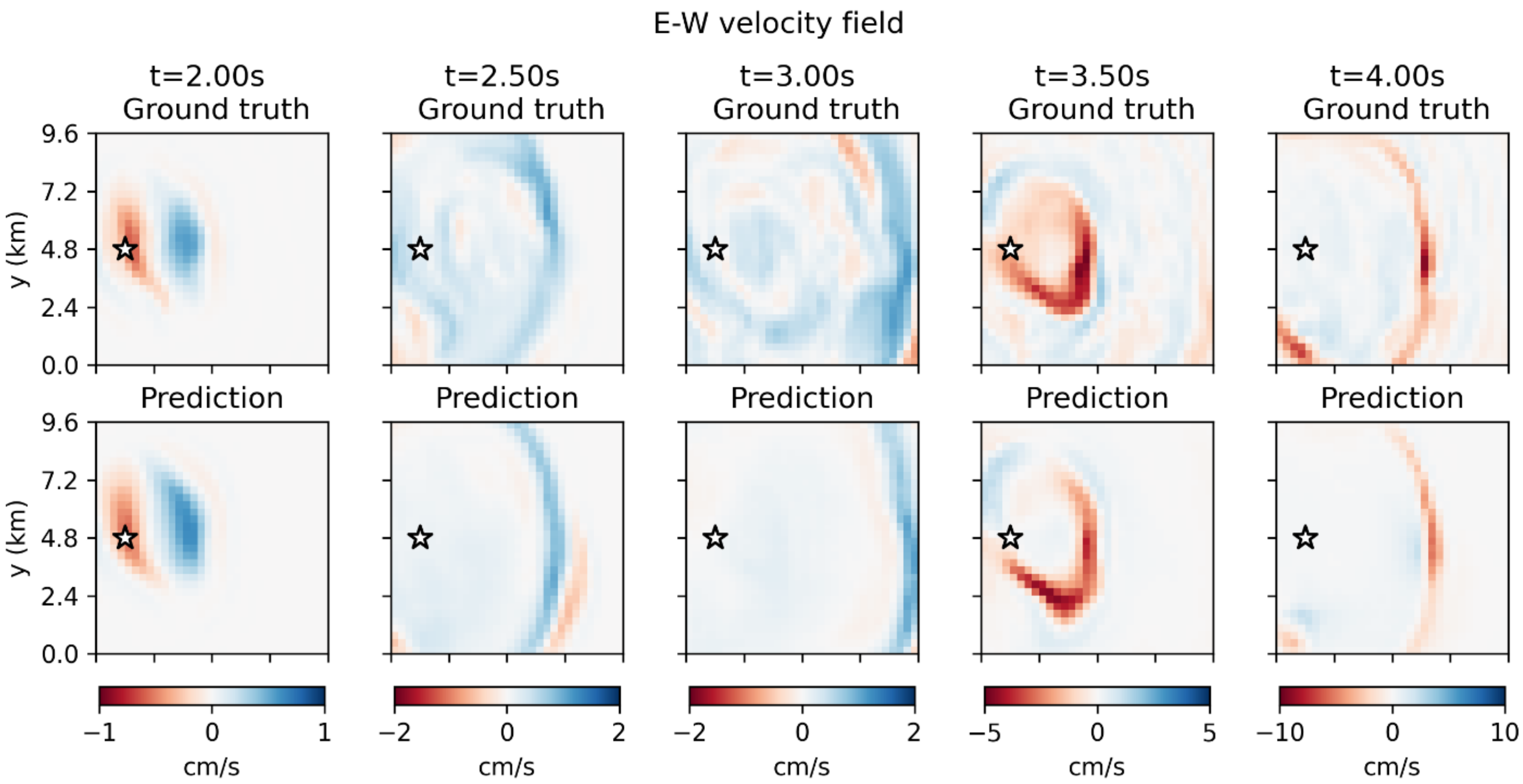


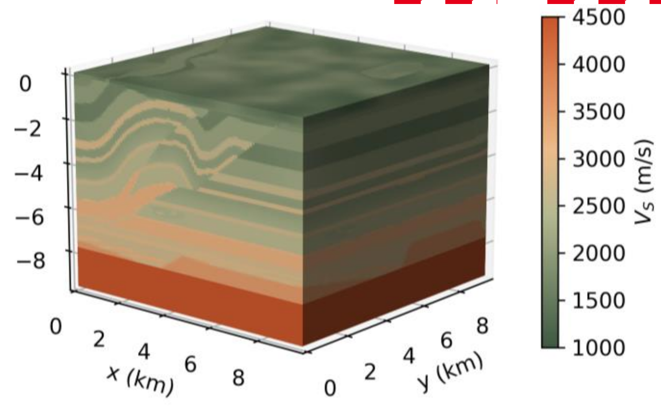
For a given geology, rotate the source



✓ Source encoding is efficient and accurate

Generalization to a realistic geology

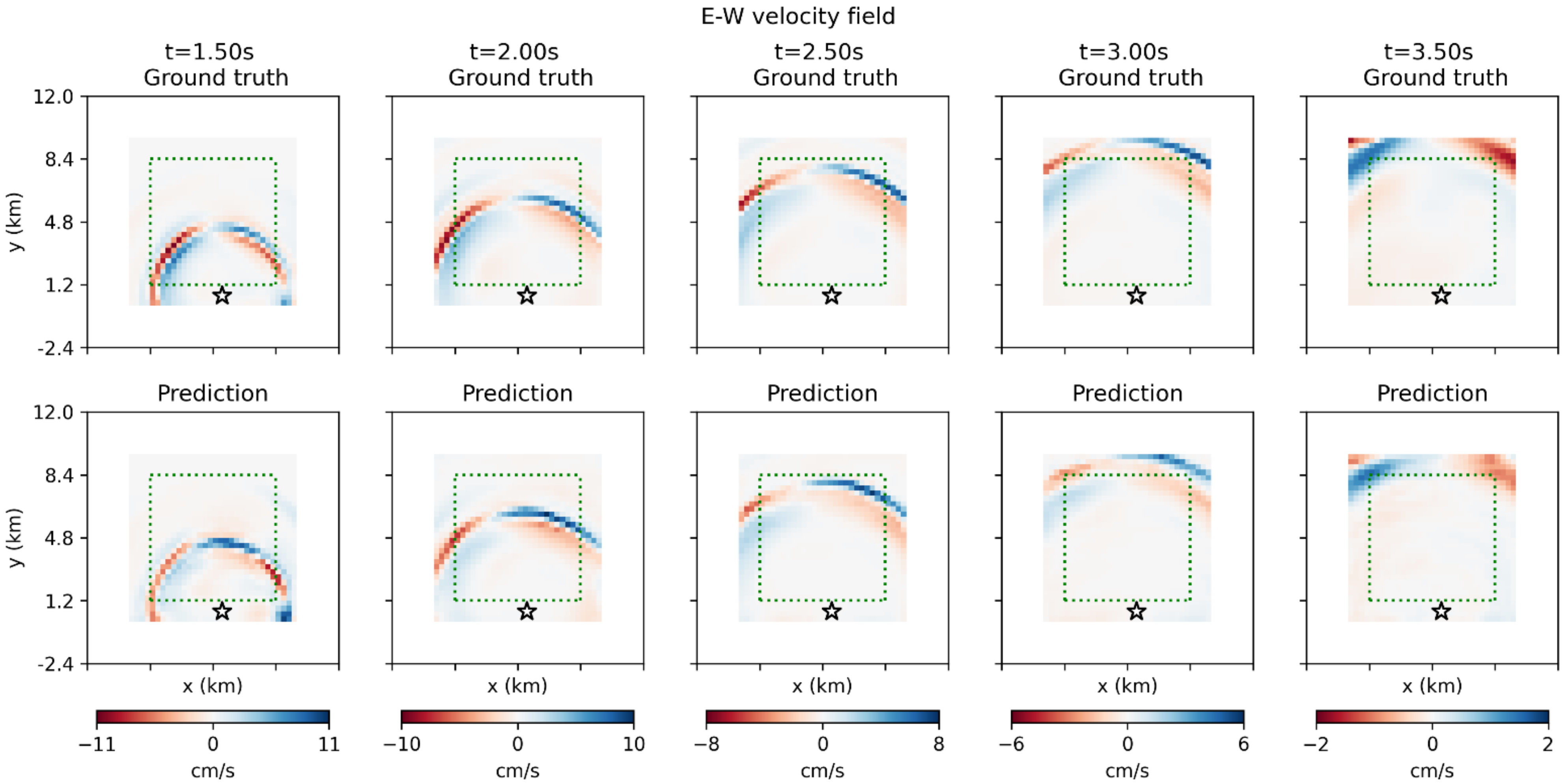




- ✓ Predictions are remarkably accurate for a geology that is far from the training dataset.
- → Slight time shift and lack of small-scale fluctuations

Lehmann et al., *in preparation*

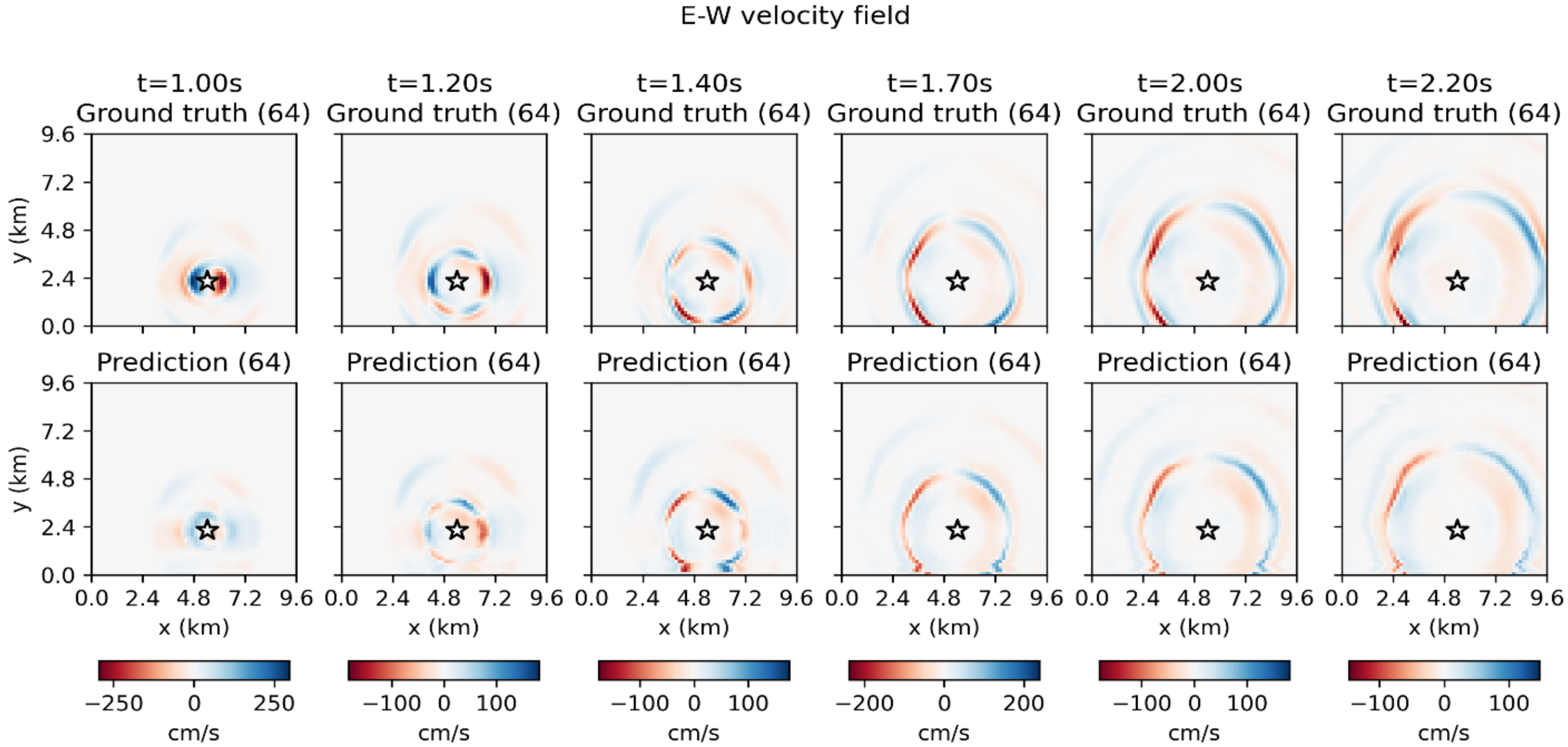
Generalization to out-of-distribution sources



✓ Accuracy is preserved when the source is slightly out of the training domain

Lehmann et al., *in preparation*

Generalization to higher resolution

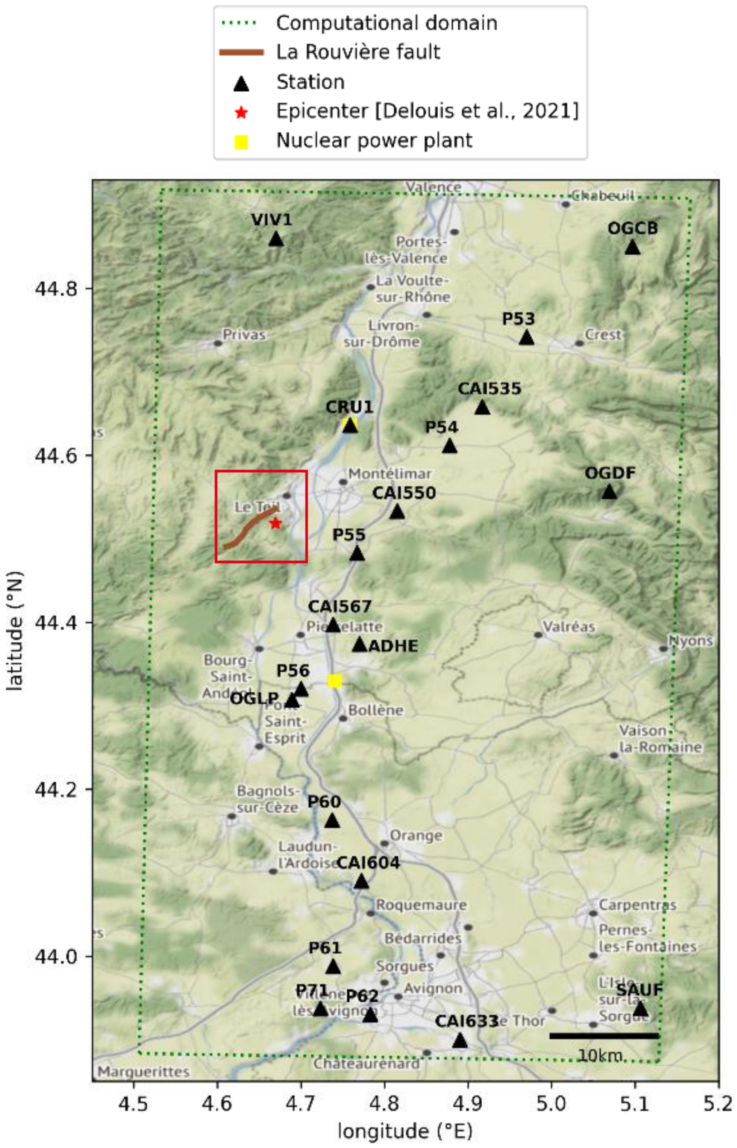


- The MIFNO can be applied to any higher resolution.

  The original FNO is resolution invariant under conditions on the frequency content of solutions [Bartolucci 2023].

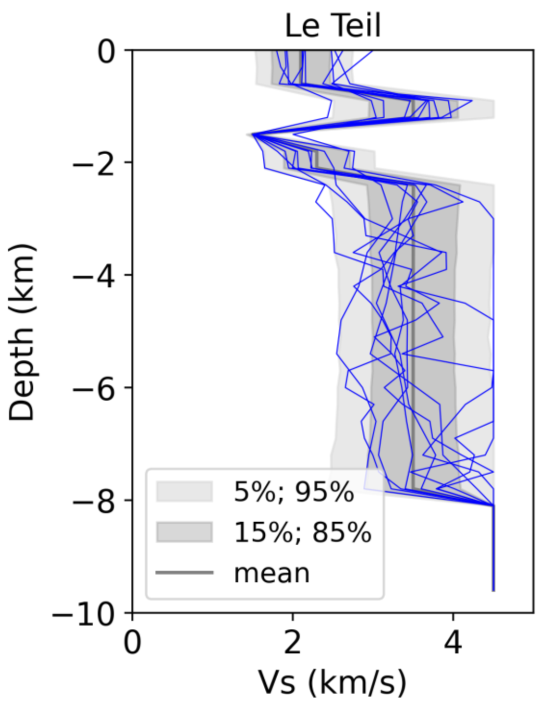
- High-resolution MIFNO can improve some features but inconsistencies remain

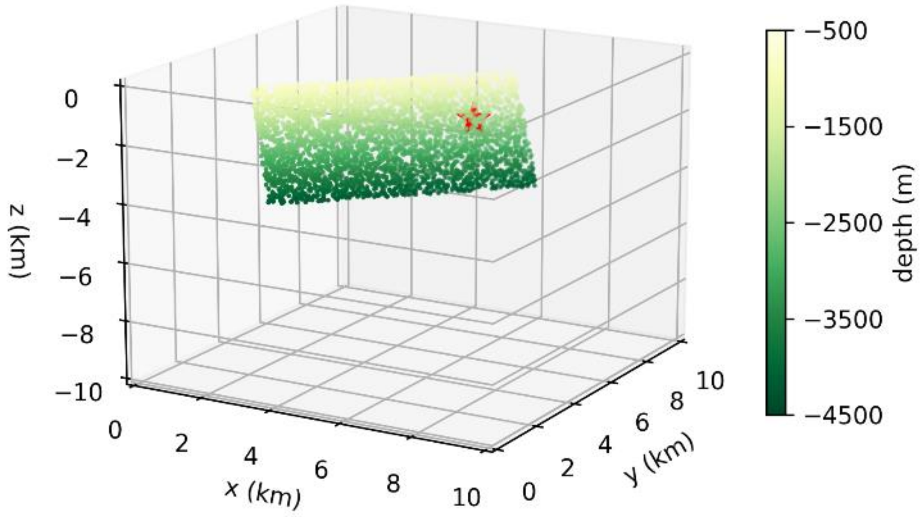
Le Teil earthquake



Design a specific (smaller) database:

- 4000 geologies built from the regional geology
- sources located along the fault plane
- source orientations from seismological inversion

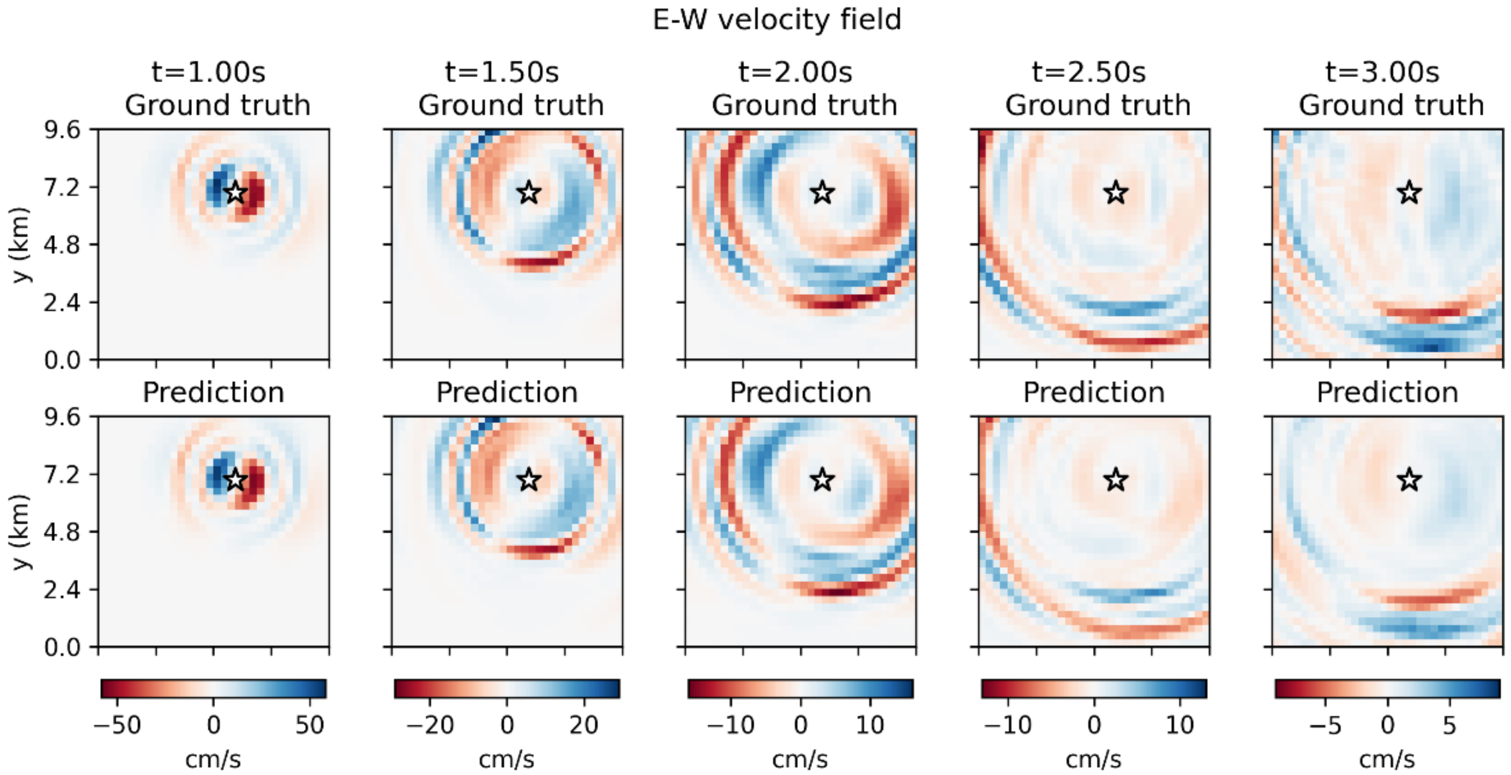


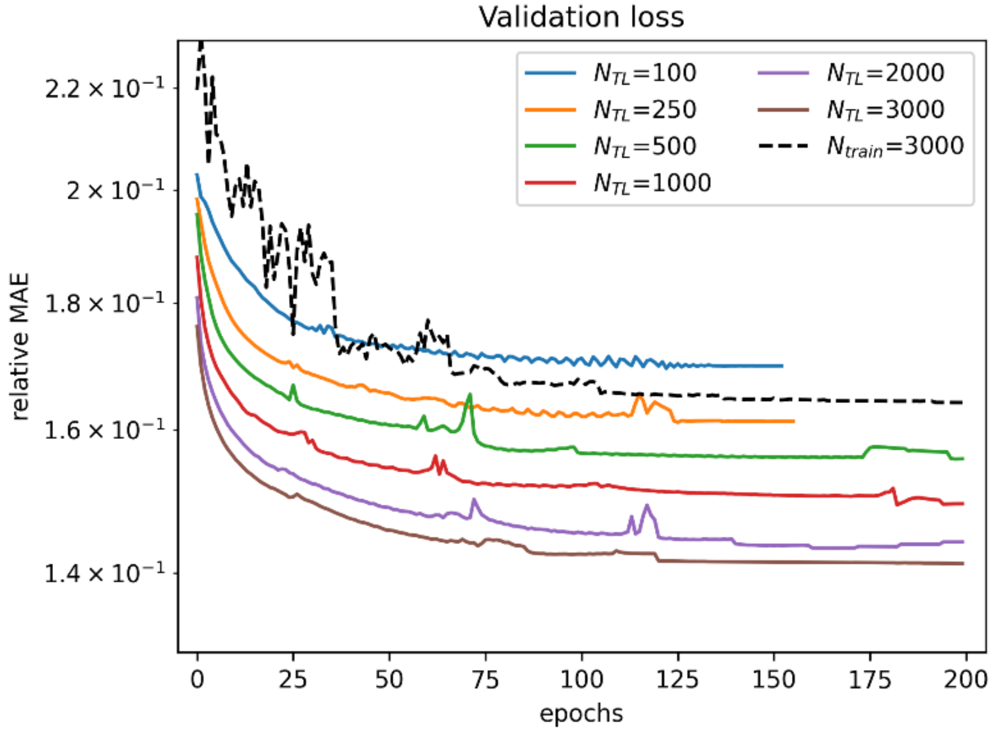


Transfer learning for UQ

1) Pretraining with 30 000 generic samples
2) Specialize with 100 to 3 000 specific samples

⇒ we need only 250 samples to achieve excellent phase accuracy and good envelope accuracy

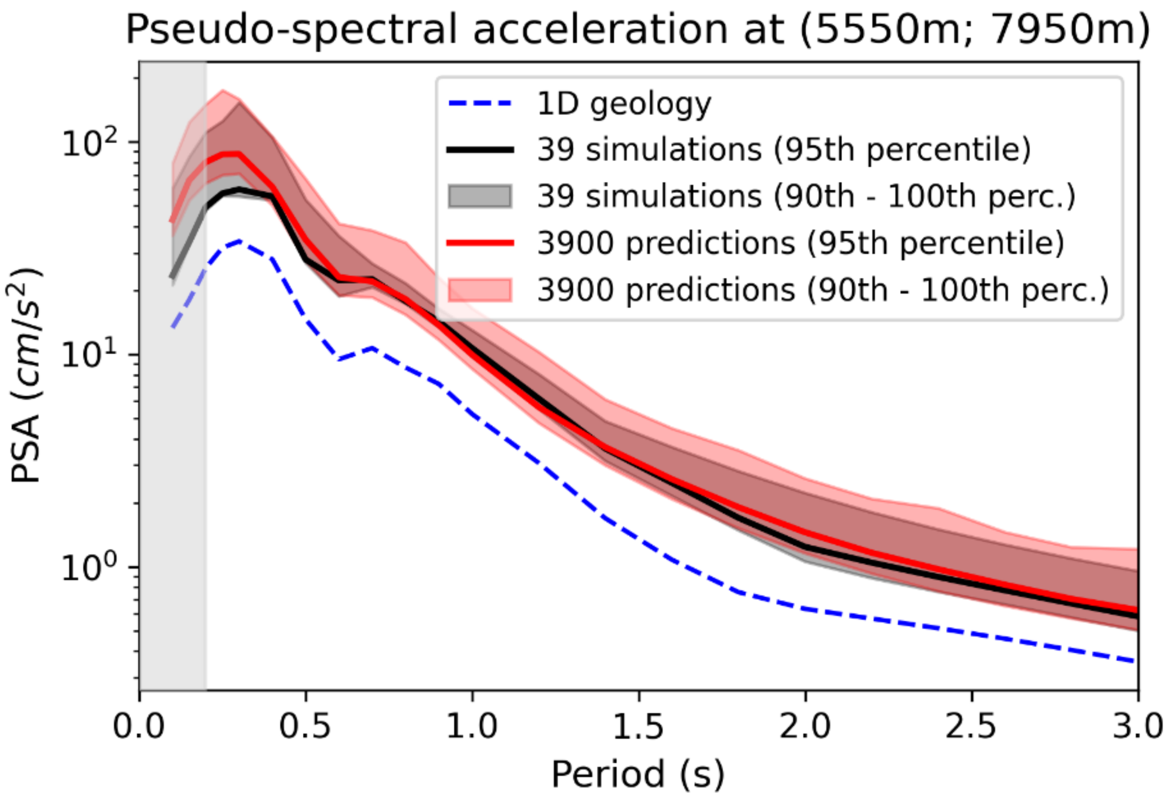




Distribution of intensity measures

We obtain fast distributions of quantities of interest, e.g. Pseudo-Spectral Acceleration (PSA)

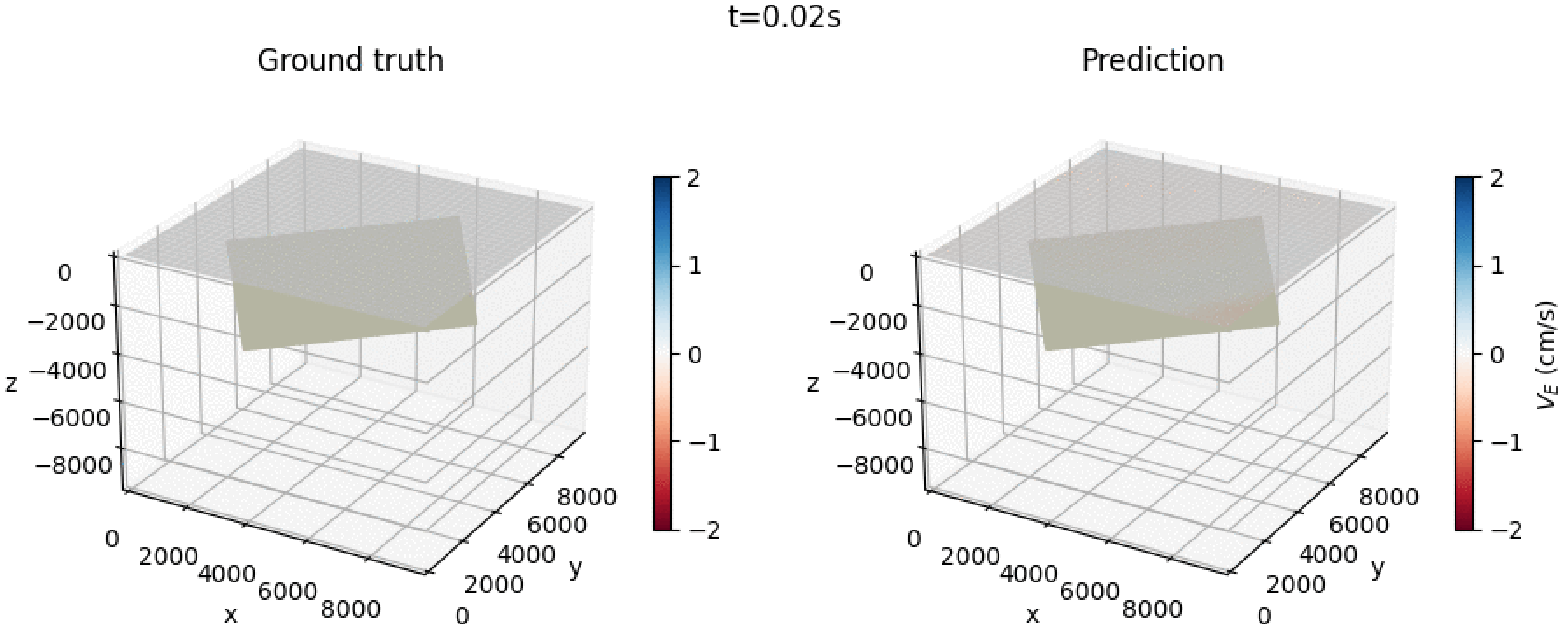
- ✓ Spread of the predicted distribution matches the simulation.
- ✓ Extreme values can be obtained with large samples.
- ✓ Predictions provide security margins.



Lehmann et al., *NeurIPS AI for Science workshop*, 2023

Conclusion

- ✓ The Multiple Input Fourier Neural Operator (MIFNO) predicts accurate ground motion for 3D geologies and various sources.
- ✓ Transfer learning is very beneficial to specialize the MIFNO to a given context.
- ✓ PSA distributions are coherent with simulations and can extend the range of extreme values
- → Constrain the predictions with observations.
- → Improve the high-resolution accuracy and extend the spatial domain.



References

Bartolucci, F., E. de Bézenac, B. Raonić, R. Molinaro, S. Mishra, et R. Alaifari. « Are Neural Operators Really Neural Operators? Frame Theory Meets Operator Learning ». arXiv, 31 mai 2023. http://arxiv.org/abs/2305.19913.

Kristeková, M., J. Kristek, et P. Moczo. « Time-Frequency Misfit and Goodness-of-Fit Criteria for Quantitative Comparison of Time Signals ». *Geophysical Journal International* 178, nº 2 (août 2009): 813-25. https://doi.org/10.1111/j.1365-246X.2009.04177.x.

Lehmann, F., F. Gatti, M. Bertin, et D. Clouteau. « Synthetic ground motions in heterogeneous geologies: the HEMEW-3D dataset for scientific machine learning ». *Earth System Science Data Discussions* 2024 (2024): 1-26. https://doi.org/10.5194/essd-2023-470.

Lehmann, F., F. Gatti, M. Bertin, et D. Clouteau. « 3D elastic wave propagation with a Factorized Fourier Neural Operator (F-FNO) ». *Computer Methods in Applied Mechanics and Engineering* 420 (15 février 2024): 116718. https://doi.org/10.1016/j.cma.2023.116718.

———. « Seismic hazard analysis with a Factorized Fourier Neural Operator (F-FNO) surrogate model enhanced by transfer learning ». In *NeurIPS 2023 AI for science workshop*, 2023. https://openreview.net/forum?id=xiNRyrBAjt.

Li, Z., N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, et A. Anandkumar. « Neural Operator: Graph Kernel Network for Partial Differential Equations ». arXiv, 6 mars 2020. http://arxiv.org/abs/2003.03485.

Li, Z., N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, et A. Anandkumar. « Fourier Neural Operator for Parametric Partial Differential Equations ». In *International Conference on Learning Representations*, 2021. https://doi.org/10.48550/arXiv.2010.08895.

Tran, A., A. Mathews, L. Xie, et C. Soon Ong. « Factorized fourier neural operators ». In *The eleventh international conference on learning representations*, 2023. https://openreview.net/forum?id=tmIiMPl4IPa.

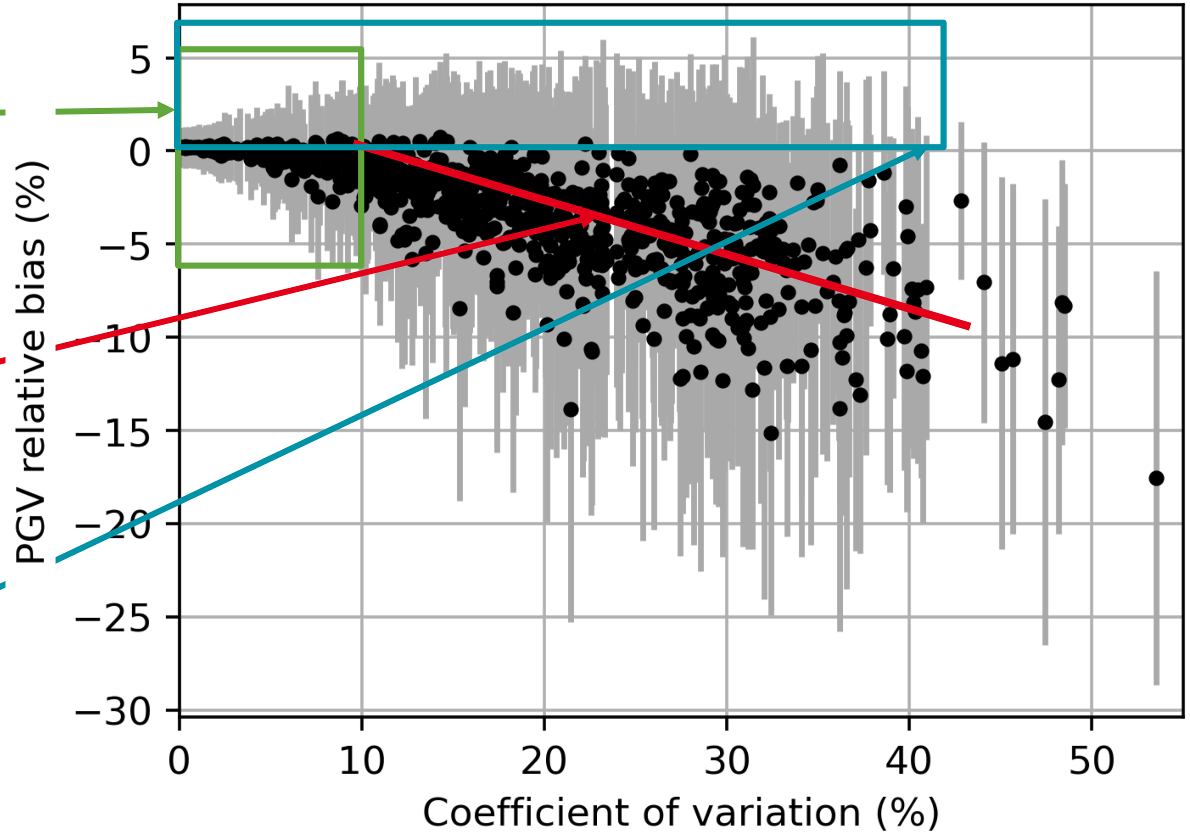
Influence of geological heterogeneities

Is the MIFNO able to predict accurate ground shaking?
→ Peak Ground Velocity estimation (PGV)

✓ The MIFNO predicts accurate PGV when geologies are not very heterogeneous (coef. var. < 10%)

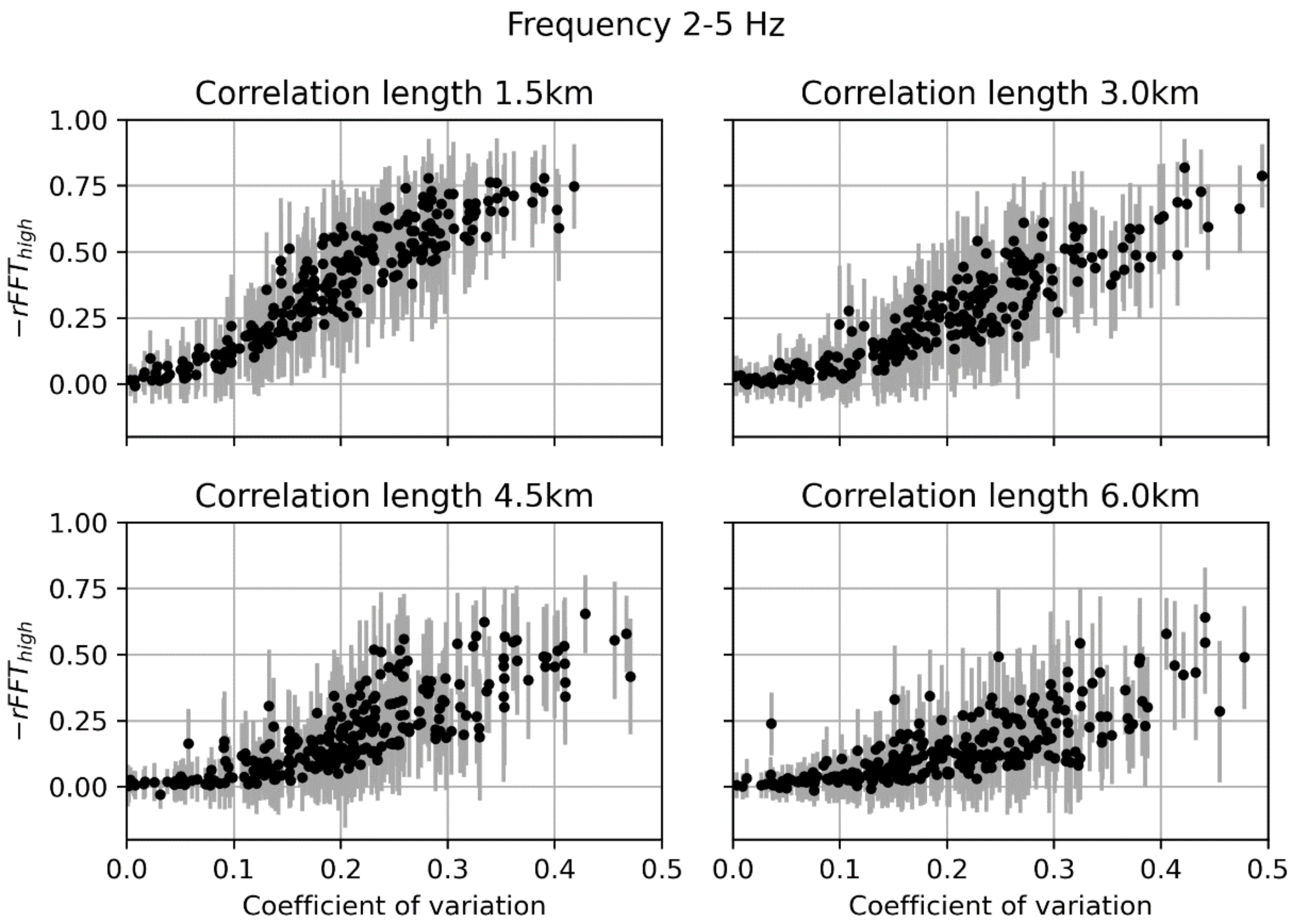
✗ The MIFNO tends to underestimate PGV when geologies are very heterogeneous

✓ You cannot always loose: some PGV will be overestimated

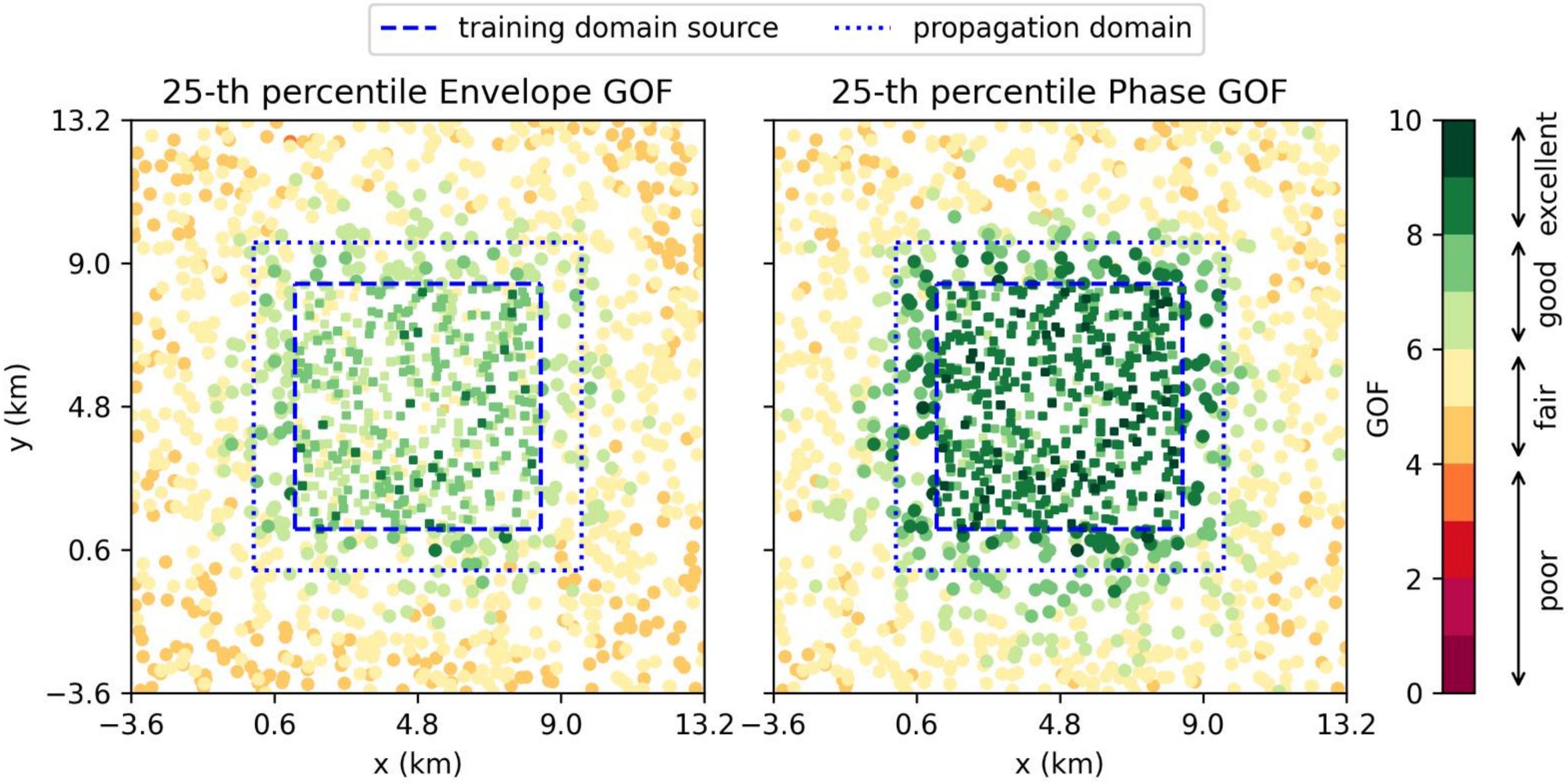


Lehmann et al., *NeurIPS AI4Science workshop*, 2023

Influence of geological heterogeneities



Generalization to out-of-distribution sources



Lehmann et al., *in preparation*

Le Teil transfer learning

Le Teil database with a random source along the fault plane

# samples	rRMSE	$rFFT_{low}$	$rFFT_{mid}$	$rFFT_{high}$	EG	PG
N_{train}=3000	0.40 ; 0.73	-0.49 ; -0.02	-0.60 ; -0.06	-0.67 ; -0.09	6.60 ; 8.43	8.25 ; 9.31
N_{TL}=0	0.64 ; 0.98	-0.44 ; 0.33	-0.64 ; 0.06	-0.68 ; -0.05	5.98 ; 7.69	6.30 ; 8.44
N_{TL}=100	0.41 ; 0.78	-0.44 ; 0.08	-0.56 ; 0.00	-0.66 ; -0.05	6.51 ; 8.32	8.17 ; 9.25
N_{TL}=250	0.38 ; 0.75	-0.41 ; 0.09	-0.52 ; 0.03	-0.61 ; -0.01	6.70 ; 8.47	8.38 ; 9.35
N_{TL}=500	0.37 ; 0.74	-0.38 ; 0.11	-0.49 ; 0.06	-0.58 ; 0.02	6.87 ; 8.57	8.51 ; 9.41
N_{TL}=1000	0.35 ; 0.69	-0.42 ; 0.03	-0.52 ; -0.02	-0.59 ; -0.04	6.92 ; 8.62	8.62 ; 9.46
N_{TL}=2000	0.33 ; 0.68	-0.38 ; 0.05	-0.47 ; 0.01	-0.54 ; -0.01	7.10 ; 8.72	8.72 ; 9.51
N_{TL}=3000	0.33 ; 0.68	-0.34 ; 0.08	-0.43 ; 0.04	-0.51 ; 0.02	7.20 ; 8.78	8.76 ; 9.53

Table 4.2: 1st and 3rd quartiles of the metrics computed on 700 test samples specific to the Le Teil region. (upper row): training with only 3000 specific data. In other experiments, transfer learning was used with 100 to 3000 samples (N_{TL}= number of transfer learning samples). rRMSE: relative RMSE (0 is best), $rFFT_{low}$: relative frequency bias 0-1Hz (0 is best), $rFFT_{mid}$: relative frequency bias 1-2Hz (0 is best), $rFFT_{high}$: relative frequency bias 2-5Hz (0 is best), EG: enveloppe Goodness-of-Fit (10 is best), PG: phase Goodness-of-Fit (10 is best). For frequency biases, negative values indicate underestimation.

Le Teil transfer learning