Learning the elastic wave equation with Fourier Neural Operators

Tianze Zhang, Kris Innanen and Daniel Trad
University of Calgary

Summary

Neural operators are extensions of neural networks which in supervised training learn how to map complex relationships, such as classes of PDE. Recent literature reports efforts to develop one type of these, the Fourier Neural Operator (FNO) such that it learns to create relatively general solutions to PDEs such as the Navier-Stokes equation. In this study, we seek what we believe is the first numerical instance of a Fourier Neural Operator (FNO) be trained to learn the elastic wave equation from a synthetic training data set. FNO attempts to find a manifold for elastic wave propagation. The FNO combines a linear transform, the Fourier transform, and a non-linear local activation, to produce a network with sufficient freedom to map from a general parameterization of a forward wave problem to its solution. Post-training, the FNO is observed to generate accurate elastic wave fields at approximately 10 times the speed of traditional finite difference methods on a CPU, and about a thousand times faster on a GPU.

Theory

Solving the elastic wave equation, a time-or frequency-domain multidimensional partial differential equation for one or more components of a wave field, is an essential part of seismic inversion and imaging. Standard methods like those based on finite differences and finite elements are based on discretization in space, on a grid or mesh. Derivatives are computed at or around each grid or mesh point, and so from a computational point of view, resolution and computational expense form a trade-off. A coarse grid is fast but captures lower-frequency and wavenumber behavior only; fine grids accommodate higher frequency and wavenumber wave phenomena but are computationally slow and expensive. These costs play a dominant role in the practical formulation (and use of) processes such as seismic waveform inversion, where for computational reasons we are often forced to choose between accuracy / completeness over computational expense and time. The expense of inversion caused by the expense of simulation grows with the maximum frequency desired and the maximum resolution needed, but also depth of investigation and model extent in general, and any other aspect of the problem that increases the length of the data trace being simulated or the number of source points being used. In all modern seismic imaging and inversion problems, including and especially waveform inversion, we are motivated to seek out accurate techniques for solution of elastic wave equations.

Inspired by the work introduced by Li Z [1], we modify their work and train the modified FNO to learn the elastic wave equation in both the time domain and frequency domain. We modified the FNO to make it more suitable for calculating derivatives in multiple directions. Figure 1 shows out modified method of the FNO. After our modification, an elastic FNO network could generate elastic wavefields accurately, and at a computational cost about 10 times less than a standard finite difference solver when computed on a CPU, and about a thousand times faster on a GPU.

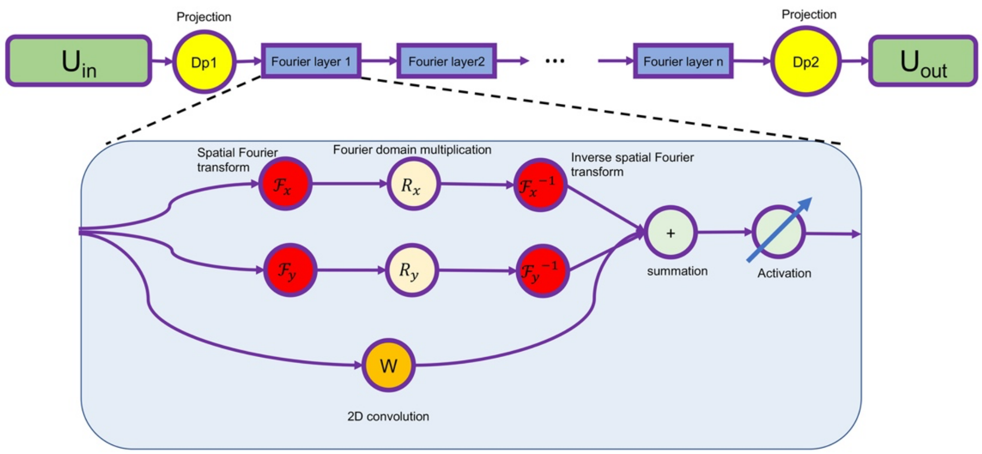

FIG1: The structure of the Fourier neural operator, aiming to train a network, which can give the rest steps of the wavefields given by the first steps of the wavefields. U_{in} and U_{out} are the input steps of the fields and the output predictions of the wavefields. Dp_1 and Dp_2 are the dimension projection layer at operates on time axis.

For time domain training, we use the wavefields generated with the stress velocity isotropic elastic wave equation, using the staggered grid finite difference method with 4-orders of accuracy in space as the training data. We generate 100 random models, and the random objects with the shape of rectangle and circle are located in the simulation area. The shape of the rectangle and the diameter of the circle are all randomly generated, and the locations of the objects are also randomly located. The VS and density models are obtained through scaling the VP model, with the relationship of $V_s = 0.7V_p$, and $\rho = 1.74V_p^{0.25}$. The sizes of these models are 84×84, and the grid lengths of the models are $\mathrm{dx} = \mathrm{dz} = 10$. The sources of the wavelet are also randomly located though out the velocity model. We use Ricker's wavelet as our source. When we are generating the training wavefields, the main frequency of the sources is also randomly generated, ranging from 5Hz-15Hz. Since the wave fields are generated with the finite difference method, we do not tend to use the sources with very high main frequencies to avoid numerical dispersion during the forward modeling. We use 20 layers of the PML boundary condition to absorb the boundary reflections.

Figure 2 shows the nine time slices of the true wavefields for one shot located in the right corner of the model. We will feed the network 50-time steps of the wavefields and train the network to generate the rest 300-time steps for the wavefields. We use 90 models out of the 100-model data set as the training models and the rest 10 models as the validation tests. The maximum epoch is 1500, and the training results are plotted in Figure 3. In this test, we will use 33 Fourier modes, and the dimension projection width is 60.

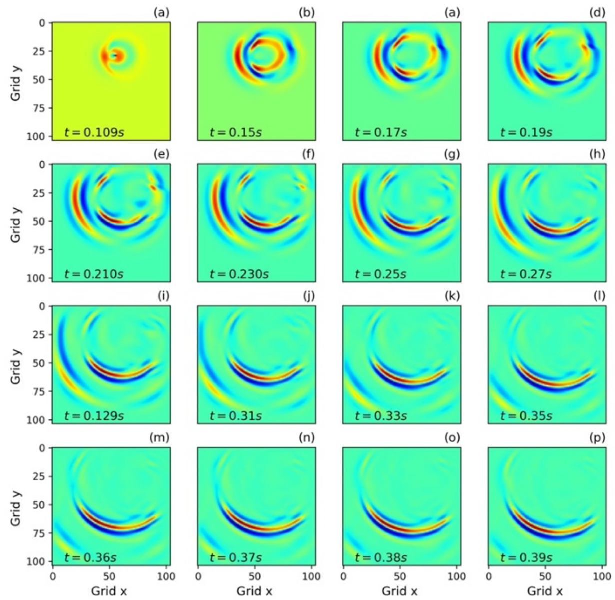

FIG2: Nine snap shots of the x component of the wavefields generated with staggered grid finite difference method of on random model, regarded as the ground truth for training.

The comparisons show that the FNO could correctly generate the phase position for different components of the wavefields, especially from Figure 3 (a)-(l). The prediction alliances well with the ground truth in Figure 2. However, with the increase of the propagation time, the accuracy of the prediction decreases, especially in Figure 3 (m)-(p). In Figure 3 (m) and 3 (n), though we have the correct phase of the waveforms, their amplitudes are distinct from the ground truth. In Figure 3 (o) and Figure 3 (p), we can also observe the incorrect source distortion influence that does not exist in Figure 2. A proper combination of the dimension projection width and the number of the Fourier model could help to release the problem.

Figure 4 shows the training in the frequency domain. We prepare 800 VP velocity models from the 3D Overthrust model and scaling them to obtain the VS and ρ model. The size of the models is 2km×2km, with grid length are dx=dz=20. We perform frequency domain forward modeling on these models, with staggered-grid finite difference method and get the frequency domain wave fields. In each frequency domain forward modeling the source location are randomly generated, and the main frequency is also randomly picked from the range of 1Hz-40Hz. Figure 4 shows the prediction results of the real and imaginary part of the fields, we could also see that the FNO could accurately predict the real and imaginary part of the fields.

Table 1 shows the Forward modeling computational performance comparison, and we can see that the fields generated with FNO is about a thousand time faster than the conventional finite difference method with acceptable accuracy.

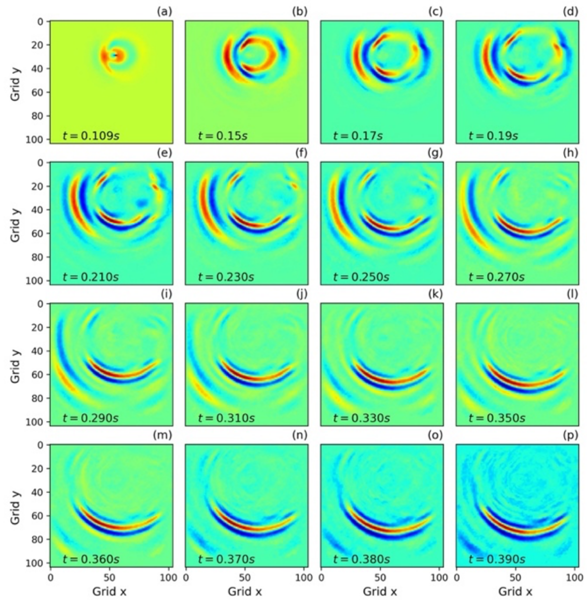

FIG3: Nine snap shots of the x component of the wavefields generated with the FNO. With dimension projection width 60 and 33 Fourier models.

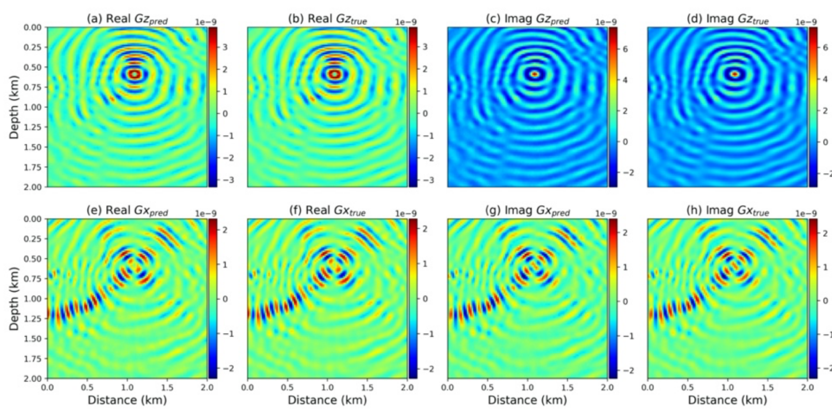

FIG4: The training results in frequency domain. The main frequency is 31Hz, and the source location is located at 0.25km in depth and 1.37km in distance of the model. (a)-(b) are the real part of the wave fields in z direction. (c)-(d) are the wavefields in the x direction. (e)-(f) are real part the fields in x direction. (g)-(h) are the imaginary part of the fields in x direction.

Conclusions

In this study, we train a fast forward modeling method with the Fourier Neural Operator (FNO). If the network is well trained, the computational speed for generating the wavefields is about a thousand times faster than the traditional finite difference method (python based). The network consists of three parts, which are two dimensions projection layers that operate on time, and several Fourier layer that learns the spatial partial derivatives. The power of the Fourier Neural operator comes from the combination of the linear operation, operators that resemble partial differential calculation (via the Fourier transform), and the non-linear local activation. The training in the time domain, the numerical tests suggest that FNO could generate promising wavefields within certain prediction steps, however, with the decreasing of accuracy as time propagates. In order to avoid this issue. We also perform the training in the frequency domain. The numerical results shows that the predictions in the frequency is also fast and promising.

Table 1: Forward modeling computational performance comparison (Time domain training)

Modeling method	CPU(s)	GPU(s)	CPU/GPU Ratio	parameters	Min loss
Finite Difference method	1.345670445	9.5749×10^{-1}	1.4	NA	NA
FNO(10, 33)	1.0359×10^{-1}	1.2264×10^{-3}	84	3113670	25.0993
FNO(20, 33)	1.3362×10^{-1}	1.2560×10^{-3}	106	12266640	11.4401
FNO(30, 33)	2.9434×10^{-1}	1.3055×10^{-3}	225	48871980	4.8973
FNO(60, 33)	4.8886×10^{-1}	2.4427×10^{-3}	200	109876520	3.4645
FNO(40, 10)	1.1607×10^{-1}	3.5792×10^{-3}	32	4564780	8.5050
FNO(40, 20)	1.5244×10^{-1}	1.7740×10^{-3}	85	18004780	5.6621
FNO(40, 40)	2.6185×10^{-1}	1.9676×10^{-3}	133	48871980	4.8973
FNO(40, 60)	3.3643×10^{-1}	2.4288×10^{-3}	138	71764780	5.4063

Acknowledgements

We thank the sponsors of CREWES for continued support. This work was funded by CREWES industrial sponsors and NSERC (Natural Science and Engineering Research Council of Canada) through the grant CRDPJ 543578-19. The first author is also supported by the Chine Scholarship Council (CSC).

References

[1] Li Z, Kovachki N, Azizzadenesheli K, et al. Fourier neural operator for parametric partial differential equations[J]. arXiv preprint arXiv:2010.08895, 2020.

[2] Raissi, M., Perdikaris, P., and Karniadakis, G. E., 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.

[3] Sun, J., Niu, Z., Innanen, K. A., Li, J., and Trad, D. O., 2020, A theory-guided deep-learning formulation and optimization of seismic waveform inversion: Geophysics, 85, No. 2, R87–R99.

[4] Kochkov, D., Smith, J. A., Alieva, A., Wang, Q., Brenner, M. P., and Hoyer, S., 2021, Machine learning–accelerated computational fluid dynamics: Proceedings of the National Academy of Sciences, 118, No. 21.