

GaborPINN: Efficient Physics-Informed Neural Networks Using Multiplicative Filtered Networks

Xinquan Huang[✉] and Tariq Alkhalifah, *Member, IEEE*

Abstract—The computation of the seismic wavefield by solving the Helmholtz equation is crucial to many practical applications, e.g., full waveform inversion (FWI). Physics-informed neural networks (PINNs) provide functional wavefield solutions represented by neural networks (NNs), but their convergence is slow. To address this problem, we propose a modified PINN using multiplicative filtered networks (MFNs), which embeds some of the known characteristics of the wavefield in training, e.g., frequency, to achieve much faster convergence. Specifically, we use the Gabor basis function due to its proven ability to represent wavefields accurately and refer to the implementation as GaborPINN. Meanwhile, we incorporate prior information on the frequency of the wavefield into the design of the method to mitigate the influence of the discontinuity of the represented wavefield by GaborPINN. The proposed method achieves up to a two-magnitude increase in the speed of convergence when compared with the conventional PINNs.

Index Terms—Gabor basis function, Helmholtz equation, partial differential equation, physics-informed neural networks (PINNs).

I. INTRODUCTION

SEISMIC wavefield simulation is a crucial and computationally intensive part of the many seismic imaging problems, e.g., reverse time migration and full waveform inversion (FWI). An efficient simulation approach is quite critical to practical applications. Compared with the time domain, frequency-domain modeling is often more efficient for applications such as FWI [1], [2]. However, the solution requires the calculation of the inverse of the impedance matrix, which consumes a lot of memory, and this problem becomes more drastic as the model size increases, like in 3-D. Furthermore, dense discretization is required for high-accuracy simulation when dealing with irregular geometry, e.g., topology and complex subsurface structures. With the recent developments in machine learning in science and engineering (so-called scientific machine learning) [3], [4], [5], [6], one type of approach, which embeds physical knowledge into the training, named physics-informed neural network (PINN), has provided the potential to solve this problem. Specifically, we can use a neural network (NN) to represent the wavefield as a function of space (and time) and train it to satisfy the governing equation. The simulation is executed based on queries, with inputs being

Manuscript received 27 May 2023; revised 10 October 2023; accepted 3 November 2023. Date of publication 7 November 2023; date of current version 16 November 2023. (*Corresponding author: Xinquan Huang*)

The authors are with the Physical Science and Engineering Division, KAUST, Thuwal 23955, Saudi Arabia (e-mail: xinquan.huang@kaust.edu.sa; tariq.alkhalifah@kaust.edu.sa).

Digital Object Identifier 10.1109/LGRS.2023.3330774

space coordinates and time and outputs being the wavefield values. This machine-learned function form of the wavefield [7], [8], [9] allows for easy handling of irregular geometry and can adapt to more complex wave equations corresponding to more complex media [10].

However, the scalability of the approach is limited by the cost of the training for PINNs [11]. Specifically, PINNs' training often provides solutions for one instance (e.g., one velocity model), and the convergence of each training may require thousands of epochs, making the total computational cost less competitive than numerical methods. So, *how can we make PINNs learn faster is an interesting, challenging, but inevitable topic*. As we know, there are three main components in PINNs: the NN architecture, the training process, and the loss function [12]. As for the training process, Huang and Alkhalifah [13] proposed a frequency upscaling and neuron splitting algorithm, resulting in a more stable and faster convergence, and Waheed et al. [14] proposed to use transfer learning to improve the computational efficiency by reducing the epochs needed for the convergence when applied to new velocity models. As for the loss function design, Xiang et al. [15] proposed a self-adaptive loss function through the adaptive weights for each loss term to adjust the collocation point samples in the domain to improve the accuracy of PINNs. Huang and Alkhalifah [16] proposed a single reference frequency loss function to improve the convergence of the multifrequency wavefield representation. While for the NN architecture design, almost all the backbone architectures used are the vanilla multilayer perception (MLP) with different activation functions. In this letter, we focus on the development of this aspect.

To make the NN fit the wavefield faster, prior knowledge should be embedded in the design of the PINN. For example, the Gabor function has been shown to effectively represent the seismic wavefield [17]. Inspired by this fact, we include the Gabor function into the NN by means of the multiplicative filtered network (MFN) [18] to accelerate the convergence of PINNs. We propose a modified PINN with MFN and we refer to this network as GaborPINN. In this framework, we represent the wavefield by a linear combination of Gabor basis functions of the input coordinates (GaborNet), in which the scale factor is determined by the frequency of the wavefield. The prior information on this combination would be beneficial for the convergence of the fitting as the seismic wavefield could naturally be represented as a linear combination of basis functions [19], e.g., Gabor basis. Although Fathony et al. [18] mentioned that this type of NNs retains some drawbacks such

as the lack of smoothness in the represented function and its gradients, we found that with the proper scale selection for the Gabor function, this problem can be avoided. We demonstrate the advantages of the method on a simple layered model extracted from the Marmousi model and also discuss the scale factor selection. Further experiments on higher frequency wavefields show that the proposed method results in faster convergence and provides higher accuracy where the vanilla PINN fails.

II. METHODOLOGY

The framework of PINNs aims to train an NN function using the governing equations of the physical system as a loss function. Here, we take the Helmholtz equation for a scattered wavefield [7] as an example

$$\frac{\omega^2}{\mathbf{v}^2} \delta \mathbf{U} + \nabla^2 \delta \mathbf{U} + \omega^2 \left(\frac{1}{\mathbf{v}^2} - \frac{1}{\mathbf{v}_0^2} \right) \mathbf{U}_0 = 0 \quad (1)$$

where \mathbf{U}_0 is the background wavefield analytically calculated for a constant velocity \mathbf{v}_0 [20]

$$\mathbf{U}_0(x, z) = \frac{i}{4} \mathbf{H}_0^{(2)} \left(\omega \sqrt{\frac{\{(x - s_x)^2 + (z - s_z)^2\}}{\mathbf{v}_0^2}} \right) \quad (2)$$

where $\mathbf{H}_0^{(2)}$ is the zero-order Hankel function of the second kind, $\delta \mathbf{U}$ is the scattered wavefield, $\delta \mathbf{U} = \mathbf{U} - \mathbf{U}_0$, \mathbf{v} is the velocity, and ω is the angular frequency. Unlike the full wavefield, the scattered wavefield helps us avoid the point source singularity [7]. We define an NN function $\Phi(\mathbf{x})$ to map from the input coordinates to the output real and imaginary parts of the scattered wavefield value for the input location, where $\mathbf{x} = \{x, z, s_x\}$ (in 2-D case, and sources are considered on the surface) represents the spatial coordinates and source location. To train an NN to satisfy the governing (1), we evaluate the PDE residuals, given the input coordinates and the output wavefield of the NN, as a loss function to train the NN. Thus, the loss function is defined as

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N \left| \frac{\omega^2}{(\mathbf{v}^i)^2} \Phi(\mathbf{x}^i) + \nabla^2 \Phi(\mathbf{x}^i) + \left(\frac{\omega^2}{(\mathbf{v}^i)^2} - \frac{\omega^2}{(\mathbf{v}_0^i)^2} \right) U_0^i \right|^2 \quad (3)$$

where U_0^i are the samples of \mathbf{U}_0 at point x^i . With this loss function, the NN can be trained as an alternative to the seismic wavefield simulation.

However, using the loss function (3), without additional constraints, may allow the network to converge to trivial solutions, e.g., a solution proportional to the negative background wavefield. Inserting $\Phi = -\mathbf{U}_0$ into (3) yields

$$\begin{aligned} \mathcal{L}_t &= \frac{1}{N} \sum_{i=1}^N \left| -\frac{\omega^2}{(\mathbf{v}^i)^2} U_0^i - \nabla^2 U_0^i + \omega^2 \left(\frac{1}{(\mathbf{v}^i)^2} - \frac{1}{(\mathbf{v}_0^i)^2} \right) U_0^i \right|_2^2 \\ &= \frac{1}{N} \sum_{i=1}^N \left| -\nabla^2 U_0^i + \omega^2 \left(-\frac{1}{(\mathbf{v}_0^i)^2} \right) U_0^i \right|_2^2. \end{aligned} \quad (4)$$

Since the background wavefield \mathbf{U}_0 satisfies the Helmholtz equation

$$\left(\frac{\omega^2}{\mathbf{v}_0^2} + \nabla^2 \right) \mathbf{U}_0(\mathbf{x}) = \mathbf{s} \quad (5)$$

where \mathbf{s} is the point source, and $\Phi = -\mathbf{U}_0$ is a trivial solution. Thus, the value of the loss \mathcal{L}_t , for most collocation points (samples that are not close to the source) in the domain, is equal to zero for the trivial solution. As the NN is trained to minimize the loss function for all the collocation points, training a network without proper initialization or other strategies (such as PINNup [13]) would push the NN to the trivial solution. In this letter, to make the training of PINN stable and avoid the trivial solution, we propose a soft constraint (penalty term) to push the NN to the right path away from the trivial solution. Considering that the value of the scattered wavefield near the source location is close to zero, while the value of the background wavefield near the source is often large, we add a penalty term that regularizes the predicted wavefield near the source (the region covering an area of almost one wavelength away from the source location). This penalty is given by

$$\mathcal{L}_{\text{reg}} = \frac{1}{N_{\text{reg}}} \sum_{i=1}^{N_{\text{reg}}} |\Phi(\mathbf{x}^i)|_2^2 \quad (6)$$

where N_{reg} is the number of samples used to calculate the penalty term. The number of training data within half of the wavelength of the source location determines this number.

In the vanilla PINN, the backbone NN is an MLP, given by the following form:

$$\mathbf{h}^{(i+1)} = \sigma(\mathbf{W}^{(i)} \mathbf{h}^{(i)} + \mathbf{b}^{(i)}), \quad i = 1, \dots, L-1 \quad (7)$$

where $\mathbf{h}^{(i)}$ is the hidden layer output of layer i , σ is a nonlinear activation function (we use the sine function here), \mathbf{W} is the weight matrix, and \mathbf{b} is the bias vector. In our method, we use the multiplicative filter network [18], which uses a different recursion that never results in the composition of nonlinear functions. The hidden layer, indexed by i , is defined as

$$\mathbf{h}^{(i+1)} = (\mathbf{W}^{(i)} \mathbf{h}^{(i)} + \mathbf{b}^{(i)}) \circ f(\mathbf{x}; \theta^{(i+1)}), \quad i = 1, \dots, k-1 \quad (8)$$

where f is the filter function parameterized by θ directly applied to the input \mathbf{x} , and \circ is an elementwise multiplication. Specially, we use the Gabor function here (GaborNet) because it often well represents the wavefield as a basis function and is given by

$$f_j(\mathbf{x}; \theta^{(i)}) = \exp \left(-\frac{\gamma_j^{(i)}}{2} \|\mathbf{x} - \mu_j^{(i)}\|_2^2 \right) \sin(\omega_j^{(i)} \mathbf{x} + \phi_j^{(i)}) \quad (9)$$

where $\theta^{(i)}$ is a group of parameters that control the shape of the Gabor kernel, including $\gamma_j^{(i)}, \mu_j^{(i)}, \omega_j^{(i)}, \phi_j^{(i)}$, and j is the index of the neurons in layer i . In this case, $\gamma_j^{(i)}$ are sampled initially from the Gamma distribution with a concentration of 1.5 and rate of 1.0, $\mu_j^{(i)}$ are initialized as random numbers from a uniform distribution on the interval $[-1, 1]$, and $\phi_j^{(i)}$ are sampled from the continuous uniform distribution from

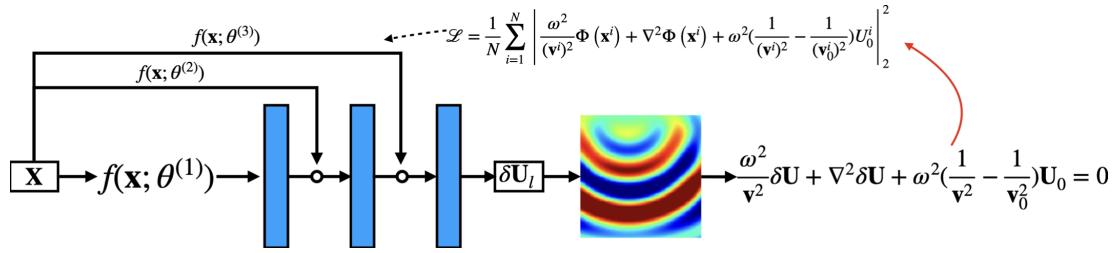


Fig. 1. GaborPINN framework for seismic wavefield simulation, where the red line points to the PDE loss calculation, the dashed line represents the backpropagation of the loss to update the NN parameters, the blue blocks are the linear projection layers, and the circles within the arrows denote the elementwise multiplication between the output of the hidden layer and the Gabor kernel.

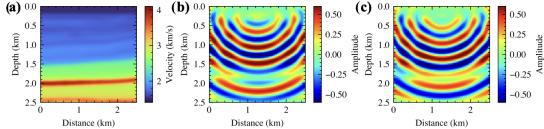


Fig. 2. (a) True velocity and (b) real and (c) imaginary parts of the 4-Hz scattered wavefield calculated numerically.

$-\pi$ to π . Thus, we refer to this implementation combined with PINN as GaborPINN, in which a diagram of it is shown in Fig. 1. For the first layer, $\mathbf{h}^{(1)} = f(\mathbf{x}; \theta^{(1)})$. Fathy et al. [18] mentioned that the MFN is generally rougher (less smooth) than the conventional MLP in representation and gradient calculation. This is because the wavenumber for the sine plane wave, $\omega_j^{(i)}$, in (9) ends up being very large. So, we propose here to connect the hyperparameter $\omega_j^{(i)}$ to the frequency of the wavefield. Specifically, we initialize $\omega_j^{(i)}$ using

$$\omega_j^{(i)} = \left\{ \sqrt{\gamma_j^{(i)}}, \sqrt{\gamma_j^{(i)}}, \sqrt{\gamma_j^{(i)}} \right\} * \omega_{\text{scale}} * \mathcal{U}\left(-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right) \quad (10)$$

where ω_{scale} is the scale factor to control the amplitude of the initialization of $\omega_j^{(i)}$, \mathcal{U} is the uniform distribution. For different input coordinates, their initializations are the same but will be updated to different values after training. We will discuss the selection of this scale factor later.

III. EXAMPLES

In this section, we will test three versions of PINNs to evaluate the convergence and accuracy improvements brought by GaborPINN. The tests are based on a simple $2.5 \times 2.5 \text{ km}^2$ layered model extracted from the Marmousi model (Fig. 2). We use 40 000 random samples from this region for training, and each sample is given by the spatial coordinates x and z for the wavefield, x_s for the location of the source near the surface, velocity v , and a constant background velocity of 1.5 km/s. Actually, the depth of the sources is fixed at 0.025 km. We train the network for a frequency-domain wavefield of 4 Hz using an Adam optimizer for 50 000 epochs. To evaluate the results, we solve the Helmholtz equation using the finite-difference method for a frequency of 4 Hz with using a fine grid spacing for accuracy to act as a reference. In the following experiments, we show the results of MLP and GaborPINN comprising three hidden layers with 256 neurons in each layer, and a larger MLP comprising three hidden layers with 512 neurons in each layer. A comparison with respect to

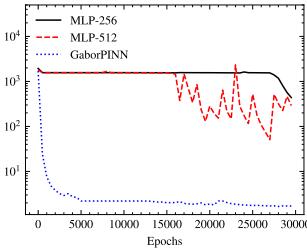


Fig. 3. Loss curves of 4-Hz test for three versions of PINNs.

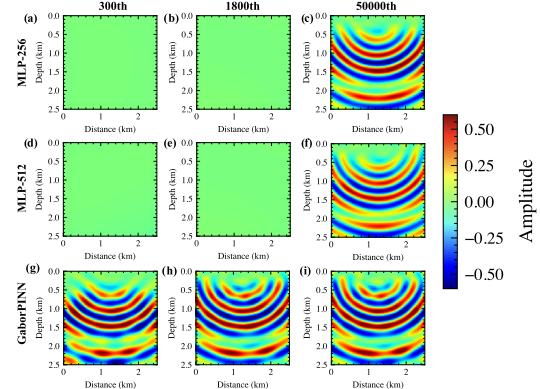


Fig. 4. Real-part predictions of the 4-Hz wavefield due to a source near the surface at 1.25 km at various epochs (a)–(i) for three versions of PINNs. (a)–(c) Results of the MLP comprising three hidden layers with 256 neurons in each layer. (d)–(f) MLP comprising three hidden layers with 512 neurons in each layer (g)–(i) Results of GaborPINN.

the network size and computational complexity is shown in Table I.

Fig. 3 shows the loss curves for these three different NNs, in which the NNs are trained by the physical loss. GaborPINN has the best convergence by far compared with the other networks. The loss function for the vanilla PINN stagnates for thousands of epochs and decreases slowly as it has no prior information related to the seismic wavefield. On the other hand, we could see that with GaborPINN, which incorporates the prior information of the wavefield into the network design, as well as injects the input coordinates at every neuron, the convergence is quite good and we could reduce the needed training epochs by two orders of magnitude.

To better understand how many training epochs are needed for GaborPINN to provide a solution, we visualize the results at the 300th, 1800th, and 50 000th epochs (Fig. 4). The GaborPINN could reconstruct the main parts of the wavefield

TABLE I
CAPACITY AND COMPLEXITY COMPARISON BETWEEN MLP AND GABORPINN

NN type	Trainable parameters	# of parameters	Computational Complexity
MLP-256	$\{\mathbf{W}^{(i)}, \mathbf{b}^{(i)}\}$	133.12 k	133.12 KMac
GaborPINN	$\{\mathbf{W}^{(i)}, \mathbf{b}^{(i)}, \gamma_j^{(i)}, \mu_j^{(i)}, \omega_j^{(i)}, \phi_j^{(i)}\}$	206.08 k	201.99 KMac
MLP-512	$\{\mathbf{W}^{(i)}, \mathbf{b}^{(i)}\}$	528.39 k	528.39 KMac

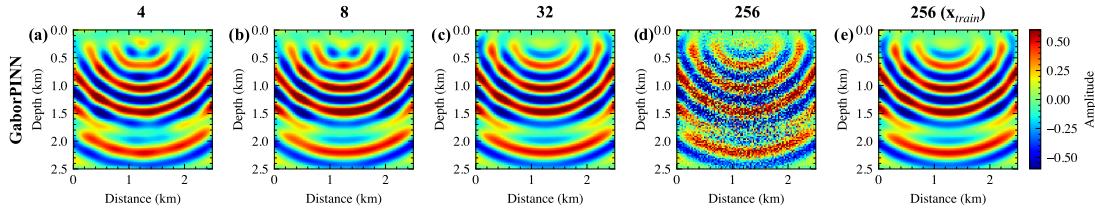


Fig. 5. Prediction results (a)–(d) of GaborPINN with different initial frequency scales applied to testing points \mathbf{x}_{test} , which are perturbed compared with the training points $\mathbf{x}_{\text{train}}$, and that of GaborNet whose scale is 256 (e) evaluated at the training points $\mathbf{x}_{\text{train}}$.

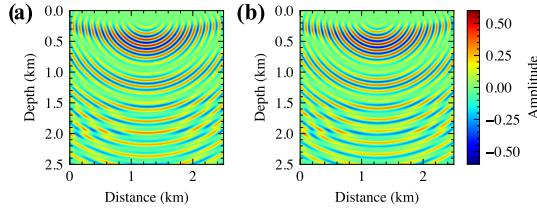


Fig. 6. (a) Real and (b) imaginary parts of the 16-Hz wavefield calculated numerically.

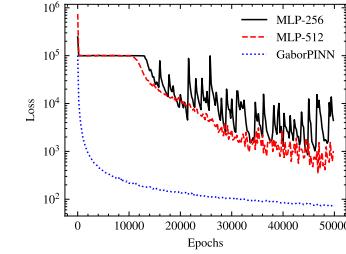


Fig. 7. Loss curves of 16-Hz test for three versions of PINNs.

within hundreds of epochs, and the details of the predictions are refined as the training progresses. However, as for the vanilla PINN, the NN learns nothing in the first 1800 epochs, which is a common problem for wavefield solutions using PINNs, as the MLP needs a lot of epochs to find the proper neuron weights to start predicting wavefields, when we start from a random initialization. Even when increasing the width of the NN in the vanilla PINN, the slow convergence persists.

The scale used here for the initialization of $\omega_j^{(i)}$ is 32, which differs from what was used in the original GaborNet. We found that a smaller value like 16 would introduce smoother, less accurate results, while a larger value would result in a failure of GaborPINN. So we test the GaborPINN for different values of initial spatial frequency (wavenumber) $\omega_j^{(i)}$ in the network under a supervised training setting on samples from a regular grid, $\mathbf{x}_{\text{train}}$ (Fig. 5), and use a grid that is slightly shifted from the training samples to show the wavefields. We found that a decrease in the value of ω_{scale} will make the representation smoother. However, for a value of 256, the predicted output wavefield is generally inaccurate. In other words, its smoothness or continuity is destroyed due to the network's focus on a high-frequency representation.

Then, we test the performance of GaborPINN in learning a 16-Hz seismic wavefield. We still use the true velocity shown in Fig. 2(a) to generate the reference results (Fig. 6) using finite difference methods. The generation of the training samples is the same as the above experiment, but we use 160 000 random samples from this region instead. We train the GaborPINN, the MLP with three hidden layers of 256 neurons in each

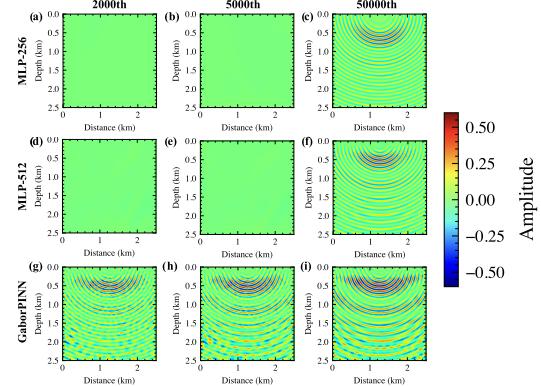


Fig. 8. Predicted real part of the 16-Hz wavefield due to a source near the surface at 1.25 km at various epochs (a)–(i) for three versions of PINN. (a)–(c) Results of the MLP comprising three hidden layers with 256 neurons in each layer. (d)–(f) MLP comprising three hidden layers with 512 neurons in each layer. (g)–(i) Results of GaborPINN.

layer, and the MLP with three hidden layers of 512 neurons in each layer, using an Adam optimizer for 50 000 epochs. We note that for the GaborPINN, the scale factor used for the initialization of the $\omega_j^{(i)}$ is 128 to fit the high-frequency nature of the wavefield. The loss curve is shown in Fig. 7. The GaborPINN still performs well for a 16-Hz wavefield, while the vanilla PINN did not converge. The predictions of the NN at different training epochs are shown in Fig. 8. The GaborPINN learns this high-frequency wavefield with limited training epochs, and the details are refined later. The vanilla PINN did not converge within 50 000 epochs. This example

demonstrates that the GaborPINN provides fast convergence and a strong capability [18] to represent wavefields. It is worth noting that the networks used for this high-frequency wavefield are the same in size as those used for the 4-Hz case, which probably hurt the conventional PINN. However, with proper frequency scaling, the GaborPINN managed to converge fast.

IV. DISCUSSION

In this letter, we proposed a modified PINN for wavefield representation using GaborNet, which we refer to as GaborPINN. With proper hyperparameter selection for GaborPINN, the convergence is fast and the prediction is good. This comes from the fact that the wavefield is inherently well-represented by a composition of Gabor basis functions. In addition, for this reason, the selection of an initial frequency scale is crucial. Unlike for image representation, which GaborNet is originally developed for, where they care about the details and the NN is trained in a supervised manner using the ground truth, PINNs with GaborNet require careful selection of the initial frequency scale. A large value will help GaborPINN fit the high-frequency details in the wavefield but will make the prediction rough, causing inaccurate second-order derivative calculations, which will harm the training of PINNs. Here, we choose the value for the scale based on the frequency of the wavefield, and as a result, we increase the initial frequency scale when we need wavefield solutions for higher frequencies. For future research, we will investigate a more intelligent or adaptive selection of the initial frequency scale.

V. CONCLUSION

We addressed the issue of the slow convergence of PINN-based wavefield solutions by incorporating Gabor basis functions. The Gabor function allowed us to incorporate the prior information on the frequency in the design of GaborPINN to accelerate the fitting of the NN to the Helmholtz equation. Unlike in image representation, we found that the proper choice of the initial frequency parameter in the Gabor function would highly affect the smoothness and continuity of the represented wavefield. Therefore, we determine this scale value based on the frequency of the wavefield. Through numerical tests, we show that the proposed approach converges much faster than vanilla PINNs. This method provides a stepping stone, from the perspective of NN design, to efficient wavefield representations for real problems using PINNs.

ACKNOWLEDGMENT

The authors would like to thank KAUST for supporting this research and Fu Wang for helpful discussions. They would also like to thank the SWAG group for the collaborative environment.

CODE AVAILABILITY

The data and accompanying codes that support the findings of this study will be available at <https://github.com/DeepWave-KAUST/PINNgabor>.

REFERENCES

- [1] K. J. Marfurt, “Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations,” *Geophysics*, vol. 49, no. 5, pp. 533–549, May 1984. [Online]. Available: <http://library.seg.org/doi/10.1190/1.1441689>
- [2] R. G. Pratt, C. Shin, and G. J. Hick, “Gauss–Newton and full Newton methods in frequency–space seismic waveform inversion,” *Geophys. J. Int.*, vol. 133, no. 2, pp. 341–362, May 1998. [Online]. Available: <https://academic.oup.com/gji/article-lookup/doi/10.1046/j.1365-246X.1998.00498.x>
- [3] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *J. Comput. Phys.*, vol. 378, pp. 686–707, Feb. 2019, doi: [10.1016/J.JCP.2018.10.045](https://doi.org/10.1016/J.JCP.2018.10.045).
- [4] Z. Li et al., “Fourier neural operator for parametric partial differential equations,” 2020, *arXiv:2010.08895*.
- [5] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, “Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators,” *Nature Mach. Intell.*, vol. 3, no. 3, pp. 218–229, Mar. 2021.
- [6] B. Li, H. Wang, S. Feng, X. Yang, and Y. Lin, “Solving seismic wave equations on variable velocity models with Fourier neural operator,” 2022, *arXiv:2209.12340*.
- [7] T. Alkhalifah, C. Song, U. B. Waheed, and Q. Hao, “Wavefield solutions from machine learned functions constrained by the Helmholtz equation,” *Artif. Intell. Geosci.*, vol. 2, pp. 11–19, Dec. 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S2666544121000241>
- [8] U. B. Waheed, “Kronecker neural networks overcome spectral bias for PINN-based wavefield computation,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.
- [9] C. Song, Y. Liu, P. Zhao, T. Zhao, J. Zou, and C. Liu, “Simulating multicomponent elastic seismic wavefield using deep learning,” *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, 2023.
- [10] C. Song, T. Alkhalifah, and U. B. Waheed, “A versatile framework to solve the Helmholtz equation using physics-informed neural networks,” *Geophys. J. Int.*, vol. 228, no. 3, pp. 1750–1762, Sep. 2021. [Online]. Available: <https://academic.oup.com/gji/article/228/3/1750/6409132>
- [11] B. Moseley, A. Markham, and T. Nissen-Meyer, “Finite basis physics-informed neural networks (FBPINNs): A scalable domain decomposition approach for solving differential equations,” 2021, *arXiv:2107.07871*.
- [12] S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, “Scientific machine learning through physics-informed neural networks: Where we are and what’s next,” *J. Sci. Comput.*, vol. 92, no. 3, Sep. 2022. [Online]. Available: <https://link.springer.com/10.1007/s10915-022-01939-z>
- [13] X. Huang and T. Alkhalifah, “PINNup: Robust neural network wavefield solutions using frequency upscaling and neuron splitting,” *J. Geophys. Res., Solid Earth*, vol. 127, no. 6, Jun. 2022, Art. no. e2021JB023703. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1029/2021JB023703>
- [14] U. B. Waheed, E. Haghhighat, T. Alkhalifah, C. Song, and Q. Hao, “PINNeik: Eikonal solution using physics-informed neural networks,” *Comput. Geosci.*, vol. 155, Oct. 2021, Art. no. 104833. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S009830042100131X>
- [15] Z. Xiang, W. Peng, X. Liu, and W. Yao, “Self-adaptive loss balanced physics-informed neural networks,” *Neurocomputing*, vol. 496, pp. 11–34, Jul. 2022. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S092523122200546X>
- [16] X. Huang and T. Alkhalifah, “Single reference frequency loss for multifrequency wavefield representation using physics-informed neural networks,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022. [Online]. Available: <https://ieeexplore.ieee.org/document/9779736/>
- [17] J. E. Womack and J. R. Cruz, “Seismic data filtering using a Gabor representation,” *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 2, pp. 467–472, Mar. 1994.
- [18] R. Fathony, A. K. Sahu, D. Willmott, and J. Z. Kolter, “Multiplicative filter networks,” in *Proc. Int. Conf. Learn. Represent.*, 2020, p. 11.
- [19] B. L. N. Kennett, “Representations of the seismic wavefield,” *Geophys. J. Int.*, vol. 118, no. 2, pp. 344–357, Aug. 1994.
- [20] P. G. Richards and K. Aki, *Quantitative Seismology: Theory and Methods*, vol. 859. New York, NY, USA: Freeman, 1980.