Surrogate Modeling of PDEs and Applications to Inverse Problems

Pelin Ersin[1], Emma Hayes[2], Peter Matthews[3], Paramjyoti Mohapatra[4]

[1] Izmir University of Economics [2] Carnegie Mellon University [3] Oxford University [4] Case Western Reserve University

Motivation

- The 2D Acoustic Wave Equation is an important equation in mechanics, modelling a plethora of phenomena

$$\frac{\delta p}{\delta t} + \kappa \cdot \Delta \mathbf{v} = f_s(\mathbf{x}, t) \qquad \frac{\delta \mathbf{v}}{\delta t} + \frac{1}{\rho}\Delta p = 0$$

$\mathbf{x} = (x, y) \in [0,1] \times [0,1]$, $t \in [0,2]$, f_s forcing term, p pressure, $\mathbf{v}$ velocity, κ viscosity, ρ density

- Discontinuous Galerkin methods are able to produce results with high accuracy, but are computationally expensive and have long run times ([3], [5])
- Neural networks are a valuable option as they are fast and have proven accurate in solving PDEs like this
- A Physics Informed Neural Network (PINN) integrates knowledge about the underlying physics of the data, providing accurate and physically consistent data even in the presence of noise and outliers [4]

Data Generation

- Our data consists of source points, time, receiver points, pressure, and x- and y-velocity
- We were given access to the codebase *WaveSims* created by AMD which implemented the Discontinuous Galerkin Method
- We were also given access to AMD's HPC systems, allowing us to generate data efficiently
- We wished to utilize all 32 threads of the CPUs to generate our data efficiently, so we implemented source-wide parallelization

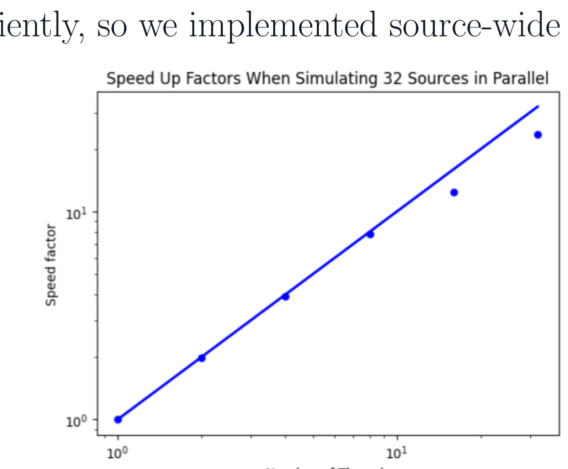


Fig. 1: Achieved 24x speedup, theoretical 32x possible

- Generated training data on 16x16, 32x32, and 64x64 receiver grids, with ground truth of 128x128 receiver grid

Data Only Neural Networks

- We trained separate networks on each of the generated grids
- Our network, $NN(\mathbf{y_s}, \mathbf{x}, t, \theta)$, takes in a source location, a receiver location, a time step, and the weights it wishes to optimize via mean squared error loss

$$\frac{1}{n}\sum_{i=1}^{n} ||\tilde{p} - p||_2^2 + ||\tilde{u} - u||_2^2 + ||\tilde{v} - v||_2^2$$

- Our network returns pressure and x- and y-velocity predictions
- Our final architecture was 5 hidden layers, 100 neurons each, 2 skip connections, *tanh* activation function
- **Interpolation:** utilized 80% of the data points
- **Extrapolation:** utilized the first 50% of the time steps

Physics Informed Neural Networks

- Incorporate the reflective boundary condition, zero initial condition, and differential operator loss into our loss function

$$L_{\text{overall}} = w_1 \cdot L_{\text{data}} + w_2 \cdot (L_{\text{DO}} + L_{\text{BC}} + L_{\text{IC}})$$
$$L_{BC} = ||\partial_x \tilde{u}||_1 + ||\partial_y \tilde{v}||_1$$
$$L_{IC} = ||\tilde{p}(x, y, 0)||_1 + ||\tilde{u}(x, y, 0)||_1 + ||\tilde{v}(x, y, 0)||_1$$
$$L_{DO} = ||\partial_t \tilde{p} + \partial_x \tilde{u} + \partial_t \tilde{v} - f_s(\boldsymbol{x}, t)||_1 + ||\partial_x \tilde{p} + \partial_t \tilde{u}||_1 + ||\partial_y \tilde{p} + \partial_t \tilde{v}||_1$$

- **Scheme 1:** Data + Physics Loss over the entire domain
- **Scheme 2:** Curriculum Learning, start training the network with small values for frequency ($\omega = 5$) and steepness ($\tau = 2$) and increment up to the desired ($\omega = 5000$, $\tau = 5$) [1]
- **Scheme 3:** Sequence to Sequence Learning, train on the time steps in an iterative process using the previous as initial conditions for the following [1]
- **Scheme 4:** Data Loss + Physics Loss over a specific region, use only data loss for the first 0.8 seconds then incorporate physics loss

Results: Data Only

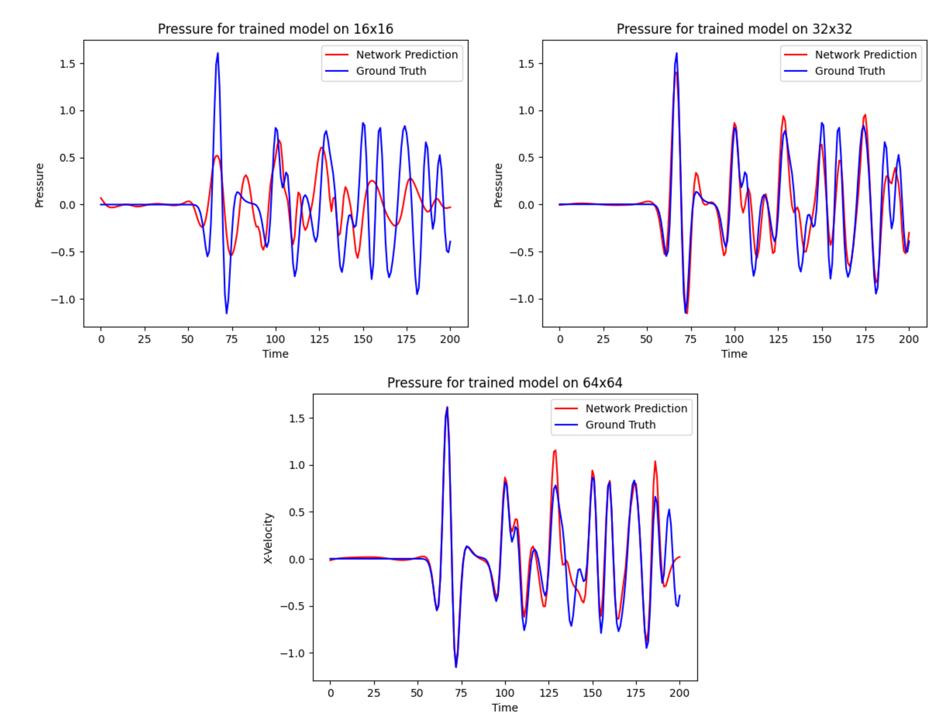


Fig. 2: Pressure Predicts for different grids

As we increase the mesh size, as we expected the predictions become more accurate, but there is a time tradeoff to consider.

	16x16	32x32	64 x 64
Execution Time	<1s	<1s	<1s
Relative Error	82.9%	27.4%	15.3%
Training Time	6 hrs	6 hrs	15hrs

Note that it takes the Galerkin Method **341 seconds** to generate the 128x128 grid, while our execution time is less than **1 second**.

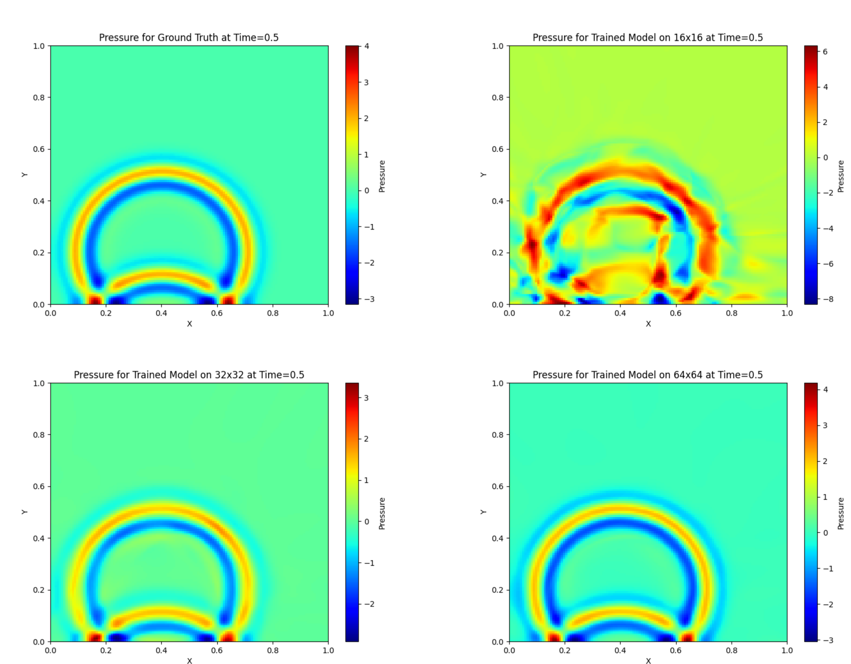
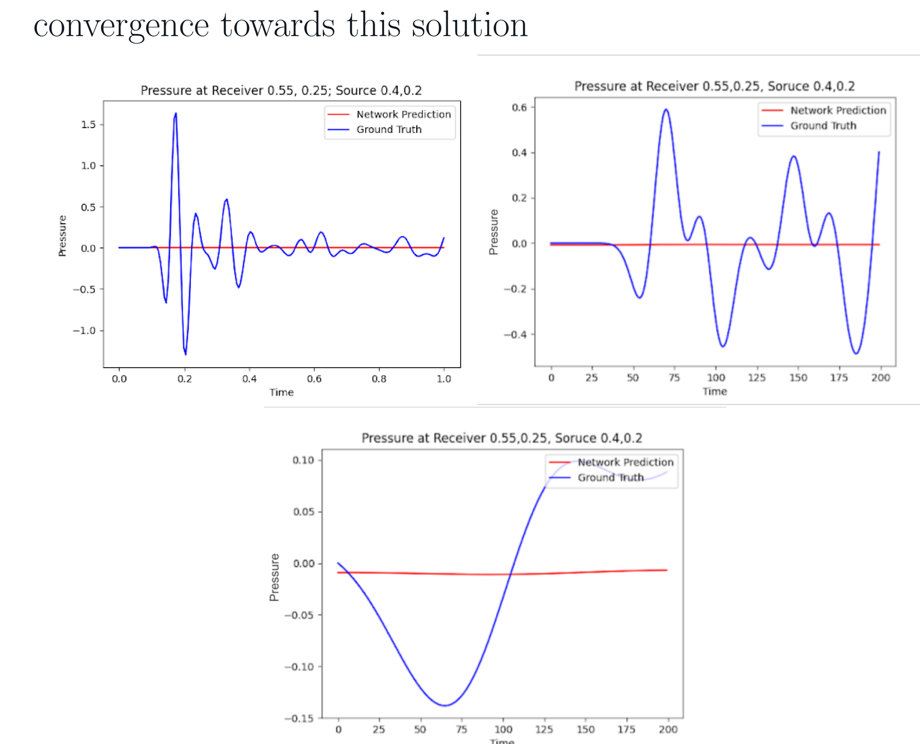


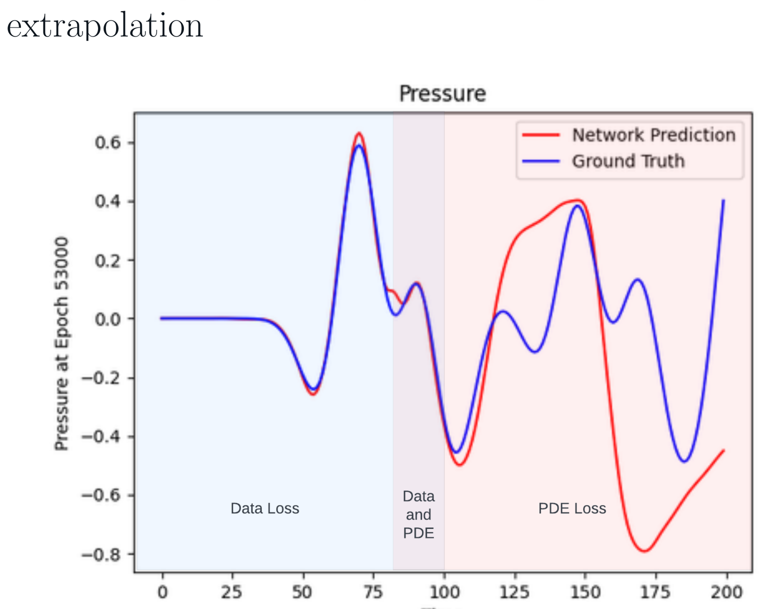
Fig. 3: Heatmap Pressure Predictions for different grids

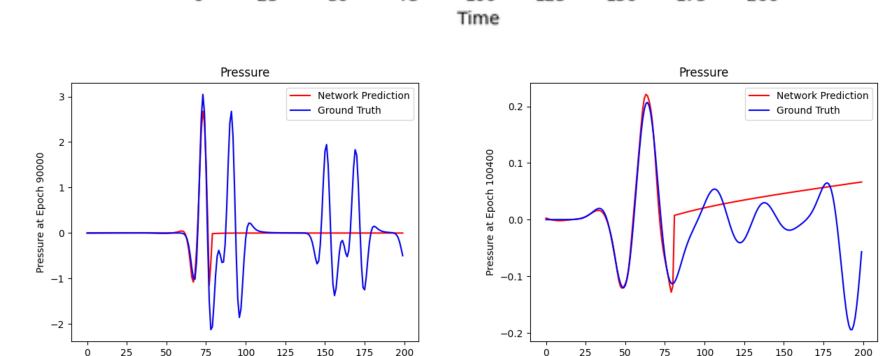
Results: Data and Physics

- **Schemes 1, 2, and 3:** The network converged to a constant zero prediction over all of space and time, indicating a strong convergence towards this solution



- **Scheme 4:** The initial data-only prediction is accurate, but the addition of physics loss gives us subpar results in the region of extrapolation





Failure Modes

- **Complex Loss Landscape:** The physics loss landscape may be too complex for our PINN to regularize using any of the existing scheme [1]
- **Forcing Term:** When our forcing term is introduced, a numerical discontinuity results, which PINNs historically struggle to capture [2]
- **Complex Parameters:** While our steepness and frequency are realistic, they may have been too complex for modelling the wave without further simplifications
- We addressed the final point through curriculum learning, and previous works have proved more successful when assuming the forcing term as an initial condition (which is less realistic)
- Addressing the loss landscape is a much more challenging task, as many regularization terms involve a differential operator which could be ill-conditioned and non-convex [1]

Inverse Problem

- If we consider the forward problem of predicting the pressure and velocity, the inverse problem is defined as inferring the source location of the forcing term, given noisy observations of pressure (or velocity) at specific receiver points
- **Approach:** Use one of our 16x16 grid data only networks as a surrogate model that can predict p and v and take advantage of its efficiency to solve the inverse problem
- We are able to sample from the posterior using a Metropolis Hastings algorithm that would not be feasible without neural network surrogates
- **Results:** These surrogates are fast enough to make MCMC sampling feasible but they are also accurate enough to identify source location with a very small mean squared error (10^{-4})
- For many sources, the true location is found at the edge of the credible estimated interval, suggesting that the posterior may be overly confident

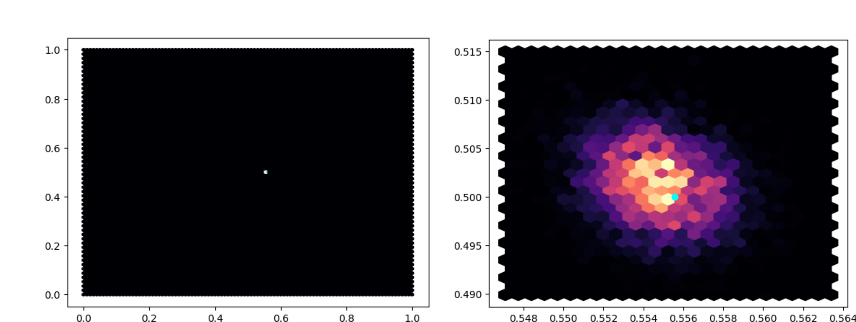


Fig. 6: $Y_s = (0.556, 0.5)$, heatmap of whole domain, then true location

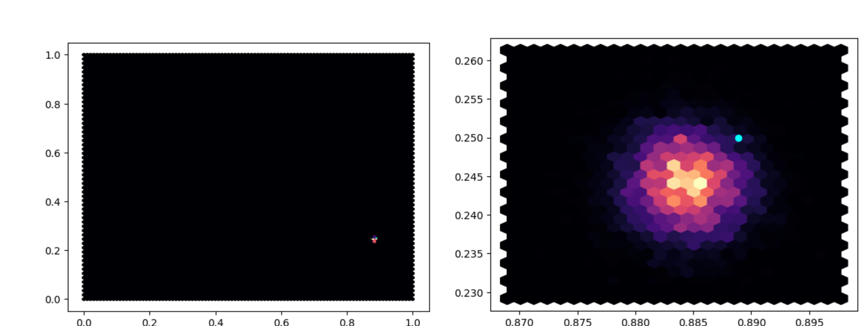


Fig. 7: $Y_s = (0.889, 0.25)$, heatmap of whole domain, then true location

Acknowledgements

This work was conducted during the RIPS program, hosted at UCLA's Institute for Pure and Applied Mathematics, and under the mentorship and guidance of Advanced Micro Devices, Inc. E.N. It is supported by the Simons Postdoctoral program at IPAM and NSF DMS 1925919 and by AFOSR MURI FA9550-21-1-0084.

References

[1] Aditi S. Krishnapriyan et al. "Characterizing possible failure modes in physics-informed neural networks". In: *Neural Information Processing Systems*. 2021. URL: https://api.semanticscholar.org/CorpusID:237386144.

[2] Li Liu. *Discontinuity Computing with Physics-Informed Neural Network*. Version 3. June 2022.

[3] Matt A. McDonald, Michael P. Lamoureux, and Gary F. Margrave. "Galerkin methods for numerical solutions of acoustic, elastic and viscoelastic wave equations". In: *CREWES Research Report* (2012).

[4] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. "Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equations". In: *Journal of Computational Physics* 378 (2019), pp. 686–707. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.10.045.

[5] F. Renac et al. "Aghora: A High-Order DG Solver for Turbulent Flow Simulations". In: *IDIHOM: Industrialization of High-Order Methods - A Top-Down Approach: Results of a Collaborative Research Project Funded by the European Union, 2010 - 2014*. Cham: Springer International Publishing, 2015, pp. 315–335. DOI: 10.1007/978-3-319-12886-3_15.