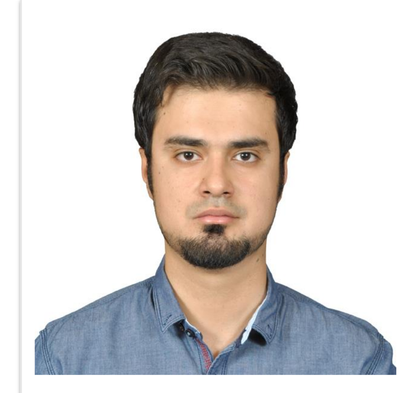
An Application of Physics Informed Recurrent Neural Networks to Structural Dynamics

Zarak Kasi[1], Barbara Simpson[2], Michael Scott[3]

[1]Oregon State University, [2]Stanford University, [1]Oregon State University



NHERI Computational Symposium 2024
February 1-2
Los Angeles, California

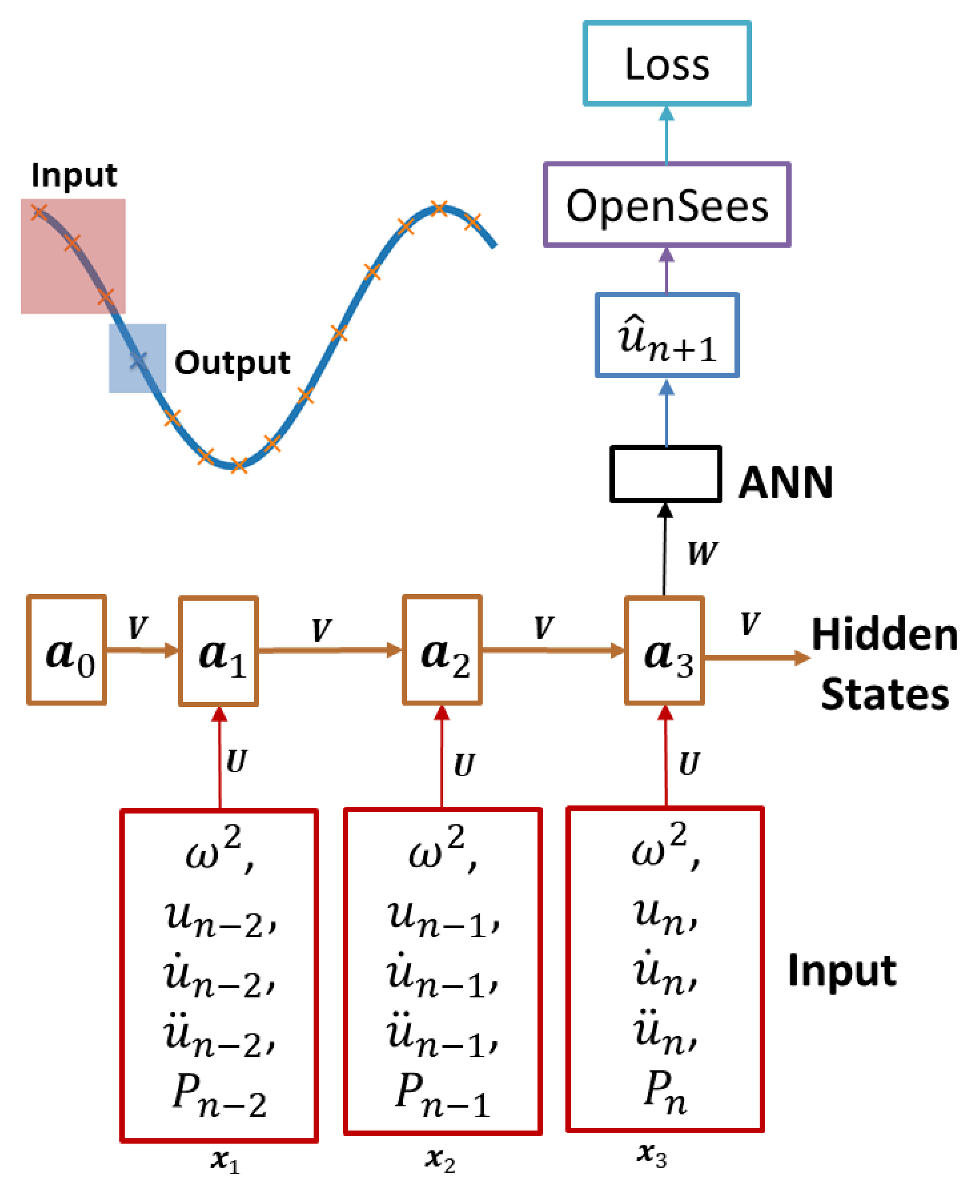
Initial Input and **Static Deformation scaling**, combined with **unitless loss functions**, greatly improve **RNN** performance in **structural dynamics.**

Introduction

- Neural Networks (NN) structural dynamics modeling can help in improved efficiency and speed.
- Physics-informed neural networks enhance DL learning by integrating physical laws.
- Our research leverages RNNs with tailored loss functions for structural dynamics response, emphasizing loss selection and data scaling importance.

Physics Informed RNN

- RNNs effectively estimate structural behavior by analyzing data over time.
- RNN weights $\{U, V, W\}$ are refined using previous input information, x to estimate future displacement.
- RNN estimations rely on input data preprocessing (data/ feature scaling) and loss normalization, impacting weight fine-tuning, especially when units are involved for uniform learning.
- Physics is embedded in the learning process through OpenSees.



Loss Function implementation

Unitless loss for unitless comparison and consistent learning

Data-driven loss (L_D)

$$L_D = \frac{1}{N}\sum_{n=1}(\hat{u}_n - u_n)^2 \qquad \widetilde{L_D} = \frac{1}{N}\sum_{n=1}\left|\frac{\hat{u}_n - u_n}{u_0}\right|$$

$$\boldsymbol{P}_u = m\ddot{\boldsymbol{u}} + c\dot{\boldsymbol{u}} + k\hat{\boldsymbol{u}}$$

Residual loss (L_R)

$$L_R = \frac{1}{N}\sqrt{(\boldsymbol{P}_u^T\boldsymbol{P}_u)} \qquad \widetilde{L_R} = \frac{1}{N}\sqrt{\left(\frac{\boldsymbol{P}_u}{\boldsymbol{k}\boldsymbol{u}_0}\right)^T\left(\frac{\boldsymbol{P}_u}{\boldsymbol{k}\boldsymbol{u}_0}\right)}$$

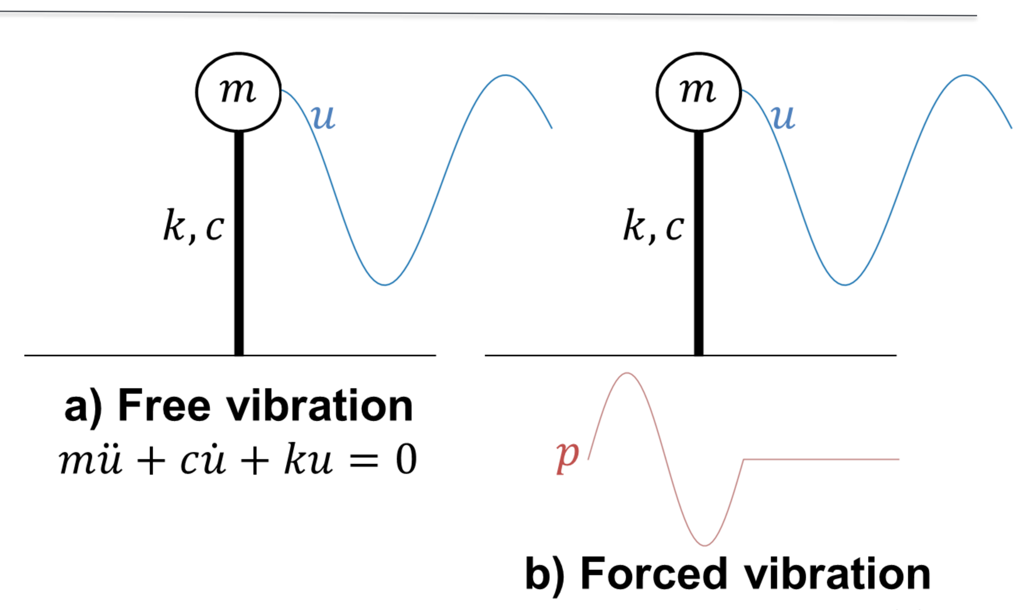
Energy loss (L_E)

$$L_E = \frac{1}{N}|(\hat{\boldsymbol{u}} - \boldsymbol{u})^T\boldsymbol{P}_u| \qquad \widetilde{L_E} = \frac{1}{N}\left|\left(\frac{\hat{\boldsymbol{u}} - \boldsymbol{u}}{\boldsymbol{u}_0}\right)^T\left(\frac{\boldsymbol{P}_u}{\boldsymbol{k}\boldsymbol{u}_0}\right)\right|$$

Step-by-Step implementation

a) RNN implementation for free-response with initial displacement

b) RNN implementation for forced vibration with trigonometric, triangle, and constant loads.



Performance Metric

Evaluated RNN model performance using the average $\overline{MAPE}$ across unique time series test files.

$$\overline{MAPE} = \frac{1}{J}\sum\{\frac{1}{N}\Sigma\left(\left|\frac{\hat{u}_n - u_n}{u_0}\right|\right)\}$$

Feature scaling

Min-max Scaling (MMS)

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} + 1$$

Initial-Input Scaling (IIS)

$$x' = \frac{x}{x_0}$$

Static Deformation Scaling (SD) (Forced Vibration and Ground motions)

$$p' = \frac{p}{p_0}, u' = \frac{u}{u_{st}}, \dot{u}' = \frac{\dot{u}}{p_0/\sqrt{km}}$$

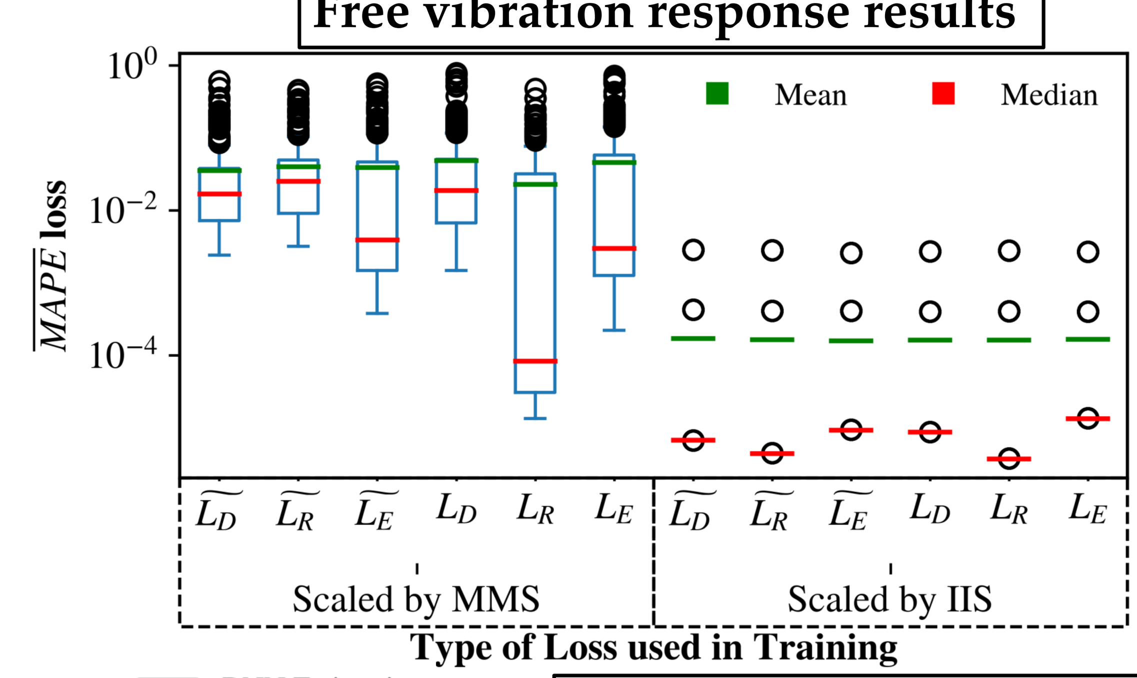
$$\ddot{u}' = \frac{\ddot{u}}{p_0/m},$$

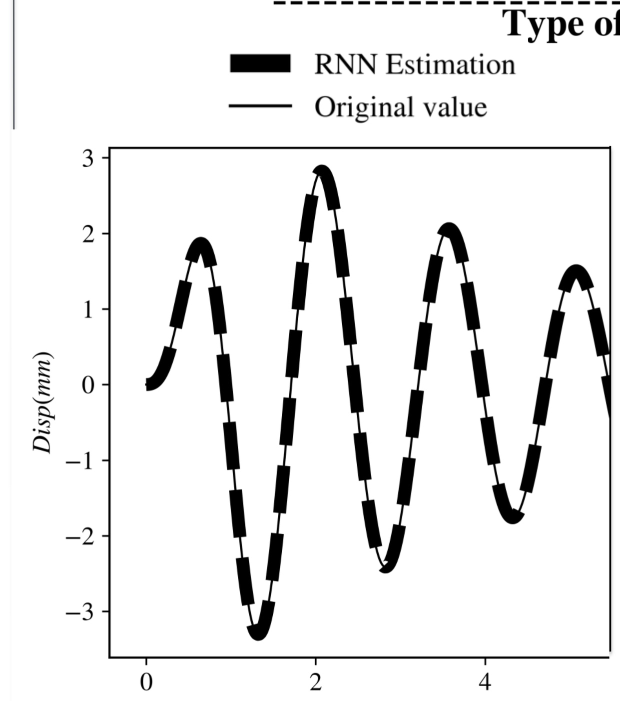
$$\omega^{2\prime} = \frac{\omega^2}{(p_0/m)/u_{st}}, u_{st} = \frac{p_0}{k}$$

Results

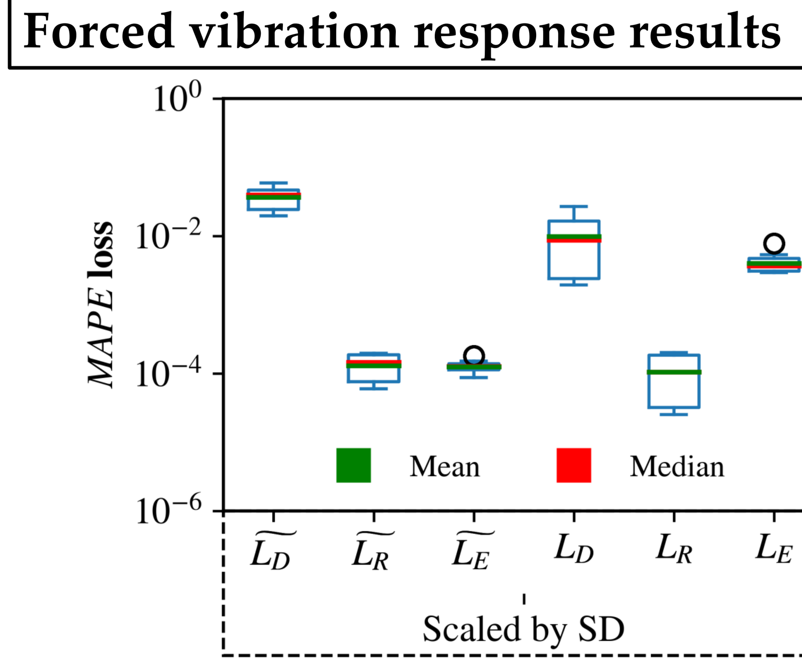
- IIS scaling has reduced variation in free-vibration response compared to MMS scaling.
- Smaller time step enhances outlier performance in IIS.
- Unitless loss functions yield comparable results for MMS and IIS scaling techniques but have less variance overall.
- SD scaling performs the best for forced vibration response.

Free vibration response results





Forced vibration response results



Conclusions

- Improved performance of RNN models with IIS or SD over MMS scaling.
- Unitless loss competes closely with unit-dependent losses, with less variability for structural dynamics applications.
- Further research is needed with more complex models such as ground motions or multi-degree of freedom systems.

References

1. Rassi. M & Perdikaris, P & Karniadakis, G. (2018). Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations. JCP
2. McKenna, F., M. H. Scott and G. L. Fenves (2010). "Nonlinear finite-element analysis software architecture using object composition." JCC 24(1): 95-107
3. Oishi, A. and G. Yagawa (2017). "Computational mechanics enhanced by deep learning." CMAME 327: 327-351
4. Peng, H., J. Yan, Y. Yu and Y. Luo (2021). Time series estimation based on deep learning for structural dynamic nonlinear prediction. Structures, Elsevier