



A Preliminary Study on Physics-Informed Machine Learning-Based Structure Health Monitoring for Beam Structures

Tran Tien Dat¹, Yasunao Matsumoto²(✉), and Ji Dang²

¹ Hanoi University of Civil Engineering, Hanoi, Vietnam
dattt@huce.edu.vn

² Saitama University, Saitama, Japan
{ymatsu,dangji}@mail.saitama-u.ac.jp

Abstract. The physics-informed neural network (PINN) was recently introduced as a potential technique to solve both forward and inverse structural dynamic problems by integrating physical differential equations and boundary conditions into the network to minimize the measurement data required for model training. However, it was only in the stage of solving the simple theoretical problems. This study applied PINN to solve an inverse problem to identify dynamic structural parameters for a prestressed concrete girder bridge built 40 years ago. The model input data were acceleration data measured by three sensors under vehicle load in two states (i.e., before and after strengthening by external prestressing cable) and combined with the partial differential equation (PDE) of beam bending and boundary conditions to minimize a loss function. The modal properties obtained from the PDE with the parameters identified from PINN were compared with those experimentally identified by Eigensystem Realization Algorithm (ERA) technique.

Keywords: Physics-informed Neural Network · Partial Differential Equation · Inversion · Eigensystem Realization Algorithm

1 Introduction

Structural Health Monitoring (SHM) is an inverse problem that involves determining the parameters of a structure based on measured data to understand its state. In recent years, machine learning has emerged as a promising tool for SHM applications [1–3]. However, traditional machine learning methods rely purely on measured data to optimize loss functions, which require a huge amount of data. Collecting enough data in different failure scenarios is challenging. Therefore, models trained on limited data tend to overfit, leading to poor generalization.

Recently, physics-informed machine learning has emerged as a promising method that combines the prior knowledge of physics with a data-driven machine learning models, which increases the generalizability and yields reliable results. Physics-Informed Neural Network (PINN) are a form of machine learning [4] that

can integrate prior domain knowledge, such as the ordinary or partial differential equation, into the training process. However, the application of PINNs in SHM is currently limited to theoretical problems [5–7]. Therefore, further studies are necessary to investigate the applicability of PINNs to both theoretical and practical issues in structural health monitoring.

The overall objective of this study was to investigate the applicability of the PINN approach in determining the parameters of structures. This objective was achieved by addressing the following tasks: 1) verifying the PINN method to determine the parameters of the partial differential equation (PDE) of theoretical beam structures using their free vibration response data; and 2) investigating the capability of the PINN approach to identify the parameters of an existing single-span girder bridge based on the damped free vibration equation and measurement data. The results were compared with those experimentally identified by Eigensystem Realization Algorithm (ERA) technique.

2 Case Study 1: Theoretical Beam Structure

2.1 Methodology

In this case study, PINN method was applied to determine the parameters of the partial differential equation of theoretical beam structure using its free vibration acceleration data. The free vibration equation of beam is given by

$$EI \frac{\partial^4 w(x, t)}{\partial x^4} + m \frac{\partial^2 w(x, t)}{\partial t^2} + c \frac{\partial w(x, t)}{\partial t} = 0 \quad (1)$$

where w , c , EI , and m are displacement, damping coefficient, stiffness, and the mass per unit length of the beam, and x and t are space-time variables of Eq. (1). In the case study, the damping coefficient c was considered as an unknown parameter, and it was identified based on the acceleration field, PDE, and boundary conditions.

To apply the method, we first created a fully connected neural network with 3 hidden layers, each with 100 nodes, and used the sinusoidal activation function for the model (see Fig. 1). The input layer has two nodes corresponding to time and space, while the output of the model has only one node corresponding to the displacement field. The total loss function used in the model is given by:

$$\mathcal{L} = \lambda_{pde} \mathcal{L}_{pde} + \lambda_{data} \mathcal{L}_{data} + \lambda_{bc1} \mathcal{L}_{bc1} + \lambda_{bc2} \mathcal{L}_{bc2} \quad (2)$$

where \mathcal{L}_{pde} and \mathcal{L}_{data} are the loss terms of PDE residual and observation data, while \mathcal{L}_{bc1} and \mathcal{L}_{bc2} are the loss terms for displacement and bending moment at the boundary. λ_{pde} , λ_{data} , λ_{bc1} and λ_{bc2} are the weights for each loss term.

To optimize the PINN model in this case study, weight values were chosen manually for each loss term, as shown in Table 1. The learning rate was set to $5e^{-4}$, and the number of epochs was set to 18000 with a batch size of 100. The value of epochs and batch size were the hyperparameters of the model.

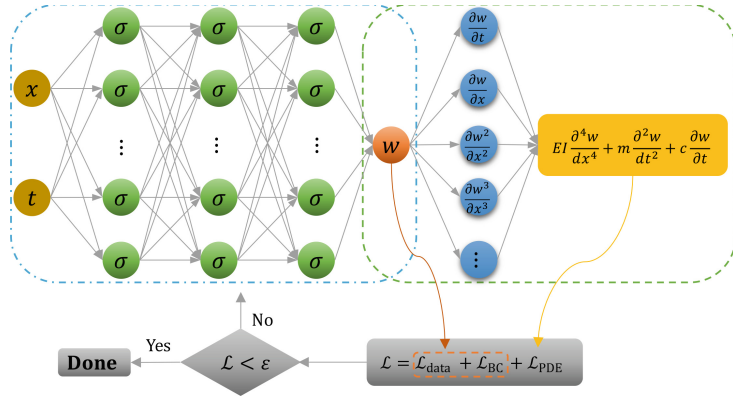


Fig. 1. Structure of PINN in case study 1

2.2 Data

Acceleration data were generated from the theoretical solution of beam structure for the first three modes. The total solution of Eq. (1) is given by

$$w(x, t) = \sum_{n=1}^3 X_n(x) \times T_n(t) \tag{3}$$

with $X_n(x) = \sin(\frac{n\pi}{L}x)$ and $T_n(t) = e^{-\xi_n\omega_n t} \cos(\omega_n t)$ with $n = 1 \rightarrow 3$.

The data from three locations shown in Fig. 2a were used for training. Figure 2b shows the acceleration data in Mode 1. Acceleration data in the first three modes were used as constraints for optimizing the PINN model, and the damping coefficient was identified for each mode.

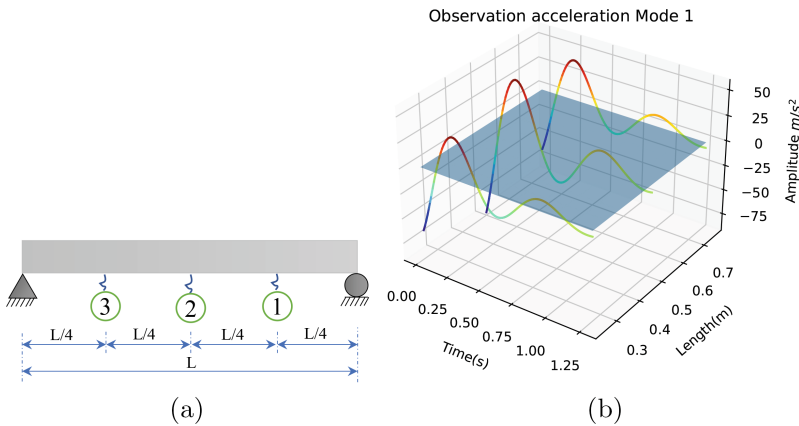


Fig. 2. a) The location of observation acceleration and b) acceleration data of Mode 1

2.3 Results

Table 1 presents the results of damping estimation for the first three modes when the acceleration is used to train the model. The table also compares the estimated damping coefficients (c) with the damping coefficients value used to generate the data. The accuracy of the estimated damping coefficient decreased from Mode 1 to Mode 3 due to the complexity of the mode shape. The weight values (λ_{data} , λ_{pde} , λ_{bc1} , λ_{bc2}) used in each component of the loss function for this case study were manually selected and significantly affected the results. In general, the results showed that the PINN model works well with limited data.

Table 1. Summary the results in first three mode shape

Mode (n)	λ_{data}	λ_{pde}	λ_{bc1}	λ_{bc2}	c_n (theory)	c_n (appr)	error %
Mode 1	1	1/50	1	250	3.95	4.03	2.00
Mode 2	1	1/50	250	250	15.79	15.08	4.49
Mode 3	1	1/50	250	250	35.53	33.56	5.55

Figure 3a shows the estimated damping coefficient at different epochs for Mode 2. The model converged around 12500 epochs, and the estimated damping coefficient (c_n approximate) was close to the theoretical value (c_n theory). Figure 3b compares the estimated mode shape with the theoretical mode shape of Mode 2.

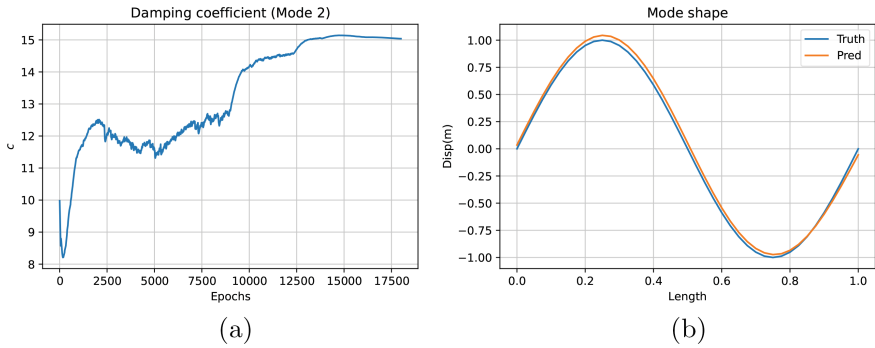


Fig. 3. a) Estimation damping coefficient at different epochs and b) comparison of mode shapes for Mode 2

3 Case Study 2: Real Bridge

3.1 Methodology

The Eq. (1) is rewritten as

$$\alpha \frac{\partial^4 w(x, t)}{\partial x^4} + \frac{\partial^2 w(x, t)}{\partial t^2} + \beta \frac{\partial w(x, t)}{\partial t} = 0 \tag{4}$$

where α represents the normalized stiffness $\alpha = \frac{EI}{m}$, and β represents normalized damping coefficient $\beta = \frac{c}{m}$, which were unknown parameters of the girder. In this case study, PINN is utilized to identify these unknown parameters. The same neural network model used in case study 1 was set up (see Fig. 4).

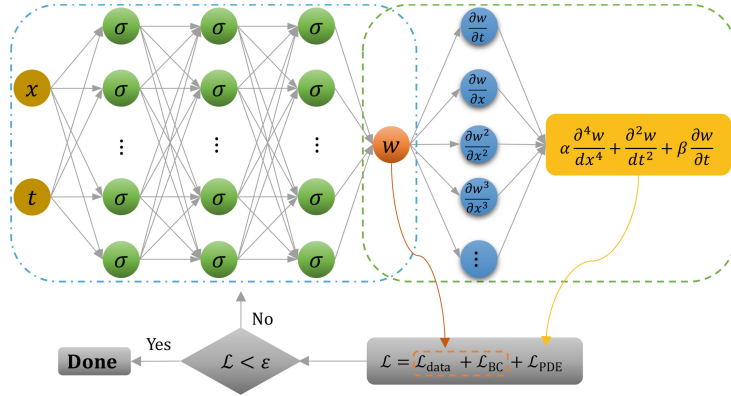


Fig. 4. Structure of PINN in the study

The time-space grid was established and subsequently utilized by the model, with the time and space coordinates serving as inputs (refer to Fig. 5). The model needs to satisfy three conditions: Eq. (4), the recorded acceleration data, and the boundary conditions. It can be described by the loss function as follows:

$$\begin{aligned}
 \mathcal{L} &= \lambda_{data} \mathcal{L}_{data} + \lambda_{pde} \mathcal{L}_{pde} + \lambda_{bc1} \mathcal{L}_{bc1} + \lambda_{bc2} \mathcal{L}_{bc2} \\
 &= \lambda_{data} \frac{1}{N_D} \sum_{x_i, t_i \in \Xi} \left| \hat{w}_{tt}(x_i, t_i) - acc(x_i, t_i) \right|^2 \\
 &+ \lambda_{pde} \frac{1}{N_G} \sum_{x_k \in \Omega; t_k \in T} \left| \alpha \hat{w}_{xxxx}(x_k, t_k) + \hat{w}_{tt}(x_k, t_k) + \beta \hat{w}_t(x_k, t_k) \right|^2 \\
 &+ \lambda_{bc1} \frac{1}{N_B} \sum_{t_i \in T} \left| \hat{w}(0, t_i) + \hat{w}(L, t_i) \right|^2 + \lambda_{bc2} \frac{1}{N_B} \sum_{t_i \in T} \left| \hat{w}_{xx}(0, t_i) + \hat{w}_{xx}(L, t_i) \right|^2
 \end{aligned} \tag{5}$$

where,

- \hat{w}_{tt} and acc are the approximation acceleration and observation accelerations. N_D is the number of observation data (red points in Fig. 5).
- N_G is the all of points in the time-space domain. α and β are unknown parameters of PDE.
- \hat{w} and \hat{w}_{xx} are the approximation of displacement and curvature at the boundary. N_B is the number of points at the boundary location (blue points in Fig. 5).

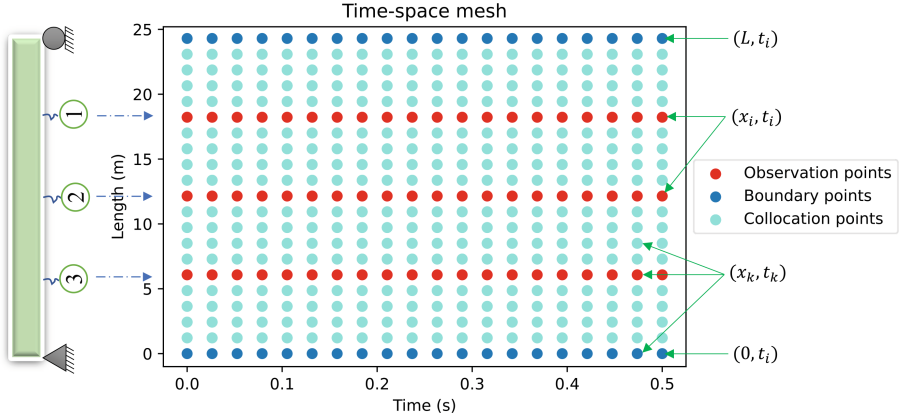


Fig. 5. Description of the points in beam mesh

The loss function includes three main components. The first component, \mathcal{L}_{data} , constrains the model’s output to satisfy the acceleration data recorded at each sensor, with a weight of $\lambda_{data} = 1$. The second component, λ_{pde} , constrains the model’s output to satisfy the PDE by minimizing the PDE’s residual. Since the number of points in the time-space domain in the grid is larger than the data points and the number of points in the boundary condition, the effect of the residual of the PDE on the loss value is the largest. To reduce the effect of \mathcal{L}_{pde} and increase the effectiveness of data and boundary conditions, the weight λ_{pde} was given as $\frac{1}{40}$. The last components, \mathcal{L}_{bc1} and \mathcal{L}_{bc2} , were used to constrain the boundary condition of the model, and the weights were set to $\lambda_{bc1} = \lambda_{bc2} = 100$.

To optimize the PINN model in this case study, a learning rate of $5e^{-4}$ was used. The number of epochs was set to 25000 and the batch size was set to 200. The values of epochs and batch size were the hyper-parameters of the model. A batch size of 200 was chosen to ensure efficient training and memory limitation.

3.2 Data

The road bridge investigated in the present study is a single-span post-tensioned PC T-girder bridge that was constructed in 1975 and crosses the Iruma River in Kawagoe, Japan. The bridge comprises of four main girders with a span length of 24.25 m and a width of 8.35 m. During a routine inspection, bending and diagonal cracks were identified in the two outer main girders (refer Fig. 6a). Subsequently, a detailed survey revealed some decreases in prestress. To improve the stress condition in the girders, it was decided to apply external prestressing to all four girders, as shown in Fig. 6b.

For monitoring the response of the structure under traffic loads, acceleration data were recorded before and after strengthening the structure with external cables under vehicle loads. Data were collected and saved every 5 min, the total measurement time was 1 h for both states.



Fig. 6. a) Bending and diagonal cracks and b) External pre-stressing arrangement in a main girder

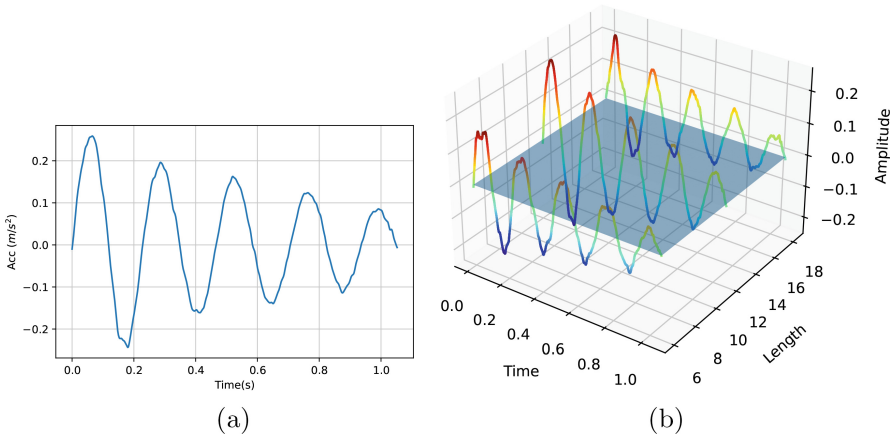


Fig. 7. a) Free vibration acceleration at the middle and b) acceleration at three sensor on the outer girder

In this study, it was assumed that the bridge behaved as a beam in free vibration, as described in Eq. (4). Therefore, the free vibration zone was manually chosen from the recorded acceleration data and used as a constraint for the PINN model. An example of the extracted free vibration from the acceleration record at the sensor located in the middle of the outer girder is shown in Fig. 7a. The extracted free vibration at three sensors in the outer girder is also shown in the Fig. 7b. Since the equation for a beam, i.e., Eq. (4), was used in this study, data from only one outer girder were used.

3.3 Results

Figure 8a and 8b show values of α and β at different epochs in one sample of measurement data before prestressing. There was no significant change in the

value α observed in both states compared with the initial value $\alpha = 2.3e^6$. However, the values of β before prestressing were significantly different from the initial value of $\beta = 1$.

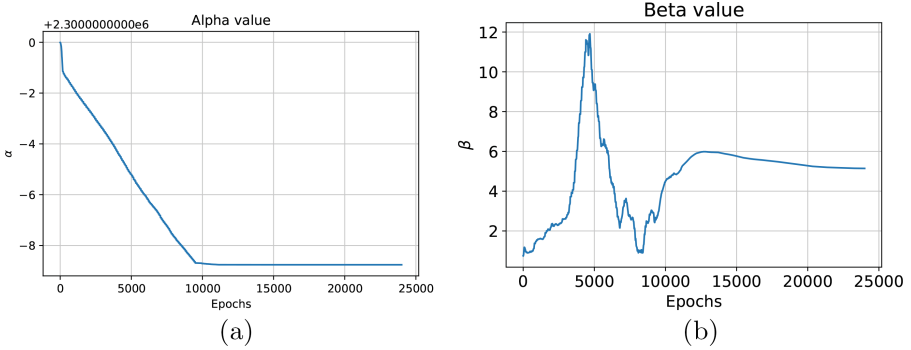


Fig. 8. a) Alpha value and b) Beta value depend on epochs

In this case study, we used four sets of free vibration acceleration data measured before and after the bridge was reinforced with prestressing cables. The free vibration data were chosen manually, and the normalized stiffness and damping coefficients for each set of data were identified and shown in Table 2.

Table 2. Summary the results of α and β

No. Sample	α		β	
	Before Prest	After Prestr	Before Prestr	After Prestr
1	2299995	2299995	0.237	1.549
2	2299995	2299991	1.65	5.143
3	2299996	2299996	7.25	3.808
4	2299993	2299983	0.0003	0.525

There is no significant change in the α value before and after the prestressing. Additionally, there is no significant change in the α value in both states when compared to the initial value of $\alpha = 2.3e^6$, which was assigned before training. With regards to β , the identified variation of beta was large and significantly different from the initial value of $\beta = 1$ in both states, except for one case.

The normalized stiffness, as found by the PINN model shown in Table 2, was substituted into Eq. (4) to theoretically calculate the natural frequencies. Due to the small effect of damping on the natural frequencies and the large variation in the identified damping, damping was neglected in the calculation of natural

frequencies. The angular frequency and natural frequency in each mode can be obtained using the equation below.

$$\omega_i = \frac{i^2 \pi^2}{L^2} \sqrt{\alpha} \implies f_i = \frac{i^2 \pi \sqrt{\alpha}}{2L^2} \quad i = 1, 2, 3 \tag{6}$$

The natural frequency of the first three modes was calculated based on Eq. (6) and shown in Table 3. The results from the PINN technique in this case study show that there are no differences between the first three natural frequencies of the beam structure before and after prestressing.

Table 3. Averages of natural frequencies (f) by using PINN technique

Mode	Before Prestressing	After Prestressing	%
	f (Hz)	f (Hz)	difference
1 st bending	4.03	4.03	0.0
2 nd bending	16.14	16.14	0.0
3 rd bending	36.31	36.31	0.0

To compare the results, the ERA technique [8] was used to identify the natural frequency in each mode. The results obtained from the ERA technique were compared with the results obtained from the PINN technique, and are shown in Table 4.

Table 4. Comparison results between PINN and ERA technique

No. Sample	PINN f(hz)		%	ERA f(hz)		%
	Before	After	difference	Before	After	difference
1 st bending	4.03	4.03	0.0	4.00	4.20	5.0
2 nd bending	16.14	16.14	0.0	14.5	14.7	1.4
3 rd bending	36.31	36.31	0.0	26.3	26.9	2.3

The natural frequencies obtained from the PINN technique were significantly different from those identified experimentally using the ERA method, as shown in Table 4. There could be several reasons for the differences in the results obtained from the PINN technique, such as: 1) the initial parameters assigned for α and β were chosen subjectively, and 2) 1D vibration equations were used to optimize the loss function. In practice, the beam structure works as a network of girders, and therefore, the results could potentially be improved by considering the effect of the stiffness of the girder network.

4 Discussion and Future Work

This study utilized an algorithm based on Physics-Informed Neural Networks (PINN) to address the inverse problem of theoretical beam bending vibration and free vibration response in an existing bridge. PINN was able to reconstruct the acceleration field of the beam structure. However, the values of (α and β) determined for the bridge did not display significant changes before and after reinforcement with prestressing cables, despite experimental evidence of changes in the natural frequency. The poor results in the real structure may have been caused by the manual selection of weights for each component in the loss function, which may not have been reasonable. Improving the results requires determining the weights automatically for each loss term to maintain a balanced interplay between the different components of the composite loss function during model training.

References

1. Sakib Ashraf Zargar and Fuh-Gwo Yuan: Impact diagnosis in stiffened structural panels using a deep learning approach. *Struct. Health Monit.* **20**(2), 681–691 (2021)
2. Yeum, C.M., Dyke, S.J.: Vision-based automated crack detection for bridge inspection. *Comput.-Aided Civil Infrastructure Eng.* **30**(10), 759–770 (2015)
3. Narazaki, Y., Hoskere, V., Hoang, T.A., Spencer, B.F.: Automated vision-based bridge component extraction using multiscale convolutional neural networks. In: *Computer Vision and Pattern Recognition* (2018)
4. Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **378**, 686–707 (2019)
5. Yuan, F.-G., Zargar, S.A., Chen, Q., Wang, S.: Machine learning for structural health monitoring: challenges and opportunities. In: Huang, H. (ed.) *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2020*, vol. 11379, pp. 1–23. International Society for Optics and Photonics, SPIE (2020)
6. Haghghat, E., Bekar, A.C., Madenci, E., Juanes, R.: Deep learning for solution and inversion of structural mechanics and vibrations. In: *Modeling and Computation in Vibration Problems*, vol. 2, 2053–2563, pp. 1-1–1-17. IOP Publishing (2021)
7. Haghghat, E., Juanes, R.: Sciann: a keras/tensorflow wrapper for scientific computations and physics-informed deep learning using artificial neural networks. *Comput. Methods Appl. Mech. Eng.* **373**, 113552 (2021)
8. Juang, J.-N., Pappa, R.S.: An eigensystem realization algorithm for modal parameter identification and model reduction. *J. Guid. Control. Dyn.* **8**(5), 620–627 (1985)