# Neural-based models ensemble for identification of the vibrating beam system

**Krzysztof Patan**[*,1] K.PATAN@ISSI.UZ.ZGORA.PL
**Maciej Patan**[†,1] M.PATAN@ISSI.UZ.ZGORA.PL
**Piotr Balik**[†,1,2] P.BALIK@STUD.UZ.ZGORA.PL
[1] *Institute of Control and Computation Engineering, University of Zielona Góra, Poland*
[2] *Doctoral School of Exact and Technical Sciences*

**Editors:** Cecília Coelho, Bernd Zimmering, M. Fernanda P. Costa, Luís L. Ferrás, Oliver Niggemann

## Abstract

The purpose of this study is to provide an effective approach to model design for transverse vibrations of the actuated cantilever beam under the regime of non-linear loads, based on available measurement data. The idea is to decompose whole system into a serial connection of a static non-linear subsystem representing electromagnetic actuation with inherent built-in magnetic hysteresis and a non-linear dynamic subsystem approximating the spatio-temporal dynamics of the vibrating beam. Then, both components can be independently modeled in terms of dedicated neural networks: feedforward network with augmented inputs providing the information on the signal gradient for the first subsystem and the multimodel neural ensemble with dedicated data fusion rule for the latter. This provides great flexibility in model design, leading to a very high accuracy of system state estimation. The decided advantage of this data-driven machine learning scheme is that incomplete knowledge of the physical model can be efficiently recovered or exchanged with the properly gathered information from input-output measurements. A physically relevant real-world application is given to illustrate the potential of the new design in the form of dynamic displacement modeling for an actuated vibrating beam system.

**Keywords:** neural modeling, spatio-temporal system, nonlinear dynamics

## 1. Introduction

Beams and cantilever beams are the main elements of structures used in different engineering applications, e.g. machine tools, civil constructions, micro-electro-mechanical systems, automotive and aerospace structures, robot manipulators (Li et al., 2022; Patan et al., 2019; Zarfam and Khaloo, 2012). The beams are subjected to static and dynamic loads, vibrations, and material fatigue. In order to guarantee safety and system performance it is necessary to analyze the beam dynamics, i.e. to obtain a spatial model of the beam.

The flexible cantilever beam is an example of a distributed-parameter system (DPS) with an infinite number of states. Its dynamics generally are represented by non-linear partial

[*] This research was funded in part by National Science Centre in Poland, grant No. 2023/48/Q/ST7/00205. For the purpose of Open Access, the author has applied a CC-BY public copyright licence to any Author Accepted Manuscript (AAM) version arising from this submission.

[†] This research was funded in part by National Science Centre in Poland, grant No. 2020/39/B/ST7/03280. For the purpose of Open Access, the author has applied a CC-BY public copyright licence to any Author Accepted Manuscript (AAM) version arising from this submission.

differential equations (PDEs). Popular procedures for obtaining spatial models of DPS are modal (Kermani, 2010) and finite-element (FE) analyses (Andreaus et al., 2007). However, these approaches require detaile physical description of sensors, actuators, and mechanical system itself. Modal analysis requires to solve descriptive PDEs. This requirement is removed in the case of FE analysis, however, information about material properties as well as boundary conditions is still needed (Fleming and Moheimani, 2003). Nowadays, a new paradigm in modeling real-world engineering systems is data-driven modeling (DDM). Prospect of not being restrained by highly complex models or an incomplete understanding of the physics of systems creates a very attractive alternative for practitioners, which leads to the rapid development of data-driven methods, especially neural networks. These can be used effectively in many engineering areas to design the input-output relations of industrial systems and plants using historical observations (Haykin, 2009). Given sufficient data, it is possible to discover previously unknown physics or/and to cope with high-dimensional problems. DDM can provide faster or computationally cheaper simulations for parameter or state estimation. However, some challenges remain. In the work of Raissi et al. (2019) proposed physics-informed neural networks (PINNs) that combine physical knowledge and data-driven models may not always achieve satisfactory accuracy. Moreover, PINN still requires a physical description of the system. A somewhat different application of neural networks was proposed by Aquilar and co-workers (Aguilar-Leal et al., 2016). Imitating the meshing procedure according to the idea of the FE method, finite-element method differential neural network. Although the developed model of the spatial structure may be very accurate, network inherits the large scale of the initial spatial mesh. Another alternative, but with state-space neural networks, was used in Patan and Patan (2022) giving the possibility to dramatically reduce size of the mesh while maintaining modeling accuracy. In this case, DPS is represented by several lumped-parameter neural models.

This research provides a full DDM for systems belonging to the specific class of non-linear DPSs with particular application to actuated vibrating beam. The idea is to treat the physics of the plant, actuation, and measurement transducers as components of the whole system, which can be efficiently reconstructed from informative measurement data. The whole system is decomposed into a serial connection of a static non-linear subsystem representing electromagnetic hysteresis-type actuation, a dynamic subsystem approximating the non-linear dynamics of the vibrating beam, and noisy measurements. Then, all the components can be independently modeled and numerically treated in terms of dedicated neural networks. Especially, a dedicated neural model ensemble reconstructing highly complex plant dynamics provides great flexibility of model design, up to high accuracy of state estimation.

Classical way to deal with magnetic material hysteresis is to use the Preisach model (Adly and Abd-El-Hafiz, 1998). However, often the question is as to what experimental data should be used for identification to provide the best hysteresis approximation possible. Neural network modeling can answer this question. In the work Adly and Abd-El-Hafiz (1998), a feedforward network was used to represent the hysteresis of a magnetic tape. Proposed approach still requires elementary hysteresis operators to excite the network. Another approach was suggested in Makaveev et al. (2001), with feedforward neural network representing relation between magnetic field and magnetic induction. However, the method requires using both extreme magnetic field and induction values, as well as a parameter distinguishing between the virgin curve and hysteresis branch. In this work we apply a very simple approach to represent hysteresis observed in the electromagnetic actuators using same feedforward structure, with two inputs only: the voltage on the electromagnet coil and its changes in subsequent time instances.

## 2. Vibrating beam laboratory stand

A vibrating beam system (shown in Figure 1) is a laboratory stand invented by Inteco Sp. z.o.o (`http://www.inteco.com.pl`) to experimentally examine the displacements and vibrations observed in mechanical constructions and large structures subjected to forces and moments. Stand renders it possible to learn about measurements of vibrations, as well as to derive controls for suppressing them. System consists of a thin flat ferromagnetic elastic beam and four electromagnetic actuators. To measure the deformation of the beam, laser distance sensor is applied. Actuators can be set manually on the dedicated rail. The distance sensor is mounted on a moving cart on its rail, driven by a servomotor. The system is controlled via RT-DAC PCI I/O board, connected to PC computer and cooperates with Matlab/Simulink software. Signals available for measurements are the voltage at the electromagnet coil [V], the current in the electromagnet coils [A], and the displacement of the vibrating beam [mm]. The control signals are the excitation of the electromagnets [V] and the position of the cart [mm]. The beam is clamped at one end and free at another. The vibrating beam is an example of a distributed-parameter system, whose transverse displacement dynamics $d(x,t)$ can be mathematically represented by the Euler-Lagrange temporal biharmonic equation:

$$\rho \frac{\partial^2 d(x,t)}{\partial t^2} + \kappa \nabla^4 d(x,t) = p(x,t), \tag{1}$$

$x$ is spatial point at the time instant $t$, $\rho$ stands for mass density per unit area and $\kappa$ denotes the elasticity coefficient. A detailed description of the considered class of DPS can be found in Polyanin (2002); Patan et al. (2019).

External excitation acting on the beam is represented by $p(x,t)$. This excitation is produced by a set of $N_{act}$ electromagnetic actuators. Since the magnetic field propagates non-linearly in space and additionally the material of the beam becomes magnetized, the specific form of $p(x,t)$ is complex and difficult to express in terms of the physical effects involved. In addition, each electromagnetic actuator can be described in the form of a non-linear relation $f(\cdot)$ between control signal $u(t)$ and current flowing in the magnet coil $i(t)$. The current flowing in the actuator generates a magnetic field that acts on the beam via $g(\cdot)$, a non-linear mapping, as follows:

$$i_j(t) = f(u_j(t)) \quad \text{for } j = 1, \ldots, N_{act} \tag{2}$$

$$p(x,t) = \sum_{j=1}^{N_{act}} g(x, i_j(t)) \tag{3}$$

The closed-form of the non-linear function would be hard to express in terms of physics and construction of the actuator device, but due to its magnetic properties, $f(\cdot)$ should represent a hysteresis curve.

The modeling objective is to design two models representing the static non-linearity Equation (2) and dynamic non-linearity Equation (1) simultaneously coping with noisy observations of system state.

### 2.1. Data acquisition

To design a high-quality model, system input data should be permanently exciting, ensuring coverage of wide range of operating conditions. Three types of excitation were proposed:

- a chirp signal with frequency range 0.05 Hz to 1 Hz, decreasing in amplitude. Two scenarios are considered, (a) decrease from 2.15 to 1 and (b) and from 3.6 to 1.85,
- ramp signals with different slopes and the amplitude from the range $[-2.25, 2.25]$,
- random steps with the amplitude selected from the range $[-2.25, 2.25]$.

The chirp signal as well as random steps are frequently used to permanently excite the system (Norgaard et al., 2000; Soderstorm and Stoica, 1989). In turn, in addition to the chirp signal, the ramp signal is used here to represent the follow-up properties of the system.

It should be emphasized that the input data were appropriately selected in such a way as to extract the deformation characteristics of the beam. The frequency of the input signal should not be too high as it can produce vibrations in the beam. Moreover, due to the characteristics of the electromagnetic actuator, one can observe that for signals with larger frequencies and relatively high amplitude, the beam can 'stick' to the actuator, producing a displacement equal to zero. Such situations should be avoided. A selected portion of the data is illustrated in Figure 2(*a*).

## 3. Data-driven modeling

The whole system can be viewed as a static non-linear subsystem representing an electromagnetic actuator and a non-linear dynamic subsystem describing the beam itself connected in series. Taking into account the system specificity, two neural network models are applied here: a feedforward neural network and a recurrent one.

### 3.1. Non-linear static subsystem

If we want to use a data-driven approach to develop a model as presented in Equation (2) only available measurement data can be used. In general, the non-linear relation between $u(k)$ and $i(k)$ is in the form of a hysteresis. In this work, we propose to use a static feedforward neural network to find the relation Equation (2). In order to properly reconstruct the hysteresis we need to take into account the change of the input in the subsequent time instances, i.e., if the voltage is increasing or decreasing. Eventually, network input space consists of two components: the control signal $u(t)$ and the control change $\Delta u(k) = u(k) - u(k-1)$, and the neural network model can be represented as follows:

$$\hat{i}(k) = \boldsymbol{V}^{M}\sigma(\boldsymbol{V}^{M-1}\sigma(\ldots\sigma(\boldsymbol{V}^{1}\bar{u}(t)))), \tag{4}$$

where $\bar{u}(k) = [u(k)\,\Delta u(k)]^{\mathsf{T}}$, $\boldsymbol{V}^{j}, j = 1, \ldots, M$ are weight matrices of the appropriate size, $M$ is the number of hidden layers, and $\sigma(\cdot)$ represents the non-linear activation function.

The model Equation (4) is shown in Figure 3($a$). It is a first-order lag external memory neural network model (Norgaard et al., 2000; Patan, 2019).

### 3.2. Non-linear dynamic subsystem

For the sake of simplicity of the considerations presented here, it is assumed that the vibrating beam is excited by one actuator only and the beam displacement is measured point-wise. Then, the model has one input (the estimation of an actuator current) and one output (the beam displacement). Such an assumption renders it possible to model the system using a lumped-parameter model. Taking into account highly non-linear dynamic properties of the vibrating beam, a recurrent neural network was used in this context:

$$\begin{aligned} \boldsymbol{x}(k) &= \sigma(\boldsymbol{W}^x \boldsymbol{x}(k-1) + \boldsymbol{W}^u \hat{i}(k) + \boldsymbol{B}^x), \\ \hat{y}(k) &= \boldsymbol{W}^y \boldsymbol{x}(k) + \boldsymbol{B}^y, \end{aligned} \tag{5}$$

where $\boldsymbol{x}(k) \in \mathbb{R}^{v\times 1}$, $\hat{i}(k) \in \mathbb{R}^1$, and $\hat{y}(k) \in \mathbb{R}^1$ are the state vector, estimated input and the estimated displacement, respectively, $\boldsymbol{W}^x \in \mathbb{R}^{v\times v}$ is state weight matrix, $\boldsymbol{W}^u \in \mathbb{R}^{v\times 1}$ stands for the input weight matrix, $\boldsymbol{W}^y \in \mathbb{R}^{1\times v}$ represents the output weight matrix, $\boldsymbol{B}^x \in \mathbb{R}^{v\times 1}$ and $\boldsymbol{B}^y \in \mathbb{R}^1$ are bias weights of the state and output layers, respectively, and $v$ is the number of hidden neurons (the model order). Finally, $\sigma(\cdot)$ is the non-linear activation function. Unfortunately, the series of preliminary experiments conducted on system dynamics demonstrated a varying behavior under different excitations.

From these impediments directly stems a motivation to search for a more sophisticated and flexible solution. As a possible remedy, we applied the concept of multimodel design, where the purpose of each model was to accurately catch the dynamics of the vibrating beam for a specific behavior represented by a particular subset of measurement data. The idea follows the so-called ensemble learning methods, where from among a significant diversity of partial models it is possible to build an aggregate model of high quality covering the whole dynamics of the system. In this spirit, the entire available data is split into several subsets taking into account a frequency range, type of excitation, or even input signal amplitude. If we assume data division into $N$ subsets, our goal is to construct $N$ models of the form Equation (5). To this end, the idea is quite common in engineering practice (Haykin, 2009).

However, the main problem when dealing with many models is to propose a method for the model output augmentation. The common approach is to generate the final model output as a mean value of outputs of the candidate models (Haykin, 2009; Patan and Patan, 2023). Unfortunately, such a data fusion rule is problematic in cases when candidate models have poor generalization abilities and perform poorly for data unseen before.

In this paper, we propose a novel augmentation method where all candidate models are evaluated first using historical data. Based on evaluation, a proper weight is assigned to each model, pointing out the contribution of that candidate to the final response of ensemble. The whole idea is shown in Figure 3($b$). All trained candidates are run in parallel.

Dealing with dynamic models, the evaluation should be done over a specific time interval. Then, it is proposed to transfer most recent output of each candidate $y_j(k)$ to the memory buffer and store it there for a specific number of time steps represented by $b_l$ (buffer length). At each time instant $k$, candidate models are evaluated but using data stored in the memory buffer assigned to them, applying the sum of squared errors as follows:

$$J_j(k) = \sum_{i=k-b_l}^{k-1} (y_d(i) - y_j(i))^2, \tag{6}$$

where $j$ is the candidate number, $y_d(i)$ represents desired outputs at the time instant $i$. It is important to notice that the past output data are easily available. The candidate model with the lowest index value $J_j(k)$ performed the best for the last of $b_l$ time steps.

The evaluation results are passed to the softmax function to normalize the quality of the candidates to a probability distribution over $N$ possible models as:

$$w_i(k) = \frac{e^{-J_i}}{\sum_{l=1}^{N} e^{-J_l}}, \qquad \text{for } i = 1, \ldots, N, \tag{7}$$

where $\boldsymbol{w}(k) = [w_1(k), w_2(k), \ldots, w_N(k)]$. The best performing model, represented by the lower value of the error function, should have the dominant role in the probability distribution of weights $\boldsymbol{w}(k)$. Therefore, as the input arguments to the softmax function, we used the inverse of the performance indexes from Equation (6) (negative argument in exponential functions in Equation (7)). Every single component of the vector $\boldsymbol{w}(k)$ constitutes the contribution of a suitable candidate model to the finally formed model output, which is derived as follows:

$$\hat{y}(k) = \boldsymbol{w}(k)\boldsymbol{y}(k)^{\mathsf{T}} = \sum_{i=1}^{N} w_i(k) y_i(k), \tag{8}$$

where $\boldsymbol{y}(k) = [y_1(k), y_2(k), \ldots, y_N(k)]$. The accuracy of the estimate $\hat{y}(k)$ depends on the buffer length. If $b_l$ is too small, performance evaluation is not accurate and one can observe chaotic switching between candidate models when deriving the output estimate. In turn, if $b_l$ is too large, the output estimator is less sensitive to the changes in operating conditions, acting similarly to moment of inertia. Thus, the selection of $b_l$ is of crucial importance here.

## 4. Experiments

### 4.1. System settings and data preprocessing

During experiments, only one actuator was used, located at the position of 40cm. A point-wise measurement was considered. Laser sensor was set at a position equal to 55cm. All signals were acquired using the RT-DAC4/PCI digital timing I/O board. The sample time was set to 0.1 second. To remove quantization noise, a simple moving-average filter of order 5, was applied to smooth the data.

Analyzing the frequency spectrum of the system output presented in Figure 2($b$), it is clear that the behavior of the system is highly non-linear. For the excitation including components with frequencies in the range from 0.01 Hz to 1 Hz, the system generates an output consisting of components with frequencies in a much broader range than input signal. Clearly, we can see vibrations induced around approximately 2.5 Hz, which are highly undesirable. As was mentioned in Section 2 the objective of this work is to catch accurately the deflection properties of the system, not the vibration ones. Then, to exclude these undesired effects, output data was filtered. To carry out this task, a low-pass IIR Butterworth filter was applied, with cut-off frequency set to 1Hz and the stop-band attenuation being 60dB. Such a designed filter was used to preprocess the beam displacement measurements.

### 4.2. Hysteresis modeling

To model the actuator, a feedforward neural network Equation (4) was used. Because the second network input is a difference between two subsequent control signal values, the training data should be treated as the time sequence. Recorded data covered four working scenarios, each representing the response of the system for a specific input signal, as described in Section 2.1. Thus, the total number of samples was equal to 287986. Data were divided into training, testing, and validation sets according to the ratio 0.4 : 0.4 : 0.2, respectively. Hyperparameters of the neural model were found through a series of experiments. We have started with a simple network consisting of two hidden layers. We increased both the number of layers and neurons to achieve the best-performing model, taking into account the mean squared error (MSE) calculated over the testing set. The specifications of the selected models investigated are listed in Table 1. The notation x–y–z means that the model was constructed using 3 hidden layers with $x$, $y$, and $z$ neurons in consecutive layers, respectively. The best performance was achieved in the neural network with 4 hidden layers (in Table 1 marked with the boldface) with MSE equal to $5.39 \cdot 10^{-4}$.

To evaluate the modeling quality, static characteristics are derived by allocating points $(u(k), i(k))$ in Cartesian coordinate system. Figure 4($a$) shows hysteresis of the actuator fed by a collection of ramp signals, where output of the model is marked with red–crosses and measurements are marked with blue–circles. To emphasize the non-linear form of this characteristic, projection giving a similar effect to the "fisheye" transformation was applied:

$$u_t(k) = \arctan(50u(k)), \quad i_t(k) = \arctan(50i(k)), \tag{9}$$

where $u_t(k)$ and $i_t(k)$ are input and output after transformation, respectively. Clearly, the model reconstructs the real static characteristic of the actuator reasonably well, however, the model response is characterized by lower variance contrary to measurements. This is caused by measurement noise, which the model filters. Figure 4($b$) shows hysteresis plotted employing a neural model fed with the chirp signal. One can see several hysteresis loops representing different levels of coercive forces resulting from the complex physics of actuators, drift due to heating, and magnetization of the beam material in close vicinity of the actuator.

### 4.3. Beam modeling

The vibrating beam, was modeled through the lumped-parameter system Equation (5). To catch the dynamics of the distributed-parameter system Equation (1), a high-order lumped model should be used. Firstly, a single neural model was tried. The training set was formed in such a way as to include data representing the system response for any excitation used (see Section 2.1). Results achieved for a model consisting of 12 hidden neurons with hyperbolic tangent activation functions, shown in Figure 5($a$) turned out to be very unsatisfactory. A model works well only for slowly varying signals or constant ones. In this case, the performance index in the form of MSE was equal to 0.1454. Also, more complex models were tested without a significant breakthrough. Then, the ensemble of models proposed in

Section 3.2 was investigated. All candidate models had the same inherent structure: $v = 20$ hidden neurons, tanh activation function and one linear output neuron. Such a kind of multimodel is called a homogenous ensemble. It means that the model architecture is the same, but the training data is different. In this study, we used 8 models. Details about the training data selected for a particular model are provided in Table 2. Each candidate model was trained for 1000 steps by the Levenberg-Marquardt algorithm using different training data but tested on all available datasets. The initial weights were set to be random values. The model quality expressed by MSE is shown in the last column of Table 2.

Obviously, the generalization abilities of candidate models are not so good due to the relatively high value of MSE. However, using candidate models in the framework of the model ensemble as proposed in Section 3.2 gives MSE equal to $5.22 \cdot 10^{-4}$. Then, a fusion of candidate models renders it possible to achieve good performance of the model with pretty good generalization properties. In the case considered, we used a memory buffer $b_l = 10$. This value was selected as a compromise between accuracy and sensitivity, as discussed in Section 3.2. The quality of the one-step ahead output prediction of the proposed ensemble is illustrated in Figure $5(b)$ where the measured beam displacement is marked with the red–solid line and the ensemble output with the blue–dashed line. The model follows the required displacement closely, giving a strong argument for the correct modeling of the beam. Moreover, the model response is not biased much, as the mean value of the model output is equal to 0.0901 and mean of the displacement measurements is 0.0882. This is a clear confirmation that the model generalizes data well.

To make a definite assessment of the proposed multimodel approach, it was compared with simpler method. The widely used approach in this context is to derive the ensemble response as the average of the responses of all candidate models. The quality of multimodel expressed by MSE was equal to 0.0803. Clearly, averaging responses of models listed in Table 2 improves the modeling over single models, but the results are still much worse than the result provided by the multimodel proposed in this work. Moreover, the averaged model is firmly biased as its response mean is 0.1509 which is clearly shown in Figure $5(c)$. Summarizing, data fusion by means of averaging is obviously not acceptable here.

## 5. Concluding remarks

In this paper a novel approach for data-driven modeling for a particular class of distributed-parameter systems was proposed, with special attention to the application of the vibrating beam system. Due to the non-linear nature of the considered laboratory stand, recurrent neural networks were used to represent the dynamics of the system. However, it was observed that a single neural network, which is a lumped-parameter system, was not able to represent spatio-temporal characteristics of the plant. As a remedy, the multimodel approach was suggested with a novel mechanism to aggregate sub-model responses. Using the recent output data, the method provides weights representing the contribution of each sub-model to the final response of the model ensemble. The proposed ensemble has a homogenous structure, but its parts were trained on different training sets. Thus, the ensemble is characterized by pretty well generalization properties which were confirmed empirically. The ultimate advantage of the proposed scheme is that it is easy to upgrade by incorporating models trained using recently recorded data or developed using different actuators-sensors settings. The important part of the modeling process is to properly represent electromagnetic actuators described by a highly non-linear static characteristic. The proposed model in the form of a feedforward multi-layer network with properly prepared input space is able to achieve pretty high approximation accuracy. At the same time, the structure of the model is quite simple and easy to train, contrary to alternative models reported in the literature. However, some effort should be paid to the proper, automatic selection of the model hyperparameters as well as dividing the data into training and testing sets in order to provide high-quality models.

## References

A.A. Adly and S.K. Abd-El-Hafiz. Using neural networks in the identification of preisach-type hysteresis models. *IEEE Transactions on Magnetics*, 34(3):629–635, 1998. doi: 10.1109/20.668057.

O. Aguilar-Leal, R.Q. Fuentes-Aguilar, I. Chairez, A. García-González, and J.C. Huegel. Distributed parameter system identification using finite element differential neural networks. *Applied Soft Computing*, 43:633–642, 2016. ISSN 1568-4946. doi: https://doi.org/10.1016/j.asoc.2016.01.004.

Ugo Andreaus, Paolo Casini, and Fabrizio Vestroni. Non-linear dynamics of a cracked cantilever beam under harmonic excitation. *International Journal of Non-Linear Mechanics*, 42(3):566–575, 2007. doi: https://doi.org/10.1016/j.ijnonlinmec.2006.08.007.

A.J. Fleming and S.O.R. Moheimani. Spatial system identification of a simply supported beam and a trapezoidal cantilever plate. *IEEE Transactions on Control Systems Technology*, 11(5):726–736, 2003. doi: 10.1109/TCST.2003.816415.

S. Haykin. *Neural Networks and Learning Machines, 3rd edition*. Prentice-Hall, New Jersey, 2009.

Mehrdad R. Kermani. Analytic modal analyses of a partially strengthened timoshenko beam. *IEEE Transactions on Control Systems Technology*, 18(4):850–858, 2010. doi: 10.1109/TCST.2009.2030175.

Chen Li, Jijun Xiong, and Debo Wang. A novel capacitive microwave power sensor based on double mems cantilever beams. *IEEE Sensors Journal*, 22(12):11803–11809, 2022. doi: 10.1109/JSEN.2022.3174556.

Dimitre Makaveev, Luc Dupré, Marc De Wulf, and Jan Melkebeek. Modeling of quasistatic magnetic hysteresis with feed-forward neural networks. *Journal of Applied Physics*, 89 (11):6737–6739, 2001. doi: 10.1063/1.1361268.

M. Norgaard, O. Ravn, N.K. Poulsen, and L.K. Hansen. *Neural Networks for Modelling and Control of Dynamic Systems*. Springer, London, 2000.

Krzysztof Patan. *Robust and Fault-Tolerant Control. Neural Network-Based Solutions*. Springer Nature, Cham, Switzerland, 2019.

Krzysztof Patan and Maciej Patan. Neural-network-based models ensemble for identification in distributed-parameter systems with application to elastic materials modeling. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 01–08, 2022. doi: 10.1109/IJCNN55064.2022.9892489.

Krzysztof Patan and Maciej Patan. Fault-tolerant design of non-linear iterative learning control using neural networks. *Engineering Applications of Artificial Intelligence*, 124: 106501, 2023. doi: https://doi.org/10.1016/j.engappai.2023.106501.

Maciej Patan, Kamil Klimkowicz, Robert Maniarski, Krzysztof Patan, and Eric Rogers. Iterative learning control of the displacements of a cantilever beam. In *58th IEEE Conference on Decision and Control, CDC 2019, Nice, France*, pages 5593–5598, 2019. doi: 10.1109/CDC40024.2019.9030205.

Andrei D. Polyanin. *Linear Partial Differential Equations for Engineers and Scientists*. Chapman and Hall/CRC, New York, 2002.

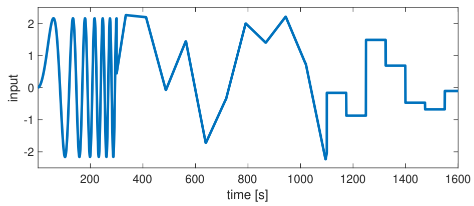
M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. doi: https://doi.org/10.1016/j.jcp.2018.10.045.

Torsten Soderstorm and Petre Stoica. *System identification*. Prentice Hall, Englewood Cliffs, NJ, 1989.

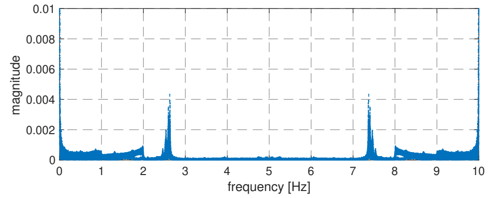
R. Zarfam and A.R. Khaloo. Vibration control of beams on elastic foundation under a moving vehicle and random lateral excitations. *Journal of Sound and Vibration*, 331(6): 1217–1232, 2012. doi: https://doi.org/10.1016/j.jsv.2011.11.001.

## Appendix A. Figures and Tables


Figure 1: Vibration beam laboratory stand.

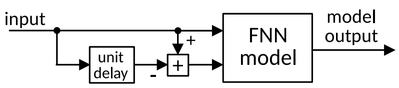



(a) A portion of the input signal.

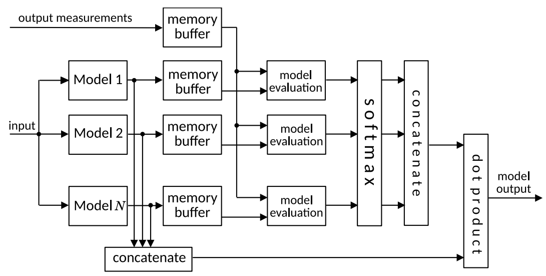



(b) Spectrum of the vibrating beam displacement.

Figure 2: Signals used in experiment.

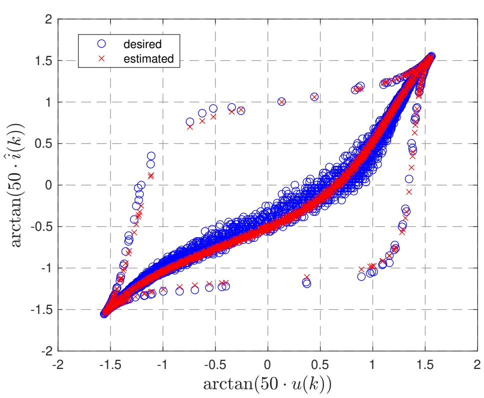



(a) Hysteresis modeling.

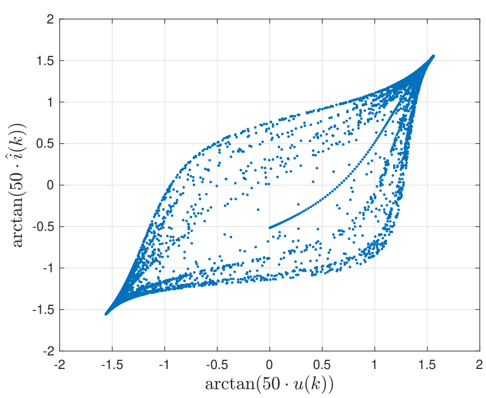



(b) Proposed dynamic multimodel.

Figure 3: Block schemes for subsystems.

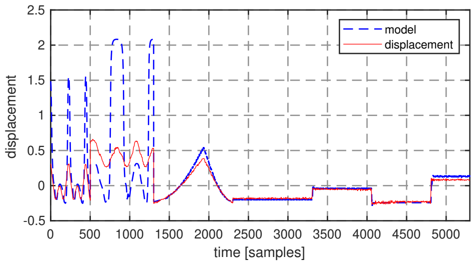



(a) Static characteristic of the actuator for a collection of ramp inputs.

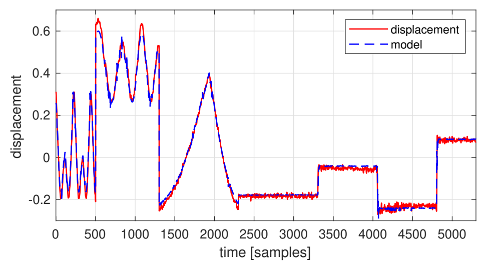



(b) Reconstruction by the neural model for the chirp signal.

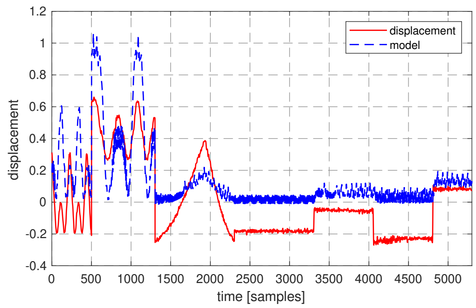
Figure 4: Hysteresis modeling results.




$(a)$ Single model.




$(b)$ Models ensemble




$(c)$ Averaging model.

Figure 5: Vibrating beam modeling results.

Table 1: Specification of selected neural models.

| No. | topology<br>units in layers | activation function<br>hidden/output | MSE |
|---|---|---|---|
| 1 | 5–3 | tanh/linear | 0.102 |
| 2 | 7–5 | tanh/linear | 0.0011 |
| 3 | 12–8–4 | tanh/linear | 0.000557 |
| 4 | **15–10–7–3** | tanh/linear | **0.000539** |

Table 2: Specification of models in the ensemble.

| No. | training data specification | MSE |
|---|---|---|
| 1 | chirp: freq. $0.01 - 0.05$Hz, amp. 1.5A | 0.19 |
| 2 | chirp: freq. $0.05 - 0.15$Hz, amp. 1.5A | 0.1749 |
| 3 | chirp: freq. $0.15 - 0.3$Hz, amp. 1.5A | 0.1557 |
| 4 | chirp: freq. $0.01 - 0.05$Hz, amp. 2.5A | 0.1802 |
| 5 | chirp: freq. $0.05 - 0.15$Hz, amp. 2.5A | 0.1393 |
| 6 | chirp: freq. $0.15 - 0.3$Hz, amp. 2.5A | 0.1616 |
| 7 | ramp collection | 0.1428 |
| 8 | random steps | 0.1787 |