



Fidelity Assessment of Real-Time Hybrid Substructure Testing: a Review and the Application of Artificial Neural Networks

C. Insam¹ · D. J. Rixen¹

Received: 4 December 2020 / Accepted: 22 March 2021 / Published online: 27 April 2021
© The Author(s) 2021

Abstract

Real-Time Hybrid Substructure (RTHS) testing is a commonly used method to investigate the dynamical influence of a component on a mechanical system. In RTHS, a part of the dynamical system is tested experimentally, while the remaining structure is simulated numerically in a co-simulation. There are several error sources in the RTHS loop that distort the test outcome. To investigate the reliability of the test, the fidelity of the test must be quantified. In many engineering applications, however, there is no reference solution available to which the test outcome can be validated against. This work reviews currently existing accuracy measures used in RTHS. Furthermore, using Artificial Neural Networks (ANN) to predict the fidelity of the RTHS test outcome when no reference solution is available is proposed. Appropriate input features for the network, such as dynamic properties of the system and existing error indicators, are discussed. ANN training was performed on a data set from a virtual RTHS (vRTHS) simulation of a dynamical system with contact. The training process was successful, meaning that the correlation between the ANN prediction and the true fidelity value was $> 99\%$. Then, the network was applied to data of experimental RTHS tests of the same dynamical system and achieved a correlation of 98% , which proves that the relation found by the ANN captured the relation between the chosen input features and the error measure. The application of the trained ANN to data from a linear vRTHS test revealed that further improvement of the network and the choice of input features is necessary. This work suggests that ANNs could be a meaningful tool to predict the fidelity of the RTHS test outcome in the absence of a reference solution, especially if more data from different RTHS tests were aggregated to train them.

Keywords Real-time hybrid testing/simulation · Artificial neural networks · Fidelity · Accuracy measures

Introduction

In Real Time Hybrid Substructure (RTHS) Testing, a dynamical system is analyzed by splitting it into a numerically simulated and an experimentally tested part. The substructures are coupled in real-time by a so-called transfer system that exchanges displacement/velocity and force information (flow and effort) between them [1]. The idea of Hybrid Substructuring was first proposed by Hakuno et al. [2] (in Japanese, briefly summarized in English in [3]). However, it took until the early 1990s and the work of Nakashima et al. [4] before more interest arose to RTHS. Since then, a

lot of research has been carried out, barriers overcome and the method applied to many engineering problems. Nevertheless, some important challenges still remain to apply RTHS in broader engineering practice. One of them is the establishment of fidelity measures, which this work deals with.

A schematic of RTHS is shown in Fig. 1.¹ The parts of a dynamical system that are relatively easy to simulate or not available as hardware are simulated numerically (in blue) and the parts that are difficult to model or critical are tested experimentally (in green) [5]. Using a numerical time integration algorithm, the dynamics of the numerical substructure are solved for one time step and the interface displacement z is commanded to the actuator. The controlled actuator performs the movement z' (different from z in practice) and moves the experimental part, which

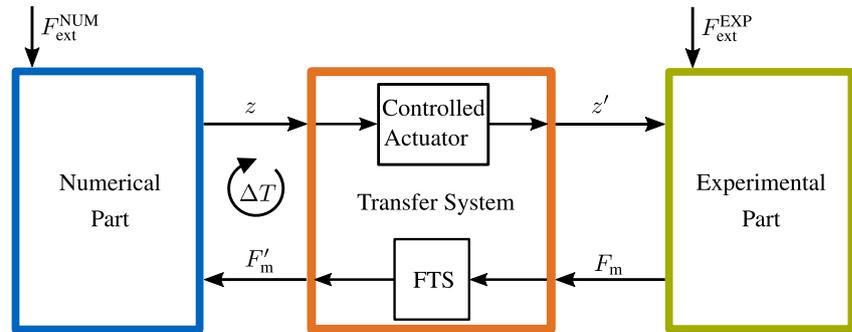
✉ C. Insam
christina.insam@tum.de

¹ Chair of Applied Mechanics, Faculty of Mechanical Engineering, Technical University of Munich, Boltzmannstr. 15, 85748 Garching, Germany

¹In the figure, one numerical and one experimental substructure are shown. In general, there can be multiple experimental substructures, though.



Fig. 1 Coupling in RTHS with the exchange of displacement and force information by the transfer system. The coupling is performed with a fixed time step ΔT .



is mounted at the end effector of the actuator. A force-torque sensor (FTS) measures the restoring forces F_m of the experimental part (measured forces F'_m). The measured interface forces are input to the numerical simulation. There might also be external forces $F_{\text{ext}}^{\text{NUM}}/F_{\text{ext}}^{\text{EXP}}$ that excite the numerical or respectively the experimental substructure. The whole loop is carried out with a fixed time step ΔT (e.g. 1 ms). Ideally, the transfer system manages to synchronize the numerical and experimental substructures, which means equilibrium and compatibility ($F'_m = F_m, z = z'$).

In an RTHS setup, there are multiple sources of errors that can lead to a loss of test fidelity or even instability of the test. According to [6–8], errors can be classified into systematic (epistemic) and random (aleatoric) errors. The latter comprise noise in the displacement or force measurement as well as truncation in electrical signals at the analog to digital conversion [8]. Even though noise has a low amplitude, its high frequency content can excite higher modes in lightly damped structural systems [6]. An appropriate choice of numerical integration algorithm can eliminate this effect. In contrast to random errors, systematic errors appear with a regular pattern. Potential sources of systematic errors can be found in all parts of the RTHS loop [6, 9–12]:

- Numerical Part
 - modeling inaccuracies
 - numerical integration algorithm and time step selected
- Transfer System and Experimental Part
 - actuator control errors
 - transfer dynamics of the FTS
 - sensor miscalibration
 - communication and computational delays
 - flexibility in the reaction frame

A common assumption in literature is that the detrimental effects of systematic errors are larger than those of random errors. The systematic errors themselves are often dominated by the transfer dynamics of the controlled

actuator [13, 14]. The reason is that the introduced time lag can be in the order of a few milliseconds (e.g. 5 – 10 ms or even more), while the other systematic errors sum up to a time lag that might be smaller than 1 ms.

The success of an RTHS test, i.e. the test fidelity and stability, does not only depend on the amount of these errors. Rather, the susceptibility of an RTHS test to these errors depends i.a. on the synchronization at the interface, the partitioning of the dynamical system (splitting ratio of mass and stiffness into numerical/experimental part), the RTHS test's fastest eigenfrequency and the amount of damping in the system [8, 15]. This means that it depends on these mentioned factors whether a certain amount of error (e.g. communication delay of 2 ms) renders the test inaccurate or even unstable.

For RTHS to be applied as testing procedure for even more applications, it is essential that the test results are accurate and can be trusted [16]. Therefore, there is a need for fidelity measures that tell the user how well the test emulates the true dynamical behavior of the investigated system (numerical plus experimental part), i.e. how much the inescapable errors distort the results. In some instances, there exists a so-called reference solution, which is the true dynamic behavior, and an error measure can be built between the RTHS test outcome and the reference solution. This kind of error is termed reference error in this article. The reference solution is either a numerical simulation of the full dynamical system or an experimental test of it. In civil engineering, for example, the dynamics of a whole building can be investigated in full-scale using shake table test rigs [17–19]. However, if the experimental part is cumbersome to model or the numerical part not yet available as hardware, there is no reference solution available. Nevertheless, accuracy measures are needed to assess the fidelity of the RTHS test outcome and since the above mentioned factors all contribute to the success of an RTHS test, the definition of fidelity measures without having a reference solution at hand is a nontrivial problem.

In this work, a review about state-of-the-art accuracy measures is given. Furthermore, the application of Artificial Neural Networks (ANNs) to predict the test fidelity is

proposed. More specifically, the potential of an ANN to learn the relation between measurable quantities during an RTHS test and the test fidelity is investigated. The manuscript is structured as follows: Section 1 presents the state-of-the-art accuracy measures with their pros and cons. It additionally gives a brief introduction to ANNs. Then, in Section 1, the RTHS test used to generate the training data and the implementation of the Neural Network are presented. Section 1 shows and discusses the results and Section 1 summarizes the findings and gives an outlook on future work.

State of the Art

RTHS tests can become unstable if they are susceptible to random/systematic errors and/or if the amount of these errors is large. Instability should be avoided since the hardware (experimental part, transfer system) might be damaged or the user endangered. In this work, RTHS tests are considered where errors lead to inaccuracy of the test, but not instability. Methods to predict the stability of RTHS tests are presented in e.g. [12, 14, 20, 21].

Accuracy of RTHS Tests

This section presents state-of-the-art accuracy measures.² Since many factors contribute to the fidelity of a performed RTHS test, the establishment of accuracy measures is complex. Research in this field has been a niche in the community of RTHS and the earlier pseudodynamic testing.³ A good overview about existing accuracy measures that all have their pros and cons is provided in [23]. They also provide a MATLAB Toolkit with the implementation of many existing accuracy measures.

Accuracy measures

Common accuracy measures consider the desynchronization at the interface as the main contribution to the final error. A common assumption is that the model of the numerical part is accurate or that the modeling errors are negligible in contrast to other errors in the RTHS test [13, 14]. This

²The terms *accuracy* and *fidelity* are often used interchangeably. In this paper, *fidelity* refers to how accurately the dynamic behavior is mimicked by the RTHS test (see e.g. [1]). The term *accuracy* is used in a broader sense, i.e. for any quantities that measure the test performance (for example the actuator tracking performance).

³In pseudodynamic testing, the real-time condition is dropped and the experimental part is excited with extended time scale. Therefore, rate-dependent effects (such as damper/inertia forces) are not captured and need to be estimated numerically. [22]

assumption holds if the numerical substructure has linear dynamic behavior. For nonlinear numerical substructures, the proper modeling and time integration algorithm is of primordial importance, though (see e.g. [24]).

Tracking of the actuator This paragraph presents accuracy measures that are based on the interface compatibility, i.e. the transfer behavior from the actuator displacement command z to the true performed movement z' . No matter how well an actuator is controlled, it will never be ideal, i.e. $z = z'$. Rather, the true movement of the actuator has a (frequency-dependent) phase shift and amplitude error. Following [7], the phase error influences the stability of the RTHS test and the amplitude error its accuracy, because it changes the amount of damping in the system (overshoot dissipates energy, undershoot introduces energy).

The tracking performance can be quantified using the time domain signals of the gap $g = z - z'$. In literature one finds the relative root-mean-square (RMS) error $e_{\text{rel,track}}$, the maximum tracking error (MTE) $e_{\text{MTE,track}}$ and the tracking peak $e_{\text{peak,track}}$ error [8, 25–27]:

$$e_{\text{rel,track}} = \frac{RMS(g)}{RMS(z)} \quad (1)$$

$$e_{\text{MTE,track}} = \frac{\max(|g|)}{\max(z)} \quad (2)$$

$$e_{\text{peak,track}} = \left| \frac{\max(|z|) - \max(|z'|)}{\max(z)} \right| \quad (3)$$

The RMS value of a variable ξ with N samples is defined as $RMS(\xi) = \sqrt{\frac{1}{N} \sum_{i=1}^N \xi(i)^2}$. The normalization can be done with either the commanded (z) or real displacement (z'). The tracking performance can also be assessed in frequency domain by performing a Fourier transform of the signals [28]. The normalized RMS error in the frequency domain then writes

$$e_{\text{rel,track,f}} = \frac{RMS(|Z| - |Z'|)}{RMS(|Z|)}, \quad (4)$$

where the Fourier transforms of the commanded and real displacements are denoted with Z and respectively Z' and the index $(\cdot)_f$ indicates the frequency domain.

The time domain tracking performance (z , z') can be visualized in the so-called Synchronization Subspace Plot (SSP) proposed by [29]. Therein, the commanded displacement z is plotted against the measured displacement z' , see Fig. 2. In case of perfect tracking, a straight line with unit slope forms. If there is amplitude over/undershoot, the slope is smaller/larger than one. If the actuator introduces phase lag, an ellipse forms that evolves clockwise and if it introduces phase lead, the ellipse evolves counterclockwise.



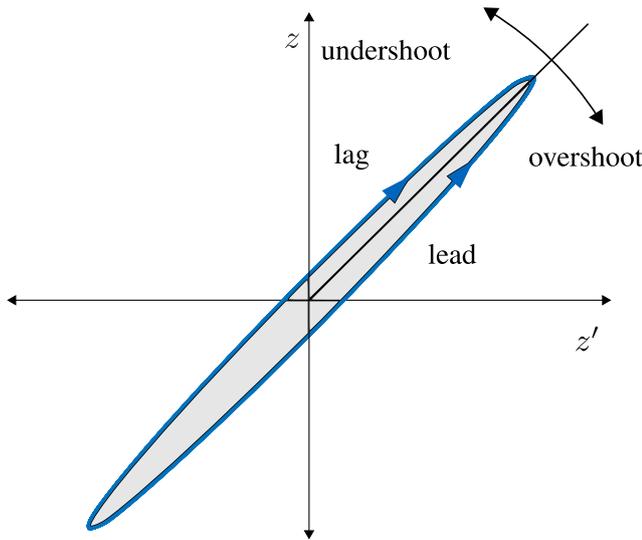


Fig. 2 Synchronization Subspace Plot (SSP). The shape of the ellipse represents the tracking performance of the actuator

Based on the SSP, [19] established the tracking and amplitude indicators (TI and Θ_{PC}). The tracking indicator corresponds to the enclosed area in the SSP plot and thus quantifies the phase lead/lag of the actuator. The amplitude indicator, in contrast, quantifies overshoot/undershoot of the actuator, as it measures the slope of the major axis of the ellipse by a principle component analysis. Following [7, 16] splitting the actuator dynamics into phase and amplitude error is necessary due to the distinct effects of the errors on the test outcome (stability vs. fidelity).

A huge disadvantage of the tracking indicator is its dependence on the magnitude of the displacement. Indeed, one can easily recognize that if the actuator delay is kept constant and the magnitude of the interface displacement doubled, also the value of the tracking indicator is doubled. This means that it can only be used as a qualitative but not as a quantitative accuracy measure. To circumvent this problem, [30] proposed the Phase and Amplitude Error Indices (PAEI) as an extension to the tracking indicator. Here, an ellipse is fit to the evolved curve in the SSP plot. Using the ellipse parameters, the phase lag and the amplitude error of the actuator can be extracted. A simplification of the implementation in [30] is presented in [31], which achieves good results, yet worse than those obtained from the PAEI. The calculation of the PAEI is possible when the SSP plot has been half passed through, i.e. at least half of the ellipse is needed to perform the ellipse fit. Note that all measures that use the SSP plot assume a constant, frequency-independent delay of the actuator. This assumption holds true for some applications, where the dynamic transfer behavior of the actuator remains constant for the frequency range of interest during the RTHS test.

In general, this is not true, however, and the phase of the actuator is frequency-dependent.

The so-called Frequency Evaluation Index (FEI) is a further accuracy measure that is based on SSPs and splits the actuator tracking error into a phase delay and an amplitude error [26, 32]. The FEI can be computed using:

$$Z_n = |Z|^l \text{ and } Z'_n = |Z'|^l \quad (5)$$

$$FEI = \frac{\sum_{j=1}^{\frac{N}{2}} \frac{Z'_j}{Z_j} \cdot Z_{n,j}}{\sum_{j=1}^{\frac{N}{2}} Z_{n,j}}, f_{eq} = \frac{\sum_{j=1}^{\frac{N}{2}} Z_{n,j} \cdot f_j}{\sum_{j=1}^{\frac{N}{2}} Z_{n,j}} \quad (6)$$

$$A = |FEI| \quad (7)$$

$$\Phi = \tan^{-1} \left(\frac{\text{Im}(FEI)}{\text{Re}(FEI)} \right), d = -\frac{\phi}{2\pi f_{eq}} \quad (8)$$

The calculation of the FEI uses the Fourier transform of the commanded and achieved displacement (Z , Z'). The Fourier transformed displacements are vectors of length $\frac{N}{2}$ and contain the frequency components up to the Nyquist frequency (half of the sampling frequency, i.e. $\frac{1}{2\Delta T}$). The frequency responses are weighted by an exponent l in (equation (5)). A choice of $l = 2$ is suitable for the application presented in [26]. Since the amplitude and phase error are frequency-dependent, an equivalent frequency f_{eq} is identified in (equation (6)). Herein, the vector f contains all frequencies (index j) from the Fourier transform up to the Nyquist frequency. The frequency spacing depends on the signal length. The equivalent frequency f_{eq} takes a scalar value and functions as a representative for all involved frequencies in the commanded signal. It is a weighted average, where the frequencies with higher peak magnitude are weighted more heavily. At this equivalent frequency, the delay value of the actuator is evaluated, see (equation (8)). The amplitude error A , the phase error Φ and the delay d of the actuator at the equivalent frequency result and are scalars.

Note that knowing the value of the amplitude and phase error indicates the size of the tracking error. Still, it is not easy to interpret the values and to estimate the influence of these errors on the test outcome.

Energy balance The transfer system ideally manages to synchronize the numerical and experimental part, i.e. achieve equilibrium and compatibility between them. Since in real experiments perfect synchronization cannot be achieved, there is energy flowing over the interface. The idea of observing the energy balance was proposed by Thewalt and Roman in [28]. This idea was worked upon by Mosqueda et al. [33, 34]. They propose the so-called Hybrid Simulation Error Monitors (HSEM), which are an estimate of the energy introduced by the transfer system

E_{error} normalized by the maximum strain energy E_{strain} ($HSEM^S$) and the input energy E_{input} ($HSEM^I$):

$$E_{\text{error}} = \int F'_m dz - \int F'_m dz' \quad (9)$$

$$E_{\text{strain}} = \frac{1}{2} z_{\text{max}}^T k_{\text{EXP}} z_{\text{max}} \text{ and } E_{\text{input}} = \int F_{\text{ext}}^{\text{NUM}} dz \quad (10)$$

$$HSEM^S = \frac{E_{\text{error}}}{E_{\text{strain}}} \text{ and } HSEM^I = \frac{E_{\text{error}}}{E_{\text{strain}} + E_{\text{input}}} \quad (11)$$

Here, the maximum deformation is denoted with z_{max} and the stiffness (in general: initial stiffness matrix) of the experimental structure with k_{EXP} . The choice of an appropriate threshold value for HSEM is ambiguous and relies on the expertise of the user, which makes it a qualitative rather than a quantitative accuracy measure [30]. Mosqueda et al. propose to determine the relationship between HSEM and an accuracy measure including the reference error to retrieve a suggestion for a quantitative HSEM threshold value. However, this relation differs for different dynamical systems. Other ideas for the choice of a threshold value include relating the error energy E_{error} to the maximum strain energy E_{strain} or the input energy E_{input} and permitting e.g. 5 % of error ($HSEM^S \leq 0.05$).

Based on HSEM, [12] proposed the Energy Error Indicator (EEI). The EEI is very similar to HSEM with the difference being that the energy balance equation considers all energies in the numerical substructure. Hence, it also includes errors in the numerical integration and not only in the transfer system. Similar to HSEM, the EEI values of two different tests cannot be compared because the errors of the transfer system are not split into amplitude and phase errors [16, 31].

Susceptibility of RTHS tests As mentioned in Section 1, the magnitude of the errors alone is not sufficient to determine the fidelity of the RTHS test outcome. Also, knowledge about the dynamical system, its eigenfrequency and its partitioning are of importance. Maghareh et al. investigated the susceptibility of RTHS tests based on their dynamic characteristics and the partitioning in the publications [15, 18, 35, 36]. They propose the Predictive Stability Indicator (PSI) and Predictive Performance Indicator (PPI). They use delay differential equations and the stability switch criterion to determine the critical delay of an RTHS system. Then, they classify experimental setups into extremely sensitive, moderately sensitive and slightly sensitive experiments. In [21], the use of robust stability analysis and the small gain theorem are proposed to assess the stability and performance of a planned RTHS test. Here, the actuator dynamics, the dynamics of the investigated system and its partitioning are taken into account. The conditions

are derived for the SISO and MIMO case. In general, when partitioning the system such that there is more experimental than numerical mass ($m_{\text{EXP}} > m_{\text{NUM}}$) and less experimental than numerical stiffness ($k_{\text{EXP}} < k_{\text{NUM}}$), the test is less sensitive to any errors. Furthermore, systems with higher damping, no matter whether in the numerical or experimental substructure, are less susceptible to any errors [8, 18, 20, 27].

Surrogate modeling For the past decade, a significant effort was put into applying uncertainty quantification and surrogate modeling to RTHS [37–39]. In contrast to the deterministic approach, uncertainties (of the dynamical properties or the actuator tracking performance) are incorporated and the propagation of uncertainties through the RTHS loop is investigated. In other words, uncertainty analysis investigates how sensitive the RTHS results are to changes in the system parameters. Sauder et al. [1] proposed using such a probabilistic approach and surrogate modeling to verify that the RTHS test fidelity is larger than a defined value. Furthermore, this approach helps to set minimum requirements for the actuator control performance. In this approach, a dynamics model of all involved parts is necessary.

Reference errors In case a reference solution z^r is available, the relative RMS reference error $e_{\text{rel,ref}}$, the maximum reference error $e_{\text{MRE,ref}}$, the peak reference error $e_{\text{peak,ref}}$ and the relative RMS reference error in frequency domain $e_{\text{rel,ref,f}}$ can be calculated similar to the tracking errors in (equation (1))–(equation (4)) by replacing z by z^r . Note that the commanded actuator displacement z is not equal to the reference solution z^r . During the hybrid simulation, compatibility between the numerical and experimental part is not exactly satisfied, i.e. $z' \approx z$. Therefore, the trajectory z of the interface during the test will be different from the reference z^r , which would be the trajectory if compatibility and equilibrium are always satisfied.

Note, that the reference solution, which is the true dynamic behavior, is either a simulation of the overall dynamical system (numerical and experimental part) or an experimental test of it. Using a pure experimental test to retrieve the reference solution is more accurate since simulation is based on models and thus inherently includes assumptions on the physics, parameters and is affected by numerical errors in the solution techniques. A disadvantage of experimentally determined reference solutions is, however, that the material properties of the used parts might differ slightly from those of the parts used in the RTHS test. Furthermore, experimental testing is not always possible due to large structural components or components that have not been manufactured yet.

Classification of accuracy measures

Existing accuracy measures can be classified according to diverse criteria. In [23], a distinction into local and global assessment measures is drawn. While local assessment measures consider the interface synchronization (equilibrium, compatibility), global assessment measures include system-level responses (error to a reference solution) that can be used to identify problems on the numerical or experimental substructure (partitioning effects, stability).

The working principle is used as basis for classification in [16, 40]. They distinguish assessment measures on the basis of the tracking error (SSP) from those based on the energy balance equation.

A classification into pre-experiment, online and post-experiment (offline) measures is presented in [18]. Pre-experiment measures are used to predict the susceptibility of RTHS tests to errors and determine the minimum requirements for the transfer system dynamics. Online measures help to terminate/interrupt a running RTHS test in case it suffers from noticeable errors. Post-experiment measures assess the quality of the test outcome with respect to a reference solution.

While all of the above classifications are meaningful, a distinction into error measures and error indicators is used in this work. Here, *error measures* refers to all accuracy measures that need the reference solution (quantify the test fidelity) for their calculation and *error indicators* comprise quantities that can be calculated solely with signals/properties measurable during the test or known beforehand.

Artificial Neural Networks

Artificial Neural Networks (ANNs) are a powerful Machine Learning method that is widely used for different kinds of problems, such as speech recognition, image classification, computer vision, forecasting in marketing/sales/financing or of electrical loads, robotics and healthcare [41–44]. The basic idea behind ANNs is to mimic the functioning of the human brain. The brain consists of neural cells that are connected to each other and form a giant connectome. A neural cell receives the signals from another neural cell through connections (synapses) via dendrites. The signal is processed and altered through the neural cell. The transformed signal is sent through the so-called axons to other neural cells. In computer science, this simple model of one neural cell is used to build a grand ANN, i.e. a network of neurons that can be trained on distinct tasks. Training means that the parameters of the network are adapted such that a given input generates a given output. A good introduction to ANNs can be found e.g. in [41]. More specifically, [45, 46] provide a review about the use

of ANNs in structural engineering and in [47] ANNs are applied to RTHS as reduction strategy to build a black box model for the numerical substructure.

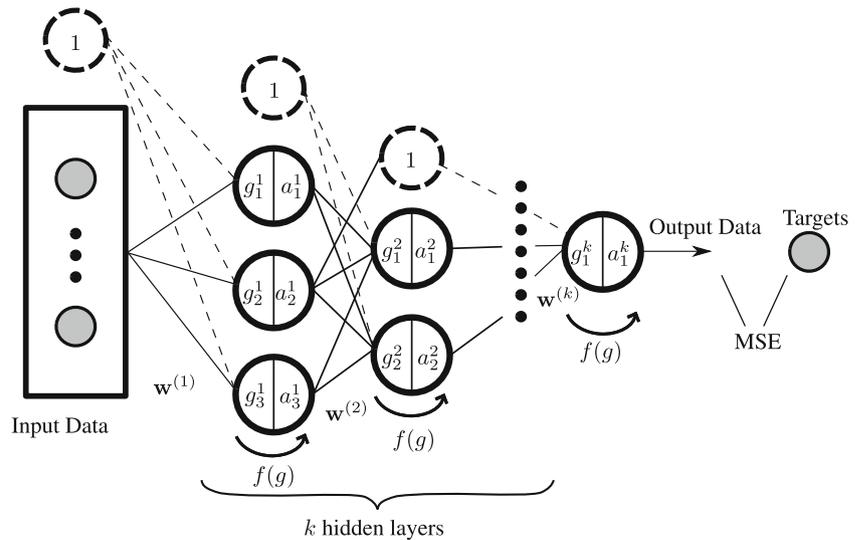
In this paper, the rather simple Feedforward Neural Network (FNN) with a supervised learning approach is used, where a set of input data and corresponding target data are available. The input data consist of many sets of input variables, which can be used for learning. The structure of a FNN is shown in Fig. 3. The input data g to a neuron are the weighted outputs of the previous layer (weights $w^{(i)}$). These inputs are then processed by an activation function $f(g)$ and form the output of the neuron a . Several activation functions are commonly used and the specific choice depends on the application. For example, if the FNN is used for a classification problem, the targets take binary values and therefore the activation function (of the output layer) should be the logistic sigmoid function (in the range of $[0, 1]$). For multi-class problems, the softmax activation function is recommended and for regression problems the identity (purely linear) activation function. In the hidden layers, there are also bias values which shift the output by a constant value and therefore it is possible to approximate more function values. [41]

During the training process, which is the learning of the relation between input and target data, the weights that connect the neurons are optimized such that the deviation between the net output and the target is minimized. The network performance, i.e. its error, is commonly measured with a mean squared error (MSE) between the net output and the target data. The algorithm to determine the weights during training is called backpropagation algorithm. It basically uses the gradient of the cost function between net outputs and targets to adapt the network weights so as to minimize the deviation between them. Each iteration of feedforward (evaluate the network) and backpropagation (updating the weights) is called an epoch. After training the network, it is assumed that the relation between the input and target data has been learned and the network can be applied to new input data in order to predict the (unknown) output [41].

A well-known problem during training of ANNs is overfitting. Often, the number of weights being optimized in the network is larger than the number of input data. To prevent the network from overfitting during training, one often used technique is early stopping. Therefore, a portion of the whole set of input data is excluded from the training procedure and called validation set. This set is forwarded through the network and the derivation from its target values is monitored. If the error on the validation set starts increasing, even though the error on the training set (data used for training) is still decreasing, the network starts to overfit the training data and the training process terminates.



Fig. 3 Principle of a FNN with input layer (grey), k hidden layers and an output layer with one target variable. The layers are fully connected and bias variables are used (dashed lines). The subscript indices represent the number of the neuron in a layer and the superscript indices denote the layer number. All weights between one layer and the next are arranged in a matrix and denoted with $w^{(i)}$



Usually, an additional data set is extracted from the input data, namely the test set. These data are only evaluated once by the ANN after training has terminated. If the deviation between the net output and the target values is small, it means that the training has worked and a relationship between the input and target values found. It is common to use a splitting ratio of 70 %, 15 % and 15 % of the whole input data into training, validation and test set.

After the training of an ANN has taken place, one might be interested to know the influence of each input variable to the net output, i.e. the importance of the variable to the prediction of the network. There exist plenty of methods to do so, see e.g. [48, 49]. In this work, the following two approaches are used to investigate the sensitivity of the network to a certain input variable: In the first approach, one input variable in each sample of the input data is set to its mean value and the change in the prediction accuracy of the network is monitored. As an example, if there are three input features and the data set consists of ten samples, the process is as follows:

- Consider input feature one, two or three.
- Calculate the mean value of the ten values of the considered input feature (one, two or three).
- Set this input feature in all ten samples to the mean value.
- Evaluate the network and record the MSE value.
- Reset the input values to their original ones and repeat the previous steps until each input feature has been investigated.
- Compare the MSE values and sort them in descending order. The first input feature is the most important, the one with the least change of MSE value is the least important.

The second approach is similar, but instead of setting the input variables successively to the mean of this input

feature, the values of each input variable are randomly permuted among the data set. In the given example, this means that the ten individual values of each input feature are randomly exchanged.

Test Setup and Implementation

In this section, the RTHS test is presented, the generation of the training data for the ANN, the implementation of the ANN and how the prediction accuracy of the ANN compared to the forecasting power of the existing accuracy measures is rated.

RTHS Test

The more input data are available, the better the ANN can be trained on a specific problem. However, it is quite time consuming to perform many RTHS tests. So, simulations of the full RTHS test, which is often referred to as vRTHS (virtual RTHS), were performed.⁴ Therein, the transfer system dynamics and experimental part are modeled too. The implementation was done using MATLAB/Simulink. A measured transfer function of the actuator, which is a Stewart Platform in this work, and its controller were implemented.⁵ The FTS was assumed to be ideal, i.e. $F'_m = F_m$. A more detailed description of the simulation environment can be found in [51]. The investigated system is a one-dimensional mass-spring-damper system experiencing contact, as

⁴Ideally, data from a real RTHS would be used for the training process as they include all possible sources of errors.

⁵Since the expected forces are small compared to the maximum force of the Stewart Platform, the dynamic behavior of the Stewart Platform is not influenced by the experimental component. Therefore, there is no need to consider control structure interaction [50].

can be seen in Fig. 4. In total, 280 vRTHS tests were performed, where the dynamical properties took the values as given in Table 1. The dynamical properties were arbitrarily chosen with the conditions that the maximum force of the used FTS was not exceeded and that the tests stayed within the limits of actuator stroke and maximum velocity. The Stewart Platform was controlled with different control parameters and with/without velocity feedforward ([52]) to alter the tracking performance of the actuator. All 280 vRTHS tests were stable.

In addition, some of the tests have also been conducted on the real test bench. Here, the same dynamical properties as for the vRTHS tests were used with the only difference being that the numerical mass had only one unique value of $m_{NUM} = 9.62$ kg. The data set of stable RTHS tests includes 100 samples.

The training of the ANN took place on the data from the presented vRTHS test with contact. Since it is also important to investigate how general the relation between input and output parameters is, the trained network (trained on the data from the vRTHS test with contact) was validated on a never seen dynamical system. This means, that a different dynamical system was chosen, RTHS simulations performed and the achieved data applied as test set to the trained network. The linear system shown in Fig. 5 with the parameters given in Table 2 was selected as dynamical system and performed vRTHS simulations. 16 samples were found, where all of the input parameters (that will be presented in Section 1) were in the range of the

Table 1 Parameters used for the vRTHS and RTHS tests of the dynamical system with contact (Fig. 4)

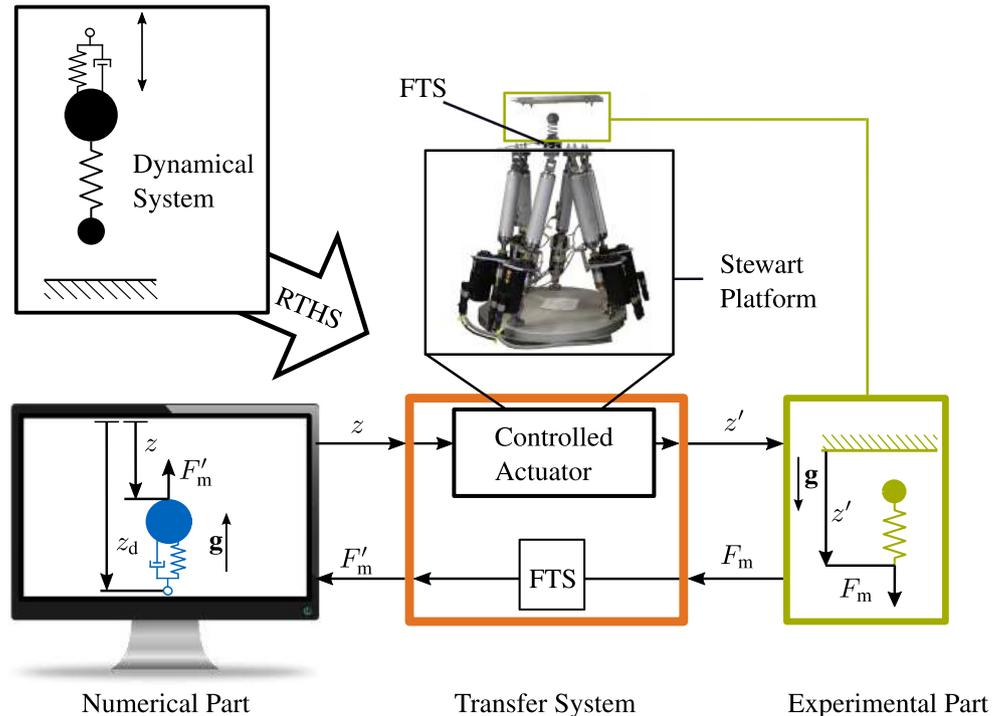
Variable	Value
h_0	[0.002 m, 0.005 m, 0.01 m]
f_d	[0.25 Hz, 0.5 Hz, 1 Hz]
Δz_d	[0.003 m, 0.005 m, 0.006 m, 0.007 m, 0.009 m]
g	$9.81 \frac{m}{s^2}$
d_{NUM}	[50, 75, 100, 150, 200, 250, 500] kg/s
k_{NUM}	[250, 500, 10^3 , $2.5 \cdot 10^3$, $5 \cdot 10^3$, $7.5 \cdot 10^3$, 10^4 , $1.5 \cdot 10^4$, $2 \cdot 10^4$, $2.5 \cdot 10^4$, $5 \cdot 10^4$, 10^5 , $5 \cdot 10^5$, $2 \cdot 10^6$] N/m
m_{NUM}	[4.81 kg, 9.62 kg, 19.24 kg]
k_{EXP}	8650 N/m
m_{EXP}	0.38 kg

input parameters of the vRTHS system with contact that were used for the training of the ANN. This is important because the network was trained on a certain range of input parameters. If the input values lie outside this range, this would mean that the network needs to extrapolate, which usually decreases confidence [41].

Choice of Input and Target Variables for ANN Training

The input features can include all parameters/values that are either properties of the dynamical system, measurable during the test or can be calculated based on measured

Fig. 4 The mass-spring-damper system is investigated using RTHS: The upper mass-spring-damper system (mass m_{NUM} , stiffness k_{NUM} and damping constant d_{NUM}) is numerically simulated and the lower mass-spring system (mass m_{EXP} and stiffness k_{EXP}) experiencing contact is investigated experimentally. The suspension moves in a cosine trajectory (z_d) with frequency f_d . The experimental mass starts from an initial height of h_0 . A Stewart Platform is used as actuator and performs the desired movement z . The amplitude of the cosine trajectory is $\frac{h_0 + \Delta z_d}{2}$



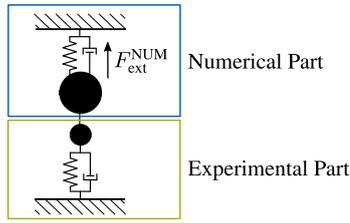


Fig. 5 This linear system consists of two mass-spring-damper systems. The numerical part consists of mass m_{NUM} , stiffness k_{NUM} and damping ratio d_{NUM} and the experimental part consists of mass m_{EXP} , stiffness k_{EXP} and damping ratio d_{EXP} . An external force of sinusoidal shape (amplitude 20 N and frequency 0.5 Hz) excites the numerical mass

quantities. The input variables should be available in all kinds of test, so e.g. the choice of h_0 or f_d as an input quantity would not be meaningful. In this study, it is assumed that the fidelity of the numerical model is high and that the time step ΔT is small enough such that the time integration scheme does not introduce a significant error. As explained in Section 1, the RTHS test fidelity does not only depend on the amount of error but also on the susceptibility of the RTHS test (partitioning, eigenfrequency, amount of damping, etc.). Hence, our input data need to include all these quantities.

The following errors explained in Section 1 were implemented as input features: the tracking errors from (equation (1)–(equation (4))), (equation (1)) was normalized with z , (equation (4)) with Z and (equation (2)) and (equation (3)) with z' , the FEI (and hence A and d), E_{error} , E_{input} and $HSEM^S$. For this kind of dynamical system an implementation of PAEI is not meaningful since the ellipse fitting starts working if the SSP plot is passed through for more than 180° , which is only after half of the test. PAEI shows its strength for dynamical systems with transient responses. During the data preparation process it was observed that using $HSEM^S$ leads to better results than using $HSEM^I$, therefore the latter was not used as input feature for the ANN training. The EEI provides the same value as $HSEM$, in our implementation.

Table 2 Parameters used for the vRTHS tests of the linear system

Variable	Value
d_{NUM}	100 kg/s
k_{NUM}	$[5 \cdot 10^3, 10^4, 5 \cdot 10^4]$ N/m
m_{NUM}	[5 kg, 9.62 kg, 20 kg]
d_{EXP}	0 kg/s
k_{EXP}	$[5 \cdot 10^3 \text{ N/m}, 8650 \text{ N/m}, 1.5 \cdot 10^4 \text{ N/m}]$
m_{EXP}	[0.2 kg, 0.5 kg]

Therefore, EEI was not used either as input feature. In preliminary investigations, it could be found that the relative RMS tracking error in time and in frequency domain (equation (1)) and (equation (4)) are fully correlated, meaning that there is a linear relationship between them. Nevertheless, both of them were used as input features.

Then, the partitioning of the system was included as input features, namely the mass ratio $\phi = \frac{m_{\text{NUM}}}{m_{\text{EXP}}}$ and the stiffness ratio $\kappa = \frac{k_{\text{NUM}}}{k_{\text{EXP}}}$. The partitioning of the damping does not change the susceptibility of the test [18]. Solely its absolute magnitude is of importance. Since the damping of the experimental part is unknown, the dimensionless critical damping value of the numerical part $\zeta = \frac{d_{\text{NUM}}}{2\sqrt{k_{\text{NUM}}m_{\text{NUM}}}}$ was selected as a further input feature. From experience and following i.a. [15], not only the absolute value of the time delay is of importance to the fidelity of the test, but its relation to the natural oscillation of the dynamical system (inverse of the eigenfrequency $\omega_{0,\text{DYN}}$). Since the eigenfrequency of the overall system is in general unknown, also the natural oscillation of the numerical system with eigenfrequency $\omega_{0,\text{NUM}}$ and the phase margin were used, which tells how much phase delay by the transfer system is permitted before the RTHS test becomes unstable [14]:

$$\tau_{\text{ratio,DYN}} = \frac{d}{\omega_{0,\text{DYN}}^{-1}} \text{ with } \omega_{0,\text{DYN}} = \sqrt{\frac{k_{\text{NUM}} + k_{\text{EXP}}}{m_{\text{NUM}} + m_{\text{EXP}}}} \quad (12)$$

$$\tau_{\text{ratio,NUM}} = \frac{d}{\omega_{0,\text{NUM}}^{-1}} \text{ with } \omega_{0,\text{NUM}} = \sqrt{\frac{k_{\text{NUM}}}{m_{\text{NUM}}}} \quad (13)$$

$$\tau_{\text{ratio,max}} = \frac{d}{PM_{\text{max}}} \text{ with } PM_{\text{max}} \text{ phase margin of the dynamical system} \quad (14)$$

In summary, 15 input features (see set 1 in Table 3) were selected that represent each vRTHS/RTHS test. All input features can be calculated based on mechanical properties or quantities that are measurable during the test. The choice of the input features is general, such that these quantities are available in any RTHS test. However, as one might have already noticed, there are still a few input variables that need knowledge about the experimental stiffness k_{EXP} , which is in general unknown. Therefore, a second set with 12 input features (set 2 in Table 3) is proposed. Note that even though $HSEM^S$ includes k_{EXP} in the calculation of the strain energy, an assumption of the order of magnitude is sufficient here (e.g. initial stiffness matrix [34]) and therefore it is present in set 2. A third set of input parameters is presented in the table and will be explained in Section 1.

As target value a reference error, i.e. an error measure that needs the reference solution (see Section 1), is chosen. There are the relative RMS reference error $e_{rel,ref}$, the maximum reference error $e_{MRE,ref}$, the peak reference error $e_{peak,ref}$ and the relative RMS reference error in frequency domain $e_{rel,ref,f}$. Since often used in literature, e.g. [34], the relative RMS reference error $e_{rel,ref}$ was chosen as the target variable. The reference solution z^r was found by setting the transfer system dynamics to 1 in the above explained vRTHS simulation (compatibility and equilibrium are satisfied), so the reference simulation was a pure simulation of the dynamical system (numerical and experimental part). The masses were considered as ideal rigid bodies, friction and damping in the experimental part were neglected and contact was modeled using the penalty method. For further details see [51].

Choice of ANN Properties

The FNN was set up and trained using the MATLAB Neural Network Toolbox (version R2020a, MathWorks). The inputs and targets were normalized to a range of $[-1, 1]$. The training was performed in batch mode and the splitting into training, validation and test set was done randomly (70 %–15 %–15 %). For better comparability the indices of the splitting were stored such that the same data were used as training/validation/test data each time the network was trained. Different topologies, meaning number of hidden layers and neurons per hidden layer were compared. When the training starts, the network weights are initialized randomly. This leads to slightly different training results and weights that are found by the backpropagation algorithm for different training runs. Therefore, each topology was trained for five times and selected the topology with the best average training performance (smallest MSE error between net output and targets after training). Here, this was one hidden layer ($k = 1$) with five neurons. The Levenberg-Marquardt algorithm was selected for training and the symmetric saturating linear activation function $f(g)$ was chosen for the hidden layer and a linear activation function for the output layer. For regression problems, it is common to use the rectified linear unit (ReLU) or the symmetric saturating linear activation function for the hidden layers. In preliminary analysis, the symmetric saturating linear activation function achieved better performance than ReLU and the training time did not increase unacceptably.

Performance Measure

The error indicators presented in Section 1 aim at identifying the test fidelity, i.e. how well the test emulates the true dynamics of the investigated system. A huge difficulty, however, is how to interpret the given quantitative values

Table 3 Input features used for training the ANN

set features	$e_{rel,track}$ (equation (1))	$e_{MTE,track}$ (equation (2))	$e_{peak,track}$ (equation (3))	$e_{rel,track,f}$ (equation (4))	A (equation (7))	d (equation (8))	E_{error} (equation (9))	E_{input} (equation (10))	$HSEM^S$ (equation (11))	ϕ	κ	ζ	$\tau_{ratio,DYN}$ (equation (12))	$\tau_{ratio,NUM}$ (equation (13))	$\tau_{ratio,max}$ (equation (14))
1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
3	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x



and to decide whether the threshold has been exceeded in the test and hence the test is unusable. Therefore, it would be of great help to know a relation between the error indicators and the error measures to at least know the magnitude of the test fidelity in terms of the reference error. Unfortunately, this relation cannot be determined easily and differs for different dynamical systems. Hence, the reference error should be predicted directly through the trained ANN.

The accuracy of the net output needs to be measured, namely how well the relative RMS reference errors (targets) are met and this is done by calculating the correlation. The net output for N_{test} test samples is denoted with a_{test} and the corresponding targets with t_{test} . Then, the correlation $\rho(a_{\text{test}}, t_{\text{test}})$ is defined as

$$\rho(a_{\text{test}}, t_{\text{test}}) = \frac{\text{cov}(a_{\text{test}}, t_{\text{test}})}{\sigma_a \sigma_t}, \tag{15}$$

with $\text{cov}(\cdot)$ being the covariance and σ_a and σ_t being the standard deviations of a_{test} and t_{test} . The correlation coefficient measures the linear dependence between the two variables and a value close to one indicates high correlation. If the network is able to learn the exact relation between input and target data, the net output a_{test} would correspond to the target values t_{test} and the correlation would be $\rho(a_{\text{test}}, t_{\text{test}}) = 1$.

Results and Discussion

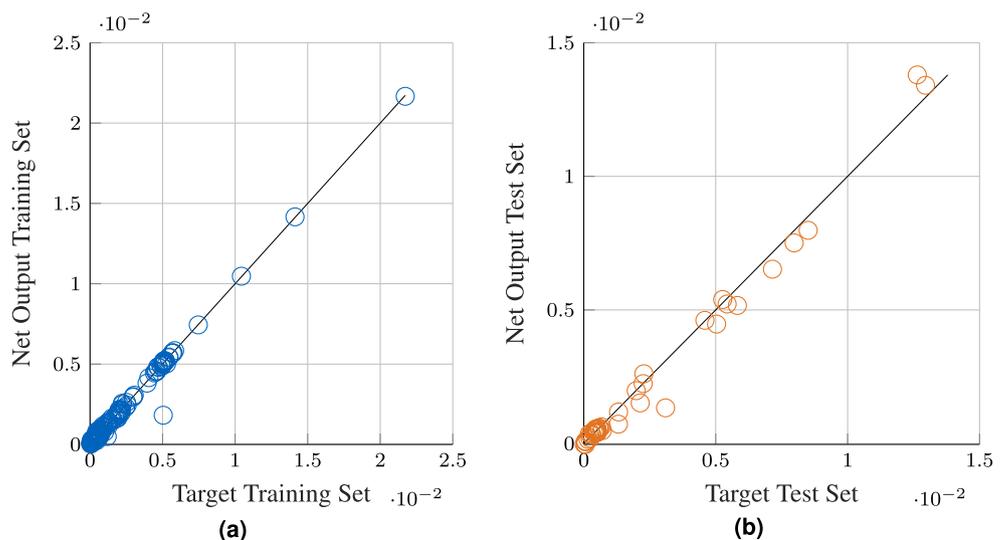
At first, the training performance of the FNN was investigated to see whether there is a relation between the available input features (see Table 3) and the targets, which is the relative RMS reference error $e_{\text{rel.ref}}$. The training was performed using the 280 data from the vRTHS test with contact

and split them into training, validation and test set. All presented input features (set 1 in the table) were used. During the training process of the FNN it can be seen that the error between the net outputs and the targets decreases for all of the three sets. The regression plots between the target values and the net outputs are shown in Fig. 6 for the training and test set (training terminated after 70 epochs). It can be seen that the training process was successful, because the circles lie on the black line which indicates a perfect prediction. The correlation between target and net output is 99.5% for the training set and 99.3% for the test set. Several network trainings were performed and the performance was high and comparable to the shown results in all cases. The MSE between the net outputs and the targets for all 280 data lay at about $4.1 \cdot 10^{-4}$ for the trainings.

Figure 7 shows the predicted and targeted relative RMS reference errors for the test set. It can be seen that the predicted values lie close to the targeted values and the magnitude of the net outputs is in the correct range. From Figs. 6 and 7 it can be concluded that there is a relation between the chosen input features and the target value, which is valid for this dynamical system (with contact and specific excitation). If there was not a consistent relation, the error on the test set would be much larger.

Until now, the input features included some knowledge about the experimental part, namely its stiffness k_{EXP} . The second set of input features in Table 3 presents all input features that are available after test and do not need knowledge about the dynamics of the experimental structure. So, a training with this set of input parameters was performed as well to investigate whether also here a relation exists. After training (also 280 input data from the vRTHS test with contact split into training, validation and test set), which took 21 epochs in this case, a correlation of 99.3% was achieved for the training data and 99.1% for the test

Fig. 6 Regression plots showing the target and predicted (net output) relative RMS reference errors. Points that lie on the black line indicate perfect prediction of the network



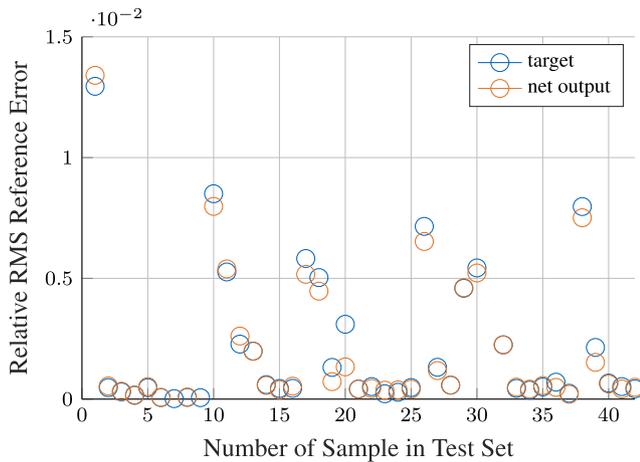


Fig. 7 The targets and net outputs are shown for all samples in the test set after termination of the training

data. The MSE had a value of about $5 \cdot 10^{-4}$. This means that some knowledge about k_{EXP} would be beneficial (set 1), since then the correlation is higher and the error smaller, but still the training is successful since the scale of the error is very small.

Then, it was investigated whether the trained network is able to predict the relative RMS reference error in the real RTHS test, i.e. with a physical transfer system and experimental part. The modeling of the physical components is not perfect in the vRTHS simulations and therefore the relation between the input features and the target value might be different. The regression plot and the network prediction are shown in Fig. 8. The network that was trained on the input feature set 2 was used and all 100 samples of the real RTHS test were fed through it. The correlation between the targets and net outputs is 98%. From the figures it is obvious, that the range of the magnitude is again predicted very well. The relative deviation is high for samples 85–98, where the network clearly underestimates the real reference error. For the remaining samples the prediction fits the target values well. There is no clear trend whether the net outputs over- or underestimate the target values. From this result it can be concluded that, in this specific RTHS setup, effects like noise and friction do not distort the prediction power. Therefore, the chosen input features are able to make a good prediction and are general such that these effects are not wrongly included in the learning process.

Next, a sensitivity analysis was performed to investigate the importance of each input feature to the accuracy of the net output. The importance of each input feature was compared for several trained networks for sets 1 and 2 of input features using the presented methods of setting one input feature to its mean or randomly permuting it (see

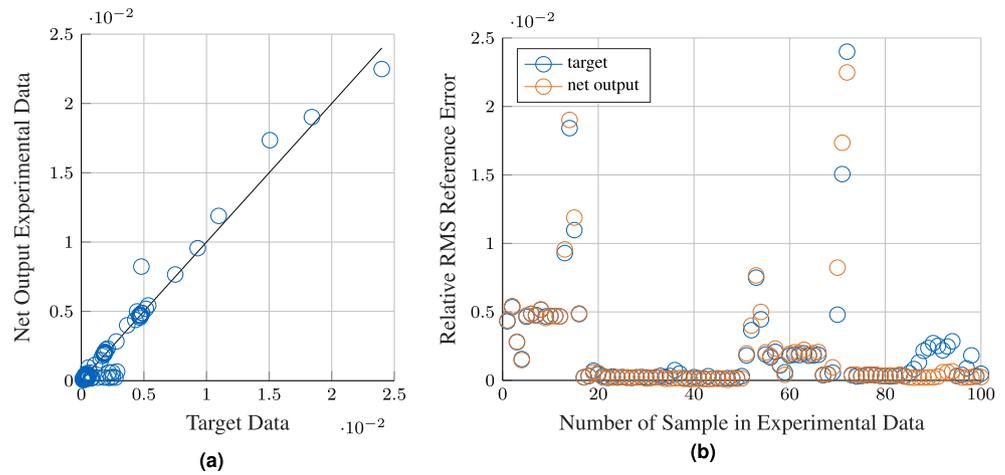
Section 1).⁶ Even though the training data were always the same, one cannot pinpoint an individual parameter that is always the most important. However, there are some variables that are always ranked more important than others. The parameters that are always more important are: $e_{rel,track}$, $e_{MTE,track}$, $e_{rel,track,f}$, d , $\tau_{ratio,NUM}$, $\tau_{ratio,DYN}$ and E_{error} . The features that are always ranked less important are: ϕ , ξ , A , $E_{track,peak}$, $\tau_{ratio,max}$, $HSEMS^S$, E_{input} . Also, network trainings were performed where only the important input features (set 3 in Table 3) were used. The training performance was still good (MSE at about $5.8 \cdot 10^{-4}$), but not as good as when input feature sets 1 and 2 were used. Note that the determined sensitivities show their independent contributions but do not consider the interdependence that might exist between some input features [48].

Finally, the applicability of the trained network to another dynamical system was studied, namely the linear system shown in Fig. 5. If the found relation between input features and targets is general and applicable to other dynamical systems, the correlation between the net outputs and the target values would be high. The networks trained on the vRTHS system with contact (networks trained with either input feature set 1 or 2) were used and applied to the 16 data samples from the linear vRTHS test. Note that it is vitally important to keep in mind that the networks are trained on a certain range for each input feature and target value. The value of each input feature and target value of the linear vRTHS test must lie in this range. Otherwise, the network would need to extrapolate. As mentioned earlier, the trained networks differ for individual training runs even if the same input parameters and the same topology are used. This is because of the random initialization at the beginning of the training and the different local minima that the backpropagation algorithm finds. The results were that some of the trained networks from different training runs could achieve correlations of up to 69% between the net outputs and targets. This is higher than any correlation between an input feature and the target, meaning that the prediction power of the network is higher than for an individual input feature alone (existing error indicators). However, there are also many networks, where the correlation was very low, i.e. close to zero or even negative (the network predicted negative relative RMS reference errors). So, the training is not very robust and a network that has been trained on one specific dynamical system is not *per se* applicable to other dynamical systems. Figure 9 shows the regression plot and prediction for a trained network (input features 2) where a high correlation of 65% was achieved. There are many reasons why the prediction power is so low and volatile for the trained networks:

⁶Both methods yielded the same results.



Fig. 8 The trained network (trained on vRTHS) applied to the data from the experimental RTHS tests



- Even though both dynamical systems are one dimensional and (piecewise) linear, contact occurs in the system that is used for the training. So, the relation between the input features and the target varies for different dynamical systems.
- Another reason might be that the selected input features do not capture all effects of the dynamical system and further possible input features need to be found.
- The target values of the linear system are in a very small range. The ANN was trained using 196 samples (70% of all available vRTHS tests), which is a rather low number. So, the resolution within the range of input features/target values is quite low, which might lead to this volatile prediction performance.
- It could be observed that networks, where the most important input features (sensitivity analysis) only included tracking properties ($e_{MTE,track}$, $e_{rel,track}$, $e_{rel,track}$) and the time delay d , had a poorer performance when predicting the error values of the linear system than the networks that are more sensitive to $\tau_{ratio,DYN}$ and $\tau_{ratio,NUM}$, too. All results are summarized in Table 4.

All the mentioned limitations could be overcome if more training data were used in the training process of the ANN. An idea would be to aggregate data from historical RTHS tests and current RTHS tests all over the world to cover a broader range of dynamical systems, values of the input features and RTHS setups (include further error sources such as measurement noise and sensor miscalibration). Ideally, RTHS tests would be used as training data where the reference solution has been found purely experimentally. The underlying idea is that the reference solution is more accurate when experimentally identified than when numerically simulated (modeling assumptions and errors of numerical time integration). With the current state of implementation, very good prediction performance was achieved for the real RTHS test, where an ANN trained on vRTHS was used. Therefore, it is proposed to perform a vRTHS test of the planned real RTHS test and approximate the dynamic behavior of the experimental part therein.

This procedure is applicable to RTHS setups where the FTS can be assumed to have ideal transfer behavior, because the current implementation does not include any

Fig. 9 The trained network (trained on vRTHS, input features 2) applied to the data from the linear vRTHS tests

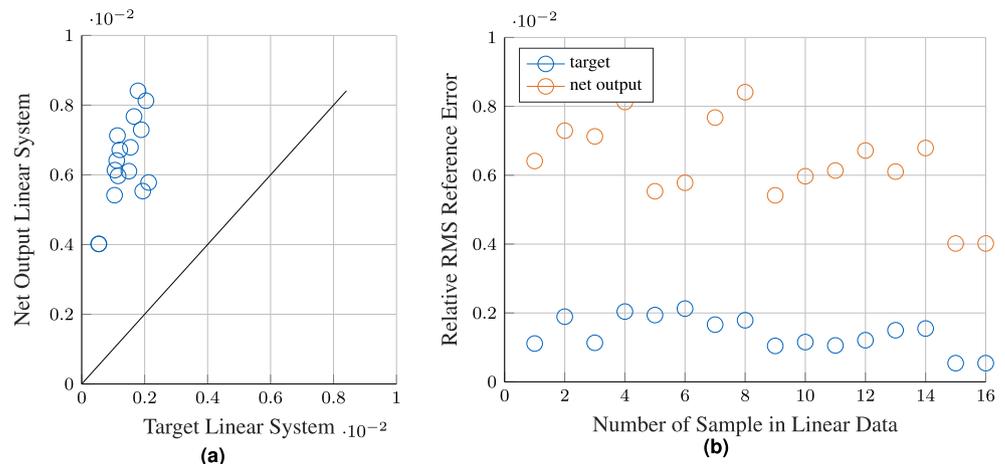


Table 4 Summary of the results

Dynamical system	Input feature set	Correlation		MSE	Notes
		Training	Test		
vRTHS with contact	1	99.5%	99.3%	$4.1 \cdot 10^{-4}$	used for training the network
	2	99.3%	99.1%	$5 \cdot 10^{-4}$	
	3	98.9%	98.9%	$5.8 \cdot 10^{-4}$	
RTHS with contact	2	98%			network applied to all samples
linear vRTHS	1 and 2	[0%, 69%]			

information about possible error in the measured forces and torques on the interface. If this method is applied to dynamical systems with multiple DoFs, it is recommended to use the most critical (fastest) eigenfrequency to calculate $\tau_{\text{ratio,NUM}}$ and $\tau_{\text{ratio,DYN}}$.

Conclusions

The contribution of this paper is twofold: firstly, an overview about existing accuracy measures with their benefits and shortcomings was given. Secondly, this work proposed the application of Artificial Neural Networks (ANNs) to predict the test fidelity based on test data such as error indicators and dynamic properties when there is no reference solution available. The potential of such an approach was investigated. Therefore, 280 training data with known reference solution were generated using a virtual RTHS simulation. The ANN was trained on that data set and the results revealed that the training process is successful, which indicates that a relation between the selected input features and the test fidelity, which was measured using the relative RMS reference error, exists. This implies that the existing error indicators capture the essential errors that determine the test fidelity. A sensitivity analysis revealed that the most important features to predict the targets are quantities that measure the tracking performance of the actuator, the equivalent time delay of the actuator and the delay relative to the period of the natural oscillation of the investigated numerical/dynamical system. The trained ANN was then applied to data from real RTHS tests of the same dynamical system and a high correlation between the net outputs and the targets was found. The application of the trained ANN to a different dynamical system, namely a linear system, revealed that the current implementation is not able to robustly predict the test fidelity for never seen dynamical systems. This implies that (i) it is not sufficient to just use training data from one dynamical system to learn a general relation and (ii) the chosen input features do not capture enough information about the system dynamics, which lead to distinct susceptibility of the RTHS test to errors.

To summarize, this work aimed at presenting the idea of using ANNs to predict the test fidelity when no reference solution is available and investigated whether ANN should be considered in principle. The underlying vision is to create a robust and powerful ANN that is able to predict the test fidelity of any (never seen) dynamical system and RTHS setup without reference solution at hand. Even though the current implementation of the ANN and the set of training data is not sufficient to already realize such a powerful ANN, this research showed that the approach itself offers some potential and the combination of available error measures and system properties could be meaningful. Still, further research is needed to achieve the goal of having a powerful and robust ANN that predicts the test fidelity of any novel RTHS test, stressing the fact that the ANN must be fed with data samples from many distinct RTHS tests (which relates this work with the field of big data). For example, historical RTHS tests and data from RTHS tests around the world need to be gathered to achieve this. Furthermore, future work could also investigate more complex dynamical Neural Networks (e.g. Recurrent Neural Networks) to predict the fidelity in each time step of the test and not only the RMS value.

A video showcasing the dynamic results of this work can be found under the following link: <https://youtu.be/rIiJ03IczM>.

Acknowledgments This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - project number 450801414.

We would like to express special thanks to the students involved in this project: Lisa Kühn and Shobhit Jain conducted the RTHS tests. Daniel Grassinger helped to prepare the input data and set up the FNN in MATLAB.

Funding Open Access funding enabled and organized by Projekt DEAL. This work was supported by the German Research Foundation (DFG), project number 450801414.

Code Availability Codes are written in MATLAB/Simulink and are available upon request.



Declarations

Conflict of Interests On behalf of all authors, the corresponding author states that there is no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Sauder T, Marelli S, Sørensen A (2019) Probabilistic robust design of control systems for high-fidelity cyberphysical testing. *Automatica* 101:111–119. <https://doi.org/10.1016/j.automatica.2018.11.040>
- Hakuno M, Shidawara M, Hara T (1969) Dynamic destructive test of a cantilever beam, controlled by an analog-computer. In: Proceedings of the Japan society of civil engineers, vol 1969.171, pp 1–9. https://doi.org/10.2208/jscej1969.1969.171_1
- Takanashi K, Nakashima M (1987) Japanese activities on online testing. In: *Journal of engineering mechanics*, vol 113.7, pp 1014–1032. [https://doi.org/10.1061/\(ASCE\)0733-9399\(1987\)113:7\(1014\)](https://doi.org/10.1061/(ASCE)0733-9399(1987)113:7(1014))
- Nakashima M, Kato H, Takaoka E (1992) Development of real-time pseudo dynamic testing. In: *Earthquake engineering & structural dynamics*, vol 21.1, pp 79–92. <https://doi.org/10.1002/eqe.4290210106>
- Plummer AR (2006) Model-in-the-loop testing. In: Proceedings of the institution of mechanical engineers, part I: journal of systems and control engineering, vol 220.3, pp 183–199. <https://doi.org/10.1243/09596518JSCE207>
- Mosqueda G, Stojadinović B, Mahin S (2005) Implementation and accuracy of continuous hybrid simulation with geographically distributed substructures. In: *Earthquake engineering research center, University of California, Berkeley, CA, USA*
- Mercan O, Ricles JM (2007) Stability and accuracy analysis of outer loop dynamics in real-time pseudodynamic testing of SDOF systems. In: *Earthquake engineering & structural dynamics*, vol 36.11, pp 1523–1543. <https://doi.org/10.1002/eqe.701>
- Lin F, Maghareh A, Dyke SJ, Lu X (2015) Experimental implementation of predictive indicators for configuring a real-time hybrid simulation. In: *Engineering structures*, vol 101, pp 427–438. ISSN: 0141–0296. <https://doi.org/10.1016/j.engstruct.2015.07.040>
- Ou G, Dyke SJ, Maghareh A, Qian Y (2013) Investigation of uncertainties associated with actuation modeling error and sensor noise on real time hybrid simulation performance. In: Proceedings of the 2013 international conference on computing, networking and communications (ICNC). IEEE Computer Society, pp 210–214. ISBN: 9781467352871. <https://doi.org/10.1109/ICCNC.2013.6504082>
- Del Carpio RM, Hashemi MJ, Mosqueda G (2017) Evaluation of integration methods for hybrid simulation of complex structural systems through collapse. In: *Earthquake engineering and engineering vibration*, vol 16, pp 745–759. <https://doi.org/10.1007/s11803-017-0411-z>
- Hashemi MJ, Mosqueda G, Lignos DG, Medina RA, Miranda E (2016) Assessment of numerical and experimental errors in hybrid simulation of framed structural systems through collapse. In: *Journal of earthquake engineering*, vol 20.6, pp 885–909. <https://doi.org/10.1080/13632469.2015.1110066>
- Ahmadzadeh M, Mosqueda G (2009) Online energy-based error indicator for the assessment of numerical and experimental errors in a hybrid simulation. In: *Engineering structures*, vol 31.9, pp 1987–1996. ISSN: 0141–0296. <https://doi.org/10.1016/j.engstruct.2009.03.002>
- Mercan O, Ricles JM (2008) Stability analysis for real-time pseudodynamic and hybrid pseudodynamic testing with multiple sources of delay. In: *Earthquake engineering & structural dynamics*, vol 37.10, pp 1269–1293. <https://doi.org/10.1002/eqe.814>
- Christenson R, Lin YZ, Emmons A, Bass B (2008) Large-scale experimental verification of semiactive control through real-time hybrid simulation. In: *Journal of structural engineering*, vol 134.4, pp 522–534. [https://doi.org/10.1061/\(ASCE\)0733-9445\(2008\)134:4\(522\)](https://doi.org/10.1061/(ASCE)0733-9445(2008)134:4(522))
- Maghareh A, Dyke SJ, Rabieniaharatbar S, Prakash A (2017) Predictive stability indicator: a novel approach to configuring a real-time hybrid simulation. In: *Earthquake engineering and structural dynamics*, vol 46.1, pp 95–116. <https://doi.org/10.1002/eqe>
- Hessabi RM, Mercan O (2010) Phase and amplitude error indices (PAEI) to assess the success of displacement based real-time testing. In: *Structures congress*, vol 2010, pp 3455–3466. [https://doi.org/10.1061/41130\(369\)312](https://doi.org/10.1061/41130(369)312)
- Blakeborough A, Williams MS, Darby AP, Williams DM (2001) The development of real-time substructure testing. In: *The royal society*, vol 359.1786, pp 1869–1891
- Maghareh A, Dyke SJ, Prakash A, Bunting GB (2014) Establishing a predictive performance indicator for real-time hybrid simulation. In: *Earthquake engineering & structural dynamics*, vol 43.15, pp 2299–2318. <https://doi.org/10.1002/eqe>
- Mercan O, Ricles JM (2009) Experimental studies on real-time testing of structures with elastomeric dampers. In: *Journal of structural engineering*, vol 135.9, pp 1124–1133. [https://doi.org/10.1061/\(ASCE\)0733-9445\(2009\)135:9\(1124\)](https://doi.org/10.1061/(ASCE)0733-9445(2009)135:9(1124))
- Chi F, Wang J, Jin F (2010) Delay-dependent stability and added damping of SDOF real-time dynamic hybrid testing. In: *Earthquake engineering and engineering vibration*, vol 9, pp 425–438. <https://doi.org/10.1007/s11803-010-0026-0>
- Botelho R, Christenson R (2015) Robust stability and performance analysis for multi-actuator real-time hybrid substructuring. In: *Dynamics of coupled structures*, vol 4, pp 1–7. ISBN: 978-3-319-15209-7. https://doi.org/10.1007/978-3-319-15209-7_1
- Pinto A, Pegon P, Magonette G, Tsionis G (2004) Pseudodynamic testing of bridges using non-linear substructuring. In: *Earthquake engineering & structural dynamics*, vol 33.11, pp 1125–1146
- Christenson R, Dyke SJ, Zhang J, Mosqueda G, Chen C, Nakata N, Laplace P, Song W, Chae Y, Marshall G, Ou G, Riascos Gonzales CA, Song C (2014) Hybrid simulation: a discussion of current assessment measures. In: Technical report, George E. Brown, Jr. Network for earthquake engineering simulation (NEES), West Lafayette, IN
- Bas EE, Moustafa MA (2020) Performance and limitations of real-time hybrid simulation with nonlinear computational substructures. In: *Experimental techniques*, vol 44, pp 715–734. <https://doi.org/10.1007/s40799-020-00385-6>



25. Chen C, Ricles JM (2009) Improving the inverse compensation method for real-time hybrid simulation through a dual compensation scheme. In: *Earthquake engineering and structural dynamics*, vol 38.10, 1237–1255. <https://doi.org/10.1002/eqe.904>
26. Xu W, Chen C, Guo T, Yang X (2017) Reliability assessment of real-time hybrid simulation under worst-case scenarios using frequency-domain evaluation indices. In: *Experimental techniques*, vol 41, pp 237–249. <https://doi.org/10.1007/s40799-017-0172-7>
27. Gao X, Castaneda N, Dyke SJ (2013) Real time hybrid simulation: from dynamic system, motion control to experimental error. In: *Earthquake engineering & structural dynamics*, vol 42.6, pp 815–832. <https://doi.org/10.1002/eqe.2246>
28. Thewalt C, Roman M (1994) Performance parameters for pseudodynamic tests. In: *Journal of structural engineering*, vol 120.9, pp 2768–2781. [https://doi.org/10.1061/\(ASCE\)0733-9445\(1994\)120:9\(2768\)](https://doi.org/10.1061/(ASCE)0733-9445(1994)120:9(2768))
29. Wallace MI, Sieber J, Neild SA, Wagg DJ, Krauskopf B (2005) Stability analysis of real-time dynamic substructuring using delay differential equation models. In: *Earthquake engineering & structural dynamics*, vol 34.15, pp 1817–1832. <https://doi.org/10.1002/eqe.513>
30. Hessabi RM, Mercan O (1186) Phase and amplitude error indices for error quantification in pseudodynamic testing. In: *Earthquake engineering and structural dynamics*, vol 41.10, pp 1347–1364. <https://doi.org/10.1002/eqe>
31. Hessabi RM, Mercan O (2013) A practical approach to the phase and amplitude error estimation for pseudodynamic (PSD) testing. In: *Structures congress*, pp 1938–1949. <https://doi.org/10.1061/9780784412848.170>
32. Guo T, Chen C, Xu W, Sanchez F (2014) A frequency response analysis approach for quantitative assessment of actuator tracking for real-time hybrid simulation. In: *Smart materials and structures*, vol 23.4, p 045042. <https://doi.org/10.1088/0964-1726/23/4/045042>
33. Mosqueda G, Stojadinović B, Mahin SA (2007) Real-time error monitoring for hybrid simulation. Part I: methodology and experimental verification. In: *Journal of structural engineering*, vol 133.8, pp 1100–1108. [https://doi.org/10.1061/\(ASCE\)0733-9445\(2007\)133:8\(1100\)](https://doi.org/10.1061/(ASCE)0733-9445(2007)133:8(1100))
34. Mosqueda G, Stojadinović B, Mahin SA (2007) Real-time error monitoring for hybrid simulation. Part II: structural response modification due to errors. In: *Journal of structural engineering*, vol 133.8, pp 1109–1117. [https://doi.org/10.1061/\(ASCE\)0733-9445\(2007\)133:8\(1109\)](https://doi.org/10.1061/(ASCE)0733-9445(2007)133:8(1109))
35. Maghareh A, Dyke SJ, Prakash A, Rhoads J (2014) Establishing a stability switch criterion for effective implementation of real-time hybrid simulation. In: *Smart structures and systems*, vol 14. <https://doi.org/10.12989/Sss.2014.14.6.1221>
36. Maghareh A (2017) Nonlinear robust framework for real-time hybrid simulation of structural systems: design, implementation, and validation. PhD Thesis. Purdue University
37. Abbiati G, Marelli S, Bursi O, Sudret B, Stojadinović B (2015) Uncertainty propagation and global sensitivity analysis in hybrid simulation using polynomial chaos expansion. In: *Proceedings of the fourth international conference on soft computing technology in civil, structural and environmental engineering*. vol 109, Fourth international conference on soft computing technology in civil, structural and environmental engineering; Conference Location: Prag, Czech Republic; Conference Date: September 1-4, 2015. Civil-Comp Press. Paper 23. <https://doi.org/10.4203/ccp.109.23>
38. Abbiati G (2017) Structural reliability analysis using deterministic hybrid simulations and adaptive kriging metamodeling. In: *Proceedings of the 16th world conference on earthquake (16WCEE)*
39. Ligeikis C, Christenson R (2020) Identifying stochastic frequency response functions using real-time hybrid substructuring, principal component analysis, and kriging metamodeling. In: *Experimental techniques*, vol 44, pp 763–786. <https://doi.org/10.1007/s40799-020-00389-2>
40. Ashasi-Sorkhabi A, Malekghasemi H, Mercan O (2015) Implementation and verification of real-time hybrid simulation (RTHS) using a shake table for research and education. In: *Journal of vibration and control*, vol 21.8, pp 1459–1472. <https://doi.org/10.1177/1077546313498616>
41. Bishop C (2006) *Pattern recognition and machine learning*. Springer
42. Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan T, Child R, Ramesh A, Ziegler DM, Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D (2020) Language models are few-shot learners. the work was presented at: 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada
43. Murray A (1995) Springer US. ISBN: 978-0-7923-9442-6. <https://doi.org/10.1007/978-1-4757-2379-3>
44. DeepMind (2021) AlphaStar: grandmaster level in StarCraft II using multi-agent reinforcement learning. <https://deepmind.com/blog/article/AlphaStar-Grandmaster-level-in-StarCraft-II-using-multi-agent-reinforcement-learning> (visited on 01/25/2021)
45. Adeli H (2001) Neural networks in civil engineering: 1989–2000. In: *Computer-aided civil and infrastructure engineering*, vol 16.2, pp 126–142. <https://doi.org/10.1111/0885-9507.00219>
46. Salehi H, Burgueño R (2018) Emerging artificial intelligence methods in structural engineering. In: *Engineering structures*, vol 171, pp 170–189. ISSN: 0141-0296. <https://doi.org/10.1016/j.engstruct.2018.05.084>
47. Bas EE, Moustafa MA, Feil-Seifer D, Blankenburg J (2021) Using a machine learning approach for computational substructure in real-time hybrid simulation. In: *Dynamic substructures*, vol 4. Springer International Publishing, pp 163–172. ISBN: 978-3-030-47630-4. https://doi.org/10.1007/978-3-030-47630-4_16
48. Hunter A, Kennedy L, Henry J, Ferguson I (2000) Application of neural networks and sensitivity analysis to improved prediction of trauma survival. In: *Computer methods and programs in biomedicine*, vol 62.1, pp 11–19. [https://doi.org/10.1016/S0169-2607\(99\)00046-2](https://doi.org/10.1016/S0169-2607(99)00046-2)
49. Gevrey M, Dimopoulos I, Lek S (2003) Review and comparison of methods to study the contribution of variables in artificial neural network models. In: *Ecological modelling*, vol 160.3, pp 249–264. [https://doi.org/10.1016/S0304-3800\(02\)00257-0](https://doi.org/10.1016/S0304-3800(02)00257-0)
50. Dyke SJ, Spencer BF, Quast P, Sain MK (1995) Role of control-structure interaction in protective system design. In: *Journal of engineering mechanics*, vol 121.2, pp 322–338. [https://doi.org/10.1061/\(ASCE\)0733-9399\(1995\)121:2\(322\)](https://doi.org/10.1061/(ASCE)0733-9399(1995)121:2(322))
51. Insam C, Kist A, Rixen DJ (2020) High fidelity real-time hybrid substructure testing using iterative learning control. In: *ISR - 52nd international symposium on robotics*
52. Insam C, Gödeli M, Klotz T, Rixen DJ (2021) Comparison of feedforward control schemes for real-time hybrid substructuring (RTHS). In: *Dynamic substructures*, vol 4. Springer International Publishing, pp 1–14. ISBN: 978-3-030-47630-4

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

