Mechanical Systems and Signal Processing 207 (2024) 110937

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

A hybrid modeling strategy for training data generation in machine learning-based structural health monitoring

Tim Vrtač, Domen Ocepek, Martin Česnik, Gregor Čepon *, Miha Boltežar

Faculty of Mechanical Engineering, University of Ljubljana, Aškerčeva 6, 1000 Ljubljana, Slovenia

ARTICLE INFO

Communicated by J.E. Mottershead

Keywords:
Structural health monitoring
Joint-damage identification
Frequency Based Substructuring
Machine learning
Training set generation

ABSTRACT

Concerning the cost- and resource-saving maintenance of assembly products, it is vital to detect any potential malfunctions, defects or structural damage at the earliest-possible stage. For this reason, considerable efforts are being put into the development of Structural Health Monitoring, a field encompassing different approaches to damage identification and capable of preventing defects and even failure. Structural Health Monitoring is often supported by machine learning, a tool for rapid and effective damage identification that can recognize patterns or changes in the data received from the structure. Despite the advances machine learning has made in recent years, obtaining a suitable data set for the efficient training of machine learning algorithms within Structural Health Monitoring remains a challenge. Currently, the data are usually obtained experimentally, with numerical or analytical models. However, the experimental approach can often be time consuming, while the reliability of numerically obtained data relies heavily on the accuracy of the numerical models in capturing the true behavior of the structure. Analytical models may be constrained by the complexity of the observed object. In this paper an alternative approach based on an experimental–numerical (i.e., hybrid) modeling approach is proposed to build a training set for Structural Health Monitoring. Frequency Based Substructuring is utilized to determine the response model of the assembled system based on the properties of its components as well as to mix experimental and numerical models, while leveraging the advantages of each. This makes it possible to generate the samples of the training set in the form of hybrid models of the structure of interest, exhibiting the realistic properties of a physical structure, with a reasonable measurement effort. Here, the approach is demonstrated for the process of joint-damage identification.

1. Introduction

In recent years, machine learning (ML) has developed rapidly and its applications have been extended to various aspects of daily life. Among others, ML approaches are gaining importance in the field of Structural Health Monitoring (SHM) [1–3]. The aim of implementing ML in SHM is to provide early damage detection and thus make structures and devices used on a daily basis safer and more reliable. A thorough understanding of the current state of a system's health makes it possible to perform predictive maintenance, which can lead to significant cost reductions compared to traditional, time-based maintenance strategies [4]. However, training an ML model that is capable of accurate damage identification requires a sufficiently large amount of training data that contains information about the damage to the structure. There are different approaches to the training of ML algorithms, two of the most common being supervised and unsupervised learning.

* Corresponding author.
E-mail address: gregor.cepon@fs.uni-lj.si (G. Čepon).

https://doi.org/10.1016/j.ymssp.2023.110937
Received 17 July 2023; Received in revised form 24 October 2023; Accepted 9 November 2023
Available online 17 November 2023

To train an unsupervised ML algorithm, only unlabeled data consisting of features of the undamaged condition of the structure is required [5]. The main task of such an approach is to measure dissimilarity between the two structural conditions [6]. The unsupervised approach is preferred when the goal is only to detect or/and localize the damage [7], due to the lesser data requirements compared to the supervised learning. Due to this advantages the unsupervised learning has been utilized in numerous applications e.g., by Sarmadi and Yuen [8] for damage detection in concrete bridges, by Entezami et al. [9,10] for early damage assessment in bridges under varying severe environmental conditions etc.

However, to obtain further information about the damage to the structure (i.e. type, extent of the damage or even the remaining useful life of the structure [11]) a supervised ML approach is required. In supervised learning additional information have to be provided to the ML model during the training process. This information is stored in form of the labeled data comprising undamaged as well as damaged samples.

In SHM different types of data obtained from real structures can be used in a damage-identification process (e.g., strain measurement, ultrasonic inspection, electromechanical impedance-based monitoring [12]), one of the most widely adopted being vibration-based SHM [13–15]. The dynamic response of the structure can be used as an input into ML algorithms in different forms, e.g., in the form of accelerations (time domain) [16,17], in the form of frequency response functions (frequency domain) [18], or the form of modal parameters, i.e., natural frequencies, mode shapes and damping, that represent a model of the structure in the modal domain [19–21].

In addition to the type of data used in the SHM process, a decision needs to be made regarding the data-collection methods. The data set, consisting of damaged (with known damage states) as well as undamaged samples, is usually obtained either by measurements made on real structures [16,22] or by numerical simulations [23–25]. Advances in measurement technology (improvements to conventional sensors as well as the development of new measurement techniques, e.g., measurements using smartphones, UAVs, self-charging sensors, connecting sensors into sensor networks, etc. [26]), have enabled the acquisition of large amounts of data. However, one of the main issues when using the experimental approach is that the various devices or structures operate in a healthy condition most of the time. This means that large amounts of data are collected for an undamaged structure, while there is a lack of information about the structure when it is subjected to different damage scenarios [27–29]. This poses a particular challenge for structures that have been only recently developed and one-of-a-kind structures and are therefore limited in terms of the availability of the data [30,31]. However, acquiring data through numerical simulations such as finite element (FE) analysis can be computationally demanding. Furthermore, due to the inherent limitations of numerical models, the obtained data often fall short of fully representing the properties of the actual structure, largely due to the uncertainty of material and geometrical parameters fed into the numerical models [30]. These drawbacks were addressed in different ways. The model's accuracy can be improved by the model updating process [32–34], but this is a demanding and time-consuming process. On the other hand, the efficiency of the synthetic data set generation was improved by using the uniform design method, which makes it possible to reduce the necessary number of simulations through a careful choice of the simulation's input parameters [35]. However, the uniform-design method demands a proper understanding of the input parameters to build a suitable training set. In addition, the substantial number of simulations required poses a challenge for ML algorithms that rely on a large volume of training samples for effective training. Another approach to reducing the high computational cost of the FE analysis was the introduction of simpler models, based on FE models, obtained using the response surface (RS) methodology. This employs statistical techniques to model the relationships between the input and response variables [36]. The problem with this approach is that the deviations of the FE model from the actual structure can be additionally amplified when using the RS methodology.

In this paper an approach to generating the ML training data set, specifically designed for vibration-based SHM, is proposed. The proposed approach targets the assembly products, composed of multiple substructures, varying in complexity, and interconnected by joints. Given that the damage to such products often occurs in the joint section, the primary focus of the proposed approach lies in identifying the damage in joints.

Each sample of the training set is designed as a hybrid model of a structure, combining experimental, numerical and analytical data of the system's components. Joints connecting the substructures are modeled analytically, which allows us to easily simulate various levels of damage severity. On the other hand, each substructure is modeled in a hybrid manner by mixing the experimental model with the numerical model of the substructure, using the dynamic substructuring approach. Such models exhibit dynamic properties, consistent with those of actual substructures and at the same time inherit a high spatial density of the numerical model. This is crucial for an adequate description of the interfaces between the substructures and the adjacent joints. By employing the dynamic substructuring approach, joint models can be coupled with hybrid models of the substructures, to obtain a valid model of the complete system. Through the ability to generate a large number of accurate dynamic models with varying degrees of joint damage, it is possible to construct a comprehensive training set for the ML. As we are only interested in detecting the damage in the joint regions between substructures, the same hybrid substructure models can be used for all the generated samples. This approach ensures computational efficiency and minimizes the experimental effort throughout the process. We will demonstrate that the resulting data set can be utilized to train an ML model that can accurately identify the location of the damaged joint and evaluate the extent of the damage.

The paper is organized as follows. First, the theory behind Frequency Based Substructuring is briefly presented, as it provides a basis for the methods used in the proposed training-set-generation approach. In Section 3, the training-set-generation procedure is presented. Finally, the method is applied to build a training set for ML-based joint-damage identification on a laboratory structure (Section 4). Subsequently, the trained ML model is tested for its effectiveness in detecting joint damage using both numerical and experimental input data.

2. Frequency based substructuring

Frequency based substructuring (FBS) enables the analysis of a complex system by leveraging the dynamic properties of its individual components, often referred to as substructures. Each substructure is represented by the corresponding admittance matrix $\mathbf{Y}^{(i)}$. When the components of the system are considered separately, there is a set of degrees of freedom (DoFs) located at the common boundary between the substructures (Fig. 1a). These are called interface DoFs, denoted with the subscript $(\star)_2$. In these DoFs the interface forces $\boldsymbol{g}$ have to be introduced. These are the forces that hold the substructures together. DoFs away from the interface are called internal DoFs. In addition to the interface forces, such a system can be subjected to the external forces $\boldsymbol{f}$ located anywhere on the assembly. In this case the system dynamics can be described using an uncoupled system of equations[1]:

$$\boldsymbol{u} = \mathbf{Y}(\boldsymbol{f} + \boldsymbol{g}), \tag{1}$$

where $\boldsymbol{u}$ is a vector of the system's responses to the excitation and $\mathbf{Y}$ is the uncoupled admittance matrix of the system.

For a system of two substructures A and B (Fig. 1), with their respective DoFs partitioned to internal DoFs $(\star)_1$, $(\star)_3$ and interface DoFs, elements of Eq. (1) can be written as[2]:

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_1^{(\mathrm{A})} \\ \boldsymbol{u}_2^{(\mathrm{A})} \\ \boldsymbol{u}_2^{(\mathrm{B})} \\ \boldsymbol{u}_3^{(\mathrm{B})} \end{bmatrix}, \ \mathbf{Y} = \begin{bmatrix} \mathbf{Y}_{11}^{(\mathrm{A})} & \mathbf{Y}_{12}^{(\mathrm{A})} & 0 & 0 \\ \mathbf{Y}_{21}^{(\mathrm{A})} & \mathbf{Y}_{22}^{(\mathrm{A})} & 0 & 0 \\ 0 & 0 & \mathbf{Y}_{22}^{(\mathrm{B})} & \mathbf{Y}_{23}^{(\mathrm{B})} \\ 0 & 0 & \mathbf{Y}_{32}^{(\mathrm{B})} & \mathbf{Y}_{33}^{(\mathrm{B})} \end{bmatrix}, \ \boldsymbol{f} = \begin{bmatrix} \boldsymbol{f}_1 \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \ \boldsymbol{g} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{g}_2^{(\mathrm{A})} \\ \boldsymbol{g}_2^{(\mathrm{B})} \\ \mathbf{0} \end{bmatrix}. \tag{2}$$

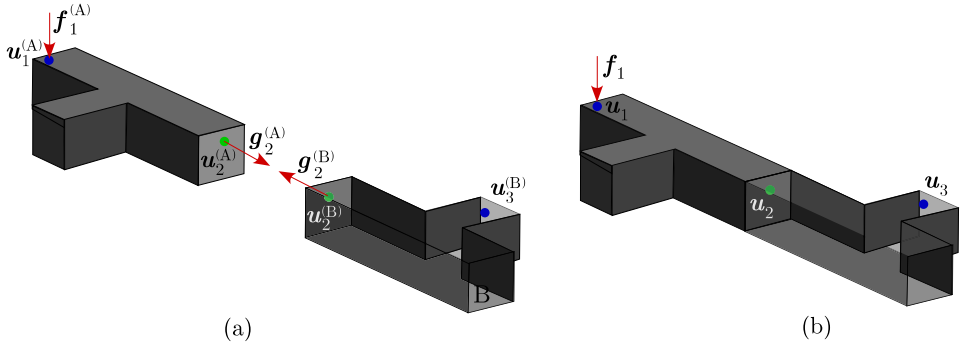

Fig. 1. System of substructures A and B in: (a) uncoupled form, (b) coupled form. (•) Interface DoFs, (•) internal DoFs.

To obtain the dynamic properties for the assembled system, the dynamic interaction between the substructures is determined by imposing compatibility and equilibrium conditions at the interface DoFs.

The compatibility condition yields the matching displacements at the interface DoFs. This can be expressed by the equation:

$$\mathbf{B}\,\boldsymbol{u} = \mathbf{0}; \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & -\mathbf{I} & \mathbf{0} \end{bmatrix}, \tag{3}$$

matrix $\mathbf{B}$ being a Boolean matrix pairing matching interface DoFs.

On the other hand, the equilibrium condition assumes the equilibrium of interface forces $\boldsymbol{g}$. Since matching interface forces $\boldsymbol{g}_2^{(\mathrm{A})}$ and $\boldsymbol{g}_2^{(\mathrm{B})}$ differ only in sign, but not in magnitude (according to Newton's third law of motion), the equilibrium condition can be satisfied by replacing the interface forces with a set of Lagrange multipliers λ denoting the magnitude of the interface forces ($\boldsymbol{g}_2^{(\mathrm{A})} = -\boldsymbol{g}_2^{(\mathrm{B})} = \lambda$):

$$\boldsymbol{g} = -\mathbf{B}^{\mathrm{T}}\lambda. \tag{4}$$

Imposing compatibility and equilibrium conditions on the admittance formulation of the uncoupled system (1) it is possible to derive the equation for the response of the coupled system [37]:

$$\boldsymbol{u} = \widetilde{\mathbf{Y}}\,\boldsymbol{f}; \quad \widetilde{\mathbf{Y}} = \mathbf{Y} - \mathbf{Y}\mathbf{B}^{\mathrm{T}}\left(\mathbf{B}\mathbf{Y}\mathbf{B}^{\mathrm{T}}\right)^{-1}\mathbf{B}\mathbf{Y}, \tag{5}$$

where $\widetilde{\mathbf{Y}}$ is a coupled admittance for the assembly AB. Eq. (5) is often referred to as the Lagrange Multiplier Frequency Based Substructuring (LM FBS) equation. Taking into account that the excitation DoFs might not always match the response DoFs, Eq. (5) can be rewritten in a more general form [38]:

$$\widetilde{\mathbf{Y}} = \mathbf{Y} - \mathbf{Y}\mathbf{B}_e^{\mathrm{T}}\left(\mathbf{B}_c\mathbf{Y}\mathbf{B}_e^{\mathrm{T}}\right)^{-1}\mathbf{B}_c\mathbf{Y}. \tag{6}$$

[1] An explicit dependency on the frequency is omitted to improve the readability of the notation, as will be the case for the remainder of the paper.

[2] For the sake of simplicity, it is assumed that the system is excited only in the internal DoFs of substructure A.

where $\mathbf{B}_c$ and $\mathbf{B}_e$ are Boolean matrices corresponding to the response DoFs and excitation DoFs, respectively.

The reverse process of dynamic coupling is dynamic decoupling. It provides us with a possibility to decouple a substructure from the entire system and obtain the dynamic properties of the remaining subsystem. This can be achieved using LM FBS equation (5) with a negative signed admittance for the substructure to be decoupled. For the case of decoupling substructure B from the coupled system AB (Fig. 2) the uncoupled admittance equals [39]:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}^{(\mathrm{AB})} & \mathbf{0} \\ \mathbf{0} & -\mathbf{Y}^{(\mathrm{B})} \end{bmatrix}. \tag{7}$$

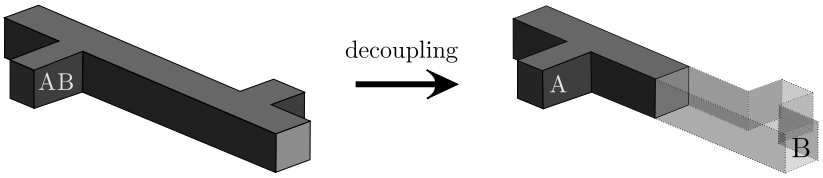

Fig. 2. Decoupling of substructure B from the coupled system AB.

3. Generation of training set using a hybrid modeling approach

To train ML algorithms that are capable of accurate damage identification, a training set that contains a sufficient number of samples to cover a wide range of possible damage scenarios is required. Obtaining training data experimentally is often a time-consuming task, while building a synthetic training set by numerical simulations is limited by the accuracy of the numerical models and can be computationally intensive for large and complex models.

In this paper an alternative methodology based on the hybrid dynamic substructuring techniques for generating a data set for structural damage identification using ML is proposed. The aim of the approach is to provide a large number of differently damaged models of the structure, along with the corresponding labels that contain information about the damage to the structure. This enables us to use the data set in the supervised machine learning approaches. The method focuses on the assembly structures consisting of multiple parts connected by joints where the damage will occur. The process of model generation consists of several steps, which are presented in Fig. 3 and explained in the following.

In the first step the hybrid models of the system parts are created, which means that each resulting model is a combination of an experimental and numerical model (Step 1 in Fig. 3). In this way the dynamic properties of the experimental model (measured at only a few points on the structure) are expanded to the high spatial resolution of a numerical model.

In the second step the analytical joint models are generated (Step 2 in Fig. 3). Such an approach to joint modeling allows us to manipulate the joint properties simply by changing the values of its parameters (mass, damping and stiffness properties) and thus to simulate the damage to individual joints simply by reducing the joint-stiffness parameters by a damage factor that determines the level of the damage.

Next, hybrid models of the assembly parts as well as the generated joint models are treated as substructures in the dynamic coupling process (step 3 in Fig. 3). By coupling adjacent substructures, a dynamic model of the complete assembly is obtained. The same hybrid models can be coupled with a large number of differently damaged joint models, which means that they are constructed only once for the entire data set. After that, each sample of the data set is obtained solely by performing dynamic coupling of substructure models with differently damaged joint models (repeating steps 2 and 3). The efficiency of substructuring methods has been demonstrated in [40,41]. Generated models form a data set, intended for use in the ML model-training process.

3.1. Step 1: Hybrid modeling using a dynamic substructuring approach

A hybrid model is the result of mixing the experimental model of an assembly part with its numerical model using the System Equivalent Model Mixing (SEMM) method [42].

The SEMM method assumes three models: the parent model $\mathbf{Y}^{(\mathrm{par})}$, which is most commonly a numerical model and determines the DoFs of the resulting hybrid model; the overlay model $\mathbf{Y}^{(\mathrm{ov})}$, which imposes its dynamic properties on the DoFs of the parent model (usually an experimental model with sparse set of DoFs), and the removed model $\mathbf{Y}^{(\mathrm{rem})}$, a submodel of the parent model that contains the DoFs of the parent model that correspond to the DoFs of the overlay model. DoFs that are common to all three models (the interface DoFs) are called boundary DoFs. The remaining DoFs are referred to as internal DoFs. The model's DoFs in individual matrices are usually partitioned in such way that the boundary DoFs follow the internal DoFs:

$$\mathbf{Y}^{(\mathrm{par})} = \begin{bmatrix} \mathbf{Y}_{ii} & \mathbf{Y}_{ib} \\ \mathbf{Y}_{bi} & \mathbf{Y}_{bb} \end{bmatrix}; \quad \mathbf{Y}^{(\mathrm{ov})} = \left[\mathbf{Y}_{bb}\right]^{(\mathrm{ov})}; \quad \mathbf{Y}^{(\mathrm{rem})} = \left[\mathbf{Y}_{bb}\right]^{(\mathrm{par})}, \tag{8}$$

where the indices i and b denote the internal and boundary DoFs, respectively.

To obtain a hybrid model, the parent model is coupled with the overlay model and the removed model is decoupled (Fig. 4).

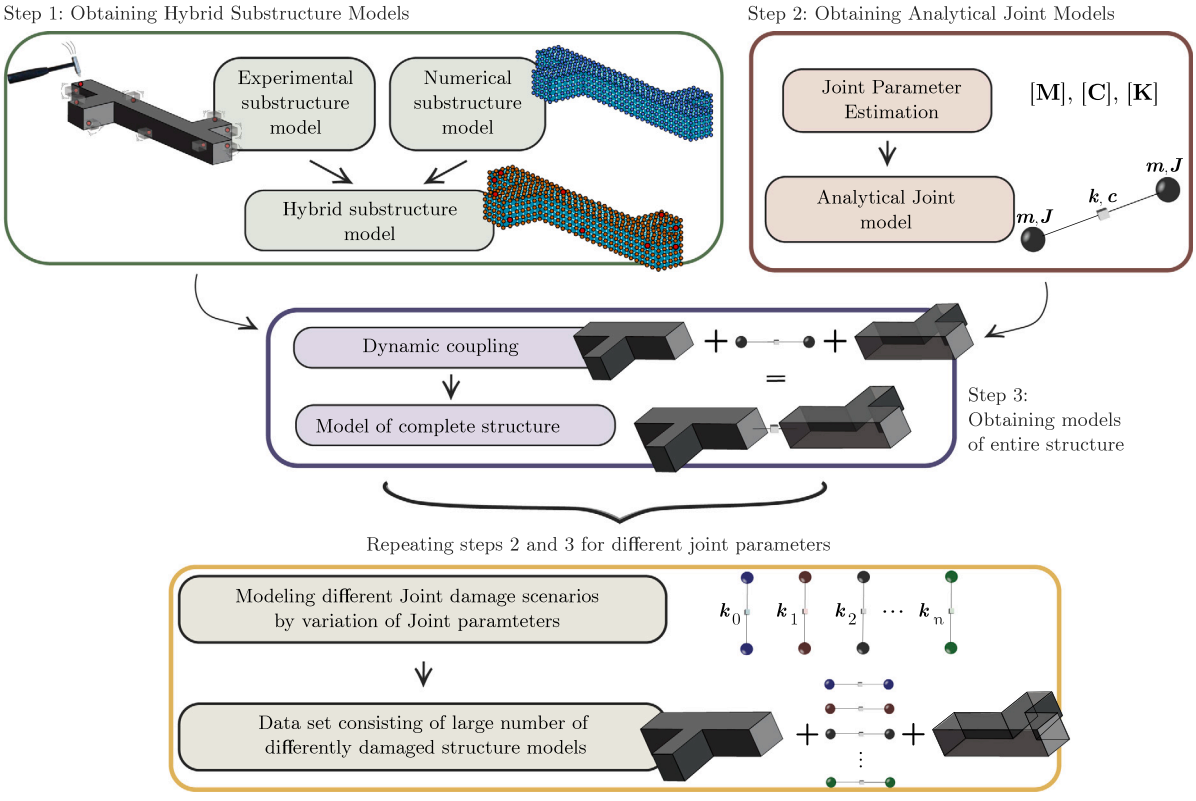

Fig. 3. Data set generation flowchart using hybrid modeling approach.

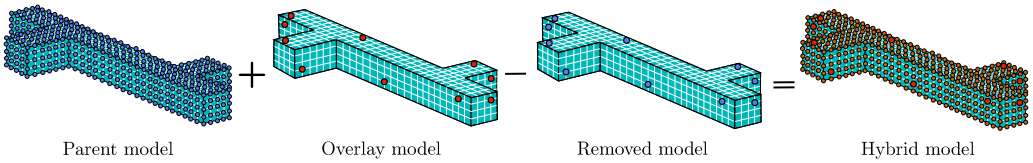

Fig. 4. SEMM method: By coupling the parent model with the overlay model and decoupling the removed model, a hybrid model is obtained.

The coupling and decoupling can involve a single step using the LM FBS equation (5) for which an uncoupled admittance matrix is defined as:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}^{(\mathrm{par})} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}^{(\mathrm{ov})} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{Y}^{(\mathrm{rem})} \end{bmatrix} \tag{9}$$

and the Boolean matrix $\mathbf{B}$ as:

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}^{(\mathrm{par})} & \mathbf{B}^{(\mathrm{ov})} & \mathbf{B}^{(\mathrm{rem})} \end{bmatrix} = \left[\begin{array}{cc|c|c} \mathbf{0} & -\mathbf{I} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} \end{array}\right]. \tag{10}$$

It turns out that using the original formulation of the SEMM method can result in spurious peaks that are a consequence of the conflicting dynamics[3] between the parent model $\mathbf{Y}^{(\mathrm{par})}$ and the overlay model $\mathbf{Y}^{(\mathrm{ov})}$ [43]. This problem can be solved by expanding the removed model definition to the internal DoFs, expanding the decoupling interface. This extension of the original SEMM method is referred to as an extended-interface SEMM and can be expressed using the following equation:

$$\mathbf{Y}^{(\mathrm{SEMM})} = [\mathbf{Y}]^{(\mathrm{par})} - \begin{bmatrix} \mathbf{Y}_{ik_e} & \mathbf{Y}_{ib} \\ \mathbf{Y}_{bk_e} & \mathbf{Y}_{bb} \end{bmatrix}^{(\mathrm{par})} \left(\begin{bmatrix} \mathbf{Y}_{bk_e} & \mathbf{Y}_{bb} \end{bmatrix}^{(\mathrm{rem})} \right)^{+} \left(\mathbf{Y}_{bb}^{(\mathrm{rem})} - \mathbf{Y}^{(\mathrm{ov})} \right) \left(\begin{bmatrix} \mathbf{Y}_{k_c b} \\ \mathbf{Y}_{bb} \end{bmatrix}^{(\mathrm{rem})} \right)^{+} \begin{bmatrix} \mathbf{Y}_{k_c i} & \mathbf{Y}_{k_c b} \\ \mathbf{Y}_{bi} & \mathbf{Y}_{bb} \end{bmatrix}^{(\mathrm{par})}, \tag{11}$$

[3] The parent and the overlay model observe and control slightly different dynamic spaces, which is due to the slight differences between the two models.

where the indices k_c and k_e denote the (internal) output and input DoFs that are included in the removed model, respectively.

By performing measurements at a few points on physical structures (to obtain experimental models) and creating equivalent numerical models, an extended SEMM method can be used to obtain hybrid models of the individual assembly parts. These exhibit properties of the real structure and at the same time contain a high DoF density of the numerical model. Another important advantage of this approach is that the numerical model does not need to thoroughly match the experimental model (introduction of so-called quasi-equivalent numerical models) as the dynamic of the parent model is entirely decoupled[4] and the properties of the experimental model are extended to all the parent DoFs, as shown by Kodrič et al. in [44]. This provides us with detailed descriptions of the interfaces between the individual substructures required for the successful dynamic coupling of hybrid models with corresponding joint models.

3.2. Phase 2: Joint models

Having generated the hybrid models of each assembly part, in the following phase the joint models connecting the physical substructures are obtained. Each joint model consists of two points of mass, treated as 6-DoF systems (resulting in a 12-DoF joint model). Spatial dimensions of the joint are assumed negligible. The connection between these two points of mass is determined by the damping and stiffness properties of the joint (Fig. 5). These properties can be arranged in the joint's mass matrix $\mathbf{M}^{(\mathrm{J})} \in \mathbb{R}^{12\times12}$, the damping matrix $\mathbf{C}^{(\mathrm{J})} \in \mathbb{R}^{12\times12}$ and the stiffness matrix $\mathbf{K}^{(\mathrm{J})} \in \mathbb{R}^{12\times12}$:

$$\mathbf{M}^{(\mathrm{J})} = \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} \end{bmatrix}, \quad \mathbf{C}^{(\mathrm{J})} = \begin{bmatrix} \mathbf{C} & -\mathbf{C} \\ -\mathbf{C} & \mathbf{C} \end{bmatrix}, \quad \mathbf{K}^{(\mathrm{J})} = \begin{bmatrix} \mathbf{K} & -\mathbf{K} \\ -\mathbf{K} & \mathbf{K} \end{bmatrix}, \tag{12}$$

$\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$ being submatrices of $\mathbf{M}^{(\mathrm{J})}$, $\mathbf{C}^{(\mathrm{J})}$, and $\mathbf{K}^{(\mathrm{J})}$, respectively, defined as follows:

$$\mathbf{M} = \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & J_x & 0 & 0 \\ 0 & 0 & 0 & 0 & J_y & 0 \\ 0 & 0 & 0 & 0 & 0 & J_z \end{bmatrix}; \ \mathbf{C} = \begin{bmatrix} c_x & 0 & 0 & 0 & 0 & 0 \\ 0 & c_y & 0 & 0 & 0 & 0 \\ 0 & 0 & c_z & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{tx} & 0 & 0 \\ 0 & 0 & 0 & 0 & c_{ty} & 0 \\ 0 & 0 & 0 & 0 & 0 & c_{tz} \end{bmatrix}; \ \mathbf{K} = \begin{bmatrix} k_x & 0 & 0 & 0 & 0 & 0 \\ 0 & k_y & 0 & 0 & 0 & 0 \\ 0 & 0 & k_z & 0 & 0 & 0 \\ 0 & 0 & 0 & k_{tx} & 0 & 0 \\ 0 & 0 & 0 & 0 & k_{ty} & 0 \\ 0 & 0 & 0 & 0 & 0 & k_{tz} \end{bmatrix}, \tag{13}$$

where m is the joint mass and J_x, J_y, J_z are its moments of inertia with respect to the axes x, y, and z, respectively.

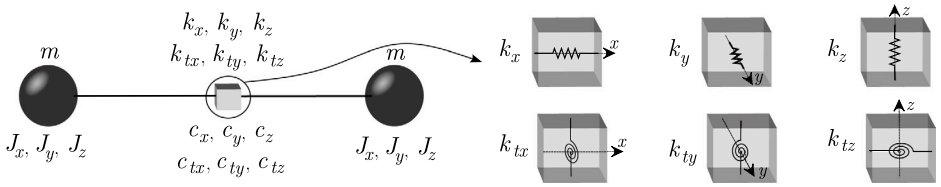

Fig. 5. Joint model.

To be able to dynamically couple the joint with neighboring hybrid part models, the matrices in Eq. (12) are used to calculate the admittance matrix of the joint $\mathbf{Y}^{(\mathrm{J})}$ as:

$$\mathbf{Y}^{(\mathrm{J})} = \left[-\mathbf{M}^{(\mathrm{J})}\omega^2 + \mathbf{C}^{(\mathrm{J})}\mathrm{i}\omega + \mathbf{K}^{(\mathrm{J})}\right]^{-1}. \tag{14}$$

Knowledge of the undamaged state is vital for damage-identification tasks [45]; therefore, care must be taken when selecting the properties of the undamaged joint (elements of matrices $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$). The joint parameters were fitted manually in such a manner that the dynamically coupled structure (hybrid part models coupled with joints) fit the measured assembly's response.

After obtaining the joint properties of an undamaged structure, joint damage is simulated as a decrease in the joint's stiffness. By varying the individual translational or rotational stiffness parameters of the undamaged joint, it is possible to simulate various damage scenarios and thus generate a large number of differently damaged joint models. The level of the joint's damage is determined by the vector of damage factors $\boldsymbol{\beta} = \begin{bmatrix}\beta_1 & \beta_2 & \dots & \beta_6\end{bmatrix}^{\mathrm{T}}$, where $\beta_i = [0, 1]$ for $i = 1 \dots 6$. The value $\beta_i = 0$ denotes the undamaged state, and the value of β_i=1 denotes the highest degree of damage. Damage factors $\boldsymbol{\beta}$ are applied to each submatrix $\mathbf{K}_{\mathrm{udmg}}$ of the stiffness matrix of the undamaged joint $\mathbf{K}^{(\mathrm{J})}_{\mathrm{udmg}}$ to obtain the submatrix $\mathbf{K}_{\mathrm{dmg}}$ of the stiffness matrix of the damaged joint $\mathbf{K}^{(\mathrm{J})}_{\mathrm{dmg}}$ using the following equation [46]:

$$\mathbf{K}_{\mathrm{dmg}} = (\mathbf{I} - \mathrm{diag}\,(\boldsymbol{\beta}))\,\mathbf{K}_{\mathrm{udmg}}, \tag{15}$$

where $\mathbf{I} \in \mathbb{R}^{6\times6}$ is an identity matrix.

[4] As long as observed and controlled dynamics in parent and overlay models are similar.

In some cases, the upper damage modeling strategy can be simplified under assumption that the properties of the joint change the same in all directions. In this case, Eq. (15) can be simplified to:

$$\mathbf{K_{dmg}} = (1-\beta)\mathbf{K}_{\mathrm{udmg}}, \tag{16}$$

where β is a single damage factor common for all the stiffness parameters in $\mathbf{K}_{\mathrm{udmg}}$ matrix.

Inserting the resulting matrix into the stiffness matrix of the damaged joint $\mathbf{K}^{(\mathrm{J})}_{\mathrm{dmg}}$ (Eq. (12)) and using $\mathbf{K}^{(\mathrm{J})}_{\mathrm{dmg}}$ in Eq. (14), gives us the admittance matrix of the damaged joint $\mathbf{Y}^{(\mathrm{J})}_{\mathrm{dmg}}$. The generated joint models are later used in the coupling process for the generation of differently damaged models of a complete structure.

3.3. Phase 3: Dynamic coupling of substructures

Obtaining hybrid substructure models, as well as joint admittance matrices, all the models to be used in the dynamic coupling process were gathered. However, physical substructures described by hybrid-response models only include translational DoFs at the interface, while joint interfaces are described by both translational and rotational DoFs. This non-collocation means that the models cannot be coupled directly. This problem is tackled using an approach called virtual point transformation [47].

3.3.1. Virtual point transformation

Virtual point transformation (VPT) makes it possible to project the displacements as well as forces observed at the DoFs in the vicinity of the interface into a single node with six DoFs (3 translations and 3 rotations), called a virtual point (VP) (Fig. 6). The relations between the observable displacements (stored in a vector of displacements $\boldsymbol{u} \in \mathbb{R}^n$) and the VP displacements (stored in a vector of displacements $\boldsymbol{q} \in \mathbb{R}^m$; $m \leq n$[5]) are described by mutual kinematic relations, stored in the interface deformation mode (IDM) matrix $\mathbf{R}_{\mathrm{u}} \in \mathbb{R}^{n\times m}$[6] [48]:

$$\boldsymbol{u} = \mathbf{R}_{\mathrm{u}}\boldsymbol{q} + \boldsymbol{\mu}, \tag{17}$$

where $\boldsymbol{\mu}$ is a vector of residual displacements, which cannot be described by the IDM matrix $\mathbf{R}_{\mathrm{u}}$. In order to minimize the residuals $\boldsymbol{\mu}$, Eq. (17) is solved in a least-square sense. Taking this into account, the equation describing the displacements of the VP can be derived:

$$\boldsymbol{q} = \mathbf{T}_{\mathrm{u}}\boldsymbol{u}; \quad \mathbf{T}_{\mathrm{u}} = \left(\mathbf{R_u}^{\mathrm{T}}\mathbf{R}_{\mathrm{u}}\right)^{-1}\mathbf{R}_{\mathrm{u}}^{\mathrm{T}}, \tag{18}$$

where $\mathbf{T}_{\mathrm{u}}$ is a displacement-transformation matrix. Similarly, the forces affecting the interface DoFs can be projected into the VP, using the force-transformation matrix:

$$\mathbf{T}_{\mathrm{f}}^{\mathrm{T}} = \mathbf{R_f}\left(\mathbf{R_f}^{\mathrm{T}}\mathbf{R_f}\right)^{-1}, \tag{19}$$

where $\mathbf{R}_{\mathrm{f}}$ is an IDM matrix connecting the loads acting on the DoFs belonging to the structure to the loads acting on the VP. Transformation matrices $\mathbf{T}_{\mathrm{u}}$ and $\mathbf{T}_{\mathrm{f}}$ can be used to transform the admittance matrix containing the interface DoFs into an admittance matrix containing VP DoFs, using the following equation [47]:

$$\mathbf{Y}_{\mathrm{VPT}} = \mathbf{T}_{\mathrm{u}}\ \mathbf{Y}\ \mathbf{T}_{\mathrm{f}}^{\mathrm{T}}. \tag{20}$$

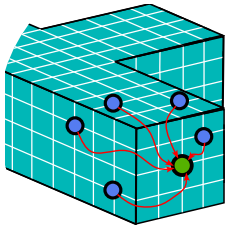

Fig. 6. Virtual point transformation. (•) DoFs to be used in VPT, (•) VP.

Performing VPT on the DoFs in the vicinity of the interface of hybrid substructures, corresponding admittance matrices $\mathbf{Y}^{(i)}_{\mathrm{VPT}}$ are obtained. These contain DoFs collocated with those of the corresponding mass point of the joint model. Thus, the joint model can

[5] For the purpose of a better fit, an over-determination is recommended (number of DoFs to be transformed into VP n greater than the number of VP DoFs m).

[6] IDM matrix $\mathbf{R}$ contains a system of equations connecting the translations of each DoF in the vicinity of the interface to the chosen set of DoFs of the VP; these are usually translations and rotations (rigid IDMs), but extension, torsion, skewing, and bending modes (flexible IDMs) can also be included [48].

be coupled to both adjacent hybrid substructure models simultaneously (Fig. 7), using the LM FBS Eq. (6). In the described case, the uncoupled admittance matrix in the LM FBS Eq. (6) takes the following form:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}^{(\mathrm{A})}_{\mathrm{VPT},ii} & \mathbf{Y}^{(\mathrm{A})}_{\mathrm{VPT},ib} & 0 & 0 & 0 & 0 \\ \mathbf{Y}^{(\mathrm{A})}_{\mathrm{VPT},bi} & \mathbf{Y}^{(\mathrm{A})}_{\mathrm{VPT},bb} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{Y}^{(\mathrm{J})}_{m_1 m_1} & \mathbf{Y}^{(\mathrm{J})}_{m_1 m_2} & 0 & 0 \\ 0 & 0 & \mathbf{Y}^{(\mathrm{J})}_{m_2 m_1} & \mathbf{Y}^{(\mathrm{J})}_{m_2 m_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{Y}^{(\mathrm{B})}_{\mathrm{VPT},ii} & \mathbf{Y}^{(\mathrm{B})}_{\mathrm{VPT},ib} \\ 0 & 0 & 0 & 0 & \mathbf{Y}^{(\mathrm{B})}_{\mathrm{VPT},bi} & \mathbf{Y}^{(\mathrm{B})}_{\mathrm{VPT},bb} \end{bmatrix}, \tag{21}$$

and the Boolean matrices $\mathbf{B}_c$ and $\mathbf{B}_e$, connecting matching pairs of rows and columns of $\mathbf{Y}$, respectively, are shaped as follows:

$$\mathbf{B}_c = \begin{bmatrix} \mathbf{0} & \mathbf{I} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & -\mathbf{I} \end{bmatrix}; \quad \mathbf{B}_e = \begin{bmatrix} \mathbf{0} & \mathbf{I} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & -\mathbf{I} \end{bmatrix}. \tag{22}$$

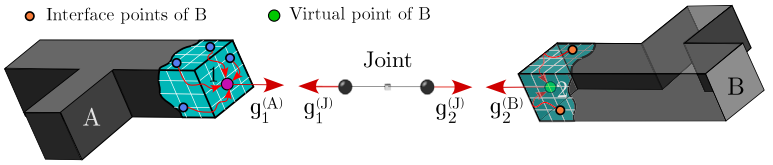

Fig. 7. Dynamic coupling of the VP of substructure A to the first point of mass of the joint model and the VP of substructure B to the second point of mass of the joint model. (•) DoFs of A to be used in VPT, (•) VP of A, (•) DoFs of B to be used in VPT, (•) VP of B.

3.4. Generation of training data set

Hybrid substructure models are coupled to the n differently damaged joint models (Fig. 8) using the LM FBS equation (5), with n denoting the number of samples contained in the data set. The models of the substructures stay the same throughout the entire process, making the proposed method computationally efficient.

For supervised machine learning it is mandatory to keep a record of the information about the damage corresponding to each generated sample (e.g., location, type and extent of the joint damage). When describing the joint's damage, the damage level can be treated as a continuous variable (numerical value of the damage factor β) or a categorical variable (values of β can be separated into intervals of, e.g., a 10% stiffness drop, each interval representing an individual output class). On the other hand, the location of the damage is limited to the finite number of joints, making the location a categorical variable.

The resulting hybrid data set can be used to train various machine learning models, which can then be used for damage identification based on the dynamic properties of the structure, which is demonstrated in the following.

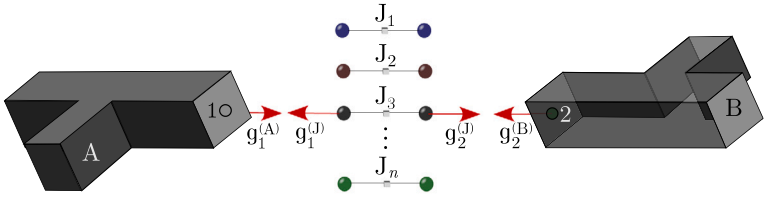

Fig. 8. Dynamic coupling of substructures A and B with n different joint models ($J_1, \ldots, J_n$).

4. Case study

4.1. Structure of interest

The observed structure consisted of four substructures (Fig. 9). The individual substructures had either shape A or B, each of two shapes appearing twice in the entire construction. Substructure A was followed by substructure B, B was followed by A, rotated by 90° counter-clockwise, followed by substructure B, rotated by 90° counter-clockwise. Each pair of substructures was connected with a model of joint J_i for $i = 1 \ldots 3$, as shown in Fig. 9.

Physical substructure models consisted of multiple sub-components, welded together, while numerical models were modeled as single-piece objects.

General properties of substructures including dimensions, mass of the physical and corresponding numerical model (mass of the sensors included) as well as the hysteretic damping factors η for the first five modes, following the rigid body modes (RBMs) are presented in the Table 1. A negligible damping ($\eta_{\mathrm{RBM}} \approx 0$) is assumed for RBMs and a hysteretic damping factor of $\eta = 0.003$ for the remaining 45 modes, accounted for in the modal superposition when calculating FRFs.

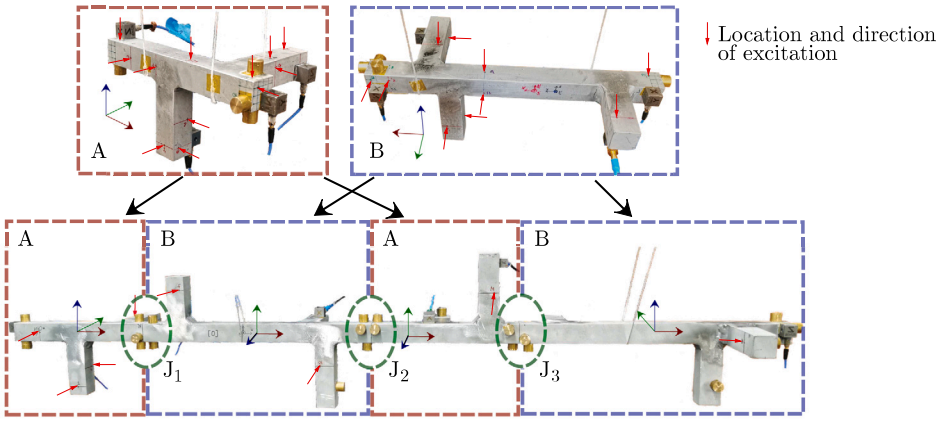

Fig. 9. Experimental structure consisting of substructures A and B (each appearing twice), connected over joints J_1, J_2, and J_3.

Table 1
Properties of substructures A and B.

	Dimensions [mm]	m_{exp} [g]	m_{num} [g]	η_1	η_2	η_3	η_4	η_5
Substructure A	$150 \times 90 \times 80$	361	363	0.0263	0.0129	0.0241	0.0124	0.0192
Substructure B	$250 \times 140 \times 80$	537	529	0.0104	0.0073	0.0132	0.0216	0.0051

4.2. Experimental setup

For the purpose of generating a data set, first, the experimental response models of parts A and B (Fig. 9) were obtained. The excitation force was applied to structure A at 19 impact points, using a modal hammer (PCB 086E80), and the structure's response was measured at 11 response points, using triaxial accelerometers (PCB 356A32 and Dytran 3273A2; Fig. 10). Measurements were used to estimate the corresponding FRFs, resulting in the admittance matrix $\mathbf{Y}_{exp}^{(A)} \in \mathbb{C}^{33\times19}$. On the other hand, structure B was excited at 18 points and responses were measured at 11 points, resulting in the matrix $\mathbf{Y}_{exp}^{(B)} \in \mathbb{C}^{33\times18}$. Excitation points were homogeneously distributed over individual structures (Fig. 9). During the measurement, both substructures were suspended on strings, simulating free boundary conditions (Fig. 10). Due to the large number of response DoFs, we were unable to measure all the response points simultaneously. For this reason measurements were conducted according to the roving sensor approach, i.e. in multiple rounds, between which the accelerometers were moved between different observed locations. To account for the changes in structure's mass distribution due to the accelerometer relocation, the (gold-colored) brass masses were attached to the locations without accelerometers in the given measurement sequence. Thus, the dynamic model of the structure remained consistent throughout entire measurement process. Some response points observed on substructures A and B were located at the surfaces to which an adjacent substructure was attached after assembling into the system ABAB (Fig. 9). The mass of accelerometers at these locations was not accounted for in the assembled system, as mass of the assembly was considerably higher then that of the missing sensors ($m_{ABAB} \approx 1760$ g compared to the mass of 6 missing sensors with the mass of an individual accelerometer $m_{acc} \approx 5.5$ g).

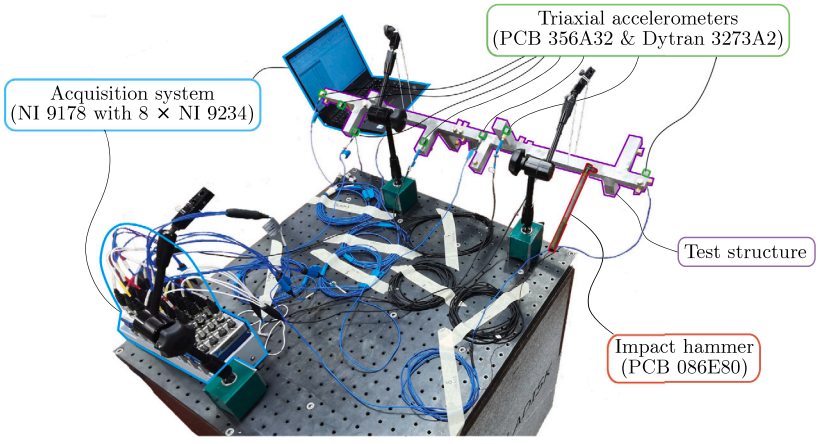

Fig. 10. Experimental setup.

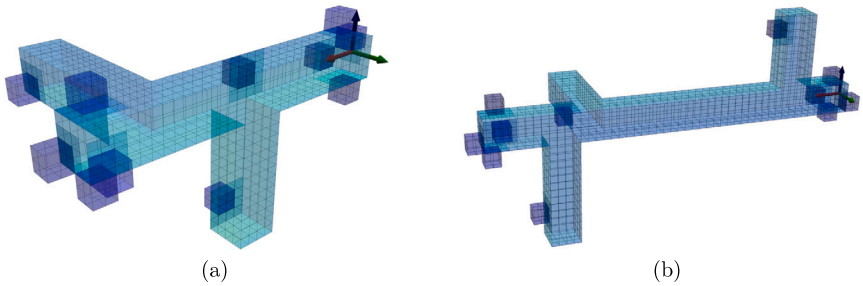

Fig. 11. Numerical models of (a) substructure A and (b) substructure B.

4.3. Data set generation

First, the experimental response models of parts A and B (Fig. 9) were obtained. Besides the experimental model, the numerical models are required for the SEMM method. The dynamic properties of the numerical models of the substructures A and B (Fig. 11) were obtained with the modal analysis performed using the FE method. As the models resulting from the FE analysis contain large numbers of DoFs, to avoid a great deal of computational complexity, the frequency response functions were calculated only for the limited set of DoFs of the numerical models. The chosen response DoFs matched the response DoFs of the experimental models, but the excitation DoFs were extended to 47 DoFs for substructure A and to 46 DoFs for substructure B, resulting in the admittance matrices $\mathbf{Y}_{\text{num}}^{(\text{A})} \in \mathbb{C}^{33\times 47}$ and $\mathbf{Y}_{\text{num}}^{(\text{B})} \in \mathbb{C}^{33\times 46}$, respectively.

The FRFs obtained with the experiment contained a substantial level of noise in a lower frequency range. For this reason they were blended with numerical results at lower frequencies, using the trust function as proposed in Klaassen et al. [42] to obtain the combined admittance $\mathbf{Y}_{\text{tf}}$. For a smoother transition between the numerical and experimental FRFs, the trust function was assigned the shape of the sigmoid function (Fig. 12). Fig. 12 also shows some discrepancies between the experimental and numerical models in the higher frequency range, which can be attributed to the modeling of numerical models of components as single-piece objects, while the sub-components of the physical model are in fact welded together.

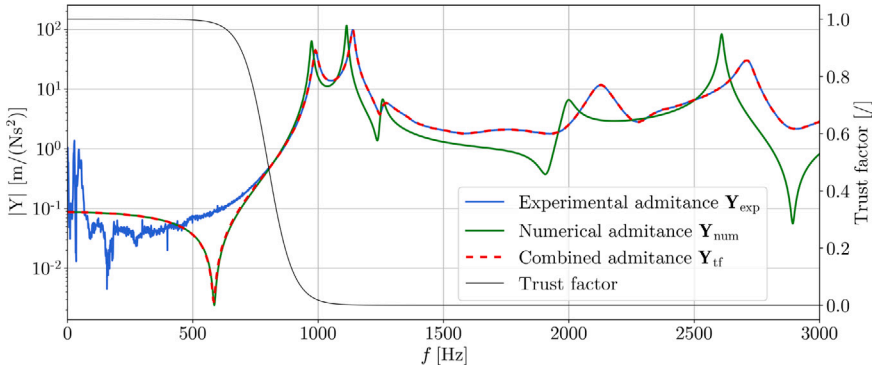

Fig. 12. Mixing of numerical and experimental FRFs of substructure B, using a trust function.

Combined admittances $\mathbf{Y}_{\text{tf}}^{(\text{A})}$ and $\mathbf{Y}_{\text{tf}}^{(\text{B})}$ of substructures A and B, respectively, were used as overlay models in the SEMM method, while the parent and removed models were based on numerical models $\mathbf{Y}_{\text{num}}^{(\text{A})}$ and $\mathbf{Y}_{\text{num}}^{(\text{B})}$. To obtain the hybrid models $\mathbf{Y}_{\text{SEMM}}^{(\text{A})}$ and $\mathbf{Y}_{\text{SEMM}}^{(\text{B})}$, a fully-extended version of the SEMM method (Eq. (11)) was used, meaning that parent models, as well as removed models, were described by the entire admittance matrix $\mathbf{Y}_{\text{num}}^{(\text{A})}$ for substructure A and the entire admittance matrix $\mathbf{Y}_{\text{num}}^{(\text{B})}$ for substructure B. The resulting hybrid models preserved the properties of the real structure, while extending them to all the DoFs of the parent models (Fig. 13). There is still a high degree of similarity between the FRFs of the hybrid models and the FRFs of the numerical models, but the locations of the resonance peaks were shifted towards the natural frequencies of the experimental model. It should also be noted that the damping properties were affected by the SEMM method.

To achieve collocation at the interface DoFs of the each substructure model with those of the adjacent joint models, required for dynamic coupling, the VPT was performed. The experimental models of substructures A and B contained four interface response points (12 DoFs) and four excitation points (4 DoFs), which was not sufficient for a successful VPT. Using the SEMM method, interface excitation DoFs were extended to 12 DoFs, which was sufficient to transform the interface response DoFs as well as the excitation DoFs into a VP with six excitation DoFs and six response DoFs using Eq. (20). By performing VPT (using an implementation in the Python package *pyFBS* [49]) at all joint-adjacent substructure interfaces, the final substructure models, which could be dynamically coupled to the different joint models, were obtained.

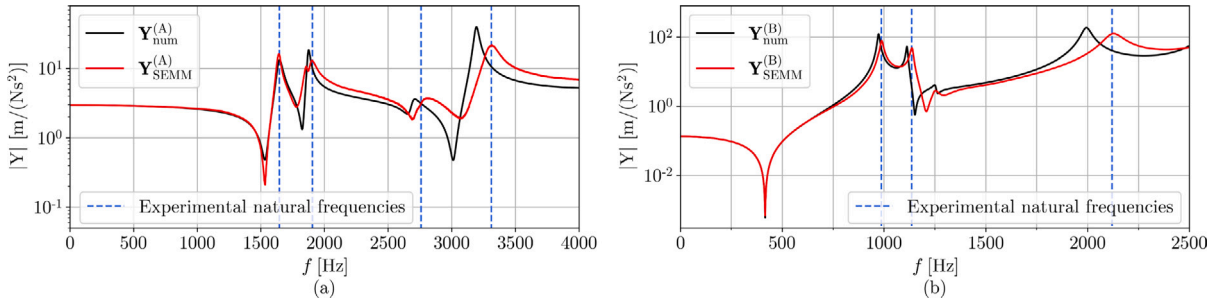

Fig. 13. FRF of a hybrid model ($\mathbf{Y}_{SEMM}$) of (a) substructure A and (b) substructure B, compared to the corresponding FRF of the numerical model ($\mathbf{Y}_{num}$).

Next, the joint models were built in a manner described in Section 3.2. First, the parameters of the undamaged joints were obtained by manually fitting the elements of the mass matrix $\mathbf{M}^{(J)}$, the damping matrix $\mathbf{C}^{(J)}$, and the stiffness matrix $\mathbf{K}^{(J)}_{udmg}$ so that the response of the coupled system could fit the measured assembly response. Joint dimensions as well as joint mass were assumed negligible compared to the size/mass of the system. For the undamaged joint, elements of damping matrix were assigned the value 0.003 and the elements of stiffness matrix were assigned value 8×10^{12} N/m for translational DoFs (k_x, k_y and k_z) and value 8×10^{10} N m/rad for rotational DoFs (k_{tx}, k_{ty} and k_{tz}). These joint parameters were inserted into Eq. (14) to obtain the admittance matrix of the undamaged joint. The models of the undamaged joints ($\mathbf{Y}^{(J_1)}$, $\mathbf{Y}^{(J_2)}$ and $\mathbf{Y}^{(J_3)}$) were used in the dynamic coupling process, resulting in the model of the complete, undamaged structure. The results of the dynamic coupling are compared to the FRF measured in the assembly in Fig. 14. Slight shifts in the natural frequencies can be observed, due to discrepancies between the physical models of the substructures and the physical model of the entire structure, which can be, to a large extent, assigned to the differences in the welds binding the sub-components of each structure. On the other hand, it can be seen that the damping properties of the coupled model fit well with those of the physical model of the complete system.

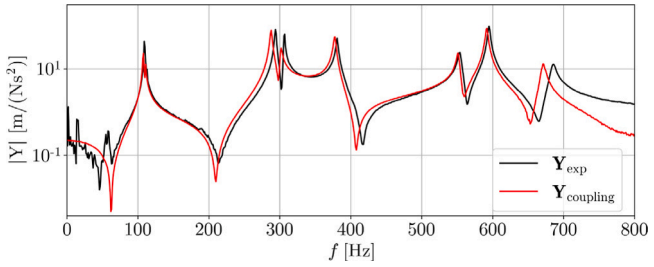

Fig. 14. FRF obtained by dynamic coupling ($\mathbf{Y}_{coupling}$) compared to the measured assembly FRF ($\mathbf{Y}_{exp}$).

Having obtained the undamaged joint parameters, it was possible to determine the damage factors β for the damaged joints. The same damage factors were assumed for the entire $\mathbf{K}^{(J)}_{dmg}$ matrix of the damaged joint ($\beta_1 = \beta_2 = \beta_3 = \cdots = \beta_6 = \beta$), meaning that, for the ith sample of the data set, Eq. (15) can be simplified to:

$$\mathbf{K}^{(J)}_{dmg,i} = (1-\beta_i)\mathbf{K}^{(J)}_{udmg}. \tag{23}$$

Thus, each damaged joint was labeled with its corresponding factor β_i and the location of the damaged joint on the structure (J_1, J_2 or J_3; Fig. 9). In this case the damage identification was considered a classification problem; therefore, the damage factors β were arranged into six classes, the first one being the class of the undamaged joint ($\beta = 0$) and the following classes representing an interval of 10% increase in β factor each (0%–10%, 10%–20%, 20%–30%, 30%–40%, and 40%–50%). Each damage class could occur at one of three joint locations, which meant that there were 16 states of the structure's health altogether (5 damage levels at 3 different locations and a class for undamaged structures). Altogether, 10,008 joint models were generated using the equation of motion (14).

In the final phase of synthetic data set generation, all the joint models were coupled with the hybrid substructure models, using the LM FBS Eq. (6) in such a way that all possible combinations of damage location (either joint J_1, J_2, or J_3) and damage level (β from 0 to 50%) were evenly distributed across all 10,008 samples.

4.4. Data analysis and feature extraction

To efficiently use generated data in ML model training, the features of the data that contain information about joint damage had to be identified. Fig. 15 shows how the damage to one of the joints (damage factor $\beta = 0.45$, alternately at joint J_1, J_2, or J_3, while the other two joints remain intact) affects the dynamic response of the whole structure. The frequency region of interest was limited to the interval from 1 to 800 Hz, as a great deal of assembly dynamics is described in this region. It can be observed that

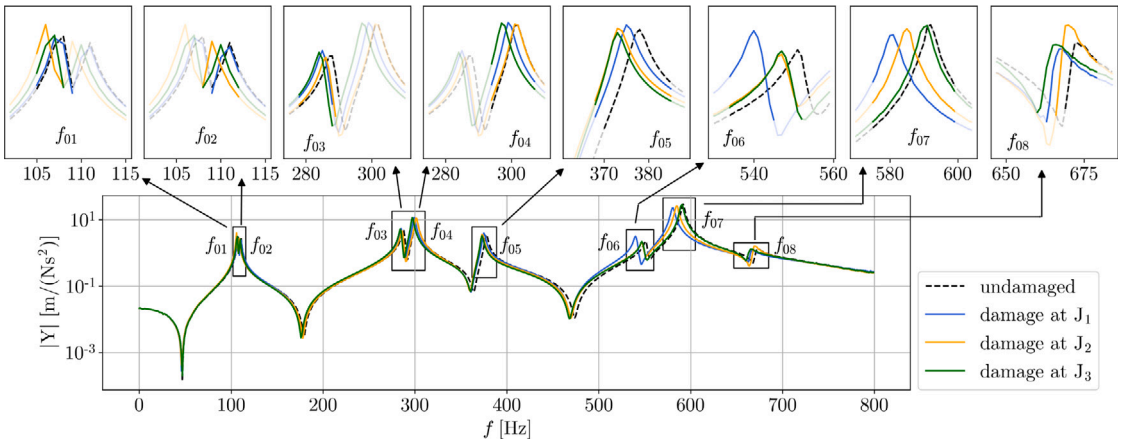

Fig. 15. Effect of location of the damaged joint ($\beta = 0.45$) on a structure's dynamic response. Changes are most visible in the vicinity of the resonance peaks corresponding to natural frequencies f_{0i}.

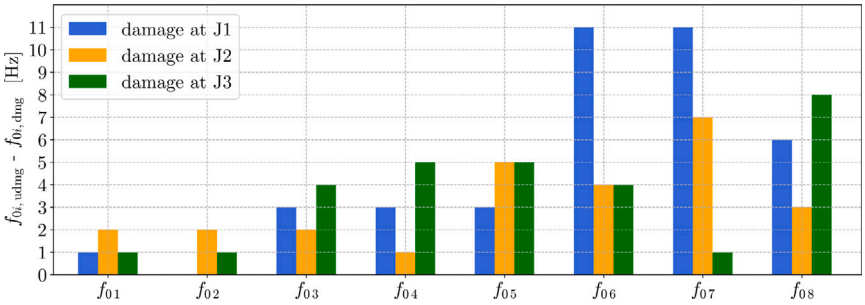

Fig. 16. Difference between natural frequencies of the undamaged structure $f_{0i,\mathrm{udmg}}$ and natural frequencies of the damaged structure $f_{0i,\mathrm{dmg}}$ ($f_{0i,\mathrm{udmg}} - f_{0i,\mathrm{dmg}}$ for $i = 1, \ldots, 8$) depending on the location of the damaged joint (damage factor $\beta = 0.45$ at joint J_1, J_2, or J_3), while the other two joints remain undamaged.

the different damage locations result in different shifts in the individual natural frequencies. The size of these shifts is presented in the bar chart in Fig. 16. The damage at joint J_1 causes the largest decrease in the sixth and seventh natural frequencies (f_{06} and f_{07}, respectively), the damage at joint J_2 causes a significant decrease in the fifth and seventh natural frequencies, while the decrease in the sixth natural frequency is smaller than in the case of the damage at J_1. The damage at joint J_3 results in the greatest decrease in the eighth natural frequency and a negligible drop in the seventh natural frequency. Besides the decrease in natural frequencies, changes in the magnitudes of the resonance peaks are also observed, which means that joint damage also affects the mode shapes of the structure. For this reason, using a whole admittance matrix instead of only a single FRF can improve the success of the ML model at identifying a joint's damage. These observations prove that the dynamic properties of the structure contain information about the location of the damaged joint.

On the other hand, the degree to which an individual joint was damaged could be determined by how much the individual natural frequencies had decreased — the greater the damage the greater the decrease in the natural frequencies. Fig. 17 shows the changes in the natural frequencies f_{06} to f_{08} for different values of the damage factor $\beta = [0,\ 0.5]$ when only joint J_1 is damaged. The values of the natural frequencies start decreasing faster as β approaches higher values (close to 0.5). Another important thing to consider is the frequency resolution. In Fig. 17c we see that at a frequency resolution of 1 Hz the eighth natural frequency does not change in the interval $\beta = [0.014,\ 0.235]$, which can make the damage recognition in this area difficult. This limitation is especially discernible in a lower frequency range where the natural frequencies decrease only by 1 or 2 Hz, even for high β values around $\beta = 0.5$ (Fig. 16). Consequently, the data in this area provided no accurate information about the structure's health.

The presented findings can be employed for an efficient feature extraction. Each sample of the data, obtained from the coupling process, contained 6900 frequency response functions (60 output DoFs × 115 input DoFs) between the internal DoFs of hybrid models of substructures A and B. Each FRF has 800 elements (frequency lines), resulting in altogether 5.52×10^6 features. Such a large number of features can make the training process for the ML model time consuming and inefficient [50]. For this reason, the dimensionality of the training data had to be reduced as much as possible with the minimum loss of useful information, while the data processing should be as quick and simple as possible.

As the natural frequencies contained a great deal of information about the level as well as about the location of the damage, there was no need for such a large number of FRFs (6900) used for each sample. Therefore, five FRFs were chosen in which the resonance peaks in the frequency range of interest were clearly visible.

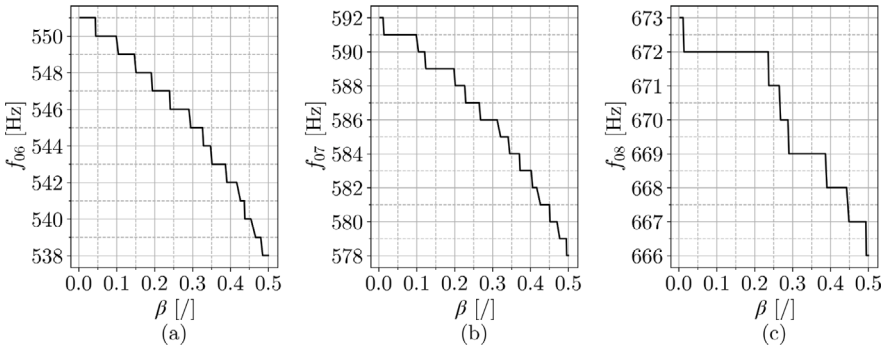

Fig. 17. Effect of the damage factor β to (a) 6th, (b) 7th and (c) 8th natural frequency of the structure (damage on the joint J_1).

Considering that the frequencies f_{01} to f_{04} exhibited only a slight decrease at a relatively high β value (Fig. 16), the frequency range of interest was limited to the interval between 360 Hz and 700 Hz. In this way, the data set was reduced by about half, but the number of features was still large (340 frequency lines × 5 FRFs per sample). For this reason, further data reduction was performed using principal component analysis (PCA), which has been widely employed for SHM applications [51,52]. PCA allows us to map the data into a lower dimension space, during which we discard linear combinations of input data with negligible contribution to the overall variance [30]. This means that the features (or their linear combinations) holding most of the information are kept within the data set with lower dimensionality from original. According to the previous findings, highest variances are expected at frequency lines in the proximity of resonance peaks (Fig. 15).

The number of principal components used determines the amount of information kept. On the other hand, a larger number of used principal components results in a higher number of data set features, and consequently in an inefficient machine learning process. By keeping the first 40 principal components (of 1700 available) the data set was sufficiently reduced and contained the information necessary for training the ML algorithm for the joint-damage identification.

As the matrix **A** contains complex values, each component of each sample will be separated into its complex and imaginary parts, which will then be considered as two distinct features, resulting in $2r$ features per sample.

4.5. Training and testing of ML model

The generated data set was used to train the Support Vector Machine (SVM)[7] classifier, which is originally a binary classification algorithm, but can be extended to a multi-class classification. SVM algorithm has several advantages compared to the other classical or deep learning algorithms for which it was utilized rather than some other ML approach. Compared to the Linear Discriminant Analysis, SVM allows non-linear boundaries between classes (through the kernel trick) [53]. Compared to the ML methods which strongly rely on statistical properties of the training data (e.g. Linear Discriminant Analysis, Quadratic Discriminant Analysis, Logistic Regression), SVM is less affected by outliers, as decision boundary in SVM only depends on the so-called support vectors (i.e. training set samples closest to the decision boundary) [54]. Compared to the neural networks (including deep learning algorithms), SVM has stronger generalization capability when limited amount of data is available [55]. In addition to these advantages, SVM also exhibits high performance on high-dimensional data [53]. In this example an implementation of the SVM in the *scikit-learn* module [56], a machine learning library in the Python programming language, was used.

The training data set was fed to the SVM algorithm, where the multi-class problem was handled according to the *one-versus-one* approach.[8] Training multiple ML models with different settings (kernel type, kernel parameters and value of tuning parameter C), and estimating the accuracy of each model based on a 10-fold cross-validation (CV) [53], a polynomial kernel of order six and the value of the tuning parameter $C = 5$ were chosen. The accuracy of the ML model was estimated in the form of the average F_1-score[9] of 10 folds. For the final model, the accuracy of F1 = 1.0 was estimated (with corresponding precision=1.0 and recall=1.0). An F_1 score of 1 indicates that all the test samples were correctly classified, which means that the model was properly trained, i.e., it was able to predict the joint damage from the input data that were not used to train the model.

However, testing the model on the test data generated with the same algorithm as the training data (resulting from the proposed approach to data set generation) is not sufficient proof of the model's accuracy. Therefore, additional testing of the trained SVM model was required. For this purpose, an additional testing data set consisting of 11 differently damaged numerical models of the whole structure was built by performing numerical simulations (FE method). These models were inserted into the SVM model to

[7] The interested reader is referred to Appendix where the SVM algorithm is further elaborated.

[8] In *one-versus-one* classification $\binom{k}{2}$ models are trained, each comparing two from the whole set of k classes. The testing sample is tested on all of the trained models and then assigned to the most frequently predicted class [53].

[9] F_1-score is defined as $(2 \cdot precision \cdot recall)/(precision + recall)$, where precision is a number of true positives divided by number of all predicted positives (true positives + false positives), and recall is a number of true positives divided by all actual positives (true positives + false negatives) [57].

obtain predictions about the structure's health for each model. In addition, the SVM model was also tested on the data obtained from measurements on several differently damaged physical structures. Note that the ML model has not been changed after being evaluated using CV; therefore, these numerical models and experimental samples serve us as a validation set for the generated ML model. Due to the smaller number of these validation models, prediction for each sample will be interpreted separately instead of calculating overall evaluation metrics.

4.6. ML model verification on numerical models

The numerical models for the validation of the trained ML model were modeled as FE models of substructures A and B, connected over bushing joints J_1, J_2, and J_3 (Fig. 18). The properties of each joint were determined by its stiffness matrix, which consists of six parameters (3 translational and 3 rotational stiffness parameters). By changing the joint parameters and performing the FE analysis, 11 models of the structure with different locations and degrees of joint damage were obtained.

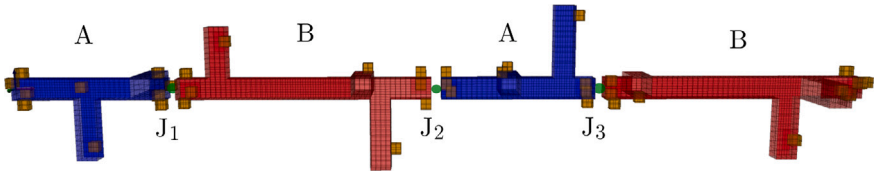

Fig. 18. Numerical model of the complete structure.

The admittance matrix of each model was processed in the same way as the training data (limiting the frequency range, PCA, see Section 4.4) and then used as an input to the trained SVM model, which made the predictions, shown in Table 2. The results show that the model had problems identifying locations for damage scenarios with a low damage factor β of 25% and below, as the damage at joint J_3 was predicted for the vast majority of those samples. However, at the highest damage level, the ML model is capable of predicting an accurate damage location. Regarding the damage level, in six out of 11 cases the predictions are accurate and in four cases the predicted damage level deviated from the reference only for a class (samples 1, 4, 7, and 8). The only sample with a larger discrepancy is sample 10 for which the damage level was underestimated for two classes. This can be explained by Fig. 19 in which it is clear that at $\beta = 0.25$, the sixth natural frequency as well as the seventh natural frequency decreased by only 1 Hz from the undamaged state (Figs. 19a and 19b). Accordingly, it is difficult for the ML model to distinguish between the lower damage levels at the joint J_3 at the limited frequency resolution of 1 Hz, despite a slightly higher decrease in the eighth natural frequency (3 Hz at $\beta = 0.25$; Fig. 19c). In general some errors in the predictions can also be ascribed to the simplified numerical models (individual parts of the system modeled as single-piece structures), while the training samples on which the SVM model was trained exhibit properties of the real structure, consisting of multiple components welded together.

Table 2
The results of the damaged-joint prediction on numerical models of damaged structures using SVM classifier.

Sample no.	1		2		3		4		5		6	
	loc.	β [%]	loc.	β [%]	loc.	β [%]	loc.	β [%]	loc.	β [%]	loc.	β [%]
Reference	/	0	J_1	5	J_1	15	J_1	25	J_1	45	J_2	5
Prediction	J_3	0–10	J_3	0–10	J_3	10–20	J_2	30–40	J_1	40–50	J_3	0–10
Sample no.	7		8		9		10		11			
	loc.	β [%]	loc.	β [%]	loc.	β [%]	loc.	β [%]	loc.	β [%]		
Reference	J_2	25	J_2	45	J_3	5	J_3	25	J_3	45		
Prediction	J_3	10–20	J_2	30–40	J_3	0–10	J_3	0–10	J_3	40–50		

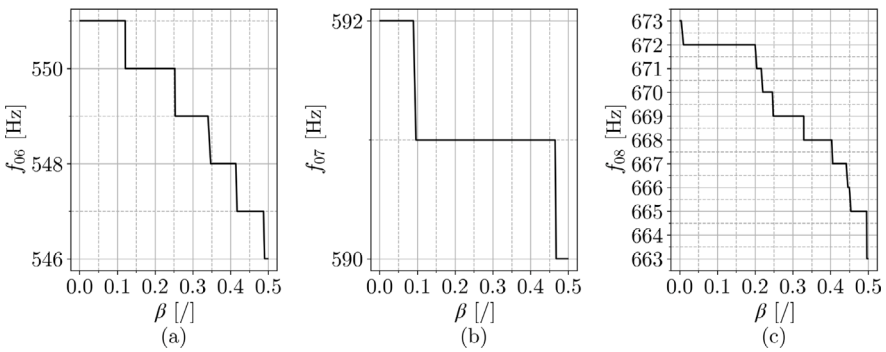

Fig. 19. Effect of the damage factor β to (a) 6th, (b) 7th and (c) 8th natural frequency of the structure (damage on the joint J_3).

4.7. ML model validation on experimental data

Finally, the trained SVM model was also validated on the physical structures. Four differently damaged experimental samples were prepared: an undamaged structure, a structure damaged at joint J_1 cut to one-quarter of the beam height (cut depth of 5 mm), a structure damaged at joint J_1 cut to one-half of the beam height (cut depth of 10 mm), and a structure damaged at joint J_2 cut to one-half of the beam height (cut depth of 10 mm) (Fig. 20). Each of these samples is an individual piece with a single-piece beam at the main axis which was then cut at the joint location, to simulate damage to the structure.

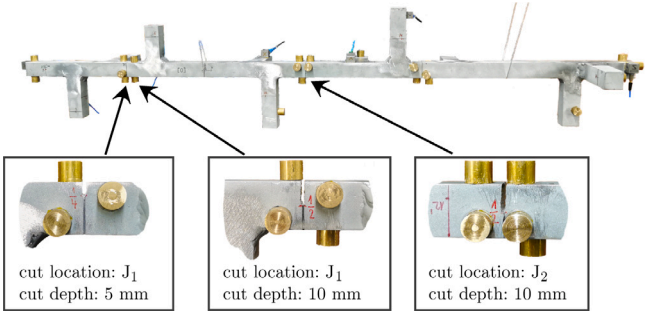

Fig. 20. Damage scenarios on experimental samples.

On each sample a series of measurements was performed to obtain five FRFs corresponding to the FRFs used in the training set (in terms of DoFs, chosen during feature extraction process; Section 4.4).

These FRFs were processed in the same way as the training data (limiting frequency range and PCA) and inserted into the trained SVM model to obtain predictions about the damage to each model (level and location). The predictions exhibited an unsatisfactory accuracy, which can be explained by comparing the FRF of the undamaged structure from the training set (obtained using the proposed approach) with the FRF from numerical simulations (used for validation of the SVM model in Section 4.6) and the experimental FRF of the undamaged structure (Fig. 21). We can see that the numerical FRF corresponds very well with the one of the training sample's FRF. On the other hand, the resonance peaks of the experimental model deviate slightly more from the ones of the training set's FRF. In case these discrepancies would be a consequence of inaccuracies in the hybrid models, they could be decreased by increasing number of measurement points (excitation points as well as response points). This would contribute to improved controllability[10] and observability[11] of experimental models and thus increase robustness of the proposed hybrid approach as well as decrease discrepancies between the hybrid and physical model. However, despite the presence of these discrepancies to some extent, they do not appear to be critical for the efficiency of the proposed methodology in this specific case. In the observed case, the discrepancies, most likely, originate from the differences in the welds between the experimental model of the whole structure and the models used in the process of hybrid data set generation. Due to these deviations, data reduction using PCA could eliminate features containing information about the experimental natural frequencies (e.g., the resonance peak at 686 Hz). Besides that, due to the resonance frequency shifts, the individual features of experimental samples can exhibit considerable discrepancies compared to the values of individual features of the training samples with the same damage properties. For these reasons, the ML model returned inaccurate predictions.

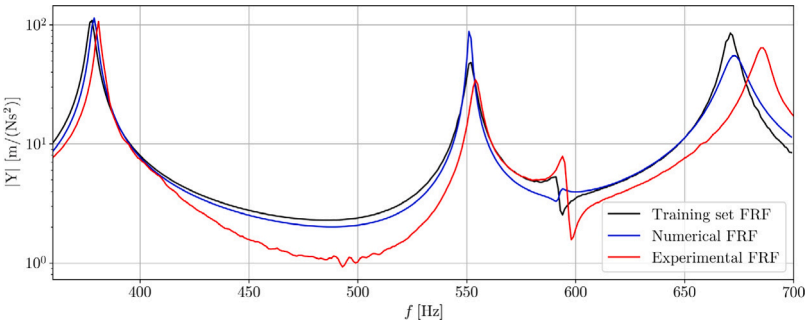

Fig. 21. Training test sample FRF of the undamaged structure compared to the experimentally obtained FRF and the numerically obtained FRF.

In order to obtain an ML model capable of identifying damage on experimental structures, despite the aforementioned discrepancies, an alternative approach to the feature extraction was explored. We performed modal identification, focusing on extraction of natural frequencies, which presents usually used data for long-term SHM [5,10]. Suitability of natural frequencies

[10] Vibration modes excited with impact hammer.

[11] Vibration modes that can be observed at measured response points.

for the purpose of damage identification originates from their tight connection to the physical properties of the structure, affected by the damage (i.e. stiffness and mass) [30]. Natural frequencies within the frequency range 360 Hz to 700 Hz were used as features for the new data set. The drawback, due to which this approach was not used from the beginning, is that in some cases, especially for structures with a high modal density, it can be difficult to automate the process of extracting the natural frequencies. On the other hand, the advantage of using natural frequencies in ML training when dealing with the experimental data is that the natural frequencies exhibit low deviations due to experimental errors (e.g., imperfect impact location or direction); there might be a substantial change in amplitude, but the locations of the resonance peaks remain unchanged. For these reasons, the resonance frequencies were identified for all 10,008 samples to build a new data set.

The new data set was used to train a new SVM model with a linear kernel and tuning parameter C=1. To test the new ML model, the 10-fold CV was performed, resulting in an average F_1 score of F_1=0.883 (with corresponding precision=0.922 and recall=0.872). A somewhat lower F_1 score can be assigned to the substantial reduction of the data set (from 1700 features for 5 FRFs with 340 frequency lines to only 8 features), in which some information was lost. Finally, the new SVM model was validated on data from measurements on physical structures. The results are shown in Table 3. These results can be interpreted by comparing the dynamics of experimental samples to the dynamics of the training data samples. As the damage to the joints J_1 and J_2 affects 6th and 7th natural frequency most significantly (Fig. 16), the differences between these natural frequencies are displayed in Fig. 22. It can be noticed that 6th natural frequency of the undamaged experimental sample and of the experimental sample with 5 mm cut as joint J_1 were even slightly higher then the one of the training sample, which explains why both samples were categorized as undamaged. The sample with deeper cut at joint J_1 causes expected decrease in both 6th and 7th natural frequency, which results in correctly predicted damage location. On the other hand, the sample with 10 mm cut at the joint J_2 results in expected decrease of 7th natural frequency, while the 6th natural frequency decreased only slightly compared to the undamaged experimental sample, which is a reason for the lower predicted damage level.

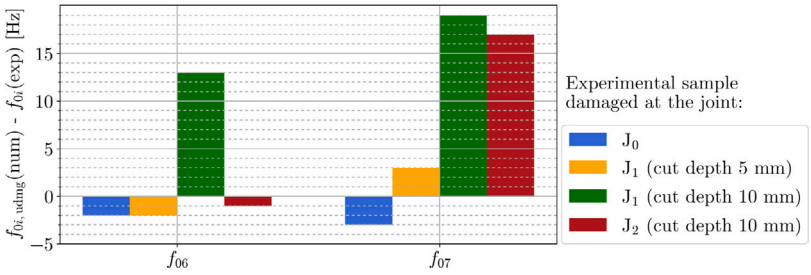

Fig. 22. Difference between the 6th and 7th natural frequencies of the undamaged structure of the training data and of the experimental samples with different locations and levels of the joint damage.

When analyzing the results, it should be taken into account that the data set samples used for training the ML model only included damage scenarios with the same damage factor β for all the stiffness parameters of the joint (the damage affects structure's properties the same in all directions). Yet the damage scenarios of real samples are more complex as the damage affects the properties of the complete structure in each direction differently. However, the proposed approach allows us to determine the damage factors β_i for each stiffness parameter separately and thus to describe the joint damage more accurately. Such an extension of the training set could improve the performance of the ML model in identifying damage location and severity, but also extend the model's ability to identify different types of joint damage. On the other hand, in this case, an ML model should be much more complex, and a larger data set would be required to successfully train the model.

Table 3
Results of joint damage prediction on experimental models of damaged structures using SVM classifier.

Sample no.	1		2		3		4	
	loc.	cut depth [mm]	loc.	cut depth [mm]	loc.	cut depth [mm]	loc.	cut depth [mm]
Reference	/	0	J_1	5	J_1	10	J_2	10
	loc.	β [%]	loc.	β [%]	loc.	β [%]	loc.	β [%]
Prediction	/	0	/	0	J_1	40–50	J_2	20–30

5. Conclusions

In this paper, a novel approach to generating a synthetic training data set for the purpose of SHM is proposed. The proposed approach is focused on identification of damage in joint sections of assembly products. The assembly is analyzed based on the properties of its substructures. Individual parts of the system are modeled as hybrid models obtained using the SEMM method, which allows mixing of numerical and experimental models of substructures. This way, properties of the real structure are extended to the high spatial density of the numerical model, which is necessary for the dynamic coupling of substructure models with joint models. Accuracy of the SEMM method is therefore a key prerequisite to obtain accurate substructure models. As the SEMM method has been successfully applied to numerous applications in literature [58–60], we believe that the proposed approach of generating

a training set for joint damage identification can be applied to any system for which SEMM method can be applied to obtain hybrid models of substructures.

Damage to the analytically modeled joints between the individual substructures is simulated by changing the model's stiffness properties. By coupling the hybrid substructure models with different joint models, an ML training set is obtained, comprised of differently damaged samples. As the approach is focused on identifying the damage in joints, the hybrid part models can be built only once for the entire training set, which makes the proposed approach computationally efficient.

The proposed approach was used to build an ML training set for joint-damage identification on a laboratory test structure. Using generated data, feature extraction was performed (limiting the frequency range and PCA) and an SVM model was trained. The resulting ML model was tested on both numerical models and physical models of the whole structure. Regarding the numerical models, the ML algorithm successfully identified damage level for most of the samples and location of the damage for the samples with higher levels of damage.

Testing on experimentally obtained data revealed that the differences between the measured data and the training-set data were too distinct to allow successful damage detection using the same feature-extraction approach. Therefore, a different approach to feature extraction was taken (still using the data set obtained using the proposed approach). The natural frequencies of each sample were identified and used to train a new SVM model. The new ML model was able to correctly recognize the undamaged sample, as well as location and increased damage level on heavily damaged samples. Nevertheless, the ML model was having difficulties in detecting the damage for the slightly damaged locations.

One of the key limitations of this study is that the proposed method was only tested on a laboratory structure. Testing on some real world systems would be beneficial to confirm the applicability of the proposed approach to the real-world problems.

One of the obstacles of the proposed approach is the definition of the undamaged state. Using this approach the undamaged state is determined based on the response of some sample structure for which we assume that it is undamaged (e.g. because it has just been manufactured and has not yet been exposed to any operational loads). However, due to some uncertainties in material properties and inevitable variations in manufacturing process, some discrepancies in dynamic properties between individual undamaged structures may occur. The proposed approach does not account for these variations.

The proposed approach is also limited by the assumption that only damage contributes to the changes in the dynamic response of the observed structure. For this reason the training set obtained using the proposed approach can only be used in the specific state of the environment for which it was designed. In real-life scenarios dynamic properties of the structure can be considerably affected by the environmental and operational factors (e.g. temperature, humidity, variable loads etc.) [5,10]. In the future research, the proposed approach could be extended in such manner to take these effect into the account, e.g. by generating the training set for different combinations of environmental/operational states. However, such an extension could appear much more demanding, regarding the time and resources required to build such a data set.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

The authors acknowledge financial support from the European Union's Horizon Europe research and innovation programme under the grant agreement No. 101091536 (DiCiM project).

Appendix. Support Vector Machine

Support Vector Machine (SVM) is a machine learning method for binary classification proposed by Boser et al. in [61]. The method assumes that the data set forming a p-dimensional feature space can be divided into two parts by a $(p-1)$ dimensional hyperplane (e.g., a 2D feature space can be separated by the 1D line, and a 3D feature space can be separated by the 2D plane). The method originates from the simpler maximum margin classifier, which also separated the feature space by a hyperplane, but was limited to problems with two well-separated classes separable by a linear hyperplane. The distance of the training samples $\boldsymbol{x}_i = \begin{bmatrix} x_{i1} & x_{i2} & \dots & x_{ip} \end{bmatrix}$; for $i = 1 \dots n$, closest to the hyperplane (called support vectors) to the hyperplane is called the margin (Fig. A.23). The hyperplane separating the two classes is determined in such way that the margin is maximized. In this case the position of individual samples relative to such a hyperplane is determined by the equation:

$$y_i \left(\boldsymbol{\beta} \cdot \boldsymbol{x}_i + \beta_0 \right) \geq 1, \tag{A.1}$$

where y_i is a class label belonging to the class of the ith training sample (labeled by 1 or −1), and $\boldsymbol{\beta} = \begin{bmatrix} \beta_1 & \beta_2 & \dots & \beta_p \end{bmatrix}$ together with β_0 are the hyperplane coefficients.

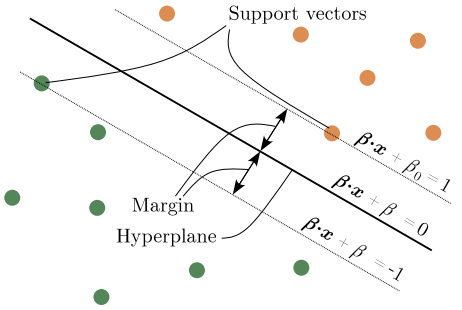

Fig. A.23. Separation of two classes of samples by a hyperplane for p=2 case.

Later, the method was extended to the non-separable problems, allowing some samples to lie on the wrong side of the hyperplane margin or even on the wrong side of the hyperplane. This was achieved by introducing the slack variables ϵ_i; $i = 1, \ldots, n$, reformulating Eq. (A.1) to:

$$y_i\left(\boldsymbol{\beta}\cdot\boldsymbol{x}_i+\beta_0\right)\geq 1-\epsilon_i. \tag{A.2}$$

It can be shown that the optimization problem of maximizing the hyperplane margin and at the same time minimizing the slack penalty $\sum_{i=1}^{n}\epsilon_i^k$ (the penalty parameter $k = 1$ is assumed) can be written in the following form:

$$\min_{\boldsymbol{\beta},\beta_0}\frac{1}{2}\|\boldsymbol{\beta}\|^2+C\sum_{i=1}^{n}\epsilon_i, \tag{A.3}$$

subjected to the constraint given by Eqs. (A.2) and:

$$\epsilon_i\geq 0 \text{ for } i\in[1,n]. \tag{A.4}$$

C is a tuning parameter which determines the trade-off between the margin maximization and the slack penalty minimization.

From the upper optimization problem a Lagrangian dual problem can be derived:

$$\max_{\boldsymbol{\alpha}}\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j\left(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j\right), \tag{A.5}$$

subjected to:

$$0\leq\alpha_i\leq C \text{ and} \tag{A.6}$$

$$\sum_{i=1}^{m}\alpha_i y_i=0 \text{ for } i\in[1,n], \tag{A.7}$$

where $\boldsymbol{\alpha}=\begin{bmatrix}\alpha_1 & \alpha_2 & \ldots & \alpha_n\end{bmatrix}^{\mathrm{T}}$ is a vector of Lagrangian variables. The dot product $\left(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j\right)$ in (A.5) can be replaced by the linear kernel $K\left(\boldsymbol{x}_i,\boldsymbol{x}_j\right)=\sum_{k=1}^{p}x_{ik}x_{jk}$:

$$\max_{\boldsymbol{\alpha}}\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j K\left(\boldsymbol{x}_i,\boldsymbol{x}_j\right). \tag{A.8}$$

The general formulation of the SVM extends the upper classifier in such way that it allows the separation of the classes by a non-linear hyperplane, implementing alternative kernel definitions, e.g., radial basis function:

$$K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\exp\left(-\frac{\|\boldsymbol{x}_i-\boldsymbol{x}_j\|}{2\sigma^2}\right), \tag{A.9}$$

σ being a positive constant, or polynomial kernel:

$$K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\left(\boldsymbol{x}_i\cdot\boldsymbol{x}_j+c\right)^d, \tag{A.10}$$

c being a positive constant, and d being a degree of polynomial kernel. When the Lagrange variables $\boldsymbol{\alpha}$ are obtained by training an SVM model, a sample $\boldsymbol{x}$ can be classified according to the following equation:

$$h(\boldsymbol{x})=\operatorname{sgn}\left(\sum_{i=1}^{n}\alpha_i y_i K(\boldsymbol{x}_i,\boldsymbol{x})+\beta_0\right), \tag{A.11}$$

with $h(\boldsymbol{x})$ being a predicted class for the sample $\boldsymbol{x}$. β_0 can be calculated from any of the training samples for which it holds that $0\leq\alpha_i\leq C$ [62]:

$$\beta_0=y_i-\sum_{j=1}^{n}\alpha_j y_j K(\boldsymbol{x}_j,\boldsymbol{x}_i). \tag{A.12}$$

The SVM classifier can be extended to the multi-class problems in different ways, e.g., using a *one-versus-one* or *one-versus-all* approach [53].

References

[1] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, A.K. Nandi, Applications of machine learning to machine fault diagnosis: A review and roadmap, Mech. Syst. Signal Process. 138 (2020) 106587, http://dx.doi.org/10.1016/j.ymssp.2019.106587.

[2] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, D.J. Inman, 1D convolutional neural networks and applications: A survey, Mech. Syst. Signal Process. 151 (2021) 107398, http://dx.doi.org/10.1016/j.ymssp.2020.107398.

[3] W. Li, R. Huang, J. Li, Y. Liao, Z. Chen, G. He, R. Yan, K. Gryllias, A perspective survey on deep transfer learning for fault diagnosis in industrial scenarios: Theories, applications and challenges, Mech. Syst. Signal Process. 167 (2022) 108487, http://dx.doi.org/10.1016/j.ymssp.2021.108487.

[4] R.K. Mobley, An Introduction to Predictive Maintenance, Elsevier, 2002.

[5] A. Entezami, H. Sarmadi, B. Behkamal, Long-term health monitoring of concrete and steel bridges under large and missing data by unsupervised meta learning, Eng. Struct. 279 (2023) 115616, http://dx.doi.org/10.1016/j.engstruct.2023.115616.

[6] A. Entezami, Statistical decision-making by distance measures, in: Structural Health Monitoring By Time Series Analysis and Statistical Distance Measures, Springer International Publishing, Cham, 2021, pp. 59–79, http://dx.doi.org/10.1007/978-3-030-66259-2_4.

[7] A. Entezami, H. Sarmadi, B. Behkamal, C.D. Michele, On continuous health monitoring of bridges under serious environmental variability by an innovative multi-task unsupervised learning method, Struct. Infrastruct. Eng. (2023) 1–19, http://dx.doi.org/10.1080/15732479.2023.2166538.

[8] H. Sarmadi, K.-V. Yuen, Structural health monitoring by a novel probabilistic machine learning method based on extreme value theory and mixture quantile modeling, Mech. Syst. Signal Process. 173 (2022) 109049, http://dx.doi.org/10.1016/j.ymssp.2022.109049.

[9] A. Entezami, H. Shariatmadar, C.D. Michele, Non-parametric empirical machine learning for short-term and long-term structural health monitoring, Struct. Health Monit. 21 (6) (2022) 2700–2718, http://dx.doi.org/10.1177/14759217211069842.

[10] A. Entezami, H. Sarmadi, B. Behkamal, A novel double-hybrid learning method for modal frequency-based damage assessment of bridge structures under different environmental variation patterns, Mech. Syst. Signal Process. 201 (2023) 110676, http://dx.doi.org/10.1016/j.ymssp.2023.110676.

[11] C.R. Farrar, K. Worden, An introduction to structural health monitoring, Phil. Trans. R. Soc. A 365 (1851) (2007) 303–315.

[12] W. Ostachowicz, R. Soman, P. Malinowski, Optimization of sensor placement for structural health monitoring: a review, Struct. Health Monit. 18 (3) (2019) 963–988, http://dx.doi.org/10.1177/1475921719825601.

[13] O. Avci, O. Abdeljaber, S. Kiranyaz, M. Hussein, M. Gabbouj, D.J. Inman, A review of vibration-based damage detection in civil structures: From traditional methods to Machine Learning and Deep Learning applications, Mech. Syst. Signal Process. 147 (2021) 107077, http://dx.doi.org/10.1016/j.ymssp.2020.107077.

[14] Y. Yang, Y. Zhang, X. Tan, Review on vibration-based structural health monitoring techniques and technical codes, Symmetry 13 (11) (2021) 1998.

[15] K. Feng, J. Ji, Q. Ni, M. Beer, A review of vibration-based gear wear monitoring and prediction techniques, Mech. Syst. Signal Process. 182 (2023) 109605, http://dx.doi.org/10.1016/j.ymssp.2022.109605.

[16] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, D.J. Inman, Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks, J. Sound Vib. 388 (2017) 154–170.

[17] Y. Bao, Z. Tang, H. Li, Y. Zhang, Computer vision and deep learning–based data anomaly detection method for structural health monitoring, Struct. Health Monit. 18 (2) (2019) 401–421.

[18] U. Dackermann, J. Li, B. Samali, Identification of member connectivity and mass changes on a two-storey framed structure using frequency response functions and artificial neural networks, J. Sound Vib. 332 (16) (2013) 3636–3653.

[19] J. Liu, Z. Lu, M. Yu, Damage identification of non-classically damped shear building by sensitivity analysis of complex modal parameter, J. Sound Vib. 438 (2019) 457–475.

[20] Y. Yang, X. Chao, J. Ming-zhen, L. Wei-dong, K. Yang, Bridge damage identification method considering road surface roughness by using indirect measurement technique, China J. Highw. Transp. 32 (1) (2019) 99.

[21] Y. Yang, L. Jin-qiu, Y. Ai-peng, L. Hui-cheng, L. Kang-hui, S. Xiao-jun, W. Qin, Bridge element bending stiffness damage identification based on new indirect measurement method, China J. Highw. Transp. 34 (2) (2021) 188.

[22] Z. Wang, Y. jin Cha, Automated damage-sensitive feature extraction using unsupervised convolutional neural networks, in: H. Sohn (Ed.), Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2018, Vol. 10598, SPIE, International Society for Optics and Photonics, 2018, p. 105981J, http://dx.doi.org/10.1117/12.2295966.

[23] G. Mariniello, T. Pastore, C. Menna, P. Festa, D. Asprone, Structural damage detection and localization using decision tree ensemble and vibration data, Comput.-Aided Civ. Infrastruct. Eng. 36 (9) (2021) 1129–1149.

[24] J. Li, H. Hao, R. Wang, L. Li, Development and application of random forest technique for element level structural damage quantification, Struct. Control Health Monit. 28 (3) (2021) e2678.

[25] Y. Deng, M. Zhang, D.-M. Feng, A.-Q. Li, Predicting fatigue damage of highway suspension bridge hangers using weigh-in-motion data and machine learning, Struct. Infrastruct. Eng. 17 (2) (2021) 233–248.

[26] M. Azimi, A.D. Eslamlou, G. Pekcan, Data-driven structural health monitoring and damage detection through deep learning: State-of-the-art review, Sensors 20 (10) (2020) http://dx.doi.org/10.3390/s20102778.

[27] M.D. Dangut, I.K. Jennions, S. King, Z. Skaf, Application of deep reinforcement learning for extremely rare failure prediction in aircraft maintenance, Mech. Syst. Signal Process. 171 (2022) 108873, http://dx.doi.org/10.1016/j.ymssp.2022.108873.

[28] W. Booyse, D.N. Wilke, S. Heyns, Deep digital twins for detection, diagnostics and prognostics, Mech. Syst. Signal Process. 140 (2020) 106612, http://dx.doi.org/10.1016/j.ymssp.2019.106612.

[29] P. Gardner, R. Fuentes, N. Dervilis, C. Mineo, S. Pierce, E. Cross, K. Worden, Machine learning at the interface of structural health monitoring and non-destructive evaluation, Phil. Trans. R. Soc. A 378 (2182) (2020) 20190581.

[30] E. Figueiredo, J. Brownjohn, Three decades of statistical pattern recognition paradigm for SHM of bridges, Struct. Health Monit. 21 (6) (2022) 3018–3054, http://dx.doi.org/10.1177/14759217221075241.

[31] F.-G. Yuan, S.A. Zargar, Q. Chen, S. Wang, Machine learning for structural health monitoring: challenges and opportunities, in: Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2020, Vol. 11379, 11379, SPIE, 2020, 1137903.

[32] A. Fernandez-Navamuel, D. Pardo, F. Magalhães, D. Zamora-Sánchez, Á.J. Omella, D. Garcia-Sanchez, Bridge damage identification under varying environmental and operational conditions combining deep learning and numerical simulations, Mech. Syst. Signal Process. 200 (2023) 110471, http://dx.doi.org/10.1016/j.ymssp.2023.110471.

[33] M. Sanayei, A. Khaloo, M. Gul, F.N. Catbas, Automated finite element model updating of a scale bridge model using measured static and modal test data, Eng. Struct. 102 (2015) 66–79.

[34] A. Suzuki, M. Kurata, X. Li, S. Shimmoto, Residual structural capacity evaluation of steel moment-resisting frames with dynamic-strain-based model updating method, Earthq. Eng. Struct. Dyn. 46 (11) (2017) 1791–1810.

[35] Q. Lu, J. Zhu, W. Zhang, Quantification of fatigue damage for structural details in slender coastal bridges using machine learning-based methods, J. Bridge Eng. 25 (7) (2020) 04020033.
[36] H. Sousa, A. Rozsas, A. Slobbe, W. Courage, A novel pro-active approach towards SHM-based bridge management supported by FE analysis and Bayesian methods, Struct. Infrastruct. Eng. 16 (2) (2020) 233–246.
[37] D. de Klerk, D.J. Rixen, S.N. Voormeeren, General framework for dynamic substructuring: History, review, and classification of techniques, AIAA J. (46(5)) (2008) 1169–1181.
[38] M. Van der Seijs, Experimental Dynamic Substructuring: Analysis and design strategies for vehicle development (Ph.D. thesis), 2016, http://dx.doi.org/10.4233/uuid:28b31294-8d53-49eb-b108-284b63edf670.
[39] S. Voormeeren, D. Rixen, A family of substructure decoupling techniques based on a dual assembly approach, Mech. Syst. Signal Process. 27 (2012) 379–396, http://dx.doi.org/10.1016/j.ymssp.2011.07.028.
[40] B. Starc, G. Čepon, M. Boltežar, The influence of washing machine-leg hardness on its dynamics response within component-mode synthesis techniques, Int. J. Mech. Sci. 127 (2017) 23–30.
[41] M. Haeussler, D. Kobus, D. Rixen, Parametric design optimization of e-compressor NVH using blocked forces and substructuring, Mech. Syst. Signal Process. 150 (2021) 107217, http://dx.doi.org/10.1016/j.ymssp.2020.107217.
[42] S. Klaassen, M. Van der Seijs, D. Klerk, System equivalent model mixing, Mech. Syst. Signal Process. 105 (2017) 90–112, http://dx.doi.org/10.1016/j.ymssp.2017.12.003.
[43] S. Klaassen, M. Van der Seijs, Introducing SEMM: A novel method for hybrid modelling, 2018.
[44] M. Kodrič, T. Bregar, G. Čepon, M. Boltežar, An expansion based on System Equivalent Model Mixing: From a limited number of points to a full-field dynamic response, Measurement 190 (2022) 110522, http://dx.doi.org/10.1016/j.measurement.2021.110522.
[45] A. Malekloo, E. Ozer, M. AlHamaydeh, M. Girolami, Machine learning and structural health monitoring overview with emerging technology and high-dimensional data source highlights, Struct. Health Monit. 21 (4) (2022) 1906–1955.
[46] T. Ritto, F. Rochinha, Digital twin, physics-based model, and machine learning applied to damage detection in structures, Mech. Syst. Signal Process. (2020).
[47] M. Van der Seijs, D. van den Bosch, D. Rixen, D. Klerk, An improved methodology for the virtual point transformation of measured frequency response functions in dynamic substructuring, 2013, pp. 4334–4347, http://dx.doi.org/10.7712/120113.4816.C1539.
[48] E. Pasma, M. Van der Seijs, S. Klaassen, M. van der Kooij, Frequency based substructuring with the virtual point transformation, flexible interface modes and a transmission simulator, 2018.
[49] T. Bregar, A. El Mahmoudi, M. Kodrič, D. Ocepek, F. Trainotti, M. Pogačar, M. Göldeli, G. Čepon, M. Boltežar, D.J. Rixen, pyFBS: A python package for frequency based substructuring, J. Open Source Softw. 7 (69) (2022) 3399.
[50] S.B. Kotsiantis, D. Kanellopoulos, P.E. Pintelas, Data preprocessing for supervised leaning, Int. J. Comput. Sci. 1 (2) (2006) 111–117.
[51] R.P. Bandara, T.H. Chan, D.P. Thambiratnam, Frequency response function based damage identification using principal component analysis and pattern recognition technique, Eng. Struct. 66 (2014) 116–128, http://dx.doi.org/10.1016/j.engstruct.2014.01.044.
[52] M. Silva, A. Santos, R. Santos, E. Figueiredo, C. Sales, J.C. Costa, Deep principal component analysis: An enhanced approach for structural damage identification, Struct. Health Monit. 18 (5–6) (2019) 1444–1463, http://dx.doi.org/10.1177/1475921718799070.
[53] G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction to Statistical Learning, Vol. 112, Springer, 2013.
[54] T. Hastie, R. Tibshirani, J.H. Friedman, J.H. Friedman, The elements of statistical learning: data mining, inference, and prediction, vol. 2, Springer, 2009.
[55] J. Yang, Y. Zhang, Y. Zhu, Intelligent fault diagnosis of rolling element bearing based on SVMs and fractal dimension, Mech. Syst. Signal Process. 21 (5) (2007) 2012–2024, http://dx.doi.org/10.1016/j.ymssp.2006.10.005.
[56] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.
[57] C. Goutte, E. Gaussier, A probabilistic interpretation of precision, recall and F-score, with implication for evaluation, in: Advances in Information Retrieval: 27th European Conference on IR Research, ECIR 2005, Santiago de Compostela, Spain, March 21-23, 2005. Proceedings 27, Springer, 2005, pp. 345–359.
[58] E. Pasma, S. Klaassen, L. Nieuwenhuijse, M. Van Der Seijs, D. Lennström, Application of system equivalent model mixing (SEMM) to model the structural dynamic properties of a complex vehicle component using numerical and experimental data, in: Proceedings of ISMA2018, 2018.
[59] Z. Saeed, S.W. Klaassen, C.M. Firrone, T.M. Berruti, D.J. Rixen, Experimental joint identification using system equivalent model mixing in a bladed disk, J. Vib. Acoust. 142 (5) (2020) 051001.
[60] M. Kodrič, T. Bregar, G. Čepon, M. Boltežar, An expansion based on System Equivalent Model Mixing: From a limited number of points to a full-field dynamic response, Measurement 190 (2022) 110522.
[61] B.E. Boser, I.M. Guyon, V.N. Vapnik, A training algorithm for optimal margin classifiers, in: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, 1992, pp. 144–152.
[62] M. Mohri, A. Rostamizadeh, A. Talwalkar, Foundations of Machine Learning, MIT Press, 2018.