MOX-Report No. 19/2024

A multi-fidelity surrogate model for structural health monitoring exploiting model order reduction and artificial neural networks

Torzoni, M.; Manzoni, A.; Mariani, S.

MOX, Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox-dmat@polimi.it https://mox.polimi.it

A multi-fidelity surrogate model for structural health monitoring exploiting model order reduction and artificial neural networks

Matteo Torzoni[a,*], Andrea Manzoni[b], Stefano Mariani[a]

[a] *Dipartimento di Ingegneria Civile e Ambientale,*
Politecnico di Milano, Piazza L. da Vinci 32, 20133 Milano, Italy
[b] *MOX, Dipartimento di Matematica,*
Politecnico di Milano, Piazza L. da Vinci 32, 20133 Milano, Italy

Abstract

Stochastic approaches to structural health monitoring (SHM) are often inevitably limited by computational constraints. For instance, for Markov chain Monte Carlo algorithms relying upon computationally expensive finite element models it is almost infeasible to sample the probability distribution of the structural state. To provide instead real-time procedures, this work proposes a non-intrusive surrogate modeling strategy, leveraging model order reduction and artificial neural networks. By relying upon a multi-fidelity (MF) framework, a composition of deep neural networks (DNNs) is devised to map damage and operational parameters onto time-dependent sensor recordings. Such an effective strategy is able to exploit datasets characterized by different fidelity levels without any prior assumption, allowing to blend a small high-fidelity (HF) dataset with a large low-fidelity (LF) dataset, ultimately alleviating the computational burden of supervised training while ensuring the accuracy of the approximated quantities of interest. The resulting surrogate model is made of an LF-DNN, which mimics sensor recordings in the undamaged condition, and of a long short-term memory HF-DNN, which adaptively refines the approximation with the effect of damage. An HF finite element model and an LF reduced order model are adopted offline to generate labeled training data of different fidelity, respectively in the presence or absence of a structural damage. Results relevant to an L-shaped cantilever beam and a portal frame railway bridge prove that the procedure efficiently provides remarkably accurate approximations, outperforming their single-fidelity counterparts. The capability of the MF-DNN to be exploited for SHM purposes is finally shown within an automated Bayesian procedure, aimed at updating the probability distribution of the structural state conditioned on sensor recordings, in the presence of operational variability and measurement noise.

Keywords: Structural health monitoring, Multi-fidelity methods, Bayesian model updating, Deep learning, Reduced-order modeling, Simulation-based damage identification.

[*] Corresponding author
Email addresses: matteo.torzoni@polimi.it (Matteo Torzoni), andrea1.manzoni@polimi.it (Andrea Manzoni), stefano.mariani@polimi.it (Stefano Mariani)

1. Introduction

The interest in predicting the behavior of physical systems has motivated the rise of accurate methods for numerically solving differential problems in engineering and applied sciences [1–3]. Numerical simulations are thus crucial to tackle several tasks including discovery, optimization and decision support. However, in many-query and outer-loop applications, such as parametric studies, statistical inference, and optimization, the computational cost of a high-fidelity (HF) model often becomes prohibitive. In these cases, a cheaper low-fidelity (LF) model can be introduced to allow for a high number of evaluations featuring a (possibly) slightly reduced accuracy [4]. Apart from a simplified physics or a coarser discretization (either in time or space), remarkable instances of LF models have also been obtained with model order reduction (MOR) strategies [5–8] and surrogate modeling techniques [9–13].

One of the most relevant challenges of modern engineering concerns the optimal management of deteriorating structural systems. In particular, the failure or non-optimized maintenance planning of civil structures can yield high safety, economic, and social costs. In this context, the evolution from classical time-based maintenance practices with scheduled inspections toward condition-based ones has been recently put forth, to reduce the lifecycle costs and to increase the system safety and availability [14–16]. This paradigm shift is empowered by the synergistic use of permanent real-time data collecting devices and systematic diagnostic activities, to enable a digital twin perspective for structural systems that are critical for either safety or operative reasons. To this aim, vibration-based structural health monitoring (SHM) techniques exploit vibration response data acquired with pervasive sensing systems to assess the damage and track its evolution [17, 18].

SHM approaches can be classified as either data-driven [19–22] or model-based [23–26]: the former rely on data to unveil the relationship between selected features and the sought damage patterns [27, 28]; the latter directly exploit the physics-based knowledge about the system response and assess damage through a model updating strategy. However, the associated inverse problem is not only ill-posed, but it also features a full range of uncertainties due to measurement noise, modeling assumptions and environmental or operational variabilities. Therefore, a probabilistic framework enabled by Bayesian model updating appears to be one of the most natural ways to address the foreseen parameter estimation problem.

Since in practical cases the posterior probability density function (pdf) of the sought parameters cannot be evaluated analytically [29, 30], it is usually approximated through sampling-based Markov chain Monte Carlo (MCMC) methods, see e.g. [31–33]. However, MCMC can incur in high computational costs due to slow convergence and repeated evaluations of an intensive forward model. To this aim, advanced samplers have been proposed to improve the convergence rate, such as the Metropolis-Hastings, transitional MCMC, Hybrid Monte Carlo algorithms and their recently proposed extensions, see e.g. [34–36]. The computational efficiency of MCMC has been improved by also means of cheap-to-evaluate data-driven surrogate models; popular strategies include polynomial response surface models, polynomial chaos, support vector machines, Kriging models and artificial neural networks, see for instance [37–41].

An intriguing opportunity to further reduce the computational cost of fitting an accurate surrogate model may arise in situations where multiple models are available to simulate the relevant structural response with varying accuracy and computational cost. This is the case of multi-fidelity (MF) methods, which blend LF with HF models to improve the accuracy of the output of interest while reducing the associated computational burden. Indeed, LF samples often supply useful information on the major trends of the problem, allowing the MF setting to outperform single-fidelity methods in terms of prediction accuracy. Motivated by the recent advances in MF methods [4], this work proposes a MF deep neural network (DNN) surrogate model tailored for SHM purposes. The premise for such MF-DNN setup is that the correlation across structural models featuring different fidelity levels can be leveraged without any prior assumption, thanks to the flexibility of neural networks, ultimately alleviating the computational burden of the offline supervised training stage while ensuring the accuracy of the approximated quantities of interest, see e.g. [42–47].

The proposed MF-DNN features a multi-level architecture defined by sequentially trained DNNs, respectively an LF and an HF part. The LF-DNN is a fully-connected model, used to mimic sensor recordings in the undamaged (baseline) condition, while the HF-DNN is a long short-term memory (LSTM) model, exploited to adaptively enrich the LF approximation with the effect of damage. Both DNNs are trained on vibration response data, numerically generated exploiting physics-based models, so that the effect of damage on the structural response can be systematically reproduced [48]. Specifically, an LF reduced-order model, obtained by relying on a proper orthogonal decomposition (POD)-Galerkin reduced basis method [7, 8, 49], and an HF finite element model, are adopted to simulate the structural response under varying operational conditions, respectively in the absence or presence of a structural damage.

The novelty of the proposed methodology stems from the specific adaptation of the MF-DNN surrogate modeling framework for SHM purposes. The proposed MF-DNN is devised to map damage and operational parameters onto sensor recordings. This task would be difficult to address with other popular MF techniques, such as the Gaussian processes-based Cokriging [50–52], mainly because the computational complexity of its training scales cubically with the total number of high- and low-fidelity training points. Moreover, Cokriging models require prior knowledge about the correlation between various fidelity levels, but can usually only capture linear correlations between them [44, 45]. On the other hand, the proposed MF-DNN: is suitable for high-dimensional problems and benefits from large LF training datasets; can make predictions for new input parameters in real-time; can learn linear and nonlinear correlations adaptively, without prior information; can handle the approximation of strongly discontinuous functions. Besides that, the proposed MF-DNN inherits the main features of MF methods, such as a reduced computational burden associated to the offline data generation phase and an improved accuracy of the approximated quantities of interest. Moreover, the setup here proposed enjoys additional features: the effect of damage in the structural response is simulated with the HF model only, which is considered to be the most accurate description enabling to account for unexperienced damage scenarios; a large amount of training data relevant to the undamaged condition can be instead simulated, once and for all, exploiting MOR to speed up the offline training phase; it is never necessary to update the LF part. The proposed framework can be useful, e.g., for model-updating purposes, or to cheaply generate large labeled datasets.

When adopted for SHM purposes, the offered strategy can be implemented by first assembling the MF datasets in an offline manner and by next processing in real time the data collected during the monitoring stage, to provide an estimation of the structural damage, if any. To get insights into all the stages of the procedure, the reminder of this paper is structured as follows, see also the roadmap shown in Fig. 1. In Sec. 2, we describe in detail the proposed surrogate modeling framework. In Sec. 3, the procedure is assessed on two test cases resembling potential monitoring scenarios, respectively related to an L-shaped cantilever beam and a railway bridge; the relevant results in terms of prediction accuracy, are discussed in comparison with a standard single-fidelity approximation. A possible way to exploit the proposed surrogate model within an MCMC-based model updating scenario is then framed in Sec. 4. Conclusions are finally drawn in Sec. 5

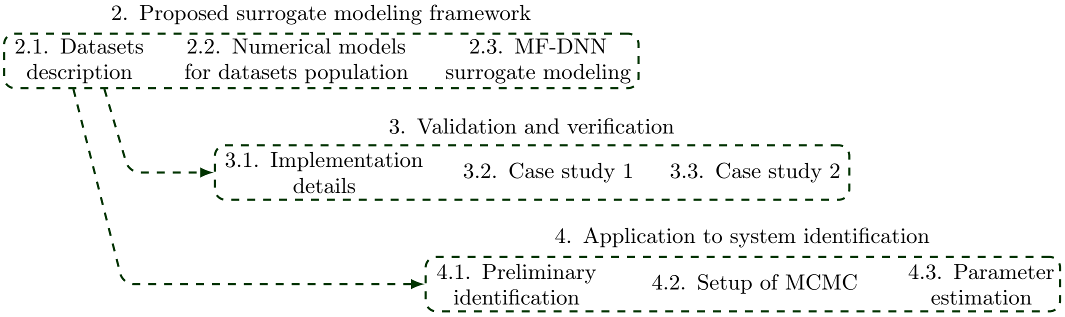

2. Proposed surrogate modeling framework
2.1. Datasets description
2.2. Numerical models for datasets population
2.3. MF-DNN surrogate modeling
3. Validation and verification
3.1. Implementation details
3.2. Case study 1
3.3. Case study 2
4. Application to system identification
4.1. Preliminary identification
4.2. Setup of MCMC
4.3. Parameter estimation

Figure 1: Roadmap of the paper.

2. MF-DNN surrogate model

In this section, we describe the MF-DNN surrogate model to map input parameters onto time series data. The MF framework is exploited to speed up the data generation phase required to fit an accurate surrogate model, while ensuring the accuracy of the approximated signals in terms of damage-sensitivity. DNNs are instead leveraged to learn the (possibly nonlinear) correlation across high-dimensional data of different fidelities without any prior knowledge. The handled LF and HF data are specified in Sec. 2.1, while the numerical models exploited for their generation are described in Sec. 2.2, together with the POD-based strategy adopted for model order reduction. The way the MF modeling is handled, the architecture of the MF-DNN, and the technical aspects related to its training and evaluation, are then discussed in Sec. 2.3.

2.1. Low/high-fidelity datasets

The LF and HF datasets, denoted as $\mathbf{D}_{\texttt{LF}}$ and $\mathbf{D}_{\texttt{HF}}$, collect simulated structural response data, respectively in the absence or presence of damage. In particular, LF data are always related to the baseline condition; HF data have to account for potential degradation processes at variance with the formerly identified damaged condition. Without loss of generality, in the following we will refer to the initial monitoring phase of an undamaged reference condition.

$\mathbf{D}_{\texttt{LF}}$ and $\mathbf{D}_{\texttt{HF}}$ are built from the assembly of $I_{\texttt{LF}}$ and $I_{\texttt{HF}}$ data instances, with $I_{\texttt{LF}} > I_{\texttt{HF}}$, as follows:

$$\mathbf{D}_{\texttt{LF}} = \{(\mathbf{x}_i^{\texttt{LF}}, \mathbf{U}_i^{\texttt{LF}})\}_{i=1}^{I_{\texttt{LF}}} \ , \quad \mathbf{D}_{\texttt{HF}} = \{(\mathbf{x}_j^{\texttt{HF}}, \mathbf{U}_j^{\texttt{HF}})\}_{j=1}^{I_{\texttt{HF}}} \ , \tag{1}$$

where $\mathbf{x}_i^{\mathtt{LF}} \in \mathbb{R}^{\mathtt{N}_{\mathtt{par}}^{\mathtt{LF}}}$ denotes the operational conditions, i.e., the $\mathtt{N}_{\mathtt{par}}^{\mathtt{LF}}$ parameters ruling the loading conditions of the corresponding LF vibration recording $\mathbf{U}_i^{\mathtt{LF}}(\mathbf{x}_i^{\mathtt{LF}}) = [\mathbf{u}_1^{\mathtt{LF}}, \ldots, \mathbf{u}_{N_u}^{\mathtt{LF}}]_i \in \mathbb{R}^{L \times N_u}$. $\mathbf{U}_i^{\mathtt{LF}}$ thus consists of N_u time series made of L sensor measurements, for instance in terms accelerations or displacements. Each recording refers to a time interval $(0, T)$, so that $L = \mathscr{f} T + 1$, where $\mathscr{f}$ is the adopted sampling rate. Each HF data instance instead accounts for the effect of damage on the HF recording $\mathbf{U}_j^{\mathtt{HF}}(\mathbf{x}_j^{\mathtt{HF}}) = [\mathbf{u}_1^{\mathtt{HF}}, \ldots, \mathbf{u}_{N_u}^{\mathtt{HF}}]_j \in \mathbb{R}^{L \times N_u}$, as the relevant $\mathtt{N}_{\mathtt{par}}^{\mathtt{HF}}$ input parameters $\mathbf{x}_j^{\mathtt{HF}} \in \mathbb{R}^{\mathtt{N}_{\mathtt{par}}^{\mathtt{HF}}}$, with $\mathtt{N}_{\mathtt{par}}^{\mathtt{HF}} > \mathtt{N}_{\mathtt{par}}^{\mathtt{LF}}$, control both the operational and the damage conditions. Referring to real-time SHM applications, the interval $(0, T)$ is assumed short enough to consider fixed operational, environmental, and damage conditions, yet long enough to not compromise the identification of the structural behavior. In the reminder of the paper, indexes i and j will be dropped for ease of notation, unless necessary.

The numerical models exploited to generate $\mathbf{D}_{\mathtt{LF}}$ and $\mathbf{D}_{\mathtt{HF}}$ differ in their approximation qualities and computational costs, as the HF model must be able to accurately reproduce a structural damage, but the proposed MF-DNN modeling strategy can also adapt to different modeling choices. Moreover, the sampling frequency $\mathscr{f}$, as well as the N_u monitored degrees of freedom (dofs), are supposed to be the same for both fidelity levels, although there are no restrictions in this respect.

2.2. Parametric numerical models for dataset generation

We first focus on the HF model, which describes the dynamic response of the structure under the applied loadings under the assumption of a linearized kinematics. By modeling the structure as a linear-elastic continuum and introducing a space discretization through a finite element method, the HF model is given by:

$$\begin{cases} \mathbf{M}_{\mathtt{HF}} \ddot{\mathbf{d}}^{\mathtt{HF}}(t) + \mathbf{C}_{\mathtt{HF}}(\mathbf{x}^{\mathtt{HF}}) \dot{\mathbf{d}}^{\mathtt{HF}}(t) + \mathbf{K}_{\mathtt{HF}}(\mathbf{x}^{\mathtt{HF}}) \mathbf{d}^{\mathtt{HF}}(t) = \mathbf{f}_{\mathtt{HF}}(t, \mathbf{x}^{\mathtt{HF}}) \ , \quad t \in (0, T) \\ \mathbf{d}^{\mathtt{HF}}(0) = \mathbf{d}_0^{\mathtt{HF}} \\ \dot{\mathbf{d}}^{\mathtt{HF}}(0) = \dot{\mathbf{d}}_0^{\mathtt{HF}} \ , \end{cases} \tag{2}$$

which is referred to as the HF full-order model (FOM) according to the number of the involved dofs. Here: $t \in (0, T)$ denotes time; $\mathbf{d}^{\mathtt{HF}}(t), \dot{\mathbf{d}}^{\mathtt{HF}}(t), \ddot{\mathbf{d}}^{\mathtt{HF}}(t) \in \mathbb{R}^{\mathscr{M}}$ are the vectors of nodal displacements, velocities and accelerations, respectively; $\mathscr{M}$ is the number of dofs; $\mathbf{M}_{\mathtt{HF}} \in \mathbb{R}^{\mathscr{M} \times \mathscr{M}}$ is the mass matrix; $\mathbf{C}_{\mathtt{HF}}(\mathbf{x}^{\mathtt{HF}}) \in \mathbb{R}^{\mathscr{M} \times \mathscr{M}}$ is the damping matrix, assembled according to the Rayleigh's model; $\mathbf{K}_{\mathtt{HF}}(\mathbf{x}^{\mathtt{HF}}) \in \mathbb{R}^{\mathscr{M} \times \mathscr{M}}$ is the stiffness matrix; $\mathbf{f}_{\mathtt{HF}}(t, \mathbf{x}^{\mathtt{HF}}) \in \mathbb{R}^{\mathscr{M}}$ is the vector of nodal forces induced by the external loadings; $\mathbf{d}_0^{\mathtt{HF}}$ and $\dot{\mathbf{d}}_0^{\mathtt{HF}}$ are the initial conditions (at $t = 0$), respectively in terms of nodal displacements and velocities. By adopting a uniform partition of the time interval $(0, T)$, the solution of problem (2) is advanced in time using the implicit Newmark integration method, to provide $\mathbf{d}_l^{\mathtt{HF}}$, $\dot{\mathbf{d}}_l^{\mathtt{HF}}$ and $\ddot{\mathbf{d}}_l^{\mathtt{HF}}$, for $l = 1, \ldots, L$.

As often assumed in the SHM literature [14, 53, 54], the structural damage is modeled as a localized reduction of the material stiffness, and this is here obtained by means of a suitable parametrization of the stiffness matrix. In concrete terms, each damage condition is parametrized through a set of variables $\boldsymbol{y} \in \mathbb{R}^3$ and $\delta \in \mathbb{R}$, collected in the vector $\mathbf{x}^{\mathtt{HF}}$, denoting the position and the magnitude of the stiffness reduction, respectively.

The LF model is a projection-based reduced-order model (ROM), built by relying on the POD-Galerkin reduced basis method [8], see e.g. [55–57]. This method is often exploited in the context of structural dynamics [7, 49, 58] because of its appealing offline-online decoupling, and availability of efficient criteria for the selection of the POD-basis functions.

The LF approximation to the solution of problem (2) in terms of displacements, is obtained by linearly combining $\mathscr{M}_{\mathtt{LF}} \ll \mathscr{M}$ POD-basis functions $\mathbf{w}_k \in \mathbb{R}^{\mathscr{M}}$, $k = 1, \dots, \mathscr{M}_{\mathtt{LF}}$, as $\mathbf{d}^{\mathtt{LF}}(t, \mathbf{x}^{\mathtt{LF}}) \approx \mathbf{W}\mathbf{r}(t, \mathbf{x}^{\mathtt{LF}})$, where $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_{\mathscr{M}_{\mathtt{LF}}}] \in \mathbb{R}^{\mathscr{M} \times \mathscr{M}_{\mathtt{LF}}}$ is the projection matrix and $\mathbf{r}(t, \mathbf{x}^{\mathtt{LF}}) \in \mathbb{R}^{\mathscr{M}_{\mathtt{LF}}}$ is the vector of unknown POD-coefficients. By enforcing the orthogonality between the residual and the subspace spanned by the first $\mathscr{M}_{\mathtt{LF}}$ POD-modes through a Galerkin projection, the following $\mathscr{M}_{\mathtt{LF}}$-dimensional dynamical system is obtained:

$$\begin{cases} \mathbf{M}_{\mathrm{r}}\ddot{\mathbf{r}}(t) + \mathbf{C}_{\mathrm{r}}\dot{\mathbf{r}}(t) + \mathbf{K}_{\mathrm{r}}\mathbf{r}(t) = \mathbf{f}_{\mathrm{r}}(t, \mathbf{x}^{\mathtt{LF}}) \ , \quad t \in (0, T) \\ \mathbf{r}(0) = \mathbf{W}^{\top}\mathbf{d}_0^{\mathtt{LF}} \\ \dot{\mathbf{r}}(0) = \mathbf{W}^{\top}\dot{\mathbf{d}}_0^{\mathtt{LF}} \ , \end{cases} \tag{3}$$

whose solution is advanced in time using the same strategy employed for the HF model, and then back-projected onto the original LF-FOM space as $\mathbf{d}^{\mathtt{LF}}(t, \mathbf{x}^{\mathtt{LF}}) \approx \mathbf{W}\mathbf{r}(t, \mathbf{x}^{\mathtt{LF}})$. Here, the reduced arrays play the same role of their HF counterparts, yet with dimension ruled by $\mathscr{M}_{\mathtt{LF}}$ instead of $\mathscr{M}$, according to the following relationships:

$$\begin{aligned} &\mathbf{M}_{\mathrm{r}} \equiv \mathbf{W}^{\top}\mathbf{M}_{\mathtt{HF}}\mathbf{W} \ , && \mathbf{C}_{\mathrm{r}} \equiv \mathbf{W}^{\top}\mathbf{C}_{\mathtt{HF}}\mathbf{W} \ , \\ &\mathbf{K}_{\mathrm{r}} \equiv \mathbf{W}^{\top}\mathbf{K}_{\mathtt{LF}}\mathbf{W} \ , && \mathbf{f}_{\mathrm{r}}(t, \mathbf{x}^{\mathtt{LF}}) \equiv \mathbf{W}^{\top}\mathbf{f}_{\mathtt{HF}}(t, \mathbf{x}^{\mathtt{LF}}). \end{aligned} \tag{4}$$

In addition to avoid accounting for the presence of the damage, the damping term is also disregarded in the LF model in problem (3). We note that it would not be necessary to disregard the damping term, and this is done for the sole purpose of better validating the MF-DNN performance on significantly different LF and HF structural responses.

The projection matrix $\mathbf{W}$ is obtained by means of POD, exploiting the so-called method of snapshots. To this aim, a LF-FOM, resembling that defined by problem (2) but not accounting for the presence of damage (through $\mathbf{K}_{\mathtt{LF}}$) and damping, is adopted to assemble a matrix $\mathbf{S} = [\mathbf{d}_1^{\mathtt{LF}}, \dots, \mathbf{d}_{\mathscr{S}}^{\mathtt{LF}}] \in \mathbb{R}^{\mathscr{M} \times \mathscr{S}}$ from $\mathscr{S}$ solution snapshots, computed by integrating in time the LF-FOM for different values of parameters $\mathbf{x}^{\mathtt{LF}}$. An optimal reduced basis of arbitrary order $\mathscr{M}_{\mathtt{LF}}$, that minimizes the projection error $\sum_{\mathscr{s}=1}^{\mathscr{S}} \|\mathbf{d}_{\mathscr{s}}^{\mathtt{LF}} - \mathbf{W}\mathbf{W}^{\top}\mathbf{d}_{\mathscr{s}}^{\mathtt{LF}}\|_2^2$, is then computed by factorizing $\mathbf{S}$ through a singular value decomposition (SVD) as:

$$\mathbf{S} = \mathbf{P}\mathbf{\Sigma}\mathbf{Z}^{\top} \ , \tag{5}$$

where: $\mathbf{P} = [\mathbf{p}_1, \dots, \mathbf{p}_{\mathscr{M}}] \in \mathbb{R}^{\mathscr{M} \times \mathscr{M}}$ and $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_{\mathscr{S}}] \in \mathbb{R}^{\mathscr{S} \times \mathscr{S}}$ are two orthogonal matrices, whose columns are the left and right singular vectors of $\mathbf{S}$, respectively; $\mathbf{\Sigma} \in \mathbb{R}^{\mathscr{M} \times \mathscr{S}}$ is a pseudo-diagonal matrix collecting the singular values $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{\mathscr{R}} \geq 0$ of $\mathbf{S}$, $\mathscr{R} = \min(\mathscr{M}, \mathscr{S})$ being the rank of $\mathbf{S}$. Considering the first $\mathscr{M}_{\mathtt{LF}} \ll \mathscr{M}$ columns of $\mathbf{P}$, the projection matrix $\mathbf{W}$ is obtained as $\mathbf{W} = [\mathbf{p}_1, \dots, \mathbf{p}_{\mathscr{M}_{\mathtt{LF}}}]$.

As each singular value quantifies the information content of $\mathbf{S}$ described by the corresponding left singular vector, the order $\mathscr{M}_{\mathtt{LF}}$ can be set by prescribing a tolerance ϵ on the fraction of energy content to be disregarded in the approximation, according to:

$$\frac{\sum_{m=1}^{\mathscr{M}_{\mathtt{LF}}} (\sigma_m)^2}{\sum_{m=1}^{\mathscr{R}} (\sigma_m)^2} \geq 1 - \epsilon^2 \ . \tag{6}$$

To populate $\mathbf{D}_{\mathtt{LF}}$ and $\mathbf{D}_{\mathtt{HF}}$ according to Eq. (1), the parametric input spaces described by $\mathbf{x}^{\mathtt{LF}}$ and $\mathbf{x}^{\mathtt{HF}}$ are assumed to be uniformly distributed and sampled via the latin hypercube rule. With reference to displacement recordings, LF and HF data instances $\mathbf{U}^{\mathtt{LF}}$ and $\mathbf{U}^{\mathtt{HF}}$ are assembled by first collecting LF and HF nodal displacements in $(0,T)$ as $\mathbf{V}_{\mathtt{LF}} = [\mathbf{d}_1^{\mathtt{LF}},\dots,\mathbf{d}_L^{\mathtt{LF}}] \in \mathbb{R}^{\mathscr{M}\times L}$ and $\mathbf{V}_{\mathtt{HF}} = [\mathbf{d}_1^{\mathtt{HF}},\dots,\mathbf{d}_L^{\mathtt{HF}}] \in \mathbb{R}^{\mathscr{M}\times L}$, from which the vibration recordings $\mathbf{U}^{\mathtt{LF}}$ and $\mathbf{U}^{\mathtt{HF}}$ are then extracted as:

$$\mathbf{U}^{\mathtt{LF}} = (\mathbf{T}\mathbf{V}_{\mathtt{LF}})^{\top} \ , \qquad \mathbf{U}^{\mathtt{HF}} = (\mathbf{T}\mathbf{V}_{\mathtt{HF}})^{\top} \ , \tag{7}$$

where $\mathbf{T} \in \mathbb{B}^{N_u \times \mathscr{M}}$ is a Boolean matrix whose (n,m)-th entry is equal to 1 only if the n-th sensor output coincides with the m-th dof.

2.3. MF-DNN surrogate: model specification and training

The MF-DNN surrogate $\mathcal{NN}_{\mathtt{MF}}$ is a composition of an LF part and an HF part, respectively termed $\mathcal{NN}_{\mathtt{LF}}$ and $\mathcal{NN}_{\mathtt{HF}}$, and is denoted as:

$$\mathcal{NN}_{\mathtt{MF}}(\mathbf{x}^{\mathtt{HF}},\mathbf{x}^{\mathtt{LF}}) = \mathcal{NN}_{\mathtt{HF}}(\mathbf{x}^{\mathtt{HF}}) \circ \mathcal{NN}_{\mathtt{LF}}(\mathbf{x}^{\mathtt{LF}}) \ , \tag{8}$$

where $\circ$ stands for function composition (see Fig. 2).

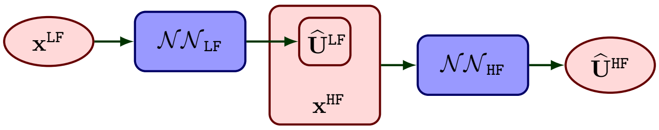

$\mathbf{x}^{\mathtt{LF}}$
$\mathcal{NN}_{\mathtt{LF}}$
$\widehat{\mathbf{U}}^{\mathtt{LF}}$
$\mathbf{x}^{\mathtt{HF}}$
$\mathcal{NN}_{\mathtt{HF}}$
$\widehat{\mathbf{U}}^{\mathtt{HF}}$

Figure 2: Scheme of the proposed MF-DNN surrogate model: red nodes denote the input/output quantities, while blue nodes refer to the learnable components of the surrogate model.

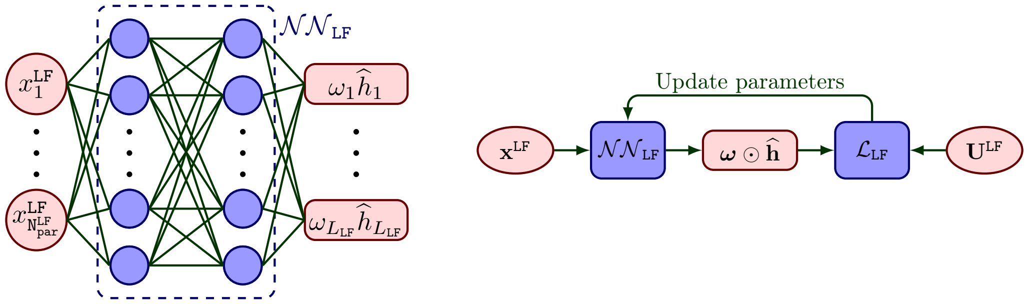
$\mathcal{NN}_{\mathtt{LF}}$ is a fully-connected DNN (see Fig. 3a) exploited to approximate the LF vibration recordings for any given LF input $\mathbf{x}^{\mathtt{LF}}$, according to:

$$\widehat{\mathbf{U}}^{\mathtt{LF}}(\mathbf{x}^{\mathtt{LF}}) = \mathtt{vec}^{-1}_{L\times N_u}[\mathbf{Y}(\frac{1}{\boldsymbol{\omega}} \odot \mathcal{NN}_{\mathtt{LF}}(\mathbf{x}^{\mathtt{LF}}))] \ , \qquad \mathcal{NN}_{\mathtt{LF}}(\mathbf{x}^{\mathtt{LF}}) = \boldsymbol{\omega} \odot \widehat{\mathbf{h}}(\mathbf{x}^{\mathtt{LF}}) \ , \tag{9}$$

where: $\mathtt{vec} : \mathbb{R}^{m\times n} \to \mathbb{R}^{mn}$ denotes the vectorization operation that converts a matrix into a column vector by stacking columns on top of one another, and $\mathtt{vec}^{-1}_{m\times n} : \mathbb{R}^{mn} \to \mathbb{R}^{m\times n}$ is its inverse; $\mathbf{Y} = [\mathbf{y}_1,\dots,\mathbf{y}_{L_{\mathtt{LF}}}] \in \mathbb{R}^{L_{\mathtt{vec}}\times L_{\mathtt{LF}}}$, with $L_{\mathtt{vec}} = LN_u$, is a matrix gathering the first $L_{\mathtt{LF}} \ll L_{\mathtt{vec}}$ POD-basis functions of the vectorized LF instances in $\mathbf{D}_{\mathtt{LF}}$, i.e. $\mathbf{D}_{\mathtt{LF}}^{\mathtt{vec}} = \{\mathtt{vec}[\mathbf{U}^{\mathtt{LF}}(\mathbf{x}^{\mathtt{LF}})]\}_{i=1}^{I_{\mathtt{LF}}}$, that is used to reduce the dimensionality of the vectorized $\mathbf{U}^{\mathtt{LF}}$ from $L_{\mathtt{vec}}$ to $L_{\mathtt{LF}}$, and therefore the number of trainable parameters of $\mathcal{NN}_{\mathtt{LF}}$; $\mathbb{R}^{L_{\mathtt{LF}}} \ni \mathbf{h}(\mathbf{x}^{\mathtt{LF}}) = [h_1(\mathbf{x}^{\mathtt{LF}}),\dots,h_{L_{\mathtt{LF}}}(\mathbf{x}^{\mathtt{LF}})] = \mathbf{Y}^{\top}\mathtt{vec}[\mathbf{U}^{\mathtt{LF}}]$ are the POD-coefficients ruling $\mathbf{U}^{\mathtt{LF}}$; hat variables denote quantities obtained from neural network approximations; $\odot$ is the (element-wise) Hadamard product; $\boldsymbol{\omega} \in \mathbb{R}^{L_{\mathtt{LF}}}$ is a vector of linearly decaying entries, that is used to weight the relative importance of the POD-basis functions as specified later in Sec. 3. Multiplying by $\boldsymbol{\omega}$ acts as a good inductive bias, see e.g. [59], encoding the notion of decreasing amount of information content described by each POD-basis function.

During training, the weights $\boldsymbol{\Omega}_{\mathtt{LF}}$ parametrizing $\mathcal{NN}_{\mathtt{LF}}$ are tuned by minimizing the following loss function:

$$\mathcal{L}_{\mathtt{LF}}(\boldsymbol{\Omega}_{\mathtt{LF}},\mathbf{D}_{\mathtt{LF}}) = \frac{1}{I_{\mathtt{LF}}}\sum_{i=1}^{I_{\mathtt{LF}}}\frac{1}{L_{\mathtt{LF}}}\|\boldsymbol{\omega}\odot(\mathbf{Y}^{\top}\mathtt{vec}[\mathbf{U}_i^{\mathtt{LF}}]) - \mathcal{NN}_{\mathtt{LF}}(\mathbf{x}_i^{\mathtt{LF}})\|_2^2 + \lambda_{\mathtt{LF}}\|\boldsymbol{\Omega}_{\mathtt{LF}}\|_2^2 \ , \tag{10}$$


$\mathcal{NN}_{\text{LF}}$
x_1^{LF}
$x_{N_{\text{par}}^{\text{LF}}}^{\text{LF}}$
$\omega_1 \widehat{h}_1$
$\omega_{L_{\text{LF}}} \widehat{h}_{L_{\text{LF}}}$
Update parameters
$\mathbf{x}^{\text{LF}}$
$\mathcal{NN}_{\text{LF}}$
$\boldsymbol{\omega} \odot \widehat{\mathbf{h}}$
$\mathcal{L}_{\text{LF}}$
$\mathbf{U}^{\text{LF}}$

(a)

(b)

Figure 3: $\mathcal{NN}_{\texttt{LF}}$ component: (a) scheme of the fully-connected model; (b) training flowchart.

where $\lambda_{\texttt{LF}}$ is the L^2 regularization rate over the model parameters $\mathbf{\Omega}_{\texttt{LF}}$. The relevant training process is sketched in Fig. 3b.

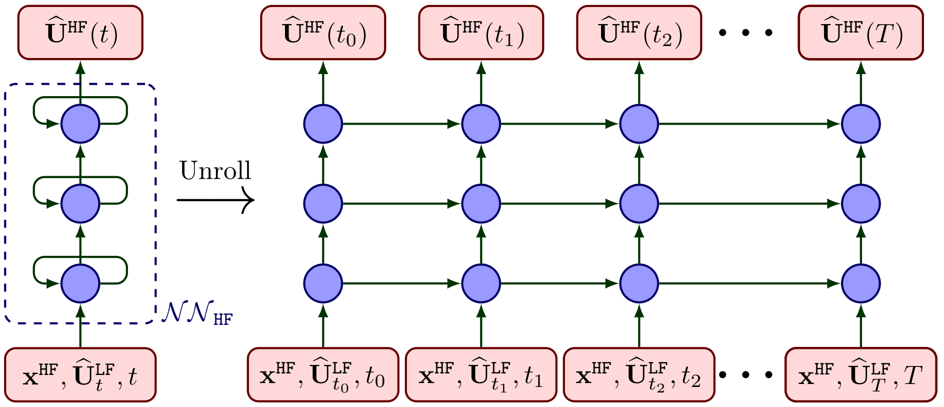
The $\mathcal{NN}_{\texttt{HF}}$ is instead a DNN built upon the LSTM model (see Fig. 4a) that, being more appropriate than fully-connected ones for time series and time-dependent problems [47, 60, 61], is used to exploit the time correlation between the two fidelity levels. Recurrent neural networks (RNNs), such as the LSTM here adopted, can handle input sequences of variable length thanks to a hidden state variable that exploits the temporal ordering in the information flow. The LSTM cell model [62] is a sophisticated RNN that has been introduced to better capture long-term dependencies, by means of an internal gating mechanism that enables a memory-like behavior [63].

$\mathcal{NN}_{\texttt{HF}}$ enriches the LF signals $\widehat{\mathbf{U}}^{\texttt{LF}}$ provided by $\mathcal{NN}_{\texttt{LF}}$ with the effects of damage and structural damping, according to the given input parameters $\mathbf{x}^{\texttt{HF}}$ (see Fig. 2), as:

$$\widehat{\mathbf{U}}^{\texttt{HF}}(\mathbf{x}^{\texttt{HF}}, \mathbf{x}^{\texttt{LF}}, t) = \mathcal{NN}_{\texttt{HF}}(\mathbf{x}^{\texttt{HF}}, \widehat{\mathbf{U}}^{\texttt{LF}}(\mathbf{x}^{\texttt{LF}}), t) \ . \tag{11}$$

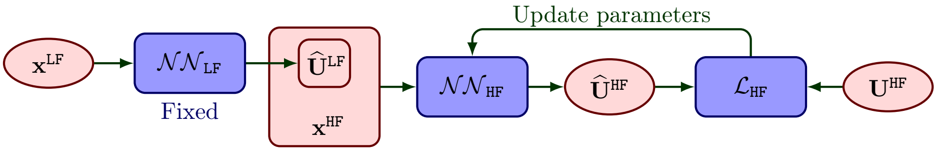
At each time step, $\mathcal{NN}_{\texttt{HF}}$ takes the HF input parameters $\mathbf{x}^{\texttt{HF}}$, the current time instant t, the corresponding LF approximation $\widehat{\mathbf{U}}_t^{\texttt{LF}}$ provided by the trained $\mathcal{NN}_{\texttt{LF}}$, and returns its HF approximation $\widehat{\mathbf{U}}^{\texttt{HF}}(t)$, see Fig. 4a. The choice of adopting an LSTM model for $\mathcal{NN}_{\texttt{HF}}$ stems from the possibility offered by the considered MF setup to exploit the temporal structure of the $\widehat{\mathbf{U}}^{\texttt{LF}}$ provided as input; on the other hand, $\mathcal{NN}_{\texttt{LF}}$ does not have access to any input time series related to the target structural response, and exploiting a fully-connected model that maps onto POD-coefficients has proved to be a more effective strategy.

As the hidden units in RNNs depend on previous time steps, a back propagation through time algorithm is necessary to compute the gradients for training. However, in order to avoid an excessive computational burden when backpropagating the error, the actual length of vibration recordings is chunked into overlapping smaller sequences treated as independent training samples. Each chunk includes $L_{\texttt{C}}$ time steps for a duration of $T_{\texttt{C}} = (L_{\texttt{C}} - 1)/\not{f}$, so that a total of $N_{\texttt{C}} = L - L_{\texttt{C}} + 1$ chunks is obtained from each recording. The chunks length $L_{\texttt{C}}$ is a hyperparameter to be chosen for optimal computational performance; $L_{\texttt{C}} = 20$ is adopted in the present work, as it provides satisfactory results.



$\widehat{\mathbf{U}}^{\mathrm{HF}}(t)$
$\widehat{\mathbf{U}}^{\mathrm{HF}}(t_0)$
$\widehat{\mathbf{U}}^{\mathrm{HF}}(t_1)$
$\widehat{\mathbf{U}}^{\mathrm{HF}}(t_2)$
$\widehat{\mathbf{U}}^{\mathrm{HF}}(T)$
Unroll
$\mathcal{NN}_{\mathtt{HF}}$
$\mathbf{x}^{\mathrm{HF}}, \widehat{\mathbf{U}}^{\mathrm{LF}}_t, t$
$\mathbf{x}^{\mathrm{HF}}, \widehat{\mathbf{U}}^{\mathrm{LF}}_{t_0}, t_0$
$\mathbf{x}^{\mathrm{HF}}, \widehat{\mathbf{U}}^{\mathrm{LF}}_{t_1}, t_1$
$\mathbf{x}^{\mathrm{HF}}, \widehat{\mathbf{U}}^{\mathrm{LF}}_{t_2}, t_2$
$\mathbf{x}^{\mathrm{HF}}, \widehat{\mathbf{U}}^{\mathrm{LF}}_T, T$

(a)



Update parameters
$\mathbf{x}^{\mathrm{LF}}$
$\mathcal{NN}_{\mathtt{LF}}$
Fixed
$\widehat{\mathbf{U}}^{\mathrm{LF}}$
$\mathbf{x}^{\mathrm{HF}}$
$\mathcal{NN}_{\mathtt{HF}}$
$\widehat{\mathbf{U}}^{\mathrm{HF}}$
$\mathcal{L}_{\mathtt{HF}}$
$\mathbf{U}^{\mathrm{HF}}$

(b)

Figure 4: $\mathcal{NN}_{\mathtt{HF}}$ component: (a) scheme of the recurrent model; (b) training flowchart.

The weights $\mathbf{\Omega}_{\mathtt{HF}}$ parametrizing $\mathcal{NN}_{\mathtt{HF}}$ are tuned by minimizing the following loss function:

$$\begin{aligned}\mathcal{L}_{\mathtt{HF}}(\mathbf{\Omega}_{\mathtt{HF}}, \mathbf{D}_{\mathtt{HF}}) = \; & \frac{1}{I_{\mathtt{HF}}} \frac{1}{N_{\mathtt{C}}} \sum_{j=1}^{I_{\mathtt{HF}}} \sum_{\tau=1}^{N_{\mathtt{C}}} \frac{1}{N_u} \frac{1}{L_{\mathtt{C}}} \| \mathtt{vec}[\mathbf{U}^{\mathrm{HF}}_{j,\tau:\tau+L_{\mathtt{C}}} - \\ & \mathcal{NN}_{\mathtt{HF}}(\mathbf{x}^{\mathrm{HF}}_j, \widehat{\mathbf{U}}^{\mathrm{LF}}_{\tau:\tau+L_{\mathtt{C}}}(\mathbf{x}^{\mathrm{LF}}_j), t_{\tau:\tau+L_{\mathtt{C}}})]\|_1 + \lambda_{\mathtt{HF}} \|\mathbf{\Omega}_{\mathtt{HF}}\|_2^2 \;,\end{aligned} \tag{12}$$

and the relevant training process is carried out as sketched in Fig. 4b, with the weights $\mathbf{\Omega}_{\mathtt{LF}}$ of $\mathcal{NN}_{\mathtt{LF}}$ kept fixed. In Eq. (12), it is to note the adoption of a mean absolute error-like term for adjusting the prediction accuracy of $\mathcal{NN}_{\mathtt{HF}}$, which has proved to be superior than a mean squared error-like one, as that exploited to train $\mathcal{NN}_{\mathtt{LF}}$. We argue that this behavior stems from the lower weighting of atypical training data instances, such as vibration recordings obtained by exciting the structure close to resonance frequencies, which counteracts the effect of data noise by exploiting a mechanism different from that of the regularization term.

The steps of the proposed MF-DNN surrogate modeling strategy for SHM purposes are outlined in Fig. 5: definition of a parametric LF-FOM; construction of a LF-ROM that takes into account the parametric and time dependence of the system by means of POD; population of the LF training dataset with synthetic vibration recordings at sensor location via LF-ROM simulations; training and validation of the LF component $\mathcal{NN}_{\mathtt{LF}}$, adopted to approximate the LF vibration recordings for any given LF input parameters; testing the generalization capabilities of $\mathcal{NN}_{\mathtt{LF}}$ on LF-FOM data; definition of a parametric HF structural model that takes into account the effects of damage (and structural damping); population of the HF training dataset; training and validation of the HF component $\mathcal{NN}_{\mathtt{HF}}$, adopted to enrich the LF approximation provided by $\mathcal{NN}_{\mathtt{LF}}$ with the effects of damage (and structural damping), for any given HF input parameters; testing the generalization capabilities of $\mathcal{NN}_{\mathtt{MF}}$.

The proposed multi-level architecture of $\mathcal{NN}_{\mathtt{MF}}$ is chosen in place of an all-in-one scheme,

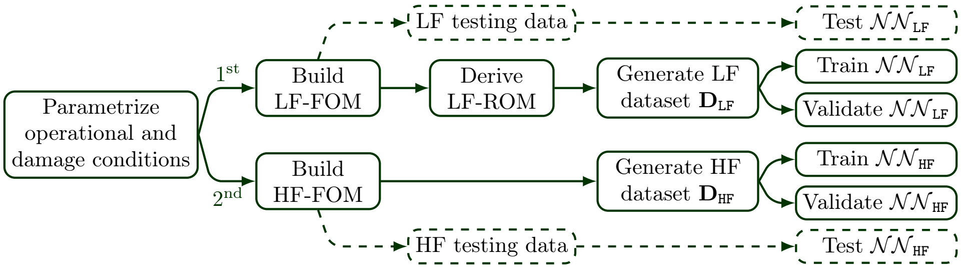

Parametrize operational and damage conditions
1st
2nd
Build LF-FOM
Build HF-FOM
Derive LF-ROM
LF testing data
HF testing data
Generate LF dataset $\mathbf{D}_{\mathtt{LF}}$
Generate HF dataset $\mathbf{D}_{\mathtt{HF}}$
Test $\mathcal{NN}_{\mathtt{LF}}$
Train $\mathcal{NN}_{\mathtt{LF}}$
Validate $\mathcal{NN}_{\mathtt{LF}}$
Train $\mathcal{NN}_{\mathtt{HF}}$
Validate $\mathcal{NN}_{\mathtt{HF}}$
Test $\mathcal{NN}_{\mathtt{HF}}$

Figure 5: MF-DNN surrogate modeling: methodology flowchart.

relying on a single DNN to approximate both LF and HF vibration data. The main reason behind this choice is that training the latter involves a nontrivial multi-objective optimization problem. For instance, in [44] an all-in-one approach has been exploited and the contributions from two fidelity levels, in terms of loss function, have been combined through a weighted sum. However, it has to be highlighted the risk of potential numerical instabilities induced by large discrepancies between the two components, as well as the need of tuning the relative mixing ratio. Moreover, in [47], an all-in-one architecture (termed “intermediate model”) trained by jointly exploiting LF and HF data was also shown to underperform alternative MF models, tuned through a sequential learning of the LF approximation and the HF refinement.

3. Results: validation and verification of surrogate modeling

In what follows, two different systems are considered as they feature a much different structural complexity. While the former one is going to be further considered in Sec. 4 to discuss the capability granted by the proposed MF scheme to identify a damage pattern, the latter is adopted to discuss the computational benefits of the strategy and next highlight possible issues regarding its generalization capabilities for systems characterized by a remarkable structural/geometrical complexity. For the considered case studies, the HF and LF numerical models are implemented in the `Matlab` environment, using the `redbKIT` library [64]. All computations are carried out on a PC featuring an `AMD Ryzen™ 9 5950X` CPU @ 3.4 GHz and 128 GB RAM. The NN architectures are implemented through the `Tensorflow`-based `Keras` API [65], and trained on a single `Nvidia GeForce RTX™ 3080` GPU card.

3.1. MF-DNN surrogate: implementation details

We start by discussing the implementation details common to both the structural systems addressed in the present study. $\mathcal{NN}_{\mathtt{LF}}$ is set as a 9-layers fully-connected DNN with residual connections, whose architecture is described in Tab. 1a. All layers feature 30 neurons, except the last two that are equipped with $2L_{\mathtt{LF}}$ and $L_{\mathtt{LF}}$ neurons, respectively. Moreover, two double-layer identity residual connections [66] are exploited to perform a progressive refinement of the outcome. Residual connections help the DNN training to converge more easily, avoiding the potential vanishing/exploding gradients issue, and mitigate the accuracy saturation problem while increasing the network depth [67, 68]. They have proved effective in enhancing the $\mathcal{NN}_{\mathtt{LF}}$ performance without adding extra parameters, when further improvements were no longer achievable through deeper architectures. No activation is applied to the last layer, while the powerful PReLU [69] is employed as activation function in all other layers.

Adopting the He's weight initialization [69], the loss function $\mathcal{L}_{\mathtt{LF}}$ is minimized using Adam [70] for a maximum of $10,000$ allowed epochs. The learning rate $\eta_{\mathtt{LF}}$ is initially set to 0.005, and decreased for 4/5 of the allowed training steps using a cosine decay schedule with weight decay equal to 0.02. The optimization is carried out by considering an 80 : 20 splitting ratio for training and validation purposes; thus, 20% of the data in $\mathbf{D}_{\mathtt{LF}}$ is randomly taken and set aside to validate the learning process by means of an early stopping strategy. The latter is adopted to prevent overfitting by interrupting learning whenever the loss function value attained on the validation set does not decrease for a prescribed number of epochs in a row. The relevant hyperparameters and training options are reported in Tab. 1b.

Table 1: $\mathcal{NN}_{\mathtt{LF}}$ - (a) adopted architecture, and (b) selected hyperparameters and training options.

(a)

Layer	Type	Output shape	Activ.	Input layer
0	Input	$(B_{\mathtt{LF}}, \mathtt{N}^{\mathtt{LF}}_{\text{par}})$	None	None
1	Dense	$(B_{\mathtt{LF}}, 30)$	PReLU	0
2	Dense	$(B_{\mathtt{LF}}, 30)$	PReLU	1
3	Dense	$(B_{\mathtt{LF}}, 30)$	PReLU	2
4	Dense	$(B_{\mathtt{LF}}, 30)$	PReLU	3
5	Add	$(B_{\mathtt{LF}}, 30)$	None	2,4
6	Dense	$(B_{\mathtt{LF}}, 30)$	PReLU	5
7	Dense	$(B_{\mathtt{LF}}, 30)$	PReLU	6
8	Dense	$(B_{\mathtt{LF}}, 30)$	PReLU	7
9	Add	$(B_{\mathtt{LF}}, 30)$	None	6,8
10	Dense	$(B_{\mathtt{LF}}, 2L_{\mathtt{LF}})$	PReLU	9
11	Dense	$(B_{\mathtt{LF}}, L_{\mathtt{LF}})$	None	10

(b)

Weight initializer:	He
Optimizer:	Adam
Batch size:	$B_{\mathtt{LF}} = 64$
Initial learning rate:	$\eta_{\mathtt{LF}} = 0.005$
Allowed epochs:	10.000
Learning schedule:	$\frac{4}{5}$ cosine decay
Weight decay:	0.02
Early stop patience:	500 epochs
Train-val split:	80 : 20

$\mathcal{NN}_{\mathtt{HF}}$ consists of four LSTM layers, respectively featuring a cell state of size $16, 16, 32, N_u$, and of a time distributed fully-connected output layer equipped with N_u neurons. No activation is applied to the dense layer, while the LSTM layers feature standard cells with sigmoidal gating functions and hyperbolic tangent (tanh) activated cell states. The architecture of $\mathcal{NN}_{\mathtt{HF}}$ is schematically reported in Tab. 2a.

In this case, the optimization is carried out using Adam together with the Xavier's weight initialization [71]. Similarly to what specified for $\mathcal{NN}_{\mathtt{LF}}$, the learning rate $\eta_{\mathtt{HF}}$ is decreased as the training advances using a cosine decay schedule, and an early stop strategy is adopted to prevent overfitting by considering an 80 : 20 splitting ratio for training and validation purposes. The relevant hyperparameters and the training options are summarized in Tab. 2b.

These architectures, as well as the relevant hyperparameters and training options, have been selected through a preliminary sensitivity study, aimed at minimizing $\mathcal{L}_{\mathtt{LF}}$ and $\mathcal{L}_{\mathtt{HF}}$ while retaining the generalization capabilities of $\mathcal{NN}_{\mathtt{LF}}$ and $\mathcal{NN}_{\mathtt{HF}}$ for the case studies considered in the next sections, paying specific attention to avoid overfitting of the training data.

3.2. L-shaped cantilever beam

The first test case considered to assess the performance of the proposed MF-DNN surrogate model is the L-shaped cantilever beam depicted in Fig. 6. The structure is assumed to be made of concrete, with mechanical properties: Young's modulus $E = 30$ GPa, Poisson's ratio $\nu = 0.2$, density $\rho = 2500$ kg/m^3. The structure is excited by a distributed vertical load $q(t)$, acting on an area of (0.3×0.3) m^2 close to its tip, see Fig. 6, and varying in time according

Table 2: $\mathcal{NN}_{\texttt{HF}}$ - (a) adopted architecture, and (b) selected hyperparameters and training options.

(a)

Layer	Type	Output shape	Activ.	Input layer
0	Input	$(B_{\texttt{HF}}, L_{\texttt{C}}, \texttt{N}_{\text{par}}^{\texttt{HF}} + N_u + 1)$	None	None
1	LSTM	$(B_{\texttt{HF}}, L_{\texttt{C}}, 16)$	Usual	0
2	LSTM	$(B_{\texttt{HF}}, L_{\texttt{C}}, 16)$	Usual	1
3	LSTM	$(B_{\texttt{HF}}, L_{\texttt{C}}, 32)$	Usual	2
4	LSTM	$(B_{\texttt{HF}}, L_{\texttt{C}}, N_u)$	Usual	3
5	Dense	$(B_{\texttt{HF}}, N_u)$ - distributed	None	4

(b)

Weight initializer:	Xavier
Optimizer:	Adam
Batch size:	$B_{\texttt{HF}} = 128$
Initial learning rate:	$\eta_{\texttt{HF}} = 0.001$
Allowed epochs:	$\{2000, 5000\}$
Learning schedule:	$\frac{4}{5}$ cosine decay
Weight decay:	0.05
Early stop patience:	100 epochs
Chunks length:	$L_{\texttt{C}} = 20$
Train-val split:	80 : 20

to $q(t) = Q \sin(2\pi f t)$, with $Q \in [1, 5]$ kPa and $f \in [10, 60]$ Hz being the load amplitude and frequency, respectively. The previous notation used to specify the ranges in which Q and f can take values, implicitly assumes that a uniform probability distribution is adopted to describe them. This first test case aims at illustrating the use of the proposed MF surrogate modeling framework against a simple potential monitoring scenario. A further purpose is to analyze the impact of different choices of the $\boldsymbol{\omega}$ weighting vector on the performance of the $\mathcal{NN}_{\texttt{LF}}$, and to identify the option providing the best results.

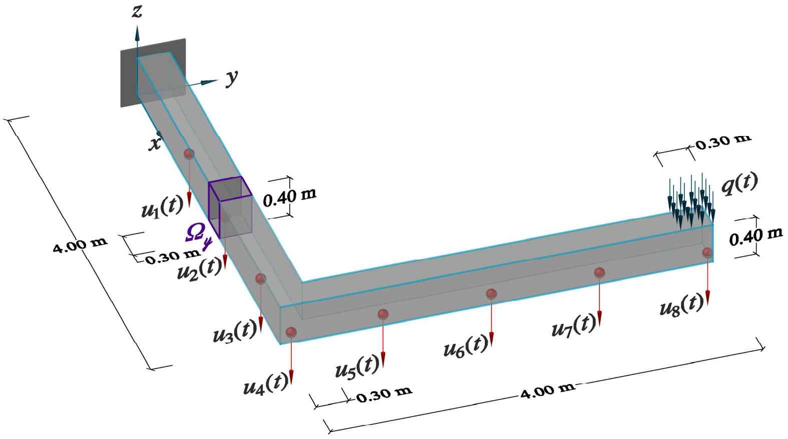


Figure 6: L-shaped cantilever beam: physics-based digital twin, with details of synthetic recordings related to displacements $u_1(t), \dots, u_8(t)$, loading condition, and damaged region $\Omega_{\boldsymbol{y}}$.

Structural displacement time histories $\mathbf{U}^{\text{LF}}(\mathbf{x}^{\text{LF}})$ and $\mathbf{U}^{\text{HF}}(\mathbf{x}^{\text{HF}})$ are supposed to be recorded by means of $N_u = 8$ dofs along the bottom surface of the structure, to mimic a monitoring system arranged as depicted in Fig. 6. Both LF and HF recordings are provided for a time interval $(0, T = 1 \text{ s})$ with an acquisition frequency of $\mathscr{f} = 200$ Hz.

The HF numerical model is obtained from a finite element discretization using linear tetrahedral elements and resulting in $\mathscr{M} = 4659$ dofs. The damping matrix is assembled according to the Rayleigh's model, to account for a 5% damping ratio on the first four structural modes. Damage is simulated by reducing the material stiffness within a subdomain $\Omega_{\boldsymbol{y}}$ of $(0.3 \times 0.3 \times 0.4)$ m^3 in size, whose position is parametrized by the coordinates of its center of mass $\boldsymbol{y} = (x_\Omega, y_\Omega)$, with either x_Ω or y_Ω varying in the range $[0.15, 3.85]$ m. The damage level is set to $\delta = 25\%$ and held constant within the time interval $(0, T = 1 \text{ s})$. Accordingly, the vector of HF input parameters is $\mathbf{x}^{\mathtt{HF}} = \{Q, f, x_\Omega, y_\Omega\}^\top$.

The projection basis $\mathbf{W}$ ruling the LF-ROM is instead computed from a snapshot matrix $\mathbf{S}$ collecting $\mathscr{S} = 40,200$ snapshots, obtained through 200 evaluations of a LF-FOM not accounting for damage and damping. Simulations are carried out for different values of LF input parameters $\mathbf{x}^{\mathtt{LF}} = \{Q, f\}^\top$ sampled via the latin hypercube rule. By prescribing a tolerance $\epsilon = 10^{-3}$ in Eq. (6), the order of the LF model is set to $\mathscr{M}_{\mathtt{LF}} = 14$, in place of the original $\mathscr{M} = 4659$ dofs. Thanks to this reduction, the computing time required to assemble each LF instance decreases from 4.90 s to 0.19 s, of which only 0.024 s is needed to advance the solution in time, entailing a speed-up of about 25.8 times.

For the case at hand, $I_{\mathtt{LF}} = 10,000$ LF data instances are collected to train $\mathcal{NN}_{\mathtt{LF}}$, while only $I_{\mathtt{HF}} = 1000$ additional HF data instances are exploited to train $\mathcal{NN}_{\mathtt{HF}}$, respectively for 10,000 and 5000 epochs. Concerning the dimensionality reduction applied to the LF recordings in order to train $\mathcal{NN}_{\mathtt{LF}}$, a projection matrix $\mathbf{Y}$ comprising $L_{\mathtt{LF}} = 104$ POD-basis functions is obtained in place of the original $L_{\mathtt{vec}} = 1608$ data points, by performing an SVD on the vectorized LF dataset $\mathbf{D}_{\mathtt{LF}}^{\mathtt{vec}}$, i.e. $\mathbf{D}_{\mathtt{LF}}^{\mathtt{vec}} = \{\mathtt{vec}[\mathbf{U}_i^{\mathtt{LF}}(\mathbf{x}_i^{\mathtt{LF}})]\}_{i=1}^{I_{\mathtt{LF}}}$, and by choosing a tolerance $\epsilon = 10^{-3}$.

Regarding the vector $\boldsymbol{\omega}$, we argue that an appropriate weighting of the POD-coefficients ruling $\mathbf{U}^{\mathtt{LF}}$, such to prioritize the regression error over the POD-coefficients encoding a higher information content, would be better than a constant $\boldsymbol{\omega}$ choice. In order to provide some insights on such a selection of $\boldsymbol{\omega}$, the performance of $\mathcal{NN}_{\mathtt{LF}}$ is systematically assessed. Results are reported in Tab. 3 considering three alternative $\boldsymbol{\omega}$ weighting vectors: a flat one, with all unit entries, so that $\boldsymbol{\omega} = \mathbf{1}$; a linearly decaying one, with entries $\omega_l = 1 - \frac{0.8}{L_{\mathtt{LF}}-1}(l-1)$, $l = 1, \ldots, L_{\mathtt{LF}}$; one ruled by the singular values of $\mathbf{D}_{\mathtt{LF}}^{\mathtt{vec}}$, according to $\boldsymbol{\omega} = (\sigma_l(\mathbf{D}_{\mathtt{LF}}^{\mathtt{vec}}))_{l=1}^{L_{\mathtt{LF}}}$. Results are reported in terms of average and minimum Pearson correlation coefficient $\mathtt{PCC}_{\mathtt{avg}}$ and $\mathtt{PCC}_{\mathtt{min}}$ between predicted and ground truth LF signals, and in terms of mean absolute error $\mathtt{MAE}$ between the same LF signals. The testing data consist of $I_{\mathtt{LF}}^{\mathtt{T}} = 40$ LF data instances, and the sensor recordings $u_1(t)$, $u_4(t)$ and $u_8(t)$ are considered to assess the $\mathcal{NN}_{\mathtt{LF}}$ performance. The vector with linearly decaying entries provides the best results, while the worst results are obtained by weighting the POD-coefficients by the corresponding singular values. Although encoding the notion of decreasing importance of the POD-modes yields an improvement in all the considered metrics, the $\mathcal{NN}_{\mathtt{LF}}$ performance is more than satisfactory even when the flat $\boldsymbol{\omega}$ is used, with attained values of $\mathtt{PCC}_{\mathtt{min}}$ above 0.95 and those of $\mathtt{PCC}_{\mathtt{avg}}$ even close to one, which suggests a strong, positive association between predicted and target LF signals. It is interesting to note that the values of $\mathtt{PCC}_{\mathtt{min}}$ also increase when the first POD-modes are prioritized, which might look surprising and is likely a symptom of the fact that ϵ can be increased (or, equivalently, $L_{\mathtt{LF}}$ can be reduced), without significantly spoiling the approximated LF signals. Adopting instead the singular values of $\mathbf{D}_{\mathtt{LF}}^{\mathtt{vec}}$ yields a degradation in all the considered metrics, with attained values of $\mathtt{PCC}_{\mathtt{min}}$ even below of 0.65, and values of $\mathtt{MAE}$ more than four times higher than those obtained with the linearly decaying $\boldsymbol{\omega}$; therefore, it can be argued that the beneficial effect of such weighting is compromised if high POD-modes are excessively penalized. According to this analysis, the linearly decaying $\boldsymbol{\omega}$ weighting vector with extreme values 1 and 0.2 is adopted in the reminder of this work.

To demonstrate the reconstruction capabilities achieved by the surrogate model, some

Table 3: L-shaped cantilever beam - $\mathtt{PCC}_{\mathtt{avg}}$, $\mathtt{PCC}_{\mathtt{min}}$ and $\mathtt{MAE}$ performance indicators, respectively measuring the average and minimum Pearson correlation coefficients, and the mean absolute error between the predicted and ground truth LF testing signals, for the $u_1(t)$, $u_4(t)$ and $u_8(t)$ monitored dofs and for three considered choices of the $\boldsymbol{\omega}$ weighting vector: flat; linearly decaying; ruled by the singular values of $\mathbf{D}_{\mathrm{LF}}^{\mathtt{vec}}$.

$\boldsymbol{\omega}$	$\mathtt{PCC}_{\mathtt{avg}}(u_1(t), u_4(t), u_8(t))$	$\mathtt{PCC}_{\mathtt{min}}(u_1(t), u_4(t), u_8(t))$	$\mathtt{MAE}(u_1(t), u_4(t), u_8(t))$
$\mathbf{1}$	$\{0.994, 0.995, 0.999\}$	$\{0.954, 0.965, 0.992\}$	$\{0.319, 2.179, 2.737\} \cdot 10^{-6}$
$(1 - \frac{0.8}{L_{\mathrm{LF}}-1}(l-1))_{l=1}^{L_{\mathrm{LF}}}$	$\{0.998, 0.998, 0.999\}$	$\{0.992, 0.986, 0.994\}$	$\{0.196, 1.481, 2.185\} \cdot 10^{-6}$
$(\sigma_l(\mathbf{D}_{\mathrm{LF}}^{\mathtt{vec}}))_{l=1}^{L_{\mathrm{LF}}}$	$\{0.884, 0.971, 0.980\}$	$\{0.628, 0.911, 0.901\}$	$\{1.193, 5.012, 8.821\} \cdot 10^{-6}$

examples of signals provided by $\mathcal{NN}_{\mathtt{MF}}$ for the monitored dofs $u_1(t)$, $u_4(t)$ and $u_8(t)$ are reported in Fig. 7. These graphs report the ground truth HF displacement time series together with the LF and HF approximations, attained for varying operational and damage conditions. From a qualitative point of view, the HF approximation always matches the target signal both in the low and high frequency components. As a comparison, the same results are also reported in Fig. 8 as obtained with the single-fidelity counterpart of $\mathcal{NN}_{\mathtt{MF}}$, termed $\mathcal{NN}_{\mathtt{HF}}^{\mathtt{only}}$ in the following; this latter is identical to $\mathcal{NN}_{\mathtt{HF}}$ except that the input channels providing $\widehat{\mathbf{U}}^{\mathrm{LF}}$ are removed, so that it cannot take advantage of $\mathcal{NN}_{\mathtt{LF}}$. The obtained results testify that preventing $\mathcal{NN}_{\mathtt{HF}}^{\mathtt{only}}$ from exploiting the correlation between HF and LF recordings has a highly negative impact on it approximation capabilities. Despite the significant discrepancy between LF approximations and target HF signals, it is interesting to note how $\mathcal{NN}_{\mathtt{LF}}$ seems to play a crucial role in keeping correct trajectories without cumulating errors in the long-time range, especially in the case of signals rich in high frequency components. To quantify the performance gain provided by the LF component, the $\mathtt{PCC}_{\mathtt{avg}}$, $\mathtt{PCC}_{\mathtt{min}}$ and $\mathtt{MAE}$ indicators between predicted and ground truth HF signals are computed for the same sample channels, and are reported in Tab. 4 as obtained against a testing set made of $I_{\mathrm{HF}}^{\mathtt{T}} = 40$ HF data instances by considering both $\mathcal{NN}_{\mathtt{HF}}$ and $\mathcal{NN}_{\mathtt{HF}}^{\mathtt{only}}$. When $\mathcal{NN}_{\mathtt{HF}}^{\mathtt{only}}$ is adopted in place of $\mathcal{NN}_{\mathtt{HF}}$, large impoverishments are shown in all the considered performance measures; for instance, the attained values of $\mathtt{PCC}_{\mathtt{min}}$, which are all above 0.95 in the MF setting, drop to 0.215, 0.467 and 0.685, respectively for dofs $u_1(t)$, $u_4(t)$ and $u_8(t)$, while the corresponding $\mathtt{MAE}$ values increase of one order of magnitude. Intuitively, this loss of accuracy is due to the fact that, $\mathcal{NN}_{\mathtt{HF}}^{\mathtt{only}}$ needs to learn not only how to enrich the LF signals with the effects of damage and structural damping, but to entirely reproduce the HF input-output behavior.

Table 4: L-shaped cantilever beam - $\mathtt{PCC}_{\mathtt{avg}}$, $\mathtt{PCC}_{\mathtt{min}}$ and $\mathtt{MAE}$ performance indicators, respectively measuring the average and minimum Pearson correlation coefficients, and the mean absolute error between predicted and ground truth HF testing signals, for the $u_1(t)$, $u_4(t)$ and $u_8(t)$ monitored dofs.

Surrogate model	$\mathtt{PCC}_{\mathtt{avg}}(u_1(t), u_4(t), u_8(t))$	$\mathtt{PCC}_{\mathtt{min}}(u_1(t), u_4(t), u_8(t))$	$\mathtt{MAE}(u_1(t), u_4(t), u_8(t))$
$\mathcal{NN}_{\mathtt{MF}}$	$\{0.997, 0.999, 0.999\}$	$\{0.987, 0.993, 0.994\}$	$\{0.804, 6.130, 9.055\} \cdot 10^{-7}$
$\mathcal{NN}_{\mathtt{HF}}^{\mathtt{only}}$	$\{0.677, 0.823, 0.908\}$	$\{0.215, 0.467, 0.685\}$	$\{0.773, 5.894, 10.815\} \cdot 10^{-6}$

Finally, we compare the overall computational times required by $\mathcal{NN}_{\mathtt{MF}}$ and the HF numerical model to approximate the $I_{\mathrm{HF}}^{\mathtt{T}} = 40$ HF testing instances. The CPU time required by the FEM solver amounts to 196 s, while, the testing GPU time required by the $\mathcal{NN}_{\mathtt{MF}}$ reduce to 0.054 s. Therefore, $\mathcal{NN}_{\mathtt{MF}}$ run considerably faster than real-time, with a speed-up of about 3630 times with respect to the HF numerical model.

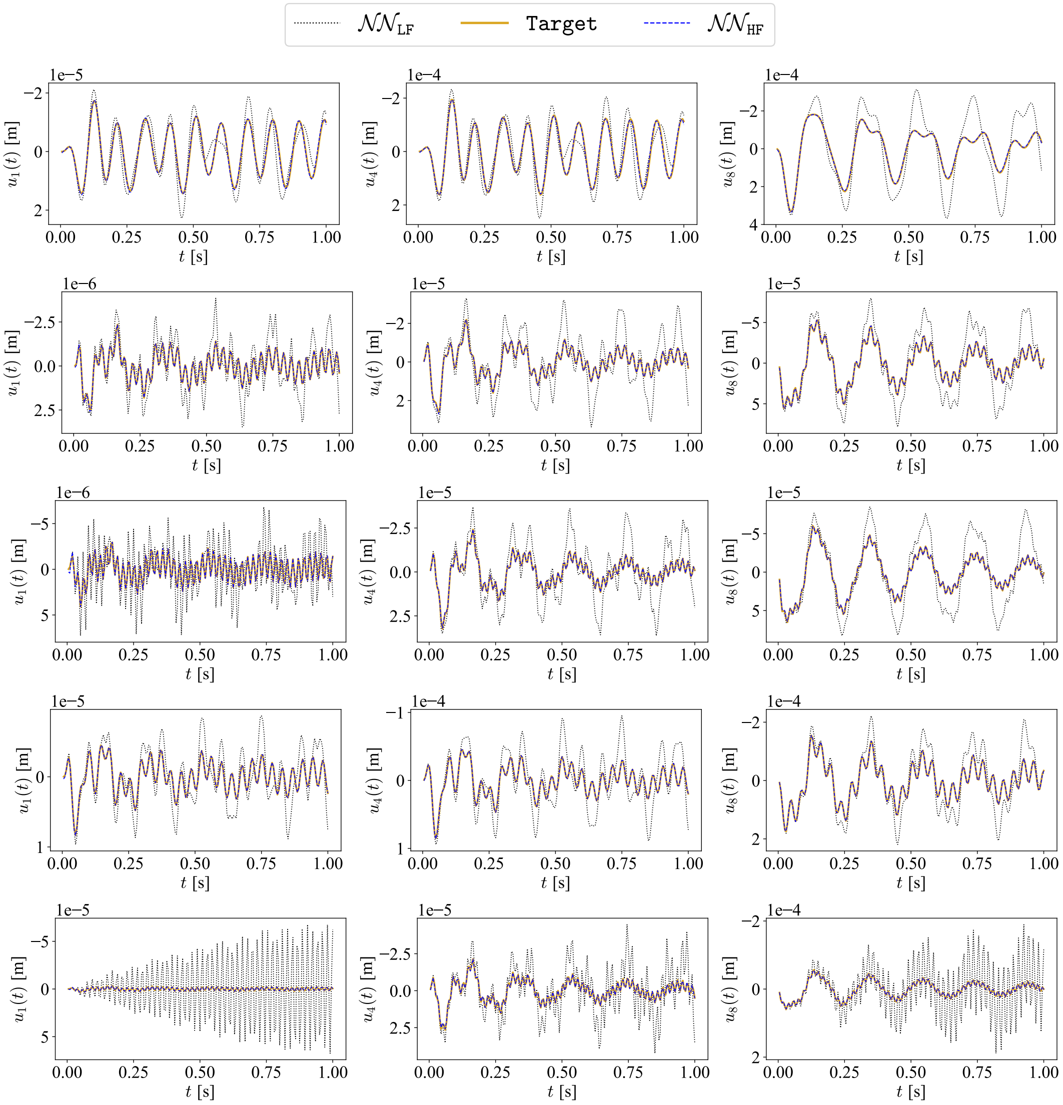


Figure 7: L-shaped cantilever beam - exemplary comparisons of $u_1(t)$, $u_4(t)$ and $u_8(t)$ displacement time histories provided by $\mathcal{NN}_{\text{LF}}$ and $\mathcal{NN}_{\text{HF}}$ with the target HF signals. Results in the five rows are obtained for values of the input parameters taken from the following vectors: $Q = \{2.37, 2.61, 4.56, 4.71, 4.90\}$ kPa; $f = \{10.32, 35.10, 46.93, 26.23, 53.26\}$ Hz; $x_\Omega = \{1.70, 3.18, 3.85, 3.85, 3.85\}$ m; $y_\Omega = \{0.15, 0.15, 0.76, 1.28, 1.93\}$ m.

3.3. Railway bridge

The second case study aims to assess the performance of the proposed surrogate modeling strategy in the more involved situation of the railway bridge depicted in Fig. 9. It is an integral concrete portal frame bridge located along the Bothnia line in the Swedish suburbs of Hörnefors, that has been already considered by the authors in [19, 56]. It features a span of 15.7 m, a free height of 4.7 m and a width of 5.9 m (edge beams excluded). The thickness of the structural elements is 0.5 m for the deck, 0.7 m for the frame walls, and 0.8 m for the wing walls. The bridge is founded on two plates connected by stay beams and supported by pile groups. The concrete is of class C35/45, whose mechanical properties are: $E = 34$ GPa,

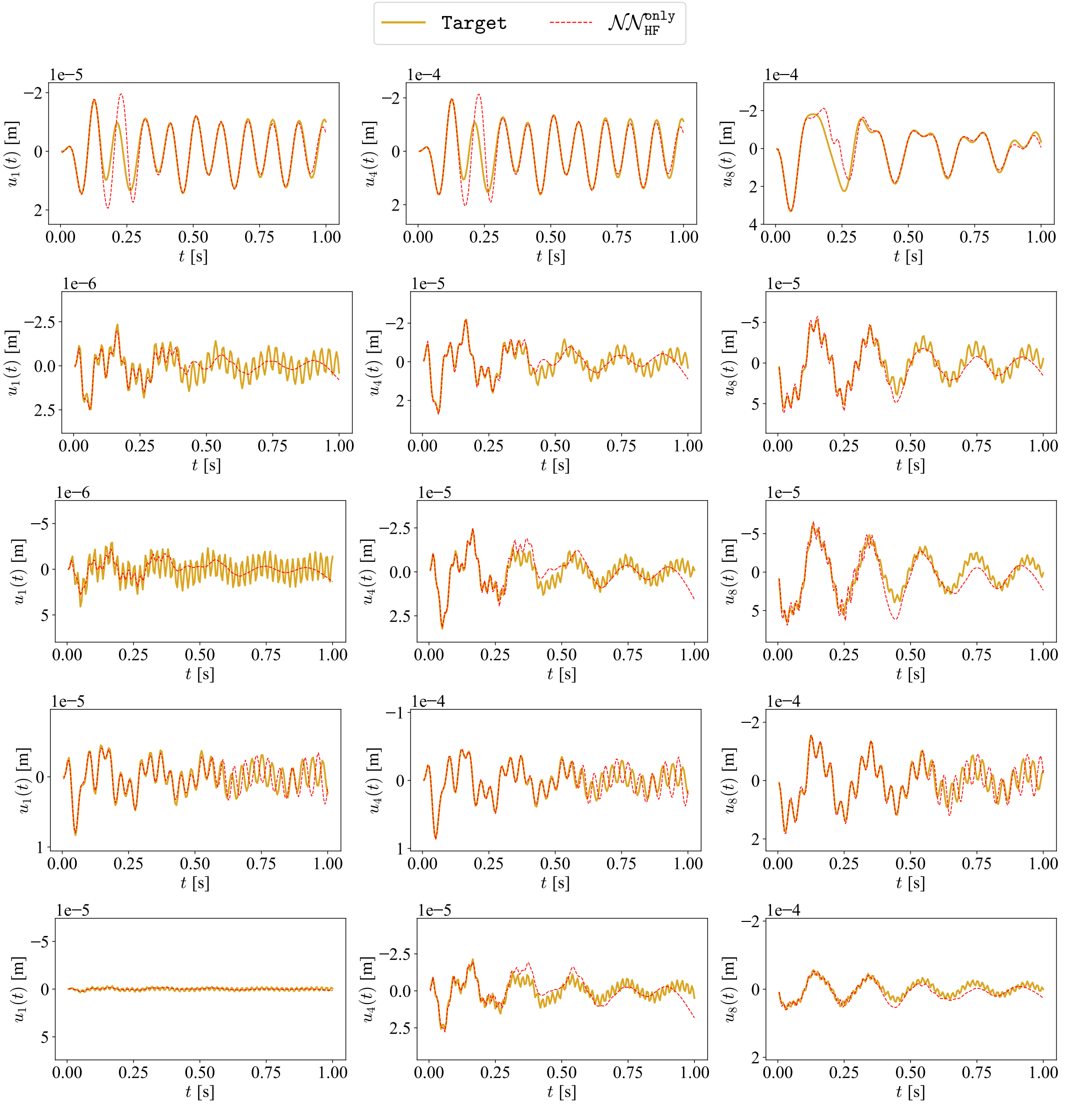


Figure 8: L-shaped cantilever beam - exemplary comparisons of $u_1(t)$, $u_4(t)$ and $u_8(t)$ displacement time histories provided by $\mathcal{NN}_{\mathtt{HF}}^{\mathtt{only}}$ with the target HF signals. Results in the five rows are obtained for values of the input parameters taken from the following vectors: $Q = \{2.57, 2.61, 4.56, 4.71, 4.90\}$ kPa; $f = \{10.32, 35.10, 46.93, 26.23, 53.26\}$ Hz; $x_\Omega = \{1.70, 3.18, 3.85, 3.85, 3.85\}$ m; $y_\Omega = \{0.15, 0.15, 0.76, 1.28, 1.93\}$ m.

$\nu = 0.2$, $\rho = 2500$ kg/m^3. The superstructure consists of a single track with sleepers spaced 0.65 m apart, resting on a ballast layer 0.6 m deep, 4.3 m wide and featuring a density $\rho_B = 1800$ kg/m^3. The geometrical and mechanical modeling data have been adapted from former research activities on the relevant soil-structure interaction, see [72, 73].

The bridge is subjected to the transit of trains of the type Gröna Tåget, at a speed $v \in [160, 215]$ km/h. Only trains composed of two wagons are considered, thus characterized by 8 axles, each one carrying a mass $\phi \in [16, 22]$ ton. The corresponding load model is described in [56], and consists of 25 equivalent distributed forces transmitted by the sleepers to the deck through the ballast layer with a slope $4:1$, according to Eurocode 1 [74].

Figure 9: Hörnefors railway bridge.

The monitoring system features $N_u = 10$ sensors and is deployed as depicted in Fig. 10, to provide LF and HF structural displacement recordings $\mathbf{U}^{\mathtt{LF}}(\mathbf{x}^{\mathtt{LF}})$ and $\mathbf{U}^{\mathtt{HF}}(\mathbf{x}^{\mathtt{HF}})$, for a time interval $(0, T = 1.5 \text{ s})$ with an acquisition frequency of $\mathscr{f} = 400$ Hz.

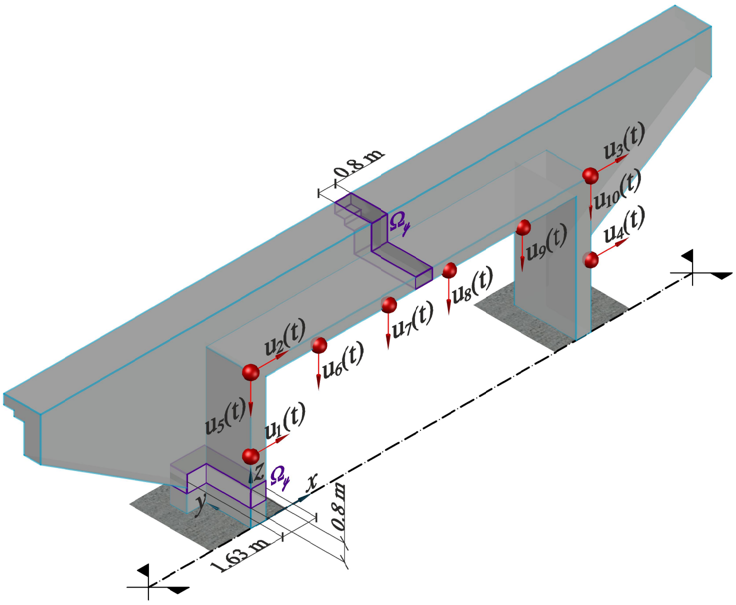


Figure 10: Railway bridge - physics-based digital twin, perspective cross-section with details of synthetic recordings related to displacements $u_1(t), \dots, u_{10}(t)$, and damaged region Ω_y.

The HF numerical model is obtained with a finite element mesh featuring a reduced element size of 0.15 m for the deck, to ensure a smooth transmission of the moving load, and of 0.80 m elsewhere, thus resulting in $\mathscr{M} = 17,292$ dofs. The effect of the ballast layer on the dynamic response of the bridge is modeled by increasing the density of the deck and of the edge beams. The embankments are modeled through distributed springs over the surfaces facing the ground, that is through a Robin mixed boundary condition (with elastic coefficient $k_{\text{Robin}} = 10^8$ N/m^3). The Rayleigh's damping matrix accounts for a 5% damping ratio on the first four structural modes. The presence of damage in the structure is modeled by means of a localized stiffness reduction that can take place anywhere over the two lateral frame walls and the deck, within subdomains $\Omega_{\boldsymbol{y}}$ featuring a different layout in the two cases (see Fig. 10). In this case, the stiffness reduction can occur with varying amplitude according to $\delta \in [20\%, 50\%]$, yet assumed fixed while a train travels across the bridge, and its position is parametrized through $\boldsymbol{y} = (x_\Omega, z_\Omega)$, with either x_Ω or z_Ω varying in the ranges $[-0.115, 16.515]$ m and $[0.4, 6.25]$ m, respectively. To summarize, the vector of HF input parameters is $\mathbf{x}^{\mathtt{HF}} = \{\upsilon, \phi, x_\Omega, z_\Omega, \delta\}^\top$.

The LF numerical model is instead obtained from a snapshot matrix $\mathbf{S}$ collecting $\mathscr{S} = 120,200$ snapshots, obtained through 200 evaluations of a LF-FOM not accounting for the presence of damage and damping, for different values of LF input parameters $\mathbf{x}^{\mathtt{LF}} = \{\upsilon, \phi\}^\top$. By setting the error tolerance to $\epsilon = 10^{-3}$, $\mathscr{M}_{\mathtt{LF}} = 312$ POD-modes are to be considered in place of the original $\mathscr{M} = 17,292$ dofs. The computing time required to assemble each LF instance decreases from 87.00 s to 11.80 s, entailing a speed-up of about 7.40 times.

The dataset $\mathbf{D}_{\mathtt{LF}}$ is built with $I_{\mathtt{LF}} = 5000$ LF data instances, generated with the LF-ROM. In order to set $\mathcal{NN}_{\mathtt{LF}}$, the projection matrix $\mathbf{Y}$ is obtained from $L_{\mathtt{LF}} = 68$ POD-basis functions, selected by performing an SVD on the vectorized LF dataset $\mathbf{D}^{\mathtt{vec}}_{\mathtt{LF}}$ and by choosing $\epsilon = 10^{-3}$ as error tolerance. On the other hand, only $I_{\mathtt{HF}} = 500$ HF data instances are computed with the HF numerical model to assemble $\mathbf{D}_{\mathtt{HF}}$. $\mathcal{NN}_{\mathtt{LF}}$ and $\mathcal{NN}_{\mathtt{HF}}$ are trained on these data for a maximum of 10,000 and 2000 allowed epochs, respectively. We remark that the DNNs are identical to those considered in Sec. 3.2, except for the size of the input/output layers, which is problem-specific.

Some results obtained with $\mathcal{NN}_{\mathtt{MF}}$ are reported in Fig. 11, for sensors $u_2(t)$, $u_4(t)$, $u_5(t)$, $u_7(t)$ and $u_{10}(t)$. Each column of the figure provides the ground truth HF displacement time series together with the LF and HF approximations, under different operational and damage conditions. From a qualitative point of view, $\mathcal{NN}_{\mathtt{MF}}$ perfectly handles the involved nonlinear correlations between LF and HF data. In particular, it should be highlighted the remarkable approximation capability of $\mathcal{NN}_{\mathtt{MF}}$ even in the presence of strongly discontinuous trajectories, as those associated to the $u_5(t)$ and $u_{10}(t)$ time histories. The same results are also reported in Fig. 12, as obtained with $\mathcal{NN}^{\mathtt{only}}_{\mathtt{HF}}$. According to what reported for the previous case study, a severe performance degradation is observed when $\mathcal{NN}_{\mathtt{LF}}$ is removed; indeed, the approximated time histories are highly unreliable for most of the time. Even if in this case the considered time histories do not show a significant high frequency content, the approximations are only piecewise accurate with sharp variations simultaneously affecting all channels. Besides the limited amount of HF data instances exploited to train $\mathcal{NN}^{\mathtt{only}}_{\mathtt{HF}}$, this behavior seems to be triggered by the occurrence of sharp discontinuities in the $u_5(t)$ and $u_{10}(t)$ time histories. The performance indicators associated to $\mathcal{NN}_{\mathtt{HF}}$, $\mathcal{NN}^{\mathtt{only}}_{\mathtt{HF}}$ and $\mathcal{NN}_{\mathtt{LF}}$ are reported in Tab. 5, as obtained for the same sample channels by considering LF and HF testing sets made of $I^{\mathtt{T}}_{\mathtt{LF}} = I^{\mathtt{T}}_{\mathtt{HF}} = 40$ data instances. The obtained results are again in agreement to what observed in the previous case study, with strong drops in the attained values of $\mathtt{PCC}_{\mathtt{avg}}$ and $\mathtt{PCC}_{\mathtt{min}}$, and an increase of almost two orders of magnitude in the $\mathtt{MAE}$ values when $\mathcal{NN}_{\mathtt{LF}}$ is removed. Moreover, by adopting the MF setting, the obtained values of $\mathtt{PCC}_{\mathtt{avg}}$ and $\mathtt{PCC}_{\mathtt{min}}$ are both extremely close to one, indicating that the nonlinear correlations between LF and HF data are perfectly assimilated by $\mathcal{NN}_{\mathtt{MF}}$.

Regarding the computational times required to approximate the $I^{\mathtt{T}}_{\mathtt{HF}} = 40$ HF data

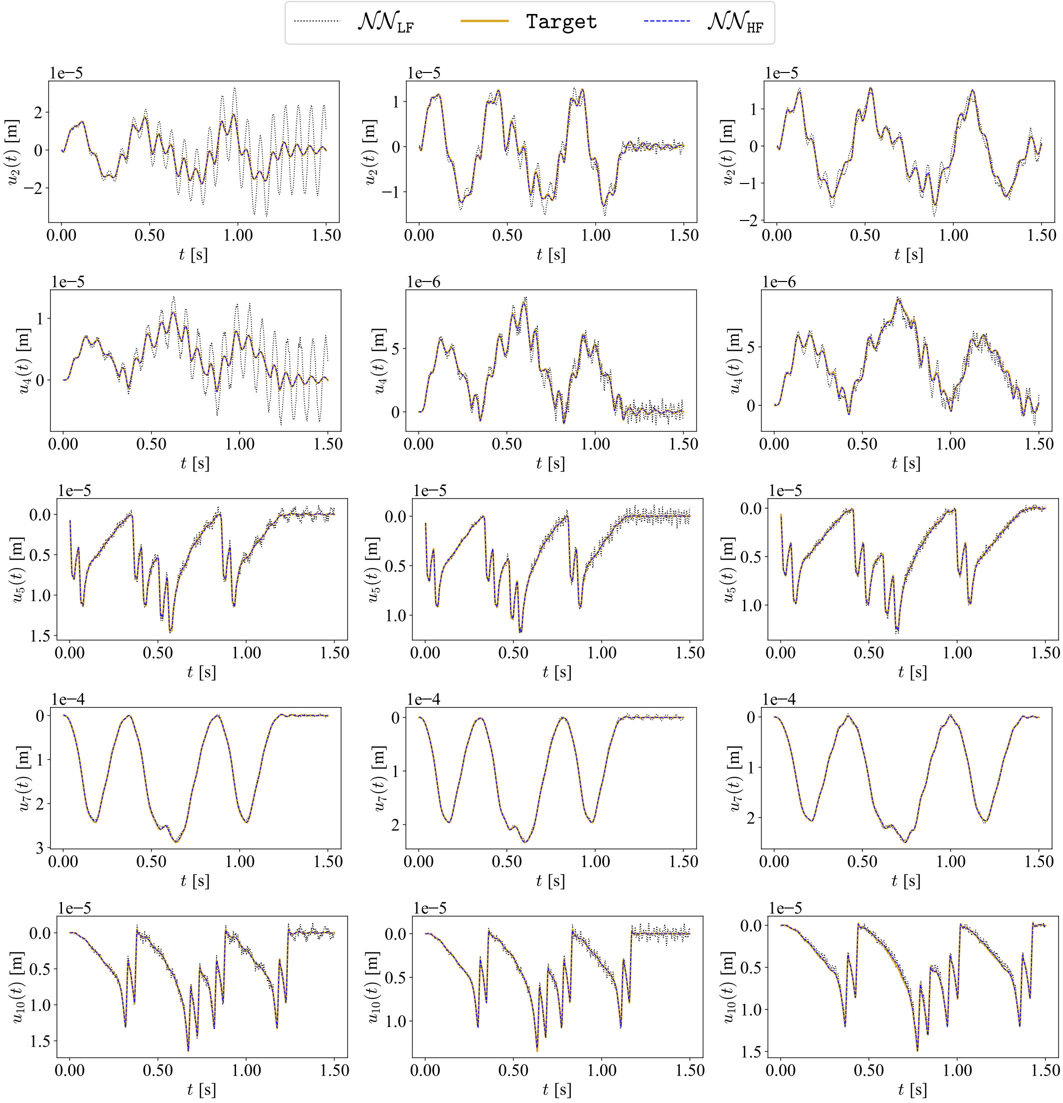


Figure 11: Railway bridge - exemplary comparisons of $u_2(t)$, $u_4(t)$, $u_5(t)$, $u_7(t)$ and $u_{10}(t)$ displacement time histories provided by $\mathcal{NN}_{\texttt{LF}}$ and $\mathcal{NN}_{\texttt{HF}}$ with the target HF signals. Results in the three columns are obtained for values of the input parameters taken from the following vectors: $\upsilon = \{192.1, 203.48, 166.59\}$ km/h; $\phi = \{20.66, 16.85, 17.95\}$ ton; $x_\Omega = \{5.97, 4.95, 16.515\}$ m; $z_\Omega = \{6.25, 6.25.2.46\}$ m; $\delta = \{0.23, 0.42, 0.30\}$%.

Table 5: Railway bridge - $\texttt{PCC}_{\texttt{avg}}$, $\texttt{PCC}_{\texttt{min}}$ and $\texttt{MAE}$ performance indicators, respectively measuring the average and minimum Pearson correlation coefficients, and the mean absolute error between predicted and ground truth HF and LF testing signals, for the $u_4(t)$, $u_5(t)$ and $u_7(t)$ monitored dofs.

Surrogate model	$\texttt{PCC}_{\texttt{avg}}(u_4(t), u_5(t), u_7(t))$	$\texttt{PCC}_{\texttt{min}}(u_4(t), u_5(t), u_7(t))$	$\texttt{MAE}(u_4(t), u_5(t), u_7(t))$
$\mathcal{NN}_{\texttt{MF}}$	$\{0.999, 0.999, 0.999\}$	$\{0.999, 0.996, 0.999\}$	$\{0.553, 0.559, 7.434\} \cdot 10^{-7}$
$\mathcal{NN}_{\texttt{HF}}^{\texttt{only}}$	$\{0.483, 0.544, 0.641\}$	$\{0.163, 0.497, 0.526\}$	$\{0.173, 0.448, 4.374\} \cdot 10^{-5}$
$\mathcal{NN}_{\texttt{LF}}$ (wrt LF)	$\{0.999, 0.999, 0.999\}$	$\{0.999, 0.999, 0.999\}$	$\{0.400, 0.446, 3.224\} \cdot 10^{-7}$

instances, the FEM solver takes 3480 s, while $\mathcal{NN}_{\texttt{MF}}$ only 0.125 s, entailing a speed-up of about 27, 840 times with respect to the HF numerical model.

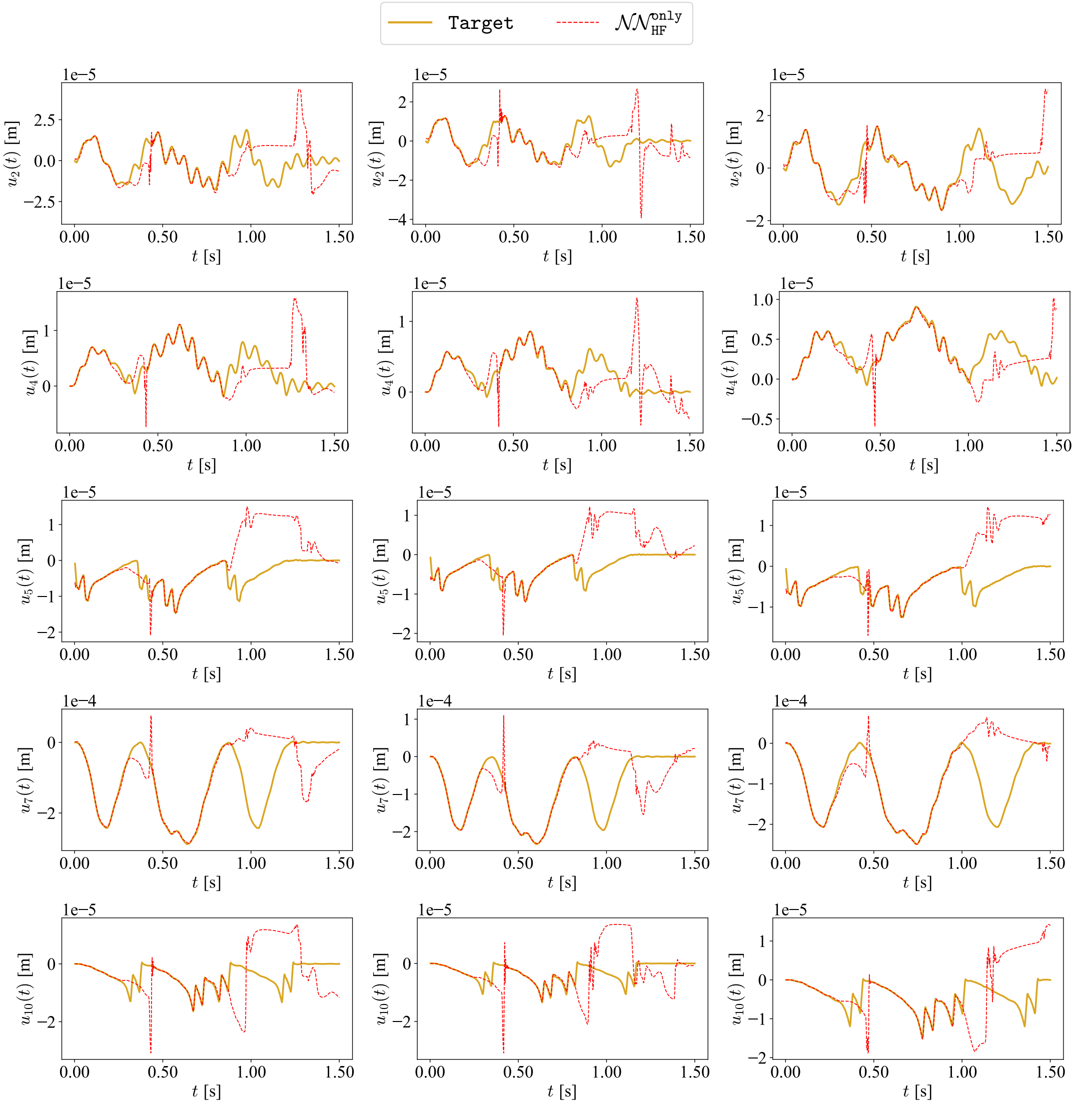


Figure 12: Railway bridge - exemplary comparisons of $u_2(t)$, $u_4(t)$, $u_5(t)$, $u_7(t)$ and $u_{10}(t)$ displacement time histories provided by $\mathcal{NN}_{\mathtt{HF}}^{\mathtt{only}}$ with the target HF signals. Results in the three columns are obtained for values of the input parameters taken from the following vectors: $v = \{192.1, 203.48, 166.59\}$ km/h; $\phi = \{20.66, 16.85, 17.95\}$ ton; $x_\Omega = \{5.97, 4.95, 16.515\}$ m; $z_\Omega = \{6.25, 6.25.2.46\}$ m; $\delta = \{0.23, 0.42, 0.30\}\%$.

Final comments can be added in relation to the robustness of the proposed approach to variations in the training datasets or to different damage scenarios. The learning stage is obviously case-specific but, as discussed here above, the proposed MF-DNN framework can effectively improve the computational performance of a SHM procedure. By increasing the structural/geometrical complexity of the monitored system, it is supposed that more (localized) damage patterns can show up. To catch them all and also track their evolution, not only in amplitude but also in shape and location, more detailed HF models are required; this would lead to more time-demanding computational models and, accordingly, to longer training stages. It should also be remarked that damage needs to be sensed via the handled recordings, independently of the physical model accuracy; hence, more and more an optimal deployment of the sensor network becomes a critical step in the design of the entire SHM strategy. However, the proposed MF setup is expected to be capable of handling even more involved parametrizations of the structural response, thus allowing for a suitable surrogate modeling of real-world applications.

4. Application to system identification

We now present a pilot application of the proposed surrogate modeling strategy within an MCMC-based model updating framework, useful to validate its approximation capabilities in a potential SHM scenario involving the L-shaped cantilever beam presented in Sec. 3.2. The vector of sought parameters is thus $\boldsymbol{\theta} = \{\theta_f, \theta_Q, \theta_\Omega\}^\top$, where the target position of Ω_y given by the coordinates of its center of mass is encoded as an abscissa $\theta_\Omega \in [0.15, 7.55]$ m running along the axis of the structure.

The posterior pdf of $\boldsymbol{\theta}$, conditioned on a batch of gathered sensor recordings, is approximated by iteratively generating a chain of samples from a proposal distribution and taking decision as to whether accept or reject each sample, on the basis of the likelihood of the current sample to represent the sensor observations by means of $\mathcal{NN}_{\texttt{MF}}$.

In addition to the damage parameter θ_Ω, the parameters θ_f and θ_Q ruling the acting load are also identified. To this aim, an informative prior knowledge on θ_f and θ_Q is preliminarily obtained by exploiting a DNN built upon a Siamese architecture [75], following a strategy similar to that proposed in [19, 76]. Besides keeping the Bayesian inference of $\boldsymbol{\theta}$ within a plausible range, this informative pdf serves also to initialize the Markov chains in high probability regions. Fig. 13 reports a high level flowchart of the online phase of the adopted SHM strategy, wherein such a preliminary identification is carried out in the lower-left node, as clarified in Sec. 4.1; the setup of the MCMC analysis is explained in Sec. 4.2 and the obtained results are finally reported and discussed in Sec. 4.3.

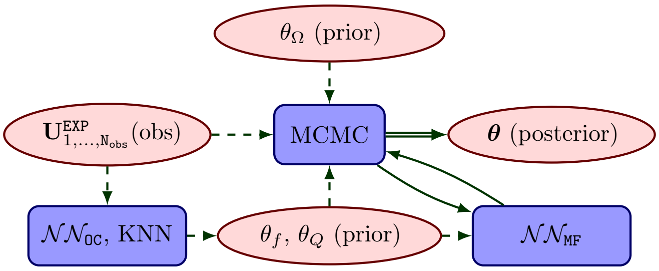

θ_Ω (prior)
$\mathbf{U}^{\texttt{EXP}}_{1,\dots,\texttt{N}_{\texttt{obs}}}$(obs)
MCMC
$\boldsymbol{\theta}$ (posterior)
$\mathcal{NN}_{\texttt{OC}}$, KNN
θ_f, θ_Q (prior)
$\mathcal{NN}_{\texttt{MF}}$

Figure 13: Flowchart of the online phase of the adopted SHM strategy. Red nodes refer to the input/output quantities, while blue nodes denote the relevant computational blocks: $\mathcal{NN}_{\texttt{OC}}$ and KNN refer to the preliminary (deterministic) identification of the parameters ruling the operational conditions; MCMC refers to the Bayesian inference procedure; $\mathcal{NN}_{\texttt{MF}}$ is the multi-fidelity surrogate model, exploited to systematically speed up the evaluation of the likelihood function while sampling the posterior pdf.

4.1. Preliminary identification of the operational conditions

In this section we detail the preliminary identification of θ_f and θ_Q, which serves to provide them with an informative prior and to initialize the Markov chains in high probability regions during the subsequent MCMC analysis. Such a preliminary assessment is based on the idea of tracing the load identification problem back to the construction (learning) of an ordered and smooth mapping of the vibration recordings onto a low-dimensional feature space. Then, for new sensor recordings, the parameters collected in $\mathbf{x}^{\mathtt{LF}}$ can be easily predicted in the low-dimensional space, by means of simple regression algorithms.

The low-dimensional mapping is driven by a fully connected neural network $\mathcal{NN}_{\mathtt{OC}}$, with the subscript $\mathtt{OC}$ standing for "operational conditions", designed to map the HF vibration recordings $\mathbf{U}^{\mathtt{HF}}(\mathbf{x}^{\mathtt{HF}})$ onto its feature representation $\boldsymbol{\hslash}(\mathbf{x}^{\mathtt{LF}}) \in \mathbb{R}^{D_{\hslash}}$ in a low-dimensional space of size $D_{\hslash}$, according to

$$\boldsymbol{\hslash}(\mathbf{x}^{\mathtt{LF}}) = \mathcal{NN}_{\mathtt{OC}}(\mathbf{Y}_{\mathtt{OC}}^{\top}\mathtt{vec}[\mathbf{U}^{\mathtt{HF}}(\mathbf{x}^{\mathtt{HF}})]) \ . \tag{13}$$

Here, $\mathbf{Y}_{\mathtt{OC}} \in \mathbb{R}^{L_{\mathtt{vec}} \times L_{\mathtt{OC}}}$ is a matrix gathering the first $L_{\mathtt{OC}} \ll L_{\mathtt{vec}}$ POD-basis functions of the vectorized HF data instances in $\mathbf{D}_{\mathtt{HF}}$, i.e. $\mathbf{D}_{\mathtt{HF}}^{\mathtt{vec}} = \{\mathtt{vec}[\mathbf{U}_j^{\mathtt{HF}}(\mathbf{x}_j^{\mathtt{HF}})]\}_{j=1}^{I_{\mathtt{HF}}}$, which is useful to perform a prior dimensionality reduction of the HF recordings; note that this is a different projection matrix from that one exploited to train $\mathcal{NN}_{\mathtt{LF}}$. In the present case, a projection matrix $\mathbf{Y}_{\mathtt{OC}}$ comprising $L_{\mathtt{OC}} = 106$ POD-basis functions is obtained in place of the original $L_{\mathtt{vec}} = 1608$ data points, by performing an SVD on the $\mathbf{D}_{\mathtt{HF}}^{\mathtt{vec}}$ and by choosing $\epsilon = 10^{-2}$ as the tolerance in Eq. (6).

In order to code the operational conditions $\mathbf{x}^{\mathtt{LF}}$ through the low-dimensional representation $\boldsymbol{\hslash}(\mathbf{x}^{\mathtt{LF}})$, we require that a suitable distance function $\mathcal{E}_{\hslash} = \mathcal{E}_{\hslash}(\boldsymbol{\hslash}_1(\mathbf{x}_1^{\mathtt{LF}}), \boldsymbol{\hslash}_2(\mathbf{x}_2^{\mathtt{LF}}))$, of any pair of mappings $\boldsymbol{\hslash}_1(\mathbf{x}_1^{\mathtt{LF}})$ and $\boldsymbol{\hslash}_2(\mathbf{x}_2^{\mathtt{LF}})$, semantically approximates the Euclidean distance $\mathcal{E}_{\mathtt{LF}}(\mathbf{x}_1^{\mathtt{LF}}, \mathbf{x}_2^{\mathtt{LF}}) = \|\mathbf{x}_1^{\mathtt{LF}} - \mathbf{x}_2^{\mathtt{LF}}\|_2$ between the associated operational conditions $\mathbf{x}_1^{\mathtt{LF}}$ and $\mathbf{x}_2^{\mathtt{LF}}$. This is achieved through the Siamese architecture [75] sketched in Fig. 14, which is made of two twins $\mathcal{NN}_{\mathtt{OC}}$ and is trained with pairwise contrastive learning [77]. The two networks are linked by the loss function $\mathcal{L}_{\mathtt{OC}}$, which means that data points are processed in pairs, yielding two outputs $\boldsymbol{\hslash}_1(\mathbf{x}_1^{\mathtt{LF}})$ and $\boldsymbol{\hslash}_2(\mathbf{x}_2^{\mathtt{LF}})$. To this aim, the dataset $\mathbf{D}_{\mathtt{HF}}$ is augmented to $\mathbf{D}_{\mathtt{HF}}^{\mathtt{P}}$ by assembling ζ_+ positive pairs, characterized by similar operational conditions, and ζ_- negative pairs, characterized by dissimilar operational conditions for each instance, according to $\mathbf{D}_{\mathtt{HF}}^{\mathtt{P}} = \{(\mathbf{U}_1^{\mathtt{HF}}(\mathbf{x}_1^{\mathtt{HF}}), \mathbf{x}_1^{\mathtt{LF}}, \mathbf{U}_2^{\mathtt{HF}}(\mathbf{x}_2^{\mathtt{HF}}), \mathbf{x}_2^{\mathtt{LF}})_j\}_{j=1}^{I_{\mathtt{HF}}^{\mathtt{P}}}$, $I_{\mathtt{HF}}^{\mathtt{P}} = I_{\mathtt{HF}}(\zeta_+ + \zeta_-)$ being the total number of pairs. Moreover, to allow for sensor noise, the training data are corrupted by adding an independent, identically distributed Gaussian noise, yielding a signal-to-noise ratio of 80.

$\mathcal{NN}_{\mathtt{OC}}$ is a 2-layers fully-connected neural network featuring only 3 neurons per layer. No activation is applied to the output layer, while the hidden layer features a softsign activation function; further insights on how to choose the size $D_{\hslash}$ of the low-dimensional space can be found, e.g., in [19].

During training, the weights $\boldsymbol{\Omega}_{\mathtt{OC}}$ parametrizing $\mathcal{NN}_{\mathtt{OC}}$ are optimized by minimizing the following square-square loss function [78]:

$$\mathcal{L}_{\mathtt{OC}}(\boldsymbol{\Omega}_{\mathtt{OC}}, \mathbf{D}_{\mathtt{HF}}^{\mathtt{P}}) = \frac{1}{I_{\mathtt{HF}}^{\mathtt{P}}} \sum_{j=1}^{I_{\mathtt{HF}}^{\mathtt{P}}} \left\{ (1-\gamma)\frac{1}{2}(\mathcal{E}_{\hslash})^2 + \gamma\frac{1}{2}\left[\max\left(0, \psi - \mathcal{E}_{\hslash}\right)\right]^2 \right\}_j + \lambda_{\mathtt{OC}}\|\boldsymbol{\Omega}_{\mathtt{OC}}\|_2^2 \ . \tag{14}$$

Herein: $\gamma = \{0, 1\}$, respectively if $\mathbf{x}_1^{\mathtt{LF}}$ and $\mathbf{x}_2^{\mathtt{LF}}$ identify a positive or a negative pair; $\psi > 0$ is a margin beyond which negative pairs do not contribute to $\mathcal{L}_{\mathtt{OC}}$. The minimization of $\mathcal{L}_{\mathtt{OC}}$ yields a distance function $\mathcal{E}_{\hslash}$, such that dissimilar pairs are kept away by at least the margin ψ, while similar pairs are pushed to be as close as possible. The metric adopted for the

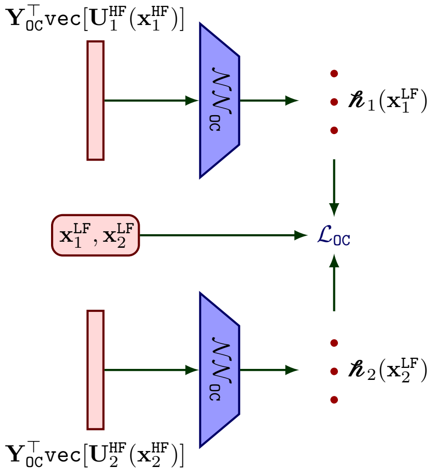

$\mathbf{Y}_{\mathtt{OC}}^{\top}\mathtt{vec}[\mathbf{U}_1^{\mathtt{HF}}(\mathbf{x}_1^{\mathtt{HF}})]$
$\mathcal{NN}_{\mathtt{OC}}$
$\boldsymbol{\hbar}_1(\mathbf{x}_1^{\mathtt{LF}})$
$\mathbf{x}_1^{\mathtt{LF}}, \mathbf{x}_2^{\mathtt{LF}}$
$\mathcal{L}_{\mathtt{OC}}$
$\mathcal{NN}_{\mathtt{OC}}$
$\boldsymbol{\hbar}_2(\mathbf{x}_2^{\mathtt{LF}})$
$\mathbf{Y}_{\mathtt{OC}}^{\top}\mathtt{vec}[\mathbf{U}_2^{\mathtt{HF}}(\mathbf{x}_2^{\mathtt{HF}})]$

Figure 14: Scheme of the Siamese architecture: red nodes denote the input/output quantities, while blue nodes refer to the tunable parameters.

low-dimensional space is the cosine similarity $S_c = \frac{\boldsymbol{\hbar}_1 \cdot \boldsymbol{\hbar}_2}{\|\boldsymbol{\hbar}_1\|_2 \|\boldsymbol{\hbar}_2\|_2}$, which measures the cosine of the angle between $\boldsymbol{\hbar}_1$ and $\boldsymbol{\hbar}_2$; the corresponding distance function exploited in $\mathcal{L}_{\mathtt{OC}}$ is given by $\mathcal{E}_{\hbar} = 1 - S_c$. This specific metric induces an auto-regularization effect that stabilizes the training, and improves the representation capability of the network; for further details, see e.g. [19].

By adopting the Xavier's weight initialization [71], the loss function $\mathcal{L}_{\mathtt{OC}}$ is minimized using the Adam optimization algorithm [70] for a maximum of 400 allowed epochs. The learning rate $\eta_{\mathtt{OC}}$ is initially set to 0.0001, and decreased for 4/5 of the allowed training steps using a cosine decay schedule with weight decay equal to 0.01. An early-stopping strategy is also used to prevent overfitting. The architecture of $\mathcal{NN}_{\mathtt{OC}}$ and its relevant hyperparameters are summarized in Tab. 6a and in Tab. 6b, respectively.

Table 6: $\mathcal{NN}_{\mathtt{OC}}$ - (a) adopted architecture, and (b) selected hyperparameters and training options.

(a)

Layer	Type	Output shape	Activ.	Input layer
0	Input	$(B_{\mathtt{OC}}, L_{\mathtt{OC}} = 106)$	None	None
1	Dense	$(B_{\mathtt{OC}}, 3)$	Softsign	0
2	Dense	$(B_{\mathtt{OC}}, D_{\hbar} = 3)$	None	1

(b)

Weight initializer:	Xavier
Optimizer:	Adam
Batch size:	$B_{\mathtt{OC}} = 32$
Initial learning rate:	$\eta_{\mathtt{OC}} = 0.0001$
Allowed epochs:	400
Learning schedule:	$\frac{4}{5}$ cosine decay
Weight decay:	0.01
Early stop patience:	50 epochs
Positive pairings:	$\zeta_+ = 10$
Negative pairings:	$\zeta_- = 10$
Similarity margin:	$\psi = 0.4$
Train-val split:	80 : 20

The preliminary identification of θ_f and θ_Q is performed in the low-dimensional space provided by $\mathcal{NN}_{\mathtt{OC}}$ through a k-nearest neighbors (KNN) regressor, similarly to [19]. To this aim, the KNN algorithm is calibrated by exploiting the low-dimensional representations of the training data, collected as $\mathbf{D}_{\mathscr{h}} = \{(\boldsymbol{\mathscr{h}}_j, \mathbf{x}_j^{\mathtt{LF}})\}_{j=1}^{I_{\mathtt{HF}}}$. The number of neighbors accounted for in the regression and the associated weighting rule are not set a-priori, but their optimal value is determined through an N-fold cross-validation strategy [79]. For the L-shaped cantilever beam case study, the maximum folds-averaged regression score, as obtained from a 10-fold cross validation, is achieved with 7 neighbors with inverse distance weights.

The load identification capabilities of $\mathcal{NN}_{\mathtt{OC}}$ are assessed by considering a testing set made of $I_{\mathtt{HF}}^{\mathtt{T}} = 40$ noisy HF data instances. The obtained results are qualitatively reported in Fig. 15, showing on the left the embeddings of $\mathbf{D}_{\mathtt{HF}}$, as provided by $\mathcal{NN}_{\mathtt{OC}}$ for a three-dimensional metric space built upon the cosine similarity. Here, $\{(\mathscr{h}_1, \mathscr{h}_2, \mathscr{h}_3)_j\}_{j=1}^{I_{\mathtt{HF}}}$ are plotted on the unit hypersphere with a color channel referring to the target values of the load frequency and of the load amplitude. Note how embeddings associated with different operational conditions differ by featuring a relative rotation in the low-dimensional space, thus ensuring that $\mathcal{NN}_{\mathtt{OC}}$ correctly encodes the varying operational conditions. In the same figure, the plots on the right report the KNN regression results obtained on the testing set; these latter yield a coefficient of determination $R^2 = \{0.987, 0.981\}$ and a $\mathtt{MAE} = \{1.28 \text{ Hz}, 0.13 \text{ kPa}\}$, respectively for the load frequency and amplitude, which stands as a quite remarkable result in light of the limited amount of HF data instances exploited to train $\mathcal{NN}_{\mathtt{OC}}$.

4.2. Setup of the MCMC analysis

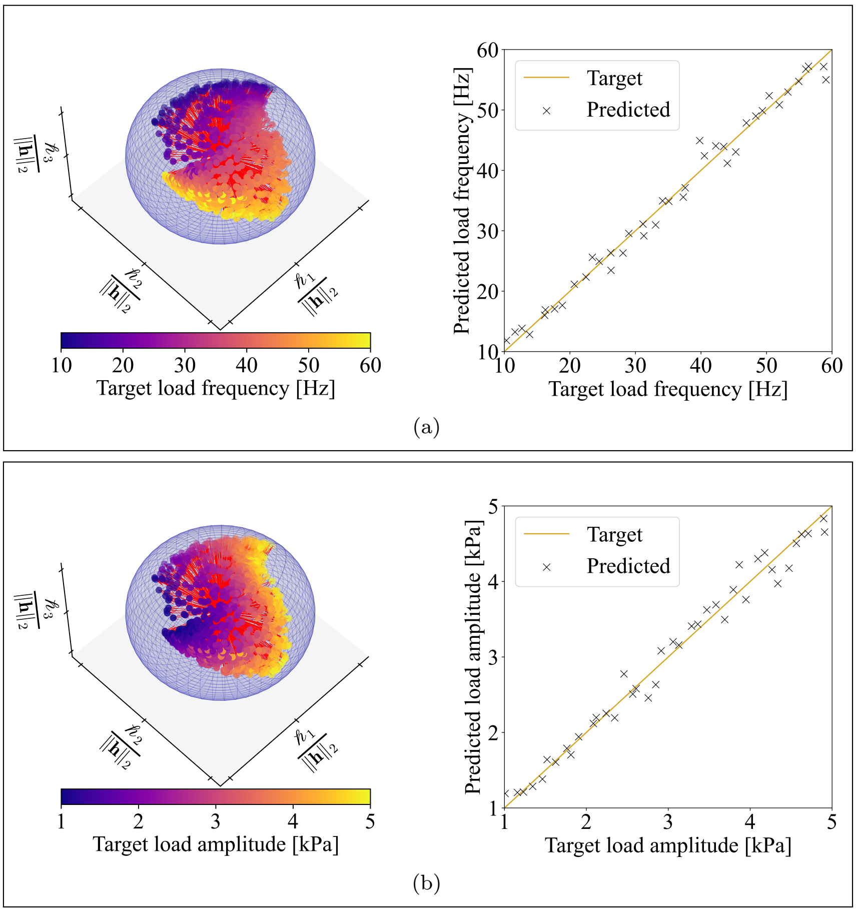
The proposed surrogate model $\mathcal{NN}_{\mathtt{MF}}$ and the feature extractor $\mathcal{NN}_{\mathtt{OC}}$ are synergically exploited within an MCMC procedure, to approximate the posterior pdf $p(\boldsymbol{\theta}|\mathbf{U}_{1,\ldots,\mathtt{N_{obs}}}^{\mathtt{EXP}}, \mathcal{NN}_{\mathtt{MF}})$ of the sought parameters $\boldsymbol{\theta} = \{\theta_f, \theta_Q, \theta_\Omega\}^\top$. The posterior is conditioned on the observed signals, or measurements $\mathbf{U}_{1,\ldots,\mathtt{N_{obs}}}^{\mathtt{EXP}}$, $\mathtt{N_{obs}}$ being the batch size of the processed observations, which is assumed small enough to assume steady operational, environmental, and damage conditions. Adopting, e.g., the Metropolis-Hastings algorithm, MCMC updates the prior pdf $p(\boldsymbol{\theta}, \mathcal{NN}_{\mathtt{MF}})$ to provide a posterior, by iteratively generating a chain of samples $\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{\mathtt{L_{chain}}}\}$ of length $\mathtt{L_{chain}}$. An acceptance rule is then used to take decision as to whether accept or reject each sample, on the basis of the likelihood of the current sample to represent $\mathbf{U}_{1,\ldots,\mathtt{N_{obs}}}^{\mathtt{EXP}}$ by means of $\mathcal{NN}_{\mathtt{MF}}$. By assuming an additive Gaussian noise uncorrelated in time to represent the internal deviation due to modeling inaccuracies and measurement noise, the likelihood function is assumed to be Gaussian too, and reads:

$$p(\mathbf{U}_{1,\ldots,\mathtt{N_{obs}}}^{\mathtt{EXP}}|\boldsymbol{\theta}, \mathcal{NN}_{\mathtt{MF}}) = \prod_{\mathscr{k}=1}^{\mathtt{N_{obs}}} c^{-1} \exp\left(- \frac{\sum_{l=1}^{L} (\mathbf{U}_{\mathscr{k}}^{\mathtt{EXP}} - \widehat{\mathbf{U}}^{\mathtt{HF}})_l^\top \boldsymbol{\Sigma}_c^{-1} (\mathbf{U}_{\mathscr{k}}^{\mathtt{EXP}} - \widehat{\mathbf{U}}^{\mathtt{HF}})_l}{2} \right). \quad (15)$$

In Eq. (15): the denominator $c = (\sqrt{2\pi})^{N_u} \sqrt{|\boldsymbol{\Sigma}_c|}$ is a normalization constant; $\boldsymbol{\Sigma}_c \in \mathbb{R}^{N_u \times N_u}$ is a covariance matrix, resulting from a prediction error model accounting for independent measurement errors and spatially correlated modeling inaccuracies, see e.g., [80]. For the $\mathscr{k}$-th observation, with $\mathscr{k} = 1, \ldots, \mathtt{N_{obs}}$, the $(\mathscr{i}, \mathscr{j})$-th element of $\boldsymbol{\Sigma}_c$ is computed as:

$$\boldsymbol{\Sigma}_c(\mathscr{i}, \mathscr{j}) = \rho_{\mathscr{i}\mathscr{j}} \left[\mathtt{RMS}(\mathbf{u}_{\mathscr{i},\mathscr{k}}^{\mathtt{EXP}} - \widehat{\mathbf{u}}_{\mathscr{i}}^{\mathtt{HF}}) - (1 - \delta_{\mathscr{i}\mathscr{j}}) \frac{\mathtt{RMS}(\mathbf{u}_{\mathscr{i},\mathscr{k}}^{\mathtt{EXP}})}{\sqrt{\mathtt{SNR}}} \right] \left[\mathtt{RMS}(\mathbf{u}_{\mathscr{j},\mathscr{k}}^{\mathtt{EXP}} - \widehat{\mathbf{u}}_{\mathscr{j}}^{\mathtt{HF}}) - (1 - \delta_{\mathscr{i}\mathscr{j}}) \frac{\mathtt{RMS}(\mathbf{u}_{\mathscr{j},\mathscr{k}}^{\mathtt{EXP}})}{\sqrt{\mathtt{SNR}}} \right]. \quad (16)$$

Herein: the root mean square $\mathtt{RMS}$ of the prediction error serves as its standard deviation, under the zero-mean assumption [29]; $\rho_{\mathscr{i}\mathscr{j}}$ is the correlation coefficient of the prediction errors at the $\mathscr{i}$-th and $\mathscr{j}$-th channels, as previously obtained by evaluating $\mathcal{NN}_{\mathtt{MF}}$ on the $I_{\mathtt{HF}}^{\mathtt{T}}$ testing



(a)

(b)

Figure 15: L-shaped cantilever beam - preliminary identification of the (a) load frequency and (b) load amplitude: (left) normalized embeddings of $\mathbf{D}_{\mathtt{HF}}$ on the unit hypersphere against the target values of the operational conditions; (right) parity plots, relevant to the outcome of the KNN regression over the $I_{\mathtt{HF}}^{\mathtt{T}} = 40$ noise-contaminated testing instances.

instances; the Kronecker delta $\delta_{i\jmath}$ serves to remove the effect of measurement noise, featuring a signal to noise ratio $\mathtt{SNR}$, from the off-diagonal elements. Due to their dependance on $\boldsymbol{\theta}$, both $\boldsymbol{\Sigma}_c$ and $\boldsymbol{\Sigma}_c^{-1}$ must be computed at each MCMC iteration. However, this does not affect the computational performance of the methodology due to the limited number of channels involved; if necessary, the computation of $\boldsymbol{\Sigma}_c^{-1}$ can be speeded up, e.g., by means of an SVD.

Even though an informative prior can provide useful insights about likely damage states, a uniform prior is chosen for the damage parameter θ_Ω to avoid possible biases in the results. $\mathcal{NN}_{\mathtt{OC}}$ is instead used to provide θ_f and θ_Q with an informative Gaussian prior, featuring a mean provided by the average KNN regression outcome on the $\mathtt{N}_{\mathtt{obs}}$ observations, and a covariance matrix previously obtained by evaluating the KNN regressor against the $I_{\mathtt{HF}}^{\mathtt{T}}$ testing instances. By adopting this prior to initialize the Markov chains, a shorter transient phase is obtained with fewer samples, along with a reduced risk of getting trapped into local maxima of $p(\mathbf{U}_{1,\dots,\mathtt{N}_{\mathtt{obs}}}^{\mathtt{EXP}}|\boldsymbol{\theta},\mathcal{NN}_{\mathtt{MF}})p(\boldsymbol{\theta},\mathcal{NN}_{\mathtt{MF}})$, when taking decisions about the acceptance/rejection of $\boldsymbol{\theta}$ candidate samples.

Fig. 16 shows a sketch of relevant MCMC procedure. Adopting the adaptive Metropolis algorithm [81], the proposal pdf is taken as Gaussian, centered at the previously accepted state and featuring a covariance matrix initialized as diagonal, with entries small enough so that the sampler gets moving, and then adapted as the sampling evolves; for the case at hand, such a covariance matrix is initialized by taking standard deviations for $\{\theta_f, \theta_Q, \theta_\Omega\}$ equal to 1% of the width of the ranges wherein they can take values.

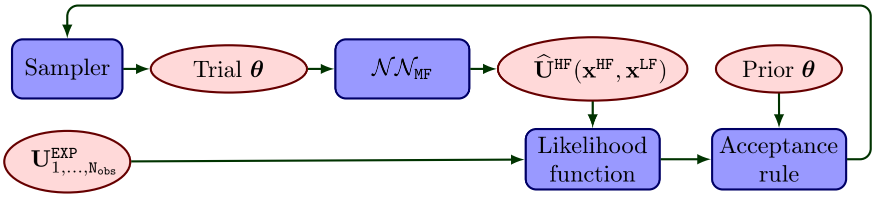

Sampler
Trial $\boldsymbol{\theta}$
$\mathcal{NN}_{\mathtt{MF}}$
$\widehat{\mathbf{U}}^{\mathtt{HF}}(\mathbf{x}^{\mathtt{HF}}, \mathbf{x}^{\mathtt{LF}})$
Prior $\boldsymbol{\theta}$
$\mathbf{U}^{\mathtt{EXP}}_{1,\ldots,\mathtt{N_{obs}}}$
Likelihood function
Acceptance rule

Figure 16: Scheme of the MCMC procedure to update the probability distribution of the structural state. Red nodes refer to the input/output quantities, while blue nodes denote the relevant computational blocks.

The estimated potential scale reduction (EPSR) metric [82], is exploited on-the-fly to monitor if a chain has converged to a steady distribution. By generating samples from multiple randomly initialized chains, the EPSR metric $\widehat{\mathscr{E}}$ tests the convergence of a multivariate chain by measuring the ratio between the estimate of the between-chain variance of samples and the average within-chain variance of samples. According to [82], convergence can deemed to be met when $\widehat{\mathscr{E}} < \mathtt{tol}$, with $\mathtt{tol} = 1.1$ being a safe tolerance value.

4.3. Parameter identification outcome

The MCMC algorithm is fed with batches of $\mathtt{N_{obs}} = 8$ noisy observations relative to the same θ_Ω, with θ_f and θ_Q featuring a stochastic scattering kept fixed within $(0, T = 1 \text{ s})$. Namely, for each data instance in the observations batch, each entry of $\mathbf{x}^{\mathtt{LF}}$ is sampled from a Gaussian pdf centered at the actual value of the parameter, and characterized by a standard deviation equal to 0.25% of the width of the range wherein it can take values.

Each MCMC simulation is carried out by generating five parallel Markov chains, that are randomly initialized from the prior and simultaneously evolved to meet the EPSR convergence criterium set to $\widehat{\mathscr{E}} \leq 1.01$. The first half of each chain is then removed to get rid of the initialization effect, and 3 out of 4 samples are discarded to reduce the within chain autocorrelation of samples.

Six MCMC analyses are carried out under different operational conditions while moving the damage position from the clamp to the free-end. Due to a high sensitivity of sensor recordings to the acting load, the parameters θ_f and θ_Q are always correctly identified with low uncertainty, i.e., the width of the corresponding 95% confidence intervals is only about 0.05 Hz and 0.15 kPa, respectively; therefore, only the results relevant to the identification of the damage position are reported and commented in the following. These latter are reported in Tab. 7 in terms of: target value; posterior mean; posterior mode; standard deviation; samples to converge. In the first three cases, which are characterized by a damage within the harm close to the clamped side, the damage position is correctly identified with a relatively low uncertainty. The quality of the estimates is highlighted by the low discrepancy between the target and the posterior mean values, which is only about 0.2 m, as well as by the relatively low values of attained standard deviation, which however increases as the damage position gets far from the clamped side. On the other hand, the last three cases featuring a damage position far from the clamped side, yield estimations still acceptable but, due to a smaller sensitivity of sensor recordings to damage, characterized by an increased discrepancy between the target and the posterior mean values, as well as a larger uncertainty. For instance, for case 5 the discrepancy between the target and the posterior mean values reaches 0.55 m, while the corresponding standard deviation, and therefore the credibility intervals, increases by 35% with respect to that of case 3. Moreover, it is to note that case 6 is characterized by a skewed distribution, testified by the increased discrepancy between the values of posterior mean and mode, yet without a larger standard deviation; and for this case, the posterior mode turns out to be a more appropriate estimate than the posterior mean. An exemplary MCMC outcome is reported in Fig. 17 for case 2, showing the generated Markov chain alongside the estimated posterior mean and credibility intervals for $\{\theta_f, \theta_Q, \theta_\Omega\}$. Here, the chains of θ_f and θ_Q are plotted over a relatively small range of values for the sake of visualization, and the posterior mean of θ_f shows a discrepancy from the target value of 0.02 Hz only, which is within the 95% confidence interval. The computing time required for the parameter estimation is about 130 s.

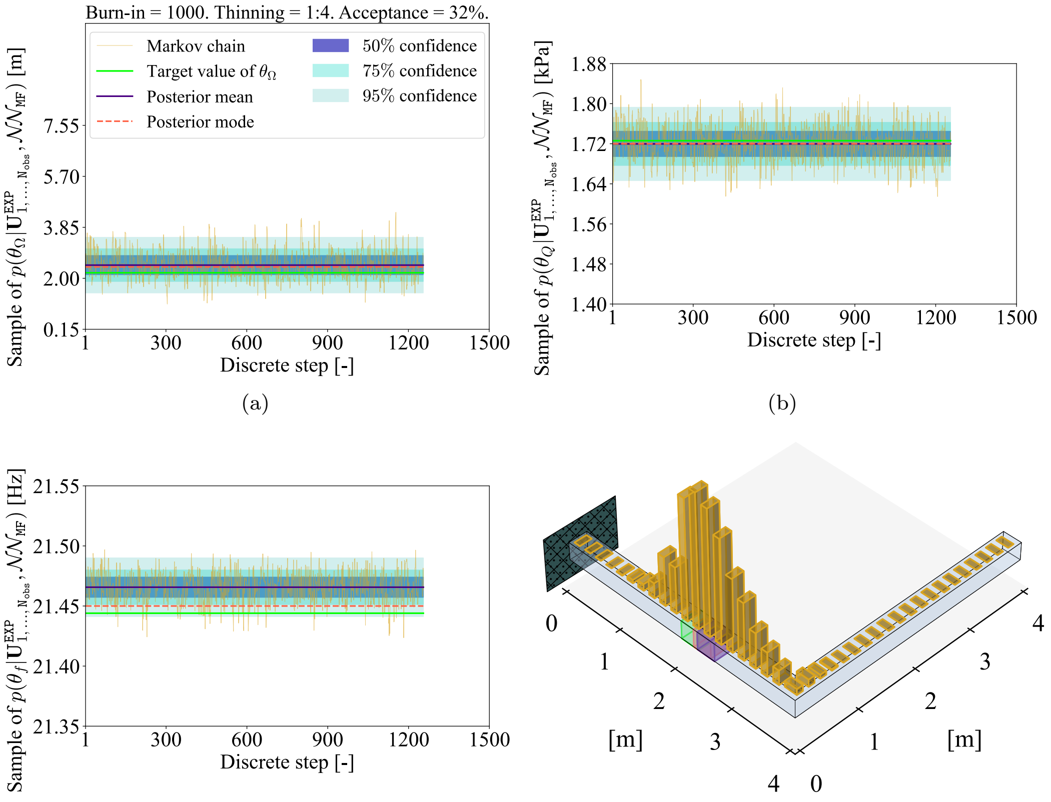
Overall, the obtained results confirm the capability of $\mathcal{NN}_{\texttt{MF}}$ to be effectively integrated within hybrid data/model-based SHM strategies. As long as the training dataset is sufficiently representative, there are no restrictions on applying the proposed surrogate modeling strategy also to large-scale structural systems. If needed, the damage identification performance can be improved by exploiting features to describe the structural response or by means of a more advanced sampler, such as the transitional MCMC or No-U-Turn algorithms [35, 36].

Table 7: Damage localization results for different operational and damage conditions, in terms of: target value; posterior mean; posterior mode; standard deviation; samples to converge.

Case	Target	$\texttt{mean}(\theta_\Omega)$	$\texttt{mode}(\theta_\Omega)$	$\texttt{stdv}(\theta_\Omega)$	$\mathrm{L}_{\texttt{chain}}$
1	0.564 m	0.631 m	0.587 m	0.170 m	2000
2	2.200 m	2.474 m	2.414 m	0.511 m	2000
3	2.888 m	3.088 m	2.844 m	0.710 m	3400
4	4.435 m	4.834 m	4.198 m	0.969 m	2000
5	5.204 m	5.759 m	5.397 m	0.962 m	3000
6	7.380 m	6.080 m	7.136 m	0.866 m	4000

5. Conclusions

This work has proposed a novel strategy to build surrogate models for structural health monitoring purposes in a non-intrusive way, exploiting model order reduction and artificial neural networks. The offered strategy relies upon a multi-fidelity setup that enables the correlation across structural models of different fidelities to be leveraged without any prior knowledge, ultimately alleviating the computational burden associated to the offline data generation phase, while ensuring the accuracy of the approximated quantities of interest. The resulting deep learning-based surrogate allows to map damage and operational parameters onto sensor recordings, as a function of time and parameters. The methodology can potentially be extended to the full-field approximation, and also adapted to frequency-domain or feature-based data.



(a)

(b)

(c)

(d)

Figure 17: Exemplary MCMC result (case 2): Markov chain, target value, posterior mean, posterior mode and credibility intervals relevant to (a) θ_Ω, (b) θ_Q and (c) θ_f; (d) histogram of the approximated, unnormalized posterior pdf $p(\theta_\Omega|\mathbf{U}^{\mathtt{EXP}}_{1,\dots,\mathtt{N_{obs}}},\mathcal{NN}_{\mathtt{MF}})$ over the admissible support, with details of the target, posterior mean and posterior mode values of θ_Ω, respectively reported as green, purple and red boxes.

The surrogate model that has been devised in the present work features a multi-level architecture characterized by two sequentially trained deep neural networks: a fully-connected model, that has been used to mimic sensor recordings in the reference state; and a long short-term memory model, which has been exploited to adaptively enrich the damage-free approximation with the effect of damage. The supervised learning stage has been enabled by an offline phase of data generation, involving numerical models of different fidelity, allowing to systematically reproduce the effect of damage on vibration response data. Moreover, the adopted framework enjoys additional features providing specific advantages for structural health monitoring purposes: the effect of damage is reproduced with a high-fidelity numerical model only, with no need of model order reduction techniques prone to lose damage-sensitivity; a large amount of low-fidelity data can be generated, once and for all, exploiting reduced-order modeling to reduce the associated computational burden; it is never necessary to update the low-fidelity component of the surrogate model. This last aspect is particularly favorable in view of a future development toward a digital twin framework, with the goal of devising a surrogate model capable of adapting to the evolving structural health state. Whenever a deterioration of the structural health is detected, the digital model can be updated by adjusting only its high-fidelity part, with few new data generated using an updated numerical model.

The proposed strategy has been assessed against an L-shaped cantilever beam and a portal frame railway bridge, and the obtained results have shown that the proposed technique provides remarkably accurate approximations of the structural response, even under the effect of varying operational and damage conditions. Moreover, it has been shown to outperform its single-fidelity counterpart by orders of magnitude. The computational time required to evaluate the proposed surrogate model has been reported to be orders of magnitude smaller than that required by the corresponding high-fidelity finite element solver. It has been remarked that, depending on the structural complexity and adopted modeling choices, the training time can be affected by the need of populating informative enough training datasets. This last aspect is not only related to the number of instances collected in the training datasets, at varying operational and (potentially) environmental conditions and allowed damage scenarios, but also to the number of sensor recordings to be processed; indeed, the observational data should ensure that the damage state can be observable, an issue strictly linked to the design of efficient sensors network deployments.

The capability of the proposed strategy to be effectively integrated within structural health monitoring strategies has been assessed by means of a stochastic approach to damage localization, under the effect of measurement noise and varying operational conditions.

The next studies will be devoted to the use of the proposed surrogate modeling strategy, in order to generate large labeled datasets. Such datasets will be eventually exploited to train deep learning-based feature extractors and feature-oriented surrogate models, to be synergically leveraged within a new Markov chain Monte Carlo algorithm. Preliminary results have shown that using *learnable* features in place of raw vibration recordings, enables to largely improve the parameter identification outcomes, also providing considerable gains in terms of efficiency due to the low-dimensionality of the involved features.

Acknowledgments: This work is supported in part by the interdisciplinary Ph.D. Grant "Physics-Informed Deep Learning for Structural Health Monitoring" at Politecnico di Milano.

The authors declare no conflict of interest.

References

[1] A. Marengo, A. Patton, M. Negri, U. Perego, and A. Reali, "A rigorous and efficient explicit algorithm for irreversibility enforcement in phase-field finite element modeling of brittle crack propagation," *Comput Methods Appl Mech Eng*, vol. 387, p. 114137, 12 2021.

[2] F. Migliavacca, L. Petrini, M. Colombo, F. Auricchio, and R. Pietrabissa, "Mechanical behavior of coronary stents investigated through the finite element method," *J Biomech*, vol. 35, no. 6, pp. 803–811, 2002.

[3] L. Kwasniewski, H. Li, J. Wekezer, and J. Malachowski, "Finite element analysis of vehicle–bridge interaction," *Finite Elem Anal Des*, vol. 42, no. 11, pp. 950–959, 2006.

[4] B. Peherstorfer, K. Willcox, and M. Gunzburger, "Survey of Multifidelity Methods in Uncertainty Propagation, Inference, and Optimization," *SIAM Review*, vol. 60, no. 3, pp. 550–591, 2018.

[5] A. C. Antoulas, *Approximation of large-scale dynamical systems.* SIAM, 2005.

[6] F. Chinesta, P. Ladevèze, and E. Cueto, “A Short Review in Model Order Reduction Based on Proper Generalized Decomposition,” *Arch Comput Methods Eng*, vol. 18, no. 4, pp. 395–404, 2011.

[7] G. Kerschen, J. C. Golinval, A. Vakakis, and L. A. Bergman, “The Method of Proper Orthogonal Decomposition for Dynamical Characterization and Order Reduction of Mechanical Systems: An Overview,” *Nonlinear Dyn*, vol. 41, no. 1, pp. 147–169, 2005.

[8] A. Quarteroni, A. Manzoni, and F. Negri, *Reduced basis methods for partial differential equations: an introduction.* Springer, 2015.

[9] A. Forrester, A. Sobester, and A. Keane, *Engineering Design Via Surrogate Modelling: A Practical Guide.* John Wiley & Sons, 2008.

[10] S. Koziel and L. Leifsson, *Surrogate-based modeling and optimization.* Springer, 2013.

[11] S. Sadeghi Eshkevari, M. Takáč, S. N. Pakzad, and M. Jahani, “DynNet: Physics-based neural architecture design for nonlinear structural response modeling and prediction,” *Eng Struct*, vol. 229, p. 111582, 2021.

[12] H. Peng, J. Yan, Y. Yu, and Y. Luo, “Structural Surrogate Model and Dynamic Response Prediction with Consideration of Temporal and Spatial Evolution: An Encoder–Decoder ConvLSTM Network,” *Int J Struct Stab Dyn*, vol. 21, no. 10, p. 2150140, 2021.

[13] R. Maulik, K. Fukami, N. Ramachandra, K. Fukagata, and K. Taira, “Probabilistic neural networks for fluid flow surrogate modeling and data recovery,” *Phys Rev Fluids*, vol. 5, p. 104401, 2020.

[14] C. Farrar and K. Worden, *Structural Health Monitoring A Machine Learning Perspective.* Wiley, 2013.

[15] D. Cristiani, C. Sbarufatti, F. Cadini, and M. Giglio, “Fatigue damage diagnosis and prognosis of an aeronautical structure based on surrogate modelling and particle filter,” *Struct Health Monit*, p. 1475921720971551, 2020.

[16] A. C. Neves, I. Gonzalez, and R. Karoumi, “Development and Validation of a Data-Based SHM Method for Railway Bridges,” in *Structural Health Monitoring Based on Data Science Techniques*, pp. 95–116, Springer International Publishing, 2022.

[17] L. Rosafalco, M. Torzoni, A. Manzoni, S. Mariani, and A. Corigliano, “A Self-adaptive Hybrid Model/data-Driven Approach to SHM Based on Model Order Reduction and Deep Learning,” in *Structural Health Monitoring Based on Data Science Techniques*, pp. 165–184, Springer International Publishing, 2022.

[18] E. García-Macías and F. Ubertini, “Integrated SHM Systems: Damage Detection Through Unsupervised Learning and Data Fusion,” in *Structural Health Monitoring Based on Data Science Techniques*, pp. 247–268, Springer International Publishing, 2022.

[19] M. Torzoni, A. Manzoni, and S. Mariani, “Structural health monitoring of civil structures: a diagnostic framework powered by deep metric learning,” *Comput Struct*, vol. 271, p. 106858, 2022.

[20] K. Worden, “Structural fault detection using a novelty measure,” *J Sound Vib*, vol. 201, no. 1, pp. 85–101, 1997.

[21] L. Rosafalco, A. Manzoni, S. Mariani, and A. Corigliano, "Fully convolutional networks for structural health monitoring through multivariate time series classification," *Adv Model and Simul in Eng Sci*, vol. 7, p. 38, 2020.

[22] A. Entezami, H. Sarmadi, B. Behkamal, and S. Mariani, "Big data analytics and structural health monitoring: a statistical pattern recognition-based approach," *Sensors*, vol. 20, no. 8, p. 2328, 2020.

[23] L. Ierimonti, N. Cavalagli, E. García-Macías, I. Venanzi, and F. Ubertini, "Bayesian-Based Damage Assessment of Historical Structures Using Vibration Monitoring Data," in *International Workshop on Civil Structural Health Monitoring*, pp. 415–429, Springer, 2021.

[24] D. Cristiani, C. Sbarufatti, and M. Giglio, "Damage diagnosis and prognosis in composite double cantilever beam coupons by particle filtering and surrogate modelling," *Struct Health Monit*, vol. 20, no. 3, pp. 1030–1050, 2021.

[25] A. Kamariotis, E. Chatzi, and D. Straub, "Value of information from vibration-based structural health monitoring extracted via Bayesian model updating," *Mech Syst Signal Process*, vol. 166, p. 108465, 2022.

[26] S. Eftekhar Azam and S. Mariani, "Online damage detection in structural systems via dynamic inverse analysis: A recursive Bayesian approach," *Eng Struct*, vol. 159, pp. 28–45, 2018.

[27] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, 2006.

[28] O. Avci, O. Abdeljaber, S. Kiranyaz, M. Hussein, M. Gabbouj, and D. 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*, vol. 147, p. 107077, 2021.

[29] J. L. Beck and L. S. Katafygiotis, "Updating models and their uncertainties. I: Bayesian statistical framework," *J Eng Mech*, vol. 124, no. 4, pp. 455–461, 1998.

[30] M. Muto and J. L. Beck, "Bayesian updating and model class selection for hysteretic structural models using stochastic simulation," *J Vib Control*, vol. 14, no. 1–2, pp. 7–34, 2008.

[31] R. Hou, X. Wang, and Y. Xia, "Vibration-Based Structural Damage Detection Using Sparse Bayesian Learning Techniques," in *Structural Health Monitoring Based on Data Science Techniques*, pp. 1–25, Springer International Publishing, 2022.

[32] P. L. Green and K. Worden, "Bayesian and Markov chain Monte Carlo methods for identifying nonlinear systems in the presence of uncertainty," *Phil Trans R Soc A*, vol. 373, 2015.

[33] H. F. Lam, J. H. Yang, and S. K. Au, "Markov chain Monte Carlo-based Bayesian method for structural model updating and damage detection," *Struct Contr Health Monit*, vol. 25, no. 4, pp. 1–22, 2018.

[34] C. Andrieu and J. Thoms, "A tutorial on adaptive MCMC," *Stat Comput*, vol. 18, no. 4, pp. 343–373, 2008.

[35] W. Betz, I. Papaioannou, and D. Straub, "Transitional Markov chain Monte Carlo: observations and improvements," *J Eng Mech*, vol. 142, no. 5, p. 04016016, 2016.

[36] M. D. Hoffman and A. Gelman, "The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo," *J Mach Learn Res*, vol. 15, no. 1, pp. 1593–1623, 2014.

[37] R. Mirzazadeh, S. Eftekhar Azam, and S. Mariani, "Mechanical characterization of polysilicon MEMS: A hybrid TMCMC/POD-kriging approach," *Sensors*, vol. 18, no. 4, pp. 1–19, 2018.

[38] R. Rocchetta, M. Broggi, Q. Huchet, and E. Patelli, "On-line Bayesian model updating for structural health monitoring," *Mech Syst Signal Process*, vol. 103, pp. 174–195, 2018.

[39] E. Meeds and M. Welling, "GPS-ABC: Gaussian process surrogate approximate Bayesian computation," 2014. `https://arxiv.org/pdf/1401.2838.pdf`.

[40] E. García-Macías, L. Ierimonti, I. Venanzi, and F. Ubertini, "An innovative methodology for online surrogate-based model updating of historic buildings using monitoring data," *Int J. Archit Herit*, vol. 15, no. 1, pp. 92–112, 2021.

[41] I. N. Giannakeas, Z. Sharif Khodaei, and M. Aliabadi, "Digital clone testing platform for the assessment of SHM systems under uncertainty," *Mech Syst Signal Process*, vol. 163, p. 108150, 2022.

[42] N. Geneva and N. Zabaras, "Multi-fidelity generative deep learning turbulent flows," *Found Data Sci*, vol. 2, no. 4, pp. 391–428., 2020.

[43] X. Zhang, F. Xie, T. Ji, Z. Zhu, and Y. Zheng, "Multi-fidelity deep neural network surrogate model for aerodynamic shape optimization," *Comput Methods Appl Mech Eng*, vol. 373, p. 113485, 2021.

[44] X. Meng and G. E. Karniadakis, "A composite neural network that learns from multi-fidelity data: Application to function approximation and inverse PDE problems," *J Comput Phys*, vol. 401, p. 109020, 2020.

[45] X. Meng, H. Babaee, and G. E. Karniadakis, "Multi-fidelity Bayesian Neural Networks: Algorithms and Applications," *J Comput Phys*, vol. 438, p. 110361, 2021.

[46] M. Guo, A. Manzoni, M. Amendt, P. Conti, and J. Hesthaven, "Multi-fidelity regression using artificial neural networks: efficient approximation of parameter-dependent output quantities," *Comput Methods Appl Mech Eng*, vol. 389, p. 114378, 2022.

[47] P. Conti, M. Guo, A. Manzoni, and J. Hesthaven, "Multi-fidelity surrogate modeling using long short-term memory networks," *Comput Methods Appl Mech Eng*, vol. 404, p. 115811, 2023.

[48] T. Taddei, J. D. Penn, M. Yano, and A. T. Patera, "Simulation-based classification; a model-order-reduction approach for structural health monitoring," *Arch Comput Methods Eng*, vol. 25, pp. 23–45, 2018.

[49] G. Kerschen and J. C. Golinval, "Physical interpretation of the proper orthogonal modes using the singular value decomposition," *J Sound Vib*, vol. 249, no. 5, pp. 849–865, 2002.

[50] M. C. Kennedy and A. O'Hagan, "Predicting the Output from a Complex Computer Code When Fast Approximations Are Available," *Biometrika*, vol. 87, no. 1, pp. 1–13, 2000.

[51] M. Kast, M. Guo, and J. Hesthaven, "A non-intrusive multifidelity method for the reduced order modeling of nonlinear problems," *Comput Methods Appl Mech Eng*, vol. 364, p. 112947, 2020.

[52] J. Goh, D. Bingham, J. P. Holloway, M. J. Grosskopf, C. C. Kuranz, and E. Rutter, "Prediction and Computer Model Calibration Using Outputs From Multifidelity Simulators," *Technometrics*, vol. 55, no. 4, pp. 501–512, 2013.

[53] M. G. Kapteyn, J. V. R. Pretorius, and K. E. Willcox, "A probabilistic graphical model foundation for enabling predictive digital twins at scale," *Nat Comput Sci*, vol. 1, no. 5, pp. 337–347, 2021.

[54] A. Teughels, J. Maeck, and G. De Roeck, "Damage assessment by FE model updating using damage functions," *Comput Struct*, vol. 80, no. 25, pp. 1869–1879, 2002.

[55] L. Rosafalco, A. Manzoni, S. Mariani, and A. Corigliano, "Combined Model Order Reduction Techniques and Artificial Neural Network for Data Assimilation and Damage Detection in Structures," in *Computational Sciences and Artificial Intelligence in Industry: New Digital Technologies for Solving Future Societal and Economical Challenges*, pp. 247–259, Springer International Publishing, 2022.

[56] L. Rosafalco, M. Torzoni, A. Manzoni, S. Mariani, and A. Corigliano, "Online structural health monitoring by model order reduction and deep learning algorithms," *Comput Struct*, vol. 255, p. 106604, 2021.

[57] M. Torzoni, L. Rosafalco, A. Manzoni, S. Mariani, and A. Corigliano, "SHM under varying environmental conditions: an approach based on model order reduction and deep learning," *Comput Struct*, vol. 266, p. 106790, 2022.

[58] G. Gobat, A. Opreni, S. Fresca, A. Manzoni, and A. Frangi, "Reduced order modeling of nonlinear microstructures through proper orthogonal decomposition," *Mech Syst Signal Process*, vol. 171, p. 108864, 2022.

[59] P. Battaglia, J. B. C. Hamrick, V. Bapst, A. Sanchez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, C. Gulcehre, F. Song, A. Ballard, J. Gilmer, G. E. Dahl, A. Vaswani, K. Allen, C. Nash, V. J. Langston, C. Dyer, N. Heess, D. Wierstra, P. Kohli, M. Botvinick, O. Vinyals, Y. Li, and R. Pascanu, "Relational inductive biases, deep learning, and graph networks," 2018. `https://arxiv.org/pdf/1806.01261`.

[60] R. Zhang, Z. Chen, S. Chen, J. Zheng, O. Büyüköztürk, and H. Sun, "Deep long short-term memory networks for nonlinear structural seismic response prediction," *Comput Struct*, vol. 220, pp. 55–68, 2019.

[61] T. Simpson, N. Dervilis, and E. Chatzi, "Machine learning approach to model order reduction of nonlinear systems via autoencoder and LSTM networks," *J Eng Mech*, vol. 147, no. 10, p. 04021061, 2021.

[62] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, 1997.

[63] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning.* MIT Press, 2016. `http://www.deeplearningbook.org`.

[64] F. Negri, "redbKIT, version 2.2," 2016. `http://redbkit.github.io/redbKIT`.

[65] F. Chollet *et al.*, "Keras," 2015. `https://keras.io`.

[66] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognitn*, pp. 770–778, 2016.

[67] I. Bello, W. Fedus, X. Du, E. D. Cubuk, A. Srinivas, T.-Y. Lin, J. Shlens, and B. Zoph, "Revisiting ResNets: Improved Training and Scaling Strategies," in *Adv Neural Inf Process Syst*, vol. 34, 2021. `https://openreview.net/forum?id=dsmxf7FKiaY`.

[68] S. Jastrzebski, D. Arpit, N. Ballas, V. Verma, T. Che, and Y. Bengio, "Residual Connections Encourage Iterative Inference," in *Int Conf Learn Represent*, vol. 6, 2018. `https://openreview.net/forum?id=SJa9iHgAZ`.

[69] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," in *Proc IEEE Int Conf Comput Vis*, pp. 1026–1034, 2015.

[70] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Int Conf Learn Represent*, vol. 3, pp. 1–13, 2015.

[71] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *J Mach Learn Res*, vol. 9, pp. 249–256, 2010.

[72] M. Ülker-Kaustell, *Some aspects of the dynamic soil-structure interaction of a portal frame railway bridge.* PhD thesis, KTH Royal Institute of Technology, 2009.

[73] T. Arvidsson and J. Li, *Dynamic analysis of a portal frame railway bridge using frequency dependent soil structure interaction.* Master thesis, KTH Royal Institute of Technology, 2011.

[74] European Committee for Standardization, "Part 2: Traffic loads on bridges," in *EN 1991-2 Eurocode 1: Actions on structures*, pp. 66–74, 2003.

[75] J. Bromley, I. Guyon, Y. Lecun, E. Säckinger, and R. Shah, "Signature Verification using a "Siamese" Time Delay Neural Network," *Int. J. Pattern Recognit.*, vol. 7, p. 25, 1993.

[76] M. Torzoni, A. Manzoni, and S. Mariani, "A Deep Neural Network, Multi-fidelity Surrogate Model Approach for Bayesian Model Updating in SHM," in *European Workshop on Structural Health Monitoring*, pp. 1076–1086, Springer International Publishing, 2023.

[77] R. Hadsell, S. Chopra, and Y. Lecun, "Dimensionality reduction by learning an invariant mapping," in *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit*, pp. 1735 – 1742, 2006.

[78] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang, "A tutorial on energy-based learning," in *Predicting structured data*, p. 10, MIT Press, 2006.

[79] J. A. Goulet, *Probabilistic Machine Learning for Civil Engineers.* MIT Press, 2020.

[80] C. Papadimitriou and G. Lombaert, "The effect of prediction error correlation on optimal sensor placement in structural dynamics," *Mech Syst Signal Process*, vol. 28, pp. 105–127, 2012.

[81] H. Haario, E. Saksman, and J. Tamminen, "An adaptive Metropolis algorithm," *Bernoulli*, vol. 7, no. 2, pp. 223–242, 2001.

[82] A. Gelman and D. B. Rubin, "Inference from Iterative Simulation Using Multiple Sequences," *Stat Sci*, vol. 7, no. 4, pp. 457–472, 1992. `http://www.jstor.org/stable/2246093`.

MOX Technical Reports, last issues

Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

18/2024 Torzoni, M.; Manzoni, A.; Mariani, S.
A multi-fidelity surrogate model for structural health monitoring exploiting model order reduction and artificial neural networks

17/2024 Fois, M.; de Falco, C.; Formaggia, L.
A semi-conservative depth-averaged Material Point Method for fast flow-like landslides and mudflows

16/2024 Domanin D. A.; Pegoraro M.; Trimarchi S.; Domanin M.; Secchi P.
Persistence diagrams for exploring the shape variability of abdominal aortic aneurysms

15/2024 Vaccaro, F.; Mauri, A.G.; Perotto, S.; Brivio, S.; Spiga, S.
Modeling and simulation of electrochemical and surface diffusion effects in filamentary cation-based resistive memory devices

14/2024 Zappon, E.; Salvador, M.; Piersanti, R.; Regazzoni, F.; Dede', L.; Quarteroni, A.
An integrated heart-torso electromechanical model for the simulation of electrophysiological outputs accounting for myocardial deformation

12/2024 Zingaro, A.; Ahmad, Z.; Kholmovski, E.; Sakata, K.; Dede', L.; Morris, A.K.; Quarteroni, A.; Trayanova, N.A.
A comprehensive stroke risk assessment by combining atrial computational fluid dynamics simulations and functional patient data

Antonietti, P.F.; Corti, M.
Numerical modelling of protein misfolding in neurodegenerative diseases a computational study

11/2024 Antonietti, P.F.; Corti, M.
Numerical modelling of protein misfolding in neurodegenerative diseases: a computational study

09/2024 Leimer Saglio, C. B.: Pagani, S.; Corti, M.; Antonietti, P. F.
A high-order discontinuous Galerkin method for the numerical modeling of epileptic seizures

10/2024 Capuano E.; Regazzoni F.; Maines M.; Fornara S.; Locatelli V.; Catanzariti D.; Stella S.; Nobile F.; Del Greco M.; Vergara C.
Personalized Computational Electro-mechanics Simulations to Optimize Cardiac Resynchronization Therapy