A hybrid Structural Health Monitoring approach based on reduced-order modelling and deep learning

Luca Rosafalco[1], Alberto Corigliano[1], Andrea Manzoni[2], Stefano Mariani[1]

[1]Dipartimento di Ingegneria Civile ed Ambientale
[2]MOX, Dipartimento di Matematica

August 1st 2007, Minneapolis (Minnesota, USA):
I-35W Mississippi bridge collapse killed 13 people

July 7th 2018, Torre Annunziata (Italy):
residential building collapse killed 8 people

March 15th 2018, Miami (Florida, USA):
pedestrian bridge collapse killed 6 people

Contents

Framework:

Structural Health Monitoring (SHM) aims to detect, localize and quantify damage continuously in time.

Damage measures the degradation of structural stiffness or load bearing capacity of a structural member. A general assumption consists in **treating the structure as linear**, that is in considering the damage as temporary frozen within a certain time window.

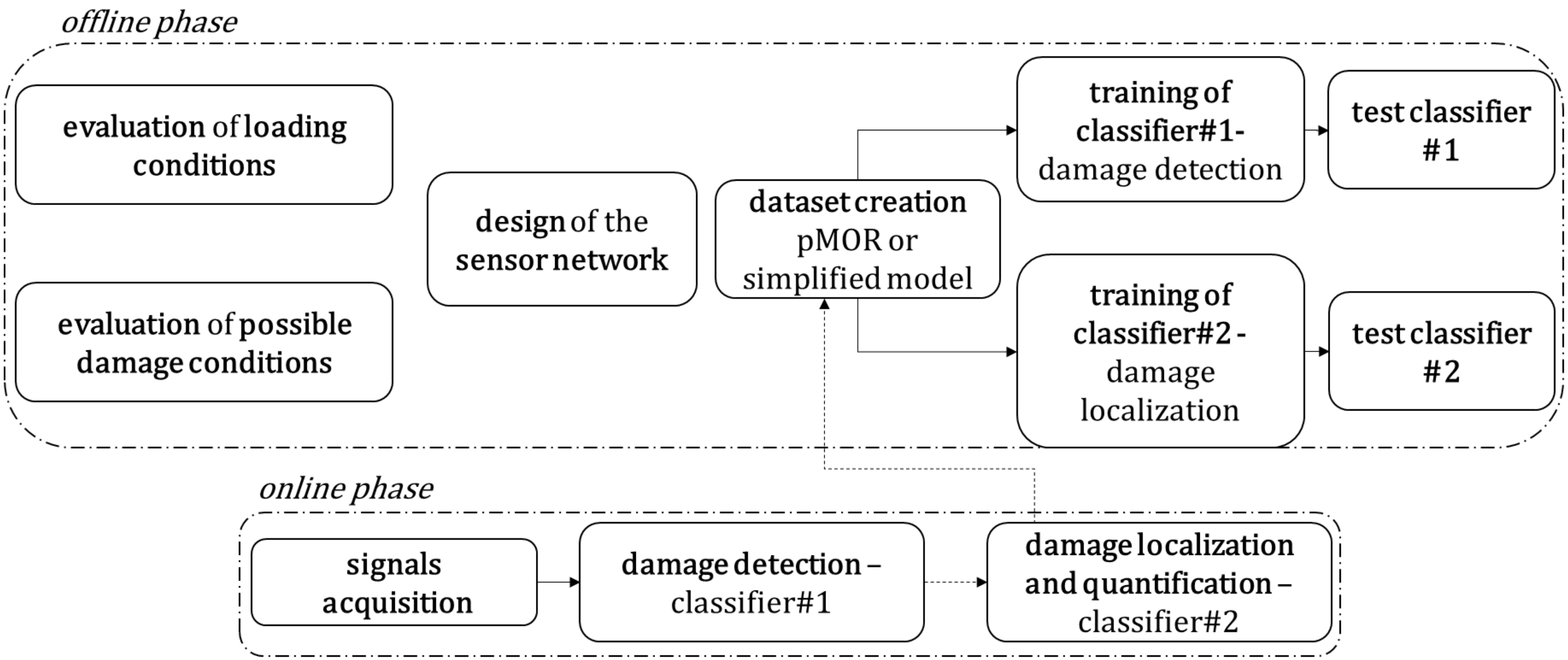
Simulation Based Classification (SBC) is the approach that treats SHM as a *classification problem*, by constructing a database of simulated structural responses under different damage scenarios.

Goals:

- **identify damage-sensitive features** from data acquired with pervasive sensor systems;
- **detect and classify the damage state of the structure**.

Proposal:

- exploit simplified models or parametric **Model Order Reduction (pMOR)** to create the *offline* dataset collecting the outcomes of the sensor system under different damage scenarios;
- train, on the built dataset, a **Fully Convolutional Network (FCN)** able to extract effective features for the classification of the assumed damage scenarios;
- analyse, through the trained classifier, the signals acquired *online* by the sensor system and perform damage detection and identification.



Proposed methodology: ***SBC + reduced/simplified models + FCN.***

Model Order Reduction (MOR) techniques aim to approximate the response of an *high-fidelity* physical system at a low computational cost by using a *low-fidelity* approximation. We consider two different reduction steps:

- **first step**: the *high-fidelity* system response $u(x,t)$ is reconstructed via a *low-fidelity* approximation $\hat{u}(x,t)$ by using the **Proper Orthogonal Decomposition (POD)** method. The *high-fidelity* problem is projected (Galerkin projection) onto the subspace spanned by the **linear combination of basis functions** $\widehat{\Phi}_{u,i}(x)$ called **Proper Orthogonal Modes (POMs)**:

$$u(x,t) \approx \hat{u}(x,t) = \sum_{i=1}^{r} \widehat{\Phi}_{u,i}(x)\,\hat{a}_i(t)$$

where r is the number of basis and $\hat{\boldsymbol{a}}(t)$ is the column vector of the unknown amplitudes of the expansion.

The set of basis functions is constructed via **Singular Value Decomposition (SVD)** from a finite set of n high-fidelity m-dimensional solutions $\boldsymbol{u}(\boldsymbol{x},t_1),\ \boldsymbol{u}(\boldsymbol{x},\boldsymbol{t}_2),\dots,\boldsymbol{u}(\boldsymbol{x},t_m)$ collected in a matrix $\boldsymbol{U}$ during a **training** phase (where x are the m nodal degrees of freedom and t_i the considered time instant).

➢ **second step**: the evolution of the internal $(F_{int}(x,t))$ and external $(F_{ext}(x,t))$ nodal forces is reconstructed using the **DEIM (Discrete Empirical Interpolation method)** algorithm.
DEIM requires to the collected basis functions to **interpolate the solution space at interpolation points called magic points**. It can be implemented by (detailed for $F_{int}(x,t)$):

- ➢ collecting a **series of snapshots** during the training phase $\boldsymbol{F_{int}}(\boldsymbol{x},t_1), \boldsymbol{F_{int}}(\boldsymbol{x},t_2), \dots \boldsymbol{F_{int}}(\boldsymbol{x},t_m)$;
- ➢ **perform a POD** from the collected snapshots getting $\widehat{\Phi}_{\mathbf{Fint},i}(\mathbf{x})$;
- ➢ determining the magic points $\boldsymbol{l}$ using an iterative (**greedy**) procedure;

The solution is reconstructed by:

$$F_{int} \approx \hat{F}_{int} = \sum_{i=1}^{r} \widehat{\Phi}_{Fint,i}\, \hat{a}_{int,i}(t)$$

where the coefficients $\hat{a}_{int,i}(t)$ are determined by solving:

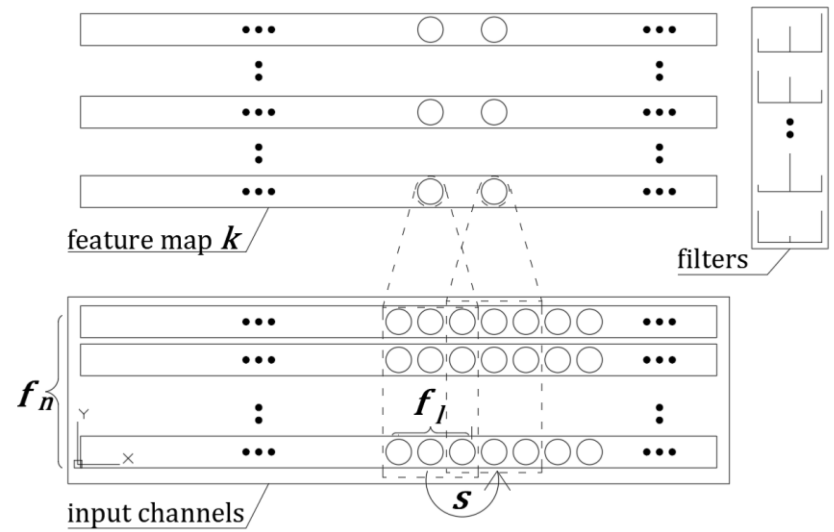
$$\sum_{i=1}^{r} \widehat{\Phi}_{Fint,i}(\boldsymbol{l})\, \hat{a}_{int,i}(t) = F_{int}(\boldsymbol{l},t)$$

Interconnected sensors provide **Multivariate Time Series**.

FCNs with 1d convolutional layers are adopted to:

- **extract features** from each single (monodimensional) time series;
- **recognise the interplay** between different times series or different measurables.
- **classify the inputs** on the base of the extracted features.

The signals acquired with the monitoring sensor system are used as the input channels of the first convolutional layer.



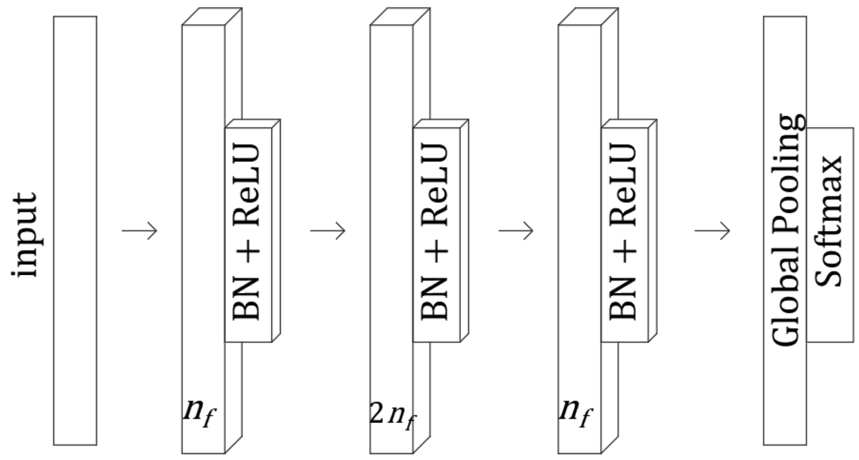
Sketch of 1D convolutional layer.
$\boldsymbol{s}$ is the striding; $\boldsymbol{f_l}$ is the kernel dimension; $\boldsymbol{f_n}$ is the number of input channels.

4 – Fully Convolutional Networks (FCNs): Single Branch Architecture

A Neural Network (NN) stacking three convolutional layers followed by a **global pooling** and a **softmax classifier** is adopted for the classification purposes.

Each convolutional layer is used together with a **Batch Normalization (BN)** and a **Rectified Linear Unit (ReLU)** activation layer.

The number of filters n_f should be chosen on the basis of the complexity of the required classification task.
The adopted filter sequence is $\{n_f, 2n_f, n_f\}$.

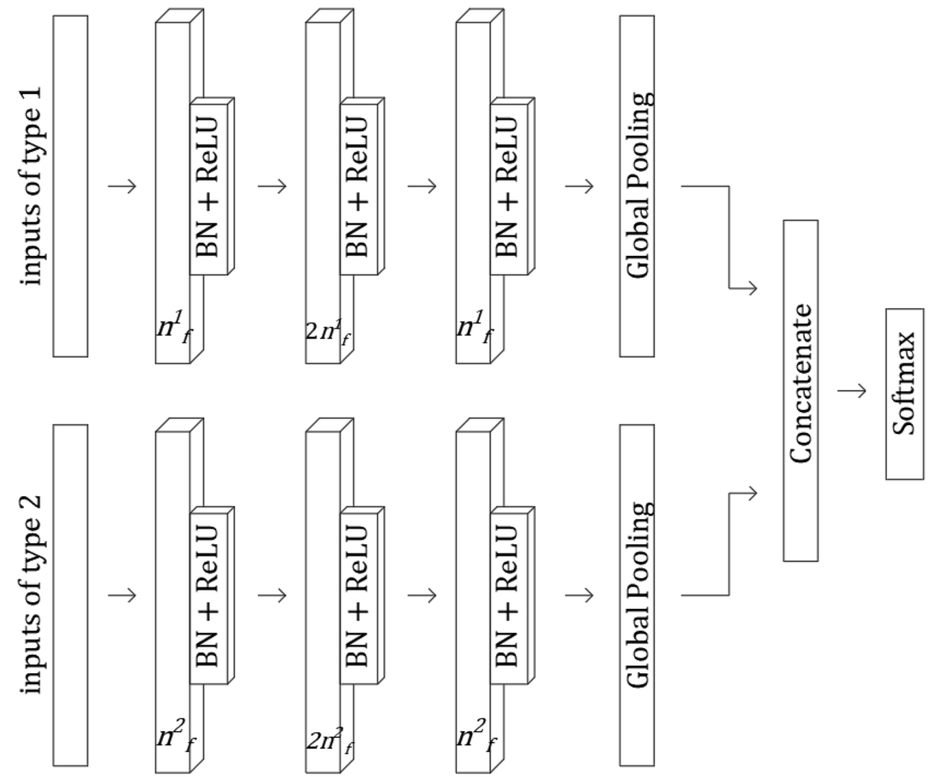


FCN single branch architecture.
n_f is the reference number of filters.

4 – Fully Convolutional Networks (FCNs): Multiple Branches Architecture

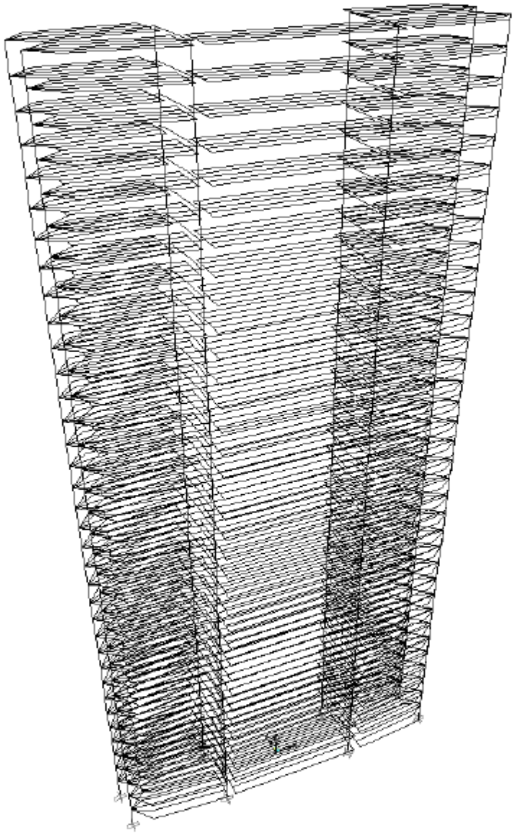
In case of different information sources, a **multiple branches architecture** is employed (a double branch architecture is shown):

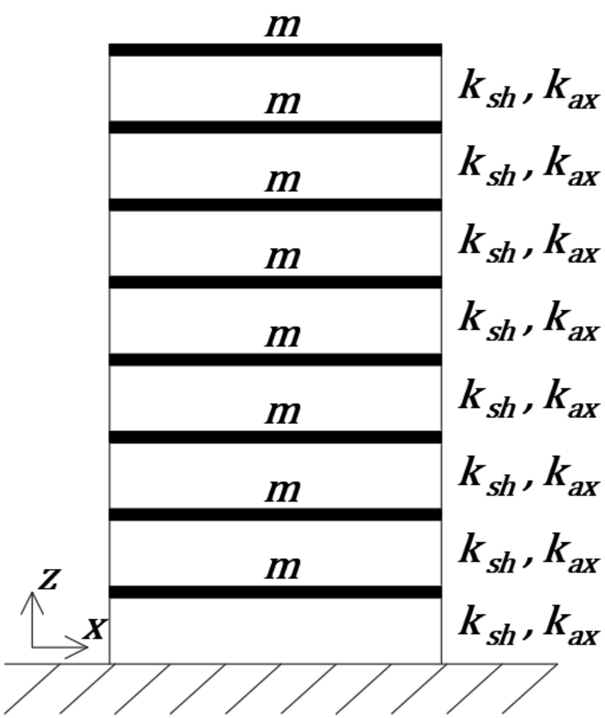
- the convolutional layer architecture is applied separately to each type of information sources;
- the data fusion on the extracted features is performed by a concatenation layer.



FCN double branch architecture.
***n_f** is the reference number of filters.*

What is a shear building model?





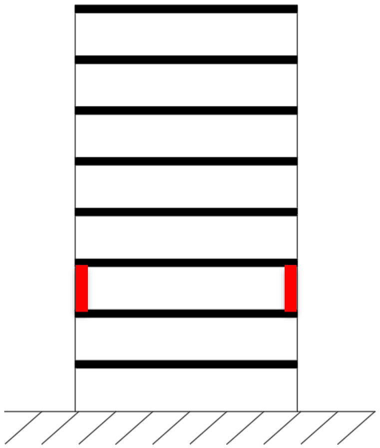
Eight story shear building model.

Simplified model – idealised eight story shear building (**Fig.6**).

- constant floor mass $\boldsymbol{m = 625}$ **t**;
- constant shear interstory stiffness $\boldsymbol{k_{sh} = 10^6}$ **kN/m**;
- constant axial interstory stiffness $\boldsymbol{k_{ax} = 10^8}$ **kN/m**;
- no damping.

Recorded signals – **displacements in x and z direction** of each story.

Hypotised damage scenarios – 25% reduction of one interstory stiffness in turn; labels ranging from 1 for the 1st-floor to 8 for the 8-th floor.



Damage scenario 3.

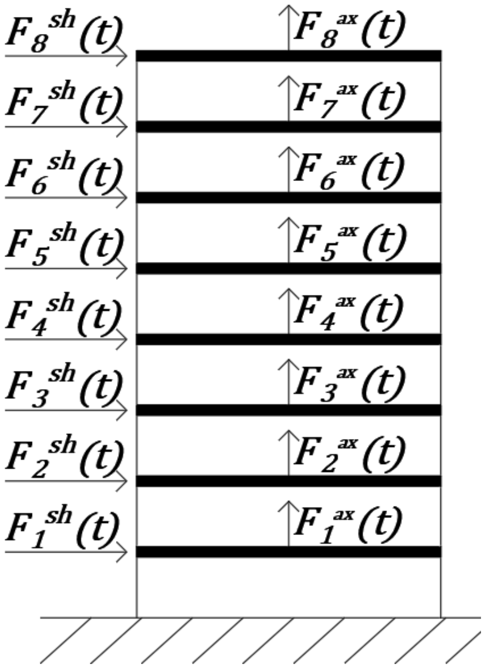
5 – Numerical results: Benchmark 1 - sinusoidal load case (1)

The loads, applied at each story, have been obtained by summing two sinusoids;

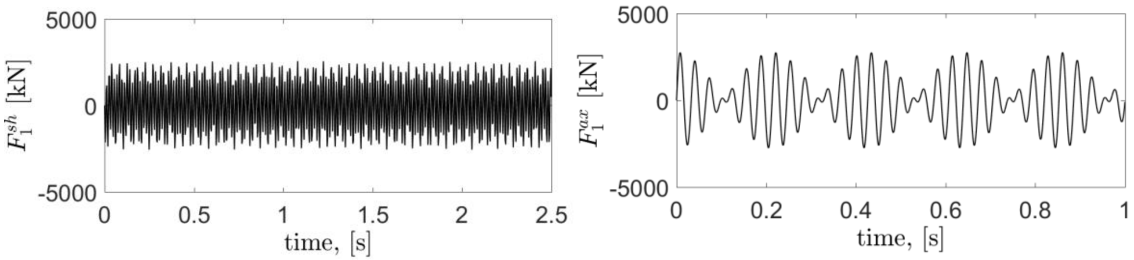
$$F_i^{sh}(t) = \sum_{j=1}^{2} \bar{F}^{sh} \gamma_{i,j}^{sh} \sin(\omega_j^{sh} t + \phi_j^{sh}) \quad F_i^{ax}(t) = \sum_{j=1}^{2} \bar{F}^{ax} \gamma_j^{ax} \sin(\omega_j^{ax} t + \phi_j^{ax})$$

where:

- γ^{sh} and γ^{ax} are scaling factors sampled from $p^{\gamma} \sim \mathcal{N}(0,1)$; γ^{sh} is multiplied by a factor dependent on the considered floor;
- ω^{sh} and ω^{ax} are the frequencies of the sinusoidal components, sampled from a discrete uniform distribution (whose values are estimate of the building structural frequencies) and scaled by a factor sampled from $p^{\gamma} \sim \mathcal{N}(0, \sqrt{2})$.

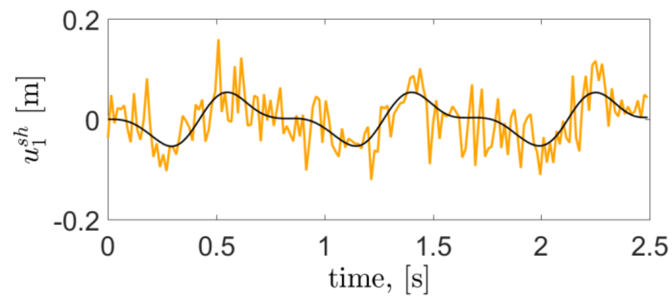


Dynamic loads applied to the model.

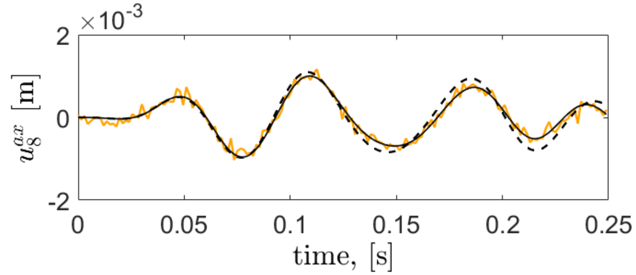


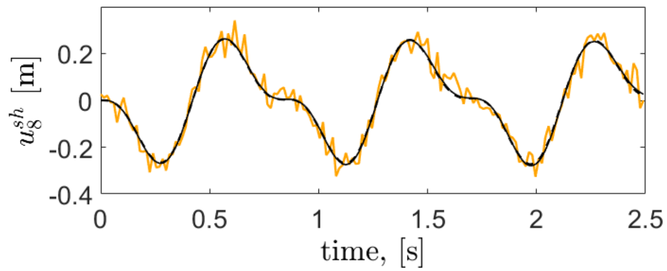
Exemplary time evolutions of the applied loads.

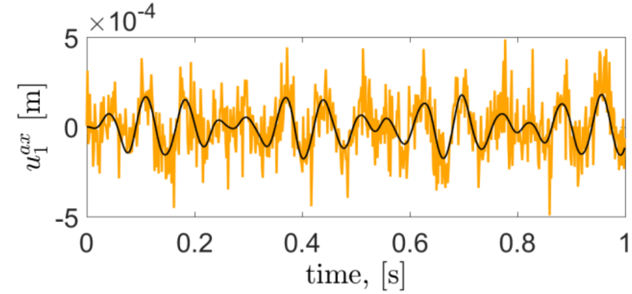
The acquired signals are corrupted with **white noise** to account for the effects of environmental and electrical disturbances. To provide different scenarios in terms of sensor accuracy, two levels of **SNR of 15 dB and 10 dB** are considered.



Orange lines: noisy acquired data; black continuous line: damage scenario 1 (left) and 8 (right); dotted lines: undamaged scenario.







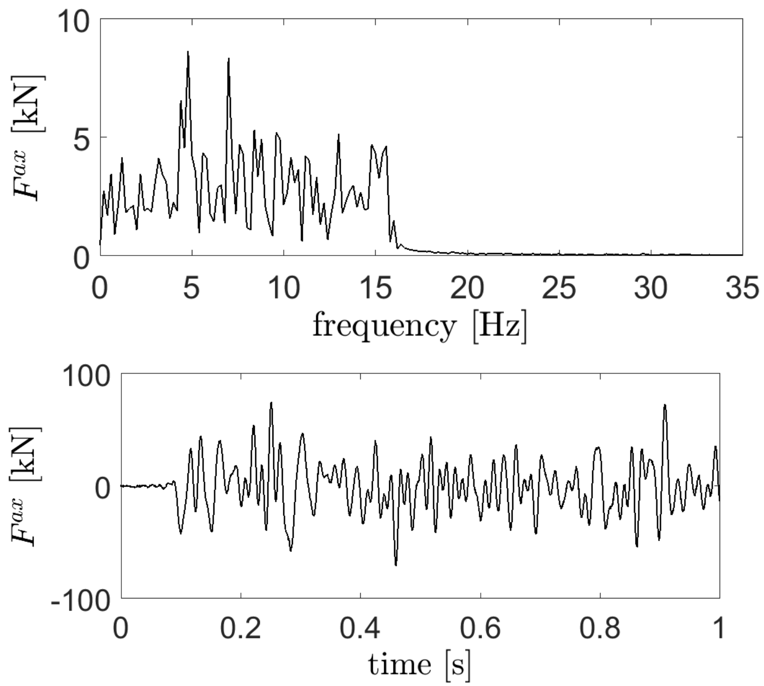
Examplary 1st floor x and z displacements for SNR=10 dB. The orange lines refer to the noisy acquired data.

We account for **random vibrations** due to low-energy seismicity (excitation type expected on site).

The applied excitations, for each direction, are the same for all the floors. Along the simulation, their values have been sampled from a Gaussian probability density function $\mathcal{N}(0,1)$ and scaled by 10^2.

The **frequency range** of the applied forces has been obtained by applying a **low-pass filter**:

- with roll-off set between 15 and 17 Hz;
- with roll-off set between 5 and 7 Hz.

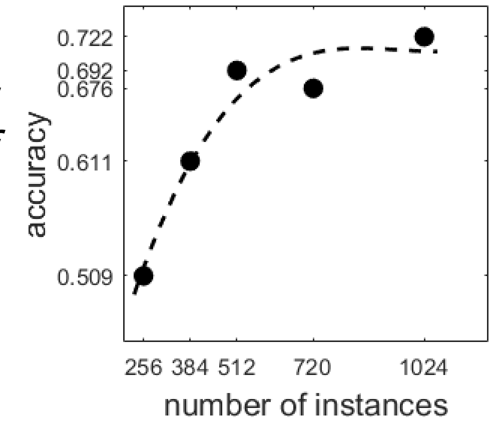


Power spectral density function and time evolution of the axial force with roll-off set between 15 and 17 Hz.

The results refer to a dataset composed by **512 instances** for each damage scenario.

In the reported table, the analysis outcomes are reported (to be compared against the ones produced by a random guess $^1/_9 = 0.111$).

Evaluation of the best number of instances for the dataset.



Target Class \ Output Class	0	1	2	3	4	5	6	7	8
0	52.3%	0.8%	3.1%	2.3%	0.8%	0.0%	0.0%	15.6%	25.0%
1	0.0%	91.4%	8.6%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
2	0.0%	0.8%	96.9%	1.6%	0.0%	0.0%	0.0%	0.0%	0.8%
3	0.0%	0.0%	2.3%	89.1%	1.6%	0.0%	0.0%	0.0%	7.0%
4	0.8%	0.0%	0.0%	8.6%	71.1%	3.1%	9.4%	4.7%	2.3%
5	1.6%	0.0%	0.0%	3.9%	3.1%	60.9%	17.2%	1.6%	11.7%
6	1.6%	0.0%	0.0%	3.9%	7.0%	12.5%	45.3%	25.8%	3.9%
7	3.1%	0.0%	0.8%	8.6%	2.3%	0.0%	8.6%	58.6%	18.0%
8	8.6%	0.0%	3.1%	6.3%	2.3%	0.0%	0.0%	9.4%	70.3%

Confusion matrix related to $SNR = 10$ dB

SNR (dB)	input signals displacement direction	accuracy
15	x	0.768
15	z	0.769
15	x and z	0.812
10	x	0.654
10	z	0.642
10	x and z	0.707

Results for different SNR and employed input signals.

In the reported table the analysis outcomes for the white noise load case are shown (to be compared against the ones produced by a random guess $^1/_9 = 0.111$).

roll-off frequencies (Hz)	input signals displ. direction	accuracy
15 – 17	x	0.998
15 – 17	z	0.997
15 – 17	x and z	0.999
5 – 7	x	0.996
5 – 7	z	0.892
5 – 7	x and z	0.998

Results for different SNR and employed input signals.

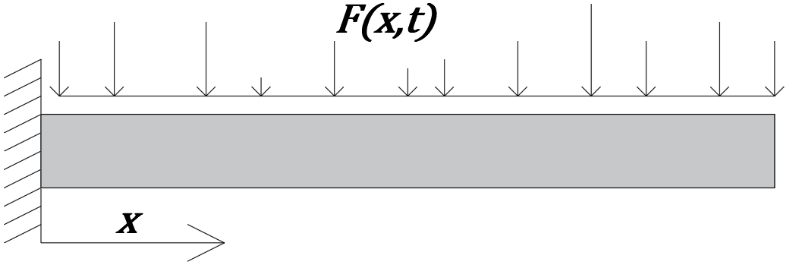
Despite of having excited just a few structural frequencies, **the NN can accomplish the classification** of the damaged scenarios almost perfectly **under the considered stochastic framework.**

The **results**, both for the sinusoidal load case and the white noise case, **are empowered by the employment of the double branch architecture**.

5 – Numerical results: Benchmark 2 - cantilever beam FE model (1)

The following data are considered (geometry depicted in **Fig.15**):

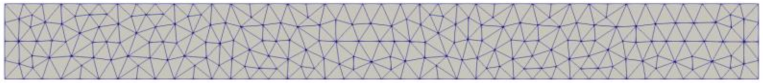
- **length**: $\mathrm{L} = 4$ m;
- **height**: $\mathrm{h} = 0.4$;
- **density**: $\boldsymbol{\rho} = 7800$ kg/ m³;
- **Poisson**: $\nu = 0.3$;
- **Young modulus**: $\mathrm{E} = 210\ 10^9$ N/m²;
- **planestress** condition;
- **load** applied at the beam upper boundary. $F(x,t)$ **sampled from a Gaussian probability density function**.



Geometry.

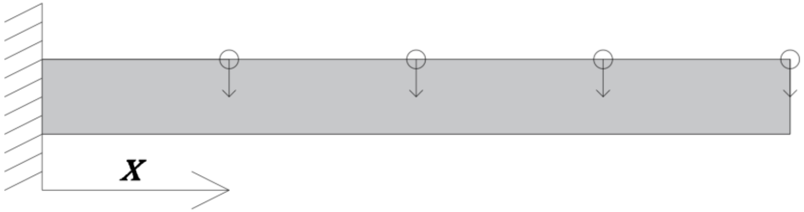
The employed **discretization** is shown in **Fig.16**:

- **CST elements**;
- **252 nodes**;
- **464 elements**;
- **integration step**: $t = 5 \cdot 10^{-4}$ s;
- **analysis time**: $t = 0.8$ s;



FE mesh.

- **the vertical displacements of the points at** $\boldsymbol{x} = [\mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}]$ of the beam upper boundary are recorded;

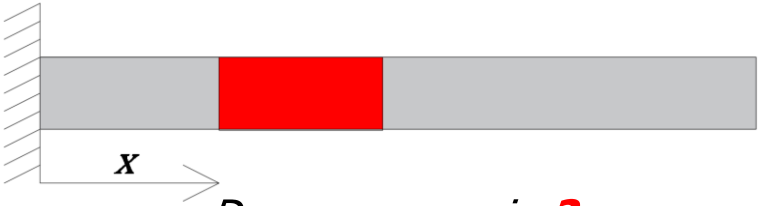


Recorded displacements.

- **Four hypothesized damage scenarios** have been simulated. They concern the 25% reduction of the beam stiffness between:

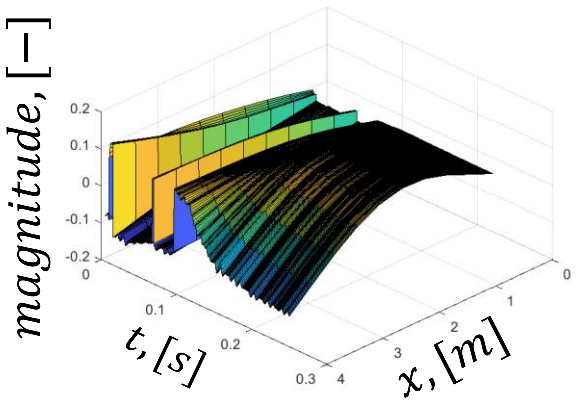
$$\begin{cases} \mathbf{0} < \boldsymbol{x_{GP}} \leq \mathbf{1} & \text{damage scenario } \mathbf{1}; \\ \mathbf{1} < \boldsymbol{x_{GP}} \leq \mathbf{2} & \text{damage scenario } \mathbf{2}; \\ \mathbf{2} < \boldsymbol{x_{GP}} \leq \mathbf{3} & \text{damage scenario } \mathbf{3}; \\ \mathbf{3} < \boldsymbol{x_{GP}} < \mathbf{4} & \text{damage scenario } \mathbf{4}; \end{cases}$$

where x_{GP} are the abscissas of the elements Gauss points.

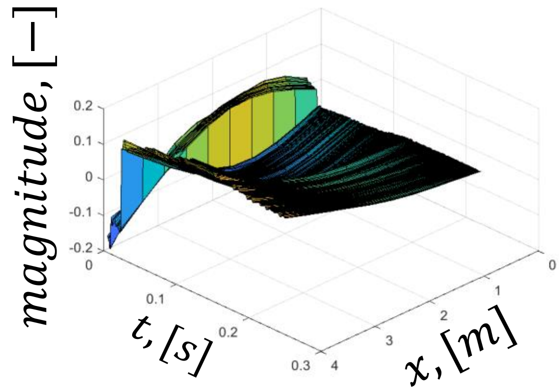


Damage scenario ***2.***

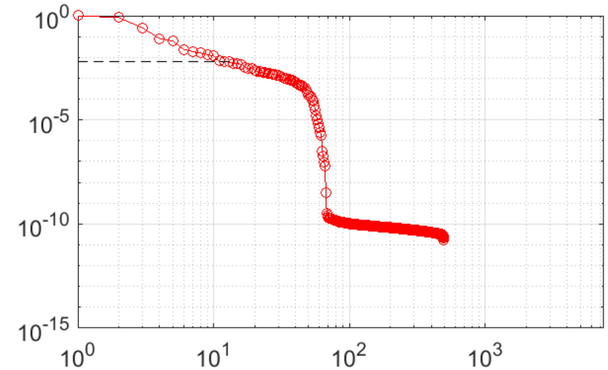
First step – reconstruction of the system response $u(x,t)$ via POD. $tol_u = 10^{-2}$. Training time $0.25\ s$. **12 selected basis function $\widehat{\Phi}_{u,i}(x)$.**



1st POM convergence.

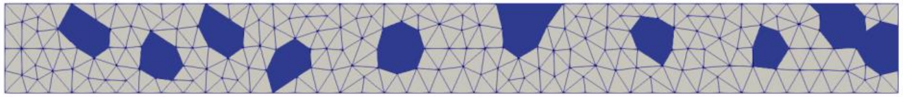


2nd POM convergence.



Energetic content of the selected POMs compared to the energetic content of the matrix of snapshots $\mathbf{U}$.

Second step – reconstruction of the time evolution of the internal and external nodal forces $F_{int}(x,t)$, $F_{ext}(x,t)$ via POD. $tol_{Fint} = 10^{-6}$, $tol_{Fext} = 10^{-6}$. Analysis training time $0.25\ s$. **13 selected basis function $\widehat{\Phi}_{Fint,i}(x)$ for $F_{int}(x,t)$, 1 selected basis function for $F_{ext}(x,t)$, $\widehat{\Phi}_{Fext,i}(x)$:**

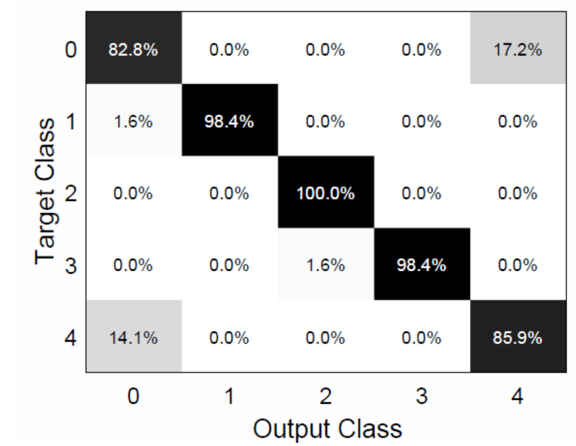


Interpolation (magic) points for F_{int}.

The **FCN is trained on a dataset whose instances are simulated using the reduced order model**. Its generalization capacities are then tested:

- **on instances generated by the reduced order model itself**, not seen during the training (in the relative confusion matrix is reported).

Except for the damage scenario 4, the **FCN is able to perform the classification task;**



Confusion matrix for instances generated by ROM.

- **on instances produced by the full order model.**

In this case, the **FCN is not able to perform the classification task** for the difficulty of the reduced model order of reproducing the dynamic response of the system. This difficulty is due to the stochastic nature of the applied excitation and to the consequent **lack or representativity of the snapshots** used for the training of the reduced order model.

- the **application of FCN** to the classification of Multivariate Time Series **exhibits very good performances** and **is noise-tolerant**.

- a **representativity problem** is encountered when a reduced order model is used to simulate a **stochastic framework**. This discrepancy biases the trained NN, that is not able to correctly classify instances generated by the full order mode. This issue is known as **domain adaptation** problem in the machine learning community.

Thank you for your attention!