Sensor placement strategy to inform decisions

Citation: Mainini, Laura, and Karen E. Willcox. "Sensor Placement Strategy to Inform Decisions." 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, 5-9 June, 2017, Denver, Colorado, American Institute of Aeronautics and Astronautics, 2017.

As Published: https://doi.org/10.2514/6.2017-3820

Publisher: American Institute of Aeronautics and Astronautics

Persistent URL: http://hdl.handle.net/1721.1/116083

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Sensor placement strategy to inform decisions

Laura Mainini *
Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139, USA
United Technologies Research Center, Cork, Ireland
and
Karen E. Willcox †
Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139, USA

This paper introduces a computational strategy to determine optimal sets of sensor locations to support real-time operational decisions. We exploit unsupervised learning strategies (specifically self-organizing maps) to identify the most informative locations to place sensors. The sensor placement procedure is then combined with a Multi-Step-Reduced Order Modeling approach that exploits the low-dimensional map between the sparse sensed data and the decisions at hand. The approach is demonstrated for the real-time assessment of an unmanned aircraft wing panel undergoing structural degradation. For this application, we compare the optimal sets of sensor locations with random placements for a variety of sensor availabilities. By adopting our placement strategy, we achieve improvements in accuracy and robustness of capability predictions, even when measured data are sparse and cover less than 10% of the reference data.

I. Introduction

The next generations of transportation systems will be able to autonomously develop operational strategies in real-time, taking into account the evolution of their health condition along with the dynamic changes of the surrounding environment. One such example is the case of autonomous aerospace vehicles that dynamically adapt their mission to the evolution of their structural state. For problems of this kind, one challenge is the limited time and resources that can be dedicated onboard to data processing. Cost and weight of state-of-the-art sensing technologies represent another relevant constraint. Furthermore, both onboard measurements and inference from sparse sensor data introduce uncertainties that can potentially affect the overall data-to-decisions process. These challenges motivate the development of strategies to optimize placement of sensors.

To tackle these challenges, in previous works[1–3] we proposed a methodological framework combining a *Sense-Infer-Plan-Act* formulation of the decision process with a *Multi-Step Reduced Order Modeling* (MultiStep-ROM) strategy for efficient real-time computing (Figure 1). Specifically, we model the flow of information that processes data to make decisions as a Sense-Infer-Plan-Act flow associated with measurements (quantities that can be monitored with sensors and carry information about the state of the system) and capabilities (quantities that evolve with the state of the system and constrain the decision space accordingly). Then, we adopt MultiStep-ROM to map from measurements to capabilities in support of real-time operations. The MultiStep-ROM scheme, originally introduced in Ref. 1, combines reduced-order models, surrogate modeling and clustering techniques to obtain an adaptive offline-online procedure suitable for time-constrained problems.

Previous studies,[3] conducted for the assessment of structural integrity of an unmanned air-vehicle wing panel, revealed that if sensors are randomly placed and cover less than 90% of the entire wing panel, measured data may be misleading or insufficient to properly represent the state of the system. In the current paper,

*Research Affiliate, Department of Aeronautics and Astronautics; Senior Research Scientist, System Modeling and Optimization Group; AIAA Member.
†Professor, Department of Aeronautics and Astronautics, AIAA Associate Fellow.

we aim to overcome these limitations by substituting the random placement with a principled approach, specifically designed to determine the best sensor locations that support effective decision making. To do that, we investigate compressive sensing[4–15] and sensor placement methodologies,[16–21] which are receiving attention from the scientific community for their effectiveness in compressing high-dimensional quantities for reconstruction purposes and identifying damage in structural systems, respectively. An open question concerns the possibility to employ similar techniques to either support classification tasks or directly inform decisions,[22,23] as required for Sense-Infer-Plan-Act problems.

To address this question, we propose a computational methodology to determine the most informative sensor locations and effectively assist the decision process associated with real-time operations. Our strategy adopts unsupervised learning techniques, in particular using Self-Organizing Maps, to exploit the low-dimensional structure of the physical quantities to measure. We demonstrate our approach for data-to-decisions problems including a structural monitoring task. In order to assess our methodology for classification and decision purposes, we couple the strategy with a MultiStep-ROM procedure. Our main goal is to dramatically improve accuracy and robustness of capability predictions for measurements covering less than 10% of the reference data. Ultimately, such improvements will allow for a drastic reduction in the number of sensors required to achieve reliable predictions and well-informed decisions, along with large savings in cost and weight of sensing equipment.

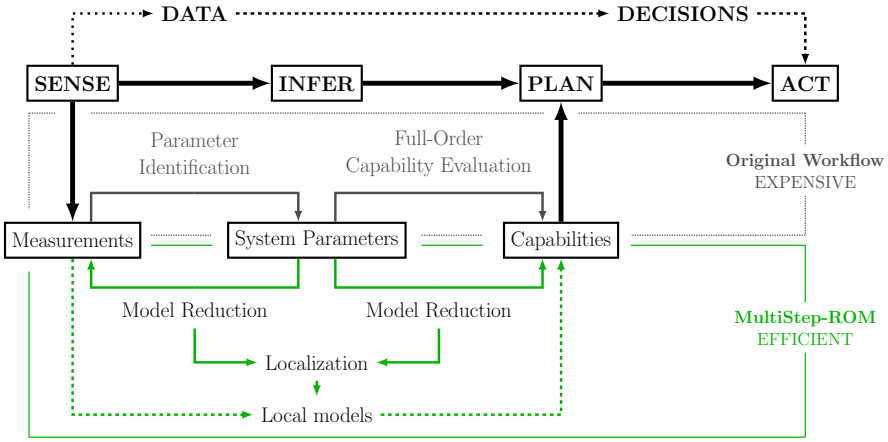


Figure 1. Sense-Infer-Plan-Act information flow and MultiStep-ROM scheme to address the time-constrained problem of mapping from measurements to capabilities in real-time.

This paper is organized as follows. Section II briefly introduces our approach and the main features of the Self-Organizing Maps, as the unsupervised learning technique at the basis of our sensor placement strategy. Section III presents the specific application to on-board structural assessment considered in this paper and discusses the main outcomes. Finally, concluding remarks are summarized in Section IV.

II. Method

This paper proposes a novel strategy to compute optimal sensor sites in support of real-time decisions. Our method relies on the use of Self-Organizing Maps[24–26] to identify the best candidate locations. The efficiency and effectiveness of SOM in compressing information for reconstruction purposes is well known[27,28] and a rich literature introduces SOM as a technique to realize an ordered form of vector quantization.[29–32] In contrast with that, this work does not focus on reconstruction goals: we adopt SOM to obtain optimal sparse measurements to inform classification and decision tasks. To do that, we wish to exploit not only a sparse representation of the physical quantities to measure/monitor, but also the low-dimensional mapping that relates capabilities to measurements, which can be computed with a MultiStep-ROM procedure (Figure 1). Section II.A briefly recalls the main stages of the MultiStep-ROM procedure adopted in this paper to map from measurements to capabilities within the Sense-Infer-Plan-Act framework. Then, Section II.B introduces our sensor placement strategy based on unsupervised learning and illustrates its integration with the mapping strategy.

A. MultiStep-ROM procedure

The MultiStep Reduced-Order Modeling strategy adopted in this paper adheres to the procedure proposed in Ref. 1, and consolidated in Ref. 2 and 3. The methodology relies on a traditional offline-online structure. During the offline phase we exploit the information provided by high-fidelity complete datasets to obtain an adaptive efficient model. The efficient model is then used during the online phase to rapidly process the information provided by sparse sensor measurements and estimate specific quantities of interest. The MultiStep procedure is named after the particular four-step structure that characterizes both offline and online computational phases.

Offline During the offline phase the information provided by high fidelity datasets is extracted and modeled through a combination of model-order reduction and localization techniques.

1. Data collection The first offline step is the collection of high-fidelity data in the form of complete snapshots of physical quantities of interest. In particular, we distinguish between quantities to be monitored online (measurements, $\mathbf{q}_m(\mathbf{x}), m = 1, ...M$) and quantities that characterize system capabilities but are not directly measured (capabilities, $\mathbf{s}_c(\mathbf{x}), c = 1, ...C$). Measurements and capabilities are all functions of system parameters $\mathbf{x}$; high-fidelity dataset are a collection of n_e-dimensional complete snapshots of quantities of interest computed for n_s different system conditions $\mathbf{x}$.

2. Projection-based model order reduction The second step computes reduced-order models for each measurement ($\tilde{\mathbf{q}}_m$) and capability ($\tilde{\mathbf{s}}_c$) through parametric Proper Orthogonal Decomposition (POD).[33–42] For each quantity we assemble the $n_e \times n_s$ matrix of complete snapshots and compute the POD basis vectors via singular value decomposition:[43,44]

$$\tilde{\mathbf{q}}_m(\mathbf{x}) = \bar{\mathbf{q}}_m + \sum_{j=1}^{n_m} \alpha_j^m(\mathbf{x})\phi_j^m \quad m = 1, ..., M; \qquad \tilde{\mathbf{s}}_c(\mathbf{x}) = \bar{\mathbf{s}}_c + \sum_{j=1}^{\ell_c} \beta_j^c(\mathbf{x})\psi_j^c \quad c = 1, ..., C. \tag{1}$$

   In Equation (1), $\bar{\mathbf{q}}_m$ and $\bar{\mathbf{s}}_c$ are the average value of $\mathbf{q}_m$ and $\mathbf{s}_c$, computed over the respective set of n_s snapshots. In the expansion terms, $\{\phi_j^m\}_{j=1}^{n_m}$ and $\{\psi_j^c\}_{j=1}^{\ell_c}$ denote the mth measurement POD modes and the cth capability POD modes, respectively; similarly, $\alpha_j^m,\ j = 1, ..., n_m$ indicates the mth measurement POD modal coefficients and $\beta_j^c,\ j = 1, ..., \ell_c$ represents the cth capability POD modal coefficients. The model reduction is obtained by truncating the POD expansion such that $n_m \ll n_e$ and $\ell_c \ll n_e$.

3. Localization The third step exploits unsupervised learning techniques to identify clusters in the space of the measurement and capability POD coefficients. The clusters empirically identify validity subdomains over which global complex phenomena can be locally represented with simple models. Our localization step exploits Self-Organizing Maps, a neural network paradigm that we also adopt in our sensor placement strategy. The SOM network is trained with n_s input points $\boldsymbol{\tau}(\mathbf{x})$ defined as follows:

$$\boldsymbol{\tau}(\mathbf{x}) = \left[\{\alpha_j^m(\mathbf{x})\}_{\substack{j=1,...,n_m \\ m=1,...,M}}, \{\beta_j^c(\mathbf{x})\}_{\substack{j=1,...,\ell_c \\ c=1,...,C}} \right] = [\boldsymbol{\alpha}(\mathbf{x}), \boldsymbol{\beta}(\mathbf{x})]\,. \tag{2}$$

   Each vector has dimension $n_{\text{pod}} = n_\alpha + n_\beta$, where $n_\alpha = \sum_{m=1}^{M} n_m$ and $n_\beta = \sum_{c=1}^{C} \ell_c$ denote the size of vectors $\boldsymbol{\alpha}(\mathbf{x})$ and $\boldsymbol{\beta}(\mathbf{x})$, respectively. SOM training identifies n_w clusters in the input space and computes the representative weight vector $\mathbf{w}_k$ of each cluster as the one for which the distance from the training points in k is minimized:

$$k = \underset{j \in 1,...,n_w}{\arg\min} \{\|\boldsymbol{\tau}_i - \mathbf{w}_j\|_\Lambda\}. \tag{3}$$

   In (3), the distance metric $\|\cdot\|_\Lambda$ is an L_2-norm scaled with normalized POD eigenvalues.[1] A the end of the training phase, the weight vectors represent cluster prototypes and embody the average properties of their training elements.

4. Local models The last offline step characterizes each subdomain k with a dedicated set of simple data-fit models. In practice, we seek for local representations of capability coefficients $\boldsymbol{\beta}$ as function of

measurement coefficients $\boldsymbol{\alpha}$. For every capability coefficient β_j^c we compute a simple response surface model $\beta_j^c(\boldsymbol{\alpha})$, valid exclusively over the kth subdomain:

$$\boldsymbol{\beta}^{(k)} \approx \tilde{\boldsymbol{\beta}}(\boldsymbol{\alpha})^{(k)} = \{\tilde{\beta}_j^c(\boldsymbol{\alpha})\}_{\substack{j=1,\dots,\ell_c \\ c=1,\dots,C}}^{(k)}, \qquad k = 1, ..., n_c, \tag{4}$$

with $\tilde{\beta}_j^c(\boldsymbol{\alpha}) \approx \beta_j^c(\mathbf{x})$ and $\boldsymbol{\alpha} = \boldsymbol{\alpha}(\mathbf{x})$ as defined in (2).

Online The online phase is the computational procedure to run on-board and assist decision making in real-time.

1. Measurement coefficients reconstruction The first online step reconstructs POD coefficients of the measured quantities of interest using information provided by incomplete snapshots $\hat{\mathbf{q}}_m$, $m = 1, ..., M$. The measurement POD basis computed offline (1) are used to estimate the coefficients via gappy POD;[45–48] M linear systems is the form

$$\mathbf{G}^m \boldsymbol{\alpha}_{\mathrm{g}}^m = \mathbf{f}^m \tag{5}$$

are solved to determine the unknown POD coefficients $\boldsymbol{\alpha}_{\mathrm{g}}^m = \left[\alpha_{\mathrm{g}1}^m, ..., \alpha_{\mathrm{g}n_m}^m\right]^\top$. In Eq. 5, the ijth entry of $\mathbf{G}^m$ is $G_{ij}^m = (\phi_i^m, \phi_j^m)_{\mathrm{g}}$, the ith entry of vector $\mathbf{f}^m$ is $f_i^m = (\hat{\mathbf{q}}_m, \phi_i^m)_{\mathrm{g}}$, and $(\cdot,\cdot)_{\mathrm{g}}$ denotes the gappy inner product which considers only those elements in the vectors that correspond to the available sensed data. In the remaining of the paper, $\boldsymbol{\alpha}_{\mathrm{g}} = \left[(\boldsymbol{\alpha}_{\mathrm{g}}^1)^\top, ..., (\boldsymbol{\alpha}_{\mathrm{g}}^M)^\top\right]$ indicates all the measurement coefficients reconstructed in this first online step.

2. Classification The second step classifies the sensed information into the closest cluster, that is, the one whose weight vector $\mathbf{w}_{k^*}$ minimizes the distance

$$k^* = \underset{j \in 1,\dots,n_w}{\arg\min} \{\|\boldsymbol{\tau}^* - \mathbf{w}_j\|_{\Lambda_\alpha}\}. \tag{6}$$

In Eq. 6, vector $\boldsymbol{\tau}^* = [\boldsymbol{\alpha}_{\mathrm{g}}, \boldsymbol{\beta}^*]$ includes the measurement coefficients $\boldsymbol{\alpha}_{\mathrm{g}}$ reconstructed via gappy POD and the vector of unknowns $\boldsymbol{\beta}^*$. The distance $\|\cdot\|_\Lambda$ is now computed only over the $\boldsymbol{\alpha}_{\mathrm{g}}$ elements of the vectors and denoted $\|\cdot\|_{\Lambda_\alpha}$.

3. Local approximation The third step fully exploits the set of local maps $\boldsymbol{\beta}^{(k^*)}$ computed offline (Eq. 4) to approximate each modal coefficient of the capabilities as a function of the reconstructed $\boldsymbol{\alpha}_{\mathrm{g}}$:

$$\beta_j^c \approx \tilde{\beta}_j^c(\boldsymbol{\alpha}_{\mathrm{g}})^{(k^*)}. \tag{7}$$

4. Capability estimate The last step exploits the POD expansion computed offline (Eq. 1) to estimate each capability $\tilde{\mathbf{s}}_c$ as a combination of its dominant modes ψ_j^c. In contrast with the offline expansion, the coefficients β_j^c are now approximated with the k^*th set of local models:

$$\tilde{\mathbf{s}}_c(\boldsymbol{\alpha}_{\mathrm{g}}) = \bar{\mathbf{s}}_c + \sum_{j=1}^{\ell_c^{k^*}} \tilde{\beta}_j^c(\boldsymbol{\alpha}_{\mathrm{g}})^{(k^*)} \psi_j^c \qquad c = 1, ..., C. \tag{8}$$

B. Sensor placement strategy

This work proposes a sensor placement strategy to compute and leverage sparse representations of measurements and capabilities in support of real time data to decision. Our particular methodology exploits Self-Organizing Maps and identifies the most informative sensor locations to monitor measurements and estimate capabilities.

Self-Organizing Maps, also referred to as Kohonen's Maps, are single-layer neural networks that rely on unsupervised competitive learning to adapt the neural nodes and compute models from a given set of input data. SOM neurons have two representations: (i) weight vectors that assume real values in the space of input data, and (ii) nodes in the topological space of the network. Usually, the input space is a high-dimensional manifold, while the topological space is a low-dimensional array (for instance, a 2D grid). Given a training

input dataset, the learning algorithm computes the weight vectors, $\boldsymbol{\omega}_j \in \mathcal{R}^{n_d}$, by progressively updating their element values. At each step, a training vector $\mathbf{t}_i$ is associated with the closest weight vector $\boldsymbol{\omega}_\pi$ in the input space

$$\pi = \underset{j \in 1,...,n_\omega}{\arg\min} \{\|\mathbf{t}_i - \boldsymbol{\omega}_j\|\}. \tag{9}$$

Then, the learning algorithm updates the closest weight vector $\boldsymbol{\omega}_\pi$ and its neighbours according to the proximity of the corresponding nodes in the topological space. Details about mathematical foundations, algorithms and possible implementations of SOMs can be found in many relevant works including Ref 25, 26 and 49.

The training process determines n_π distinct clusters of input data; each cluster is led by a weight vector (namely, the prototype) representing its centroid in the n_d-dimensional input space. The number of clusters is not imposed a priori; still, it is limited by the number n_ω of constitutive neurons (weight vectors) of the SOM network. Therefore, $n_\pi \leq n_\omega$ because it may happen that not all the n_ω vectors end up leading a cluster. The particular structure of SOM paradigms guarantees that similar points in the input space preserve their proximity in the topological space. In fact, the SOM realizes an ordered non-linear projection of the high-dimensional data onto a low-dimensional manifold.

Our strategy for sensor placement exploits SOMs to compute weight vectors in the particular input space defined by dominant modes of measured quantities of interest and geometry coordinates of the specific system to monitor. This paper proposes and discusses a first implementation of this strategy to identify a set of informative sensor locations for all the M measured quantities of interest at once. This goal is achieved by using a unique SOM network and leveraging the information content of the first dominant modes of measured quantities only. The first dominant modes of measurements ($\phi_1^m,\ m = 1, ..., M$) are computed during the offline phase of the MultiStep-ROM procedure through proper orthogonal decomposition (see Offline Step 2 in Section II.A). ϕ_1^m is a n_e-dimensional vector whose elements are associated with spatial coordinates of physical points $\mathbf{r}_i \in \mathcal{R}^{n_r}$ of the system to monitor. Now, the set of sensor locations (SSL) can be computed with the following procedure.

Input dataset definition The dataset $\mathbf{T}$ collects training input for the SOM networks and consists of n_e points $\mathbf{t}_i = \left[\phi_1^1(\mathbf{r}_i), ..., \phi_1^M(\mathbf{r}_i), \mathbf{r}_i\right]$ such that:

$$\mathbf{T} = \left[\mathbf{t}_1, ..., \mathbf{t}_{n_e}\right]^\top = \left[\phi_1^1, ..., \phi_1^M, \left[\mathbf{r}_1, ..., \mathbf{r}_{n_e}\right]^\top\right]. \tag{10}$$

$\mathbf{T}$ is a $n_e \times (M + n_r)$ matrix whose first M columns report the first POD modes of the measured quantities of interest; the remaining n_r columns record spatial coordinates of all the n_e physical grid points associated with measured values. These grid points are all eligible sites to place sensors.

Network training Dataset $\mathbf{T}$ can now be used to train SOM and compute the n_π weight vectors. At each training step, the weight vector to update is determined as per Eq 9. Specifically, we introduce the particular metric $\|\cdot\|_\rho$ to measure the distance between the input vector and the weight vectors at each training step:

$$\|\mathbf{t}_i - \boldsymbol{\omega}_j\|_\rho = \sqrt{(\mathbf{t}_i - \boldsymbol{\omega}_j)^\top \boldsymbol{\rho}\mathbf{I}\ (\mathbf{t}_i - \boldsymbol{\omega}_j)}. \tag{11}$$

Eq. 11 indicates a scaled L_2-norm where $\boldsymbol{\rho}$ is a vector of weights with $\|\boldsymbol{\rho}\|_1 = 1$ and $\mathbf{I}$ is the identity matrix. Combining Eq. 9 and 11, the closest weight vector is determined as the one for which the distance from the training vectors in cluster π is minimized:

$$\pi = \underset{j \in 1,...,n_\omega}{\arg\min} \{\|\mathbf{t}_i - \boldsymbol{\omega}_j\|_\rho\}. \tag{12}$$

The weighted distance regulates the relevance of measurement modes with respect to spatial coordinates in determining the optimal placement of sensing sites. In terms of practical applications, such a metric enables our placement strategy to account for the physical constraints related to size and accuracy of sensing technologies.

Identification of sensor locations Once training is completed, the weight vectors $\boldsymbol{\omega}_j = [\omega_1, ..., \omega_{M+n_r}]_j$ enclose the sensor locations. Specifically, weight vectors' elements corresponding to spatial coordinates (the last n_r entries) identify the most informative sites to place sensors:

$$\text{SSL} = \{[\omega_{M+1}, ..., \omega_{M+n_r}]_j\}_{j=1}^{n_c} \tag{13}$$

The remaining elements (the first M entries) encode the dominant modes of the measured quantities of interest. In addition, cardinality and radius of the clusters associated with the weight vectors/neurons may be employed in future as quantitative measures to assist dynamic policies for an adaptive real-time allocation of resources.

Sensor placement is a form of resource allocation conducted offline and exploited online. Our strategy takes most advantage of this offline-online structures when combined with a MultiStep-ROM procedure. Figure 2 illustrates the information flow that characterizes such an integrated approach. The information flow shows that the sensor placement strategy is designed to support the decisional process, rather than measurement reconstruction. Indeed, our main goal is to assist and enable real-time decisions, rather than the reconstruction of measured quantities of interest from sparse data.

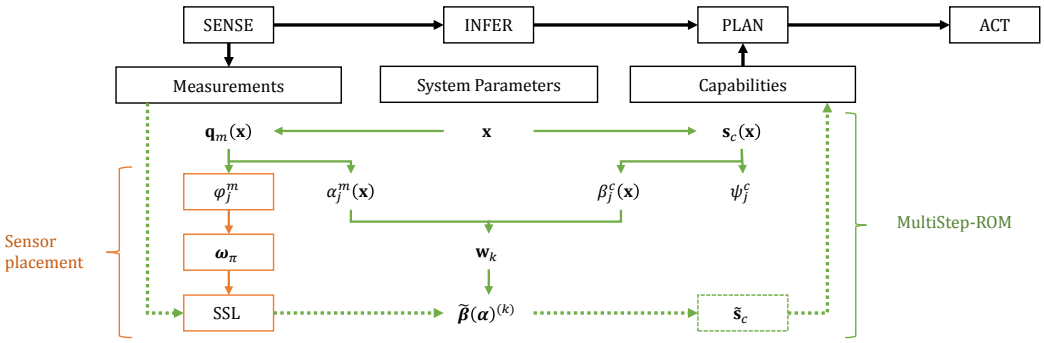


Figure 2. Sensor placement strategy (red flow) in support of decisional process. Online (dashed lines), measurements are recorded in the (sparse) SSL determined offline. These sparse measurements are used to estimate capabilities and constrain the decision space via MultiStep-ROM.

III. Application to on-board structural assessment

We demonstrate our sensor placement technique for a real-time decision problem associated with structural assessment on-board an unmanned air vehicle (UAV). We consider the specific test case of UAV composite wing panel that undergoes degradation of its structural properties for a variety of damage conditions. In particular, our application is represented by a composite square panel subjected to static loading (Figure 3). It is a 18×18 square-inch structure consisting of four plain-weave carbon-fiber plies with symmetric stacking sequence $[45°, 0°]_s$; two additional plies, with orientation $0°$ and $45°$, reinforce the borders where 24 bolts are placed to fasten the plate. To obtain the reference data, we adopt a numerical model based on the finite element method (FEM): the panel is discretized into $n_e = 3921$ laminate plate elements and clamped along all the four edges at bolt locations; the compression load is applied as uniform displacement imposed along the entire upper edge. The presence of the damage is modeled by weakening the stiffness properties of the elements affected by local degradation.

A. Problem setup

We reframe our structural assessment problem as a Sense-Infer-Plan-Act information flow associated with system parameters ($\mathbf{x}$), measured quantities of interest ($\mathbf{q}_m$), and capability quantities of interest ($\mathbf{s}_c$). System parameters define specific damage conditions, and include damage location (y_d and z_d in Figure 3) and damage size/extent (Δy and Δz as in Figure 3, and plies p affected by the damage): $\mathbf{x} = [y_d, z_d, \Delta y, \Delta z, p]$. Measured quantities of interest are the three components of strain that characterize the plane deformation of the external layer: we place our sensors over ply 4 to measure the normal components along the main

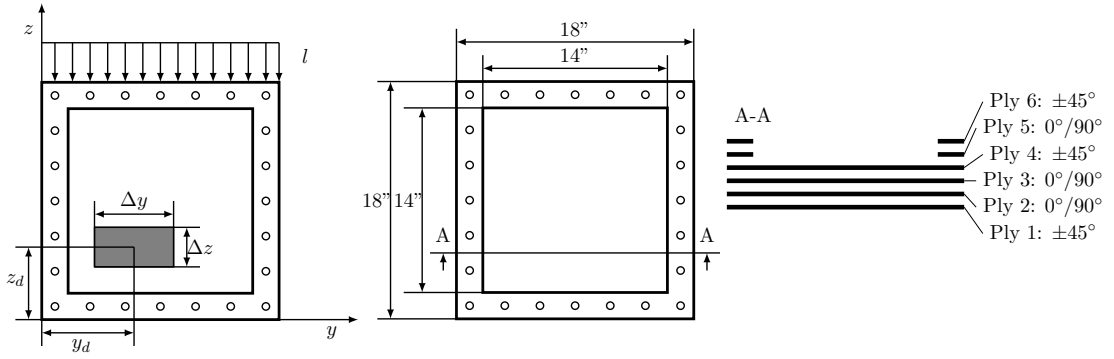


Figure 3. Panel parameters (damage location and damage size) and loading definition. Panel layout and layers sequence.

orthotropic axes of the ply ($\boldsymbol{\varepsilon}_{\mathrm{n}_1}$ and $\boldsymbol{\varepsilon}_{\mathrm{n}_2}$), and the shear component on the ply plane ($\boldsymbol{\varepsilon}_{\mathrm{s}_{12}}$). Accordingly, measurements are three vector-valued quantities reflecting a finite element representation of the strain field: $\mathbf{q}_1(\mathbf{x}) = \boldsymbol{\varepsilon}_{\mathrm{n}_1}(\mathbf{x})$, $\mathbf{q}_2(\mathbf{x}) = \boldsymbol{\varepsilon}_{\mathrm{n}_2}(\mathbf{x})$, and $\mathbf{q}_3(\mathbf{x}) = \boldsymbol{\varepsilon}_{\mathrm{s}_{12}}(\mathbf{x})$. Structural capabilities are condensed into a single vector-valued quantity including element-wise values of maximum failure index. We adopt the definition of failure index *FI* as ratio between the experienced stress and the related maximum allowable value (material strength). Then, for each discretization element, we compute the maximum failure index (**FI**) over panel layers and possible failure modes: $\mathbf{s}_1(\mathbf{x}) = \mathbf{FI}(\mathbf{x}) = \max_{\mathrm{ply,mode}} FI(\mathbf{x}, \mathrm{ply}, \mathrm{mode})$.

Given fixed loading and boundary conditions, both deformation field and failure index depend on the specific damage affecting the structural integrity of the panel. Therefore, in principle, measurements and capabilities can be determined from structural parameters by solving forward problems. However, the original Sense-Infer-Plan-Act flow includes a parameter identification step —the solution of inverse problems to compute structural parameters from strain measurements— followed by a forward problem to estimate failure index from structural parameters. This process may be very expensive and not suitable for time-constrained computations. Therefore, we adopt a MultiStep-ROM strategy to obtain capability (failure index) estimates from measurements (strain components) and support real-time decisions.

The FEM model of the panel is used to simulate the state of the structural component for a variety of damage conditions $\mathbf{x}$. In particular, we collect complete snapshots of measurements and capabilities for two distinct samples of the parameter space selected through Latin hypercube exploration. The first sample constitutes the *evaluation set* of $n_s = 3000$ different damage conditions employed offline to compute models and sensor sites. The second sample is the *validation set* used to test the online phase over $n_v = 500$ different damage conditions. For each damage condition we record $M = 3$ measurements and $C = 1$ capability snapshots in the form of column vectors of $n_e = 3921$ elements. Each vector element is associated with a FEM grid point identified by coordinates $\mathbf{r}_i = (y_i, z_i)$.

The evaluation set is used to compute the POD expansions of measurements and capabilities:

$$\begin{aligned}
\boldsymbol{\varepsilon}_{\mathrm{n}_1} &\approx \bar{\boldsymbol{\varepsilon}}_{\mathrm{n}_1} + \sum_{j=1}^{n_{\mathrm{n}_1}} \alpha_j^{\mathrm{n}_1} \phi_j^{\mathrm{n}_1}; \\
\boldsymbol{\varepsilon}_{\mathrm{n}_2} &\approx \bar{\boldsymbol{\varepsilon}}_{\mathrm{n}_2} + \sum_{j=1}^{n_{\mathrm{n}_2}} \alpha_j^{\mathrm{n}_2} \phi_j^{\mathrm{n}_2}; \qquad\qquad \mathbf{FI} \approx \bar{\mathbf{FI}} + \sum_{j=1}^{\ell} \beta_j \psi_j. \\
\boldsymbol{\varepsilon}_{\mathrm{s}_{12}} &\approx \bar{\boldsymbol{\varepsilon}}_{\mathrm{s}_{12}} + \sum_{j=1}^{n_{\mathrm{s}_{12}}} \alpha_j^{\mathrm{s}_{12}} \phi_j^{\mathrm{s}_{12}};
\end{aligned} \tag{14}$$

These modal terms are employed as described in Section II.A and illustrated in Figure 2. In particular, the POD coefficients are used offline to obtain local models for β_j in the form of linear polynomial functions of $\boldsymbol{\alpha} = [\alpha_1^{\mathrm{n}_1}, \alpha_2^{\mathrm{n}_1}, \alpha_1^{\mathrm{n}_2}, \alpha_2^{\mathrm{n}_2}, \alpha_1^{\mathrm{s}_{12}}, \alpha_2^{\mathrm{s}_{12}}]$. Then, local models $\beta_j(\boldsymbol{\alpha})$ and POD modes are employed online to estimate **FI**. POD modes $\phi_1^{\mathrm{n}_1}$, $\phi_2^{\mathrm{n}_1}$, $\phi_1^{\mathrm{n}_2}$, $\phi_2^{\mathrm{n}_2}$, $\phi_1^{\mathrm{s}_{12}}$, and $\phi_2^{\mathrm{s}_{12}}$ are used to reconstruct coefficients $\boldsymbol{\alpha}$ from incomplete measurements $\hat{\boldsymbol{\varepsilon}}_{\mathrm{n}_1}$, $\hat{\boldsymbol{\varepsilon}}_{\mathrm{n}_2}$, and $\hat{\boldsymbol{\varepsilon}}_{\mathrm{s}_{12}}$ by solving three distinct systems (5), one for each measured

quantity of interest. Reconstructed coefficients $\boldsymbol{\alpha}_{\mathrm{g}}$ are used to identify the most representative set of linear models (6) and compute approximations of capability coefficients $\tilde{\beta}_j(\boldsymbol{\alpha}_{\mathrm{g}})^{(k^*)}$. Finally, POD modes $\{\psi_j\}_{j=1}^{\ell}$ are used to compute the final estimate of capability $\tilde{\mathbf{FI}}$. The online procedure is repeated for all the 500 damage conditions of the validation set.

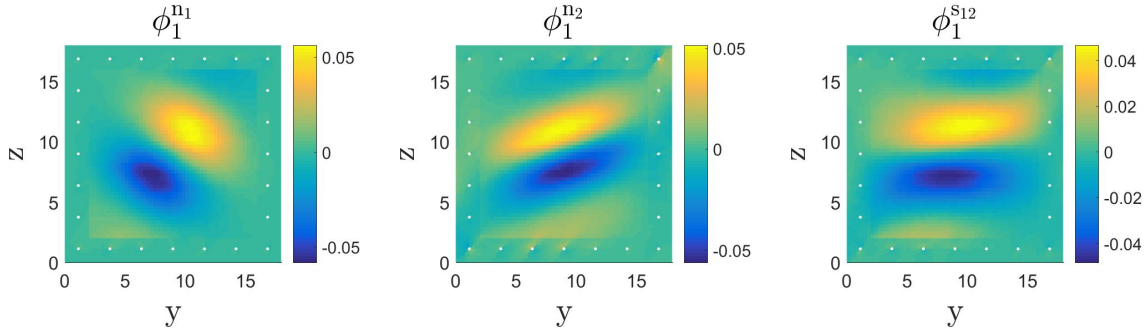
B. Sensor placement

The online measurements $\hat{\boldsymbol{\varepsilon}}_{\mathrm{n}_1}$, $\hat{\boldsymbol{\varepsilon}}_{\mathrm{n}_2}$, and $\hat{\boldsymbol{\varepsilon}}_{\mathrm{s}_{12}}$ are incomplete snapshots of strain components whose non-zero elements are measured at the grid-points indicated by the SSL. The SSL is determined offline through the sensor placement strategy proposed in Section II.B.

Input dataset definition We consider the first dominant mode of each component of strain measured over ply 4: $\phi_1^{\mathrm{n}_1}, \phi_1^{\mathrm{n}_2}$, and $\phi_1^{\mathrm{s}_{12}}$ are represented in Figure 4. All the grid points lay over the same plane, therefore two spatial coordinates are sufficient to identify their physical location. Dataset $\mathbf{T}$ is defined as follows:

$$\mathbf{T} = [\phi_1^{\mathrm{n}_1}, \phi_1^{\mathrm{n}_2}, \phi_1^{\mathrm{s}_{12}}, \mathbf{y}, \mathbf{z}] \tag{15}$$

where $\mathbf{y}$ and $\mathbf{z}$ collect y- and z-coordinates of each grid point, respectively.



(a) First POD mode for $\boldsymbol{\varepsilon}_{\mathrm{n}_1}$ (b) First POD mode for $\boldsymbol{\varepsilon}_{\mathrm{n}_2}$ (c) First POD mode for $\boldsymbol{\varepsilon}_{\mathrm{s}_{12}}$

Figure 4. First dominant mode of each component of strain measured over ply 4. Modes are computed via parametric POD from a set of 3000 different damage conditions affecting panel integrity.

Network training Sensing technologies are usually expensive and we wish to contain costs by limiting the amount of sensors needed to achieve reliable estimates of capabilities. In this paper we assess the efficiency of our placement strategy for a range of sensor availabilities, that is, for different fractions of information f_p measured online with respect to the complete data used offline. At the matter of fact, the neurons of the SOM embody the sensors that we wish to place over ply 4 in the most efficient way. Therefore, the size of SOM network (n_ω) reflects the number of available sensors. During the training phase, the sensors (neurons) are virtually moved over the panel until the most representative locations are reached.

This work considers 20 different SOMs, each providing locations for different amounts of available sensors. Table 1 reports the details about the networks and their corresponding fractions of information $f_p = \frac{n_\omega}{n_e}$. For all the SOM networks we set $\boldsymbol{\rho} = [0.30, 0.30, 0.30, 0.05, 0.05]$; this choice largely privileges the characterization of the modal behavior (0.9 overall) over the spatial distribution of the sensors (0.1). In contrast with that, assigning larger relevance to spatial coordinates would result in a more uniform distribution of sensors over the panel, but would also limit the leading role of the information content in our placement strategy. The role of $\boldsymbol{\rho}$ is to account for physical constraints related to sensor size and accuracy by preventing the SOM to locate the neurons too close to each other.

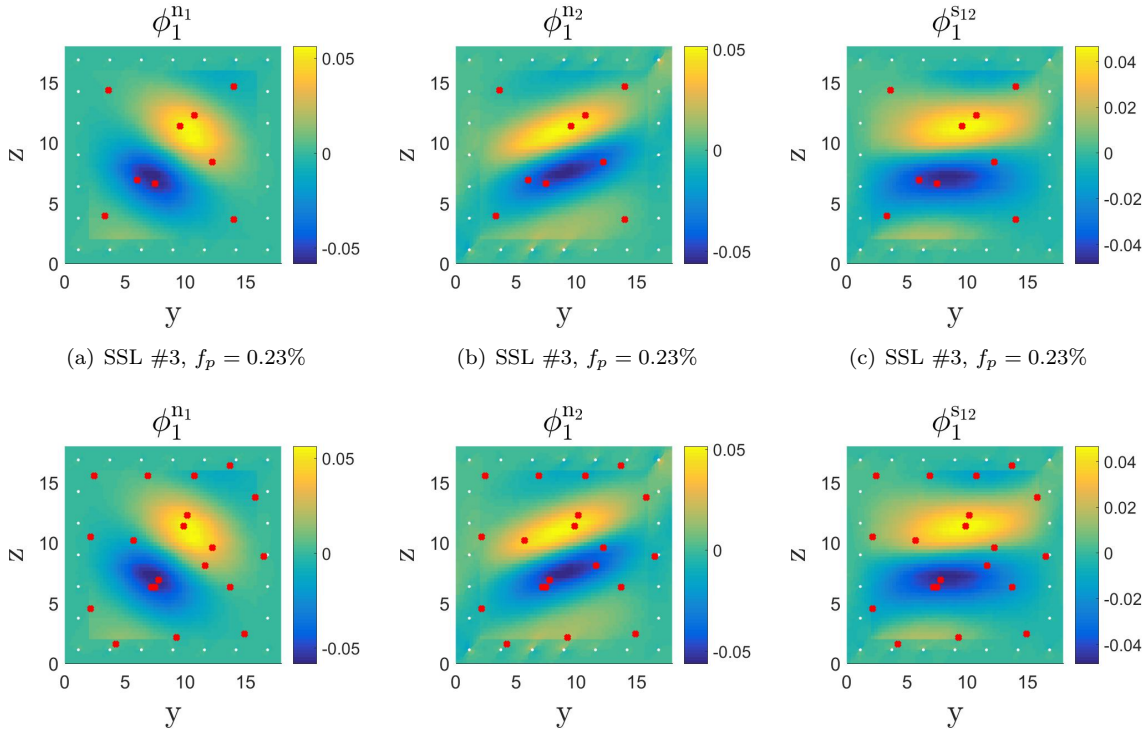
Identification of sensor locations Once training is completed the last 2 elements of the weight vectors indicate the coordinates to place the sensors. Figures 5a-c and 5d-f illustrate SSL #3 and SSL #6, respectively. As reported in Table 1, SSL #3 consists of 9 points and covers only $f_p = 0.23\%$ of the reference data; similarly, SSL #6 consists of 20 points and covers $f_p = 0.51\%$ of the complete dataset. The sensor distributions are illustrated with red dots and superimposed to the first POD modes. A row-wide comparison of the figures shows that sensors are distributed to capture extremes (maximum and minimum) and shape

SSL #	1	2	3	4	5	6	7	8	9	10
n_ω	4	6	9	12	16	20	25	30	36	42
f_p [%]	0.10	0.15	0.23	0.31	0.41	0.51	0.61	0.76	0.91	1.07

SSL #	11	12	13	14	15	16	17	18	19	20
n_ω	49	56	64	72	81	90	100	144	225	400
f_p [%]	1.25	1.43	1.63	1.81	2.07	2.29	2.55	3.65	5.74	10.13

Table 1. Characteristics of SOMs used to identify different Sets of Sensor Locations (SSLs). All the SOMs employed in this paper consists of a 2D grid of n_ω neurons

of all the main modal components at the same time: accordingly, sensor sites are close to each other in the proximity of steep slopes (in the neighborhoods of maximum and minimum).



(a) SSL #3, $f_p = 0.23\%$ (b) SSL #3, $f_p = 0.23\%$ (c) SSL #3, $f_p = 0.23\%$

(d) SSL #6, $f_p = 0.51\%$ (e) SSL #6, $f_p = 0.51\%$ (f) SSL #6, $f_p = 0.51\%$

Figure 5. The Set of Sensor Locaitons (SSLs #3 and #6 of Table 1) are represented with red dots and superimposed to the first POD modes of each measured quantity.

C. Results and discussion

In previous works[3] we extensively investigated different levels of measurements sparsity for the case of random placement of sensors. For the structural application considered in this paper, we found that for measurements with less than 90% of the entire wing panel, online estimates of capabilities are robust to both sensor locations and sensor accuracy. Conversely, if sensors cover less than the 10% of the reference grid-points, measured data may be misleading or insufficient to properly represent the state of the system. These results refer to the circles (◯) in Figures 6; the diagram displays the percentage normalized root mean square error E_c of capability estimate $\tilde{\mathbf{FI}}$ with respect to the reference value $\mathbf{FI}$ obtained with finite element simulations.

$$E_c = \frac{1}{n_v}\sum_{i=1}^{n_v}\epsilon^i \quad \text{with} \quad \epsilon^i = \frac{\|\mathbf{FI}^i - \tilde{\mathbf{FI}}^i\|_2}{(FI_{\max} - FI_{\min})\sqrt{n_e}} \times 100\%. \tag{16}$$

In Eq. 16, E_c is an average value computed over the n_v damage cases of the validation set; $FI_{\max}$ and $FI_{\max}$ are the maximum and minimum values in snapshot $\mathbf{FI}^i$, respectively. For a given fraction of measured data f_p, there are multiple points represented with the same marker and different shades of color: each point refers to a different level of noise introduced to corrupt sensor measurements. Error values in Figure 6 are averaged over 50 random samples of additive Gaussian noise contributions at a given value of standard deviation σ. Specifically, $\sigma = 10\mu\varepsilon$ covers common sensor accuracy of 1-5%, while $\sigma = 100\mu\varepsilon$ corresponds to much less probable noise of the same order of the value to measure.

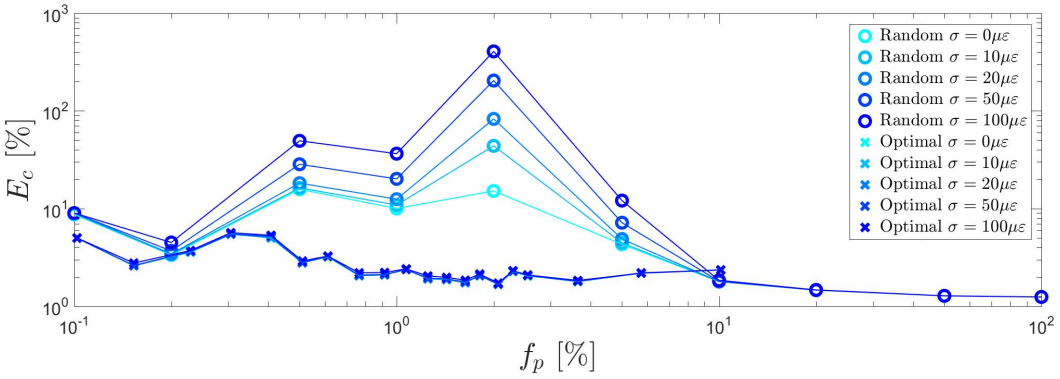


Figure 6. Percentage normalized root mean square error E_c of capability estimates with respect to the reference values obtained with finite element simulations. A comparison between values obtained for sensing sites randomly located over the plate (◯) and values obtained with our particular strategy for sensor placement (×). σ denotes the standard deviation of Gaussian noise added to corrupt the measurements.

A comparison of the two sets of data in Figure 6 (◯-marked values and ×-marked values) reveals two interesting outcomes. First, the estimate error E_c is reduced by adopting our strategy in place of a random selection of sensor sites/locations. In particular, significant improvements emerge for fractions of measurements f_p comprised between 0.5% and 5% of the reference grid-points. In this interval we observe a stable $E_c \approx 2 - 3\%$, that is two orders of magnitude smaller than the approximation error recorded for random selections of similar amounts of sensor sites. Second, for a given fraction of sensed data f_p we observe similar values of estimate error E_c for all the considered levels of noise, as the values obtained for different values of σ nearly superimpose. These results differ significantly from the case of random placement, for which error curves increase with noise. Hence, our placement strategy drastically enhances the robustness of capability estimates to measurement uncertainties.

IV. Concluding remarks

This paper introduced a new methodology to determine optimal sets of sensor locations (SSLs) to support real time decisions. The main objective is the identification of a small number of measurement sites to limit cost and weight of the sensing technologies required onboard. In addition, measured data must be rich in information content to properly support the decision task. Our strategy exploits unsupervised learning techniques to compute the most informative sites to place sensors. Specifically, we adopt Self-Organizing Maps and introduce a training metric accounting for spatial constraints related to size, shape, and accuracy of sensing devices. Finally, the methodology can be integrated with a MultiStep-ROM procedure to efficiently leverage measured data to decisions.

This work proposed and discussed an implementation of the placement strategy oriented to the identification of a unique SSL for all the quantities of interest to measure. The computed SSL is a trade-off among all the measured quantities to capture their informative content at once. The approach is demonstrated for the structural assessment of an unmanned air vehicle wing panel. For this application, the SSLs determined with our strategy are compared with random placements for a variety of sensor availabilities ranging from $f_p \approx 0.10\%$ to $f_p \approx 10\%$; the error associated to online estimates of capabilities illustrates that the SSLs determined with our placement strategy allow for dramatic improvements in terms of accuracy and robustness to uncertainty. In particular, we achieve a significant error reduction (two orders of magnitude) for small fractions of measurements f_p ranging from 0.5% to 5% of the reference grid points. Future developments

include the study of alternative implementations to enhance computational efficiency and enable a dynamic selection of the most informative locations to monitor online.

Acknowledgments

This work was supported by the U.S. Air Force Office of Scientific Research grant FA9550-16-1-0108 under the Dynamic Data Driven Application System Program, Program Manager Dr. Frederica Darema, by the MIT-SUTD Collaboration under the partnership program between Massachusetts Institute of Technology and Singapore University of Technology and Design, and by the MIT-SUTD International Design Center.

References

[1]Mainini, L. and Willcox, K., "Surrogate modeling approach to support real-time structural assessment and decision making," *AIAA Journal*, Vol. 53, No. 6, June 2015, pp. 1612–1626.

[2]Mainini, L. and Willcox, K., "Sensitivity analysis of surrogate-based methodology for real-time structural assessment," *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA SciTech*, Kissimmee, Florida, January 5-9 2015.

[3]Mainini, L. and Willcox, K. E., "Data to decisions: Real-time structural assessment from sparse measurements affected by uncertainty," *Computers & Structures*, Vol. 182, 2017, pp. 296 – 312.

[4]Donoho, D. L., "Compressed sensing," *IEEE Transactions on Information Theory*, Vol. 54, No. 11, 2006, pp. 1289–1306.

[5]Haupt, J. and Nowak, R., "Signal reconstruction from noisy random projections," *IEEE Transactions on Information Theory*, Vol. 52, No. 9, 2006, pp. 4036–4048.

[6]Candès, E. J., Romberg, J. K., and Tao, T., "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, Vol. 59, No. 8, 2006, pp. 1207–1223.

[7]Candés, E. and Romberg, J., "Sparsity and incoherence in compressive sampling," *Inverse Problems*, Vol. 23, No. 3, 2007, pp. 969.

[8]Kashin, B. and Temlyakov, V. N., "A remark on Compressed Sensing," *Mathematical Notes*, Vol. 82, No. 5, 2007, pp. 748–755.

[9]Baraniuk, R. G., "Compressive sensing," *IEEE signal processing magazine*, Vol. 24, No. 4, 2007.

[10]Donoho, D. L. and Tsaig, Y., "Fast solution of ℓ_1-norm minimization problems when the solution may be sparse," *IEEE Transactions on Information Theory*, Vol. 52, No. 4, 2008, pp. 4789–4812.

[11]Yang, A., Ganesh, A., Sastry, S., and Ma, Y., "Fast ℓ_1-minimization algorithms and an application in robust face recognition: A review," *In Proc. IEEE Int. Conf. Image Processing (ICIP), Hong Kong*, 2010.

[12]Baraniuk, R. G., Cevher, V., Duarte, M. F., and Hegde, C., "Model-Based Compressive Sensing," *IEEE Transactions on Information Theory*, Vol. 56, No. 4, 2010, pp. 1982–2001.

[13]Mathelin, L. and Gallivan, K., "A compressed sensing approach for partial differential equations with random input data," *Communications in computational physics*, Vol. 12, No. 04, 2012, pp. 919–954.

[14]Mathelin, L., Pastur, L., and Maître, O. L., "A compressed-sensing approach for closed-loop optimal control of nonlinear systems," *Theoretical and computational fluid dynamics*, Vol. 26, No. 1-4, 2012, pp. 319–337.

[15]Zhang, Y., "Theory of Compressive Sensing via ℓ_1-Minimization: a Non-RIP Analysis and Extensions," *Journal of the Operations Research Society of China*, Vol. 1, No. 1, 2013, pp. 79–105.

[16]cobb, R. G. and Liebst, B. S., "Sensor Placement and Structural Damage Identification from Minimal Sensor Information," *AIAA Journal*, Vol. 35, No. 2, February 1997, pp. 369–374.

[17]Shi, Z., Law, S., and Zhang, L. M., "Optimum sensor placement for structural damage detection," *Journal of Engineering Mechanics*, Vol. 126, No. 11, 2000, pp. 1173–1179.

[18]Worden, K. and Burrows, A., "Optimal sensor placement for fault detection," *Engineering structures*, Vol. 23, No. 8, 2001, pp. 885–901.

[19]Papadimitriou, C., "Optimal sensor placement methodology for parametric identification of structural systems," *Journal of sound and vibration*, Vol. 278, No. 4, 2004, pp. 923–947.

[20]Papadimitriou, C., "Pareto optimal sensor locations for structural identification," *Computer Methods in Applied Mechanics and Engineering*, Vol. 194, No. 12, 2005, pp. 1655–1673.

[21]Fendzi, C., Morel, J., Rebillat, M., Guskov, M., Mechbal, N., and Coffignal, G., "Optimal sensors placement to enhance damage detection in composite plates," *7th European Workshop on Structural Health Monitoring*, 2014, pp. 1–8.

[22]Brunton, B. W., Brunton, S. L., Proctor, J. L., and Kutz, J. N., "Optimal sensor placement and enhanced sparsity for classification," Sumbitted for Publication.

[23]Brunton, S. L., Tu, J. H., Bright, I., and Kutz, J. N., "Compressive Sensing and Low-Rank Libraries for Classification of Bifurcation Regimes in Nonlinear Dynamical Systems," *SIAM Journal on Applied Dynamical Systems*, Vol. 13, No. 4, 2014, pp. 1716–1732.

[24]Kohonen, T., *Self-Organizing Maps*, Springer-Verlag, New York, 3rd ed., 2001.

[25]Erwin, E., Obermayer, K., and Schulten, K., "Self-organizing maps: ordering, convergence properties and energy functions," *Biological Cybernetics*, Vol. 67, 1992, pp. 47–55.

[26]Kohonen, T., "Things you haven't heard about the self-organizing map," *Proc. IEEE Int. Conf on Neural Networks*, San Francisco, 1993, pp. 1147–1156.

[27]Kohonen, T., Oja, E., Simula, O., Visa, A., and Kangas, J., "Engineering applications of the Self-organizing map," *Proceedings of the IEEE*, Vol. 84, No. 10, October 1996, pp. 1358–1384.

[28]Bishop, C. M., *Pattern Recognition and Machine Learning*, Springer, 2006.

[29]Nasrabadi, N. M. and Feng, Y., "Vector quantization of images based upon the Kohonen self-organizing feature maps," *Neural Networks, 1988., IEEE International Conference on*, IEEE, 1988, pp. 101–108.

[30]Kohonen, T. and Somervuo, P., "Self-organizing maps of symbol strings," *Neurocomputing*, Vol. 21, No. 1, 1998, pp. 19–30.

[31]Somervuo, P. and Kohonen, T., "Self-organizing maps and learning vector quantization for feature sequences," *Neural Processing Letters*, Vol. 10, No. 2, 1999, pp. 151–159.

[32]Heskes, T., "Self-organizing maps, vector quantization, and mixture modeling," *IEEE Transactions on Neural Networks*, Vol. 12, No. 6, 2001, pp. 1299–1305.

[33]Lumley, J. L., "The Structure of Inhomogeneous Turbulent Flows," *Atmospheric Turbulence and Radio WavePropagation*, edited by Yaglom and Tatarsky, Macmillian, Moscow and Toulouse, 1967, pp. 166–178.

[34]Algazi, V. R. and Sakrison, D. J., "On the Optimality of the Karhunen-Loève expansion," *IEEE Transactions on Information Theory*, Vol. 15, No. 2, March 1969, pp. 319–321.

[35]Sirovich, L., "Turbulence and the Dynamics of Coherent Structures. Part 1–3," *Quarterly of Applied Mathematics*, Vol. 45, No. 3, October 1987, pp. 561–590.

[36]Holmes, P. J., Lumley, J. L., Berkooz, G., Mattingly, J., and Wittenberg, R. W., "Low-dimensional Models of Coherent Structures in Turbulence," *Physics Reports*, Vol. 287, No. 4, 1997, pp. 337–384.

[37]Dür, A., "On the optimality of the discrete Karhunen–Loève expansion," *SIAM Journal on Control an Optimizaton*, Vol. 36, No. 6, Novemnber 1998, pp. 1937–1939.

[38]Ly, H. V. and Tran, H. T., "Modeling and control of physical processes using proper orthogonal decomposition," *Journal of Mathematical and Computer Modeling*, 1999.

[39]Antoulas, A. C., *Approximation of Large-Scale Dynamicsl Systems*, Advances in Design and Control, SIAM, Philadelphia, PA, 2005.

[40]Rathinam, M. and L.R.Petzold, "A new look at proper orthogonal decomposition," *SIAM Journal of Numerical Analysis*, Vol. 41, No. 5, 2003, pp. 1893–1925.

[41]Schilders, W. H., van deer Vorst, H. A., and Rommes, J., *Model order reduction: Theory, research aspects and applications*, No. 13 in Mathematics in Industry, Springer-Verlag Berlin Heidelberg, 2008.

[42]Benner, P., Gugercin, S., and Willcox, K., "A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems," *SIAM Review*, Vol. 57, No. 4, 2015, pp. 483–531.

[43]Kunisch, K. and Volkwein, S., "Control of the Burgers Equation by a Reduced-Order Approach Using Proper Orthogonal Decomposition," *Journal of Optimization Theory and Applications*, Vol. 102, No. 2, August 1999, pp. 245–371.

[44]Hinze, M. and Volkwein, S., *Proper Orthogonal Decomposition Surrogate Models for Nonlinear Dynamical Systems: Error Estimates and Suboptimal Control*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 261–306.

[45]Everson, R. and Sirovich, L., "The Karhunen-Loeve Procedure for Gappy Data," *Journal of Optical Society of America*, Vol. 12, No. 8, 1995, pp. 1657–1664.

[46]Bui-Thanh, T., Damodaran, M., and Willcox, K., "Aerodynamic Data Reconstruction and Inverse Design Using Proper Orthogonal Decomposition," *AIAA Journal*, Vol. 42, No. 8, August 2004, pp. 1505–1516.

[47]Wang, M., Dutta, D., Kim, K., and Brigham, J. C., "A computationally efficient approach for inverse material characterization combining Gappy POD with direct inversion," *Computer Methods in Applied Mechanics and Engineering*, Vol. 286, 2015, pp. 373–393.

[48]Wang, M. and Brigham, J. C., "A Generalized Computationally Efficient Inverse Characterization Approach Combining Direct Inversion Solution Initialization with Gradient-based Optimization," *Comput. Mech.*, Vol. 59, No. 3, March 2017, pp. 507–521.

[49]Hastie, T., Tibshirani, R., and Friedman, J., *The elements of statistical learning: data mining, inference and prediction*, Springer Series in Statistics, Springer, 2nd ed., 2009.