HAL
open science

A Probabilistic Multi-class Classifier for Structural Health Monitoring

Nazih Mechbal, Juan Sebastian Uribe, Marc Rebillat

To cite this version:

Nazih Mechbal, Juan Sebastian Uribe, Marc Rebillat. A Probabilistic Multi-class Classifier for Structural Health Monitoring. Mechanical Systems and Signal Processing, 2015, 60-61, pp.106-123. 10.1016/j.ymssp.2015.01.017 . hal-01111025

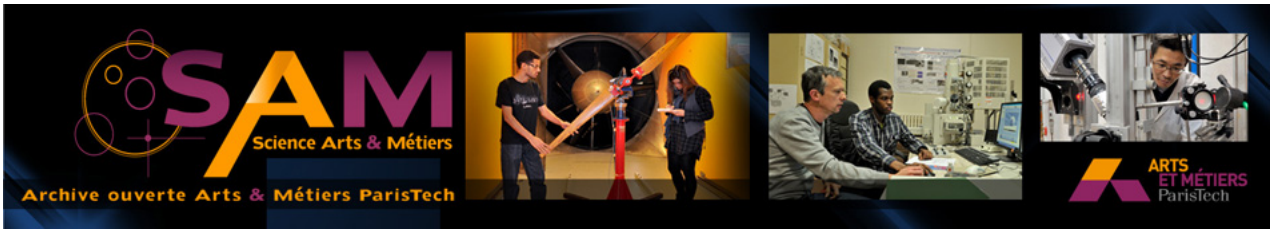
HAL Id: hal-01111025

https://hal.science/hal-01111025

Submitted on 2 Feb 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Science Arts & Métiers (SAM)

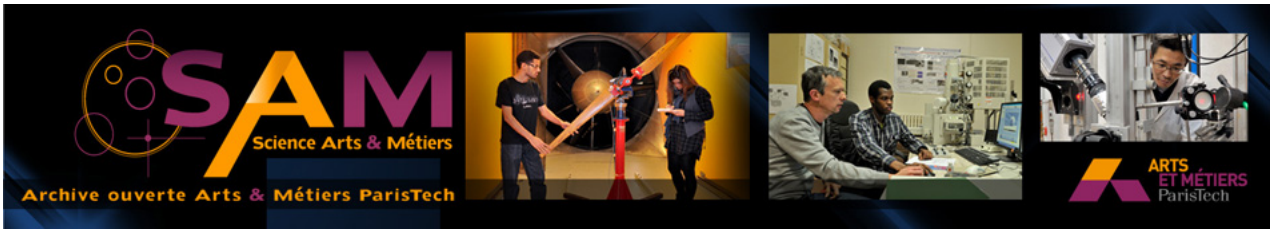
is an open access repository that collects the work of Arts et Métiers ParisTech researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: http://sam.ensam.eu
Handle ID: .http://hdl.handle.net/10985/9287

To cite this version :

Nazih MECHBAL, Juan Sebastian URIBE, Marc REBILLAT - A Probabilistic Multi-class Classifier for Structural Health Monitoring - Mechanical Systems and Signal Processing - Vol. --, n°--, p.0-25 - 2015

Any correspondence concerning this service should be sent to the repository Administrator : archiveouverte@ensam.eu



Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers ParisTech researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: http://sam.ensam.eu
Handle ID: .http://hdl.handle.net/null

To cite this version :

Nazih MECHBAL, Juan Sebastian URIBE, Marc REBILLAT - A Probabilistic Multi-class Classifier for Structural Health Monitoring - Mechanical Systems and Signal Processing - Vol. --, n°--, p.0-25 - 2015

Any correspondence concerning this service should be sent to the repository Administrator : archiveouverte@ensam.eu

A Probabilistic Multi-class Classifier for Structural Health Monitoring

Nazih Mechbal [*1], Juan Sebastian Uribe [2] and Marc Rébillat [3]

[1,2,3] *Processes and Engineering in Mechanics and Materials Laboratory*
PIMM, UMR CNRS 8006, Arts et Métiers ParisTech (ENSAM)
151, Boulevard de l'Hôpital 75013, Paris, France

Abstract. In this paper, a probabilistic multi-class pattern recognition algorithm is developed for damage detection, localization, and quantification in smart mechanical structures. As these structures can face damages of different severities located at various positions, multi-class classifiers are naturally needed in that context. Furthermore, because of the lack of available data in the damaged state and of environmental effects, the experimentally obtained damage sensitive features may differ from those learned offline by the classifier. A multiclass classifier that provides probabilities associated with each damage severity and location instead of a binary decision is thus greatly desirable in that context. To tackle this issue, we propose an original support vector machine (SVM) multi-class clustering algorithm that is based on a probabilistic decision tree (PDT) and that produces a posteriori probabilities associated with damage existence, location, and severity. Furthermore, the PDT is here built by iteratively subdividing the surface of the structure and thus takes into account the actual structure geometry. The proposed algorithm is very appealing as it combines both the computational efficiency of tree architectures and the classification accuracy of SVMs. The effectiveness of this algorithm is illustrated experimentally on a composite plate instrumented with piezoelectric elements on which damages are simulated using added masses. Damage sensitive features are computed using an active approach based on the permanent emission of non-resonant Lamb waves into the structure and on the recognition of amplitude disturbed diffraction patterns. On the basis of these damage-sensitive features, the proposed multi-class probabilistic classifier generates decisions that are in excellent agreement with the actual severities and locations of the simulated damages.

Keywords: Structural health monitoring; Support vector machine; Probabilistic decision tree; Pattern recognition; Wave diffraction patterns; Piezoelectric elements.

[*1]Corresponding author, E-mail: nazih.mechbal@ensam.eu
[2] E-mail: juan-sebastian.uribe-echeverri@ensam.eu
[3] E-mail: marc.rebillat@ensam.eu

1 Introduction

Structural health monitoring (SHM) is an emerging technology designed to automate the inspection process undertaken to assess the health condition of structures. In a smart structure, SHM systems may automatically process data, state regarding structural conditions, and highlight any need for human intervention [1]. The output of any SHM process should provide information regarding the type, the location, and the severity of the damage. Thereby, the SHM process is classically decomposed into four sequential steps: detection, localization, quantification, and prognosis. SHM involves many disciplinary fields (material science, structural analysis, signal processing, data mining, fracture mechanics, fatigue life analysis…) and has been the topic of extensive research efforts over the last thirty years. This technology is now progressing toward operational service and recent surveys have shown that even reluctant industry areas are now convinced that SHM is the key technology that will enable the transition from schedule-driven maintenance to condition-based maintenance [2]. Rresearchers are now exploring ways to adapt information technologies in order to reduce system monitoring costs while simultaneously broadening their functional capabilities. In this paper, we address the detection, localization and quantification steps of the SHM process.

1.1 Damage-sensitive features generated using Amplitude Disturbed Diffraction Pattern

SHM system operational results can be classified into four different sequential levels: detection, localization, quantification, and prognosis [1]. Several different techniques that depend on the structure's material, on the technology used for acting and sensing, on the position, size, and nature of damage may be employed to perform SHM [3]. They could be sorted in two main categories: global or local.

Among others global SHM methods, we can highlight vibration based approaches [4, 5]. These methods are categorized based on the type and nature of measured data. Significant examples include: model updating [6]; statistical time series and modal analysis [4]. In those approaches we seek to track changes in global structural parameters (mass, stiffness, flexibility, damping) and modal parameters (modal frequencies, associated damping values and mode shapes) by analyzing changes induced in the global dynamic behavior of a structure. Experimental identification of these dynamic properties indeed gives insight on the structural damage conditions, see [7, 8, 9] and references therein. Multivariate techniques have been also used in global vibration-based SHM: examples are the PCA (Principal Components Analysis) [10] and ICA [11]. In the case of nonlinear damage, specific vibration-based approaches have also been developed recently by the authors using cascade Hammerstein models [12]. An extensive overview of nonlinear global vibration-based SHM methods can be found in [13].

For local inspection, we can employ electro-mechanical impedance or displacement/strain as features indicating the presence of damage [3]. As these methods are local, their sensibility is strongly linked to the position of the sensors. In that category, we can also highlight acoustic emission or high frequency wave-based approaches that have the advantage to be sensitive to very small damages and offer the capability of propagation over a significant distance [14]. Evaluation of wave propagation on solids is one of the most successful techniques for damage monitoring [15]. One piezoelectric actuator emits periodic burst pulses exciting Lamb waves in the structure under inspection, and a set of sensors records signals, representing the respective structural responses. These signals are processed in order to extract damage related information such as

location, size, orientation, type among others [14]. One of the outstanding advantages of using Lamb waves for SHM is that such waves can travel over relatively long distance and can be used to monitor various types of damage (as delamination, disbonds, fiber breaking, impact,...) [16].

To construct measurements that are sensitive to the damage, we use in this work an active SHM approach based upon a correlation technique that relies on wave diffraction patterns recognition. Damage-sensitive features are generated thanks to the Amplitude Disturbed Diffraction Pattern (ADDP) phenomenon observed using permanent emission of selected non-resonant Lamb waves in the structure. ADDP assesses the disturbances that damage brings to the acoustic wave propagation in the structure. These disturbances depend upon the damage severity and position as well as on the frequency of the exciting signal. With an appropriate calibration procedure, it is thus possible to detect, localize, and quantify the damage using ADDP. This process has already been successfully used as a multi-touch sensing approach to tactile sensing [17]. The formulation of this process as a damage location one has been presented in [18]. In this paper, the ADDP process will be used to generate the damage-sensitive features that will allow detecting, localizing, and quantifying the damage.

1.2 The need for a probabilistic multiclass classification tool in SHM

The extraction of damage-sensitive features from measurements is a process that is most powerful when it is followed by pattern recognition [19]. Indeed, in a statistical pattern recognition paradigm for SHM, it is usually rather difficult to obtain data from damaged structures because of cost and of practical constraints. However, when such data are available, a whole new range of algorithms can be used [19] and the problem of damage detection, localization, and quantification can be cast as one of classification [20, 21] .

As the structures under study can face damages of different severities located at various positions, multi-class classifiers are naturally needed. Furthermore, because of the lack of available data in the damaged state and of the environmental effects, the experimentally obtained damage-sensitive features may differ from those learned offline by the multi-class classifier. A multiclass classifier that provides probabilities associated with damage existence and with each damage location and severity instead of a binary decision is thus greatly desirable for SHM. With such a classifier, damage locations can for example be ranked from the most probable to the least one, thus providing helpful guidance for the inspection task. Indeed, the reliability assessment of SHM systems is the key issue in ensuring their successful implementation. For example in aerospace maintenance procedure minimizing structural teardown to access regions to be monitored is one of the claimed benefits in using SHM rather than classical nondestructive evaluation (NDE) systems. Moreover, given that the diagnosis (class assignment) will be made on the basis of measured data, it is important that the chosen pattern recognition algorithm is able to accommodate a degree of imprecision commensurate with expected levels of measurement error and noise. This requires that the damage detection, localization and quantification SHM outputs' have to be assorted with a probability. This probability is then considered as a level of the trust in the SHM systems results.

In this work, where a statistical pattern recognition paradigm for SHM is considered, we propose to provide a metric helping the condition-based maintenance decisions for in-service structures. This is possible by capturing during the training phase a part of the effect of environmental variability. This will represent the probability of how likely are SHM system outputs. Moreover, to ease the practical implementation, by simply thresholding these *a posteriori* probabilities at each node, the soft-decision approach can be converted to the conventional hard-decision approach.

1.3 Towards probabilistic binary decision trees

Formally, we seek for an algorithm able to solve a multi-class data classification problem in a manner that produces confidence probabilities associated with each class. This algorithm should also ideally take into account some *a-priori* knowledge related to the structure (geometry, material…) in order to speed up the diagnostic. Moreover, in this work, we propose to use pre-defined class hierarchy. That means that the internal (non-leaf) nodes of the structured tree are chosen based on data where we built the hierarchy during training and then classify new samples by using the hierarchical tree.

Generally speaking, there exist two main types of classifiers: hard and soft [22]. Hard classifiers, such as support vector machine (SVM) [23] , build a frontier between classes. They only label new unknown points with the class associated to the side of the frontier in which they fall, without giving any idea of the certitude of the decision or the degree of membership to that class. These classifiers are very appealing, because in general they tend to give very accurate predictions. The formulation of bi-class SVMs finds the hyperplane that separates the two classes by ensuring the largest margin with respect to each of the class. Extending this formulation directly to more than two classes is generally avoided due to the complex optimization equation that it leads to. Instead, the multi-class SVM problem is often dealt with by using an ensemble of two-class SVMs. Various strategies exists to achieve such combinations, the most popular ones being *"one-against-one"* [24, 25], *"one-against-all"* [23, 26] and *Error-correcting output codes* (ECOC) [27]. A less known approach consists in dividing the problem in a hierarchical way where classes which are more similar to one another are grouped together into mata-classes, resulting in a Binary Hierarchical Classifier [28, 29]. Regarding hierarchical classification [30] where classes are organized in tree or a DAG-like structures (Directed Acrylic Graph), the damage localization classification problem can be naturally cast as a hierarchical classification problem where the classes to be predicted are organized into meta-classes (for example subdividing geometric zones to be monitored) hierarchy as a tree.

Considering a local classifier per node approach [30], Madzarov [31] has come up with a simple and intuitive method based on building a binary SVM decision tree. Actually, a class binarization is the mapping of a multiclass problem into several two-class problems in a way that allows the derivation of a prediction for the multiclass problem from the predictions of the two-class classifiers. By selecting specific features, such as the distance between gravity centers of the different classes, an automatic graph is generated where at each node a bi-class SVM is trained [31].. However, such multi-class hard classifiers only provide one predicted class without any associated score indicating the confidence of the classification. On the opposite, soft classifiers like Logistic Regression (LR) [32] are able to build probability estimations for the belonging of an incoming example to all the classes, and then with this information they choose the most likely class. However, their classification performances are in general poorer than the hard classifiers ones. We are thus interested, in a multi-class context, by the probabilities estimation offered by soft classifiers while keeping the hard classifiers proved performances [33]. Standard SVMs do not provide such probabilities, but solutions to this issue were proposed. In a bi-class context, Platt [28] proposed a method to compute the probabilities $p(class|input)$ from SVM outputs. The approach consists in training the parameters of a sigmoid function to map the SVM outputs into probabilities. The underlying idea of this so-called Probabilistic SVM classifier (PSVM) is that when the distance from an example to the frontier is large, the example will very likely belong to that class. However all these works were developed for the bi-class problem only. By adapting

Platt's method to a multi-class context by means of a binary decision tree, it is thus in principle possible to extract probabilities from SVMs outputs while keeping the demonstrated performances of hard classifiers.

This approach has been developed previously by the authors and successfully applied to automotive repairing backdrop [34]. Using damage-sensitive features generated through ADDP, this approach is extended here to detect, localize and quantify damages. Moreover, by assuming that the structure has an Euclidean geometry and topology (which is not restrictive as the structure can be subdivided in areas that satisfied this property), we propose to construct the decision tree by iteratively partitioning the structure along affine hyperplanes. By doing so, each leaf of the tree, represent, a geometric zone of the structure. This leads to a new algorithm: SVM geometric probabilistic decision tree (SVM-GPDT), where a multi-class probabilistic classifier is then built on the basis of this decision tree. Therefore, using this topological a priori knowledge, the SVM-GPDT will provide fast classification (logarithmic complexity) along with associated posterior probabilities $p(class|input)$. In a SHM context, this approach is very appealing because due to restrictions upon actually damaging structures, few data are usually available to train the classifier, and few data are needed to train this one. Furthermore, it is sufficient in several applications to only locate the most likely damaged area. For example, in aeronautical structures, it is satisfactory to give the damaged panel and an estimate of damage severity and then to use precise nondestructive testing and evaluation procedures. This algorithm thus offers an original and practical way to generate probability information related to the presence and severity of damage in each structural zone to be monitored.

The paper is organized as follows: Section 2 describes the original SVM based multi-class classification approach proposed here, including the geometric approach used to generate the damage decision tree. Section 3 validates the proposed approach for damage detection, localization, and quantification experimentally. Concluding remarks and future perspectives are drawn in the last sections.

2 A probabilistic multi-class classification algorithm

2.1 Optimized soft margins Support Vector Machines with Gaussian kernels

Classic support vector machines (SVMs) have proved to be very effective classification methods [35]. They are binary linear classification techniques which search for the hyperplane (in the hyperspace of attributes) that separates two classes in a training set. This hyperplane is found by maximizing the so-called margin, which is the distance from the hyperplane to the closest points, denoted support vectors. A common variant of classic SVM, is called *soft margin* and it consists of admitting some misclassified points in the training set to prevent the over fitting problem. However, we want to avoid too many points being misclassified. Thus we impose a penalty C that will penalize every misclassified example. C can take values in the range $0 < C \leq \infty$. A high value of C, means a strict classifier that doesn't admit many misclassified points. On the opposite, a small C means a very flexible classifier.

Formally, we have a training set $\{(\boldsymbol{x_1}, \boldsymbol{y_1}), \dots, (\boldsymbol{x_n}, \boldsymbol{y_n})\}$, where every point $\boldsymbol{x_i} = (\boldsymbol{x_{i1}}, \dots, \boldsymbol{x_{im}})$ has m attributes and one of the two possible labels $\boldsymbol{y_i} = \{-1, 1\}$. A soft margin SVM classifier will label a new unknown point $\boldsymbol{x_t}$ according to the decision function:

$$y(\mathbf{x}_t) = sgn((\boldsymbol{w}.\boldsymbol{x}_t) + w_0) \tag{1}$$

where $\boldsymbol{w}$ and w_0 are the hyperplane parameters to find on the basis of a training set.

In SVM, kernels can be used to project the data into a virtual space where it might be easier to separate them [23]. The main advantage of kernel functions is that the only operation needed to be defined in the new virtual space is the inner product $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$. Applying a kernel function together with the soft margin, a soft margin SVM classifier will label a new unknown point $\boldsymbol{x_t}$ according to the new decision function [36]:

$$y(\mathbf{x}_t) = sgn(f(\boldsymbol{x_t})) \tag{2}$$

where $f(\boldsymbol{x})$ is defined as:

$$f(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i u_i \kappa(\boldsymbol{x}_i, \boldsymbol{x}) + w_0 \tag{3}$$

The values of α_i and u_i are found by solving the following constrained optimization problem on the basis of the training examples:

$$\begin{cases} \max\limits_{\alpha_i} \left[\sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j u_i u_j \kappa(\boldsymbol{x_i}, \boldsymbol{x_j}) \right] \\ 0 \le \alpha_i \le C, \;\; i = 1, \dots, m \\ \sum_{i=1}^{m} \alpha_i u_i = 0 \end{cases} \tag{4}$$

In the case of Gaussian kernel the only tuning parameter is σ^2. A small value of σ^2 will lead to curved hyperplanes and a high value will constrain the hyperplanes to be straighter. In [37] it is shown that for some combinations of the hyperparameters (C, σ^2), the Gaussian kernel tends towards the linear kernel, which makes the Gaussian kernel the most general method and one that would work for a large range of datasets. The hyper parameters (C, σ^2) have to be optimized for every classification problem. In this work we used Lin's approach to optimize the choice of the coefficients (C, σ^2) [37].

2.2 Probabilistic Support Vector Machines (PSVM)

As stated before, SVM produce a value that is not a probability. Indeed, SVM only give a class prediction output that will be either $+1$ or -1. In order to extract the associated probabilities from SVM outputs, several approaches have been proposed [23, 28]. We will focus on Platt's approach [28] who proposed a technique that has been largely used in the literature. He builds a sigmoid function between the outputs $f(\boldsymbol{x})$ of the SVM and the probability of membership $p(y = i|\boldsymbol{x})$ to a class i, given the attributes of $\boldsymbol{x}$. A simple bi-class example is shown in Fig. 1. The sigmoid will have the following parametric expression:

$$p\,(y\ = 1|f(\boldsymbol{x})) = \frac{1}{1 + e^{af(\boldsymbol{x})+b}} \tag{5}$$

where a and b are parameters computed from the minimization of the negative log-likelihood function on a set of training examples [28],

$$-\sum_i t_i\ \log p(f(\boldsymbol{x_i}))\ + (1 - t_i\ )\log(1 - p(f(\boldsymbol{x_i})) \tag{6}$$

and t_i is the new label of the classes: $+1$ becomes t_+ and -1 becomes t_-.

This relabeling procedure is conducted so that the sigmoid fitting is softer. These new labels are computed using:

$$t_+ = \frac{N_+ + 1}{N_+ + 2} \qquad t_- = \frac{1}{N_- + 2} \tag{7}$$

where N_+ and N_- are the number of points that belong to class 1 and class 2 respectively.

The PSVM as proposed by Platt [28], uses first a SVM classifier that has to be trained with a training set s_1. Then the sigmoid parameters (a, b) have to be found. To do so, it is recommended [28] to use a second training set s_2. The sigmoid fit requires the inputs $f(\boldsymbol{x})$ and the labels t_i. We use s_2 as a test set for the classifier. Thus we obtain the output values $f(s_2)$. Knowing the labels of s_2 (number of positives ones and number of negatives ones) we can calculate t_+ and t_- for this sub-training set. With values $(f(s_2), t_i(s_2))$ we then find (a, b) by optimizing Eq. (12) using a Newton's method with backtracking. After these two steps, we have built a posterior probability estimator for the bi-class problem. Regarding the practical implementation of the algorithm, we have improved Platt's method [28] by introducing modifications based on ideas sketched by Lin and coauthors in [38]. The new algorithm is more precise and more efficient than Platt's one in terms of computational cost.

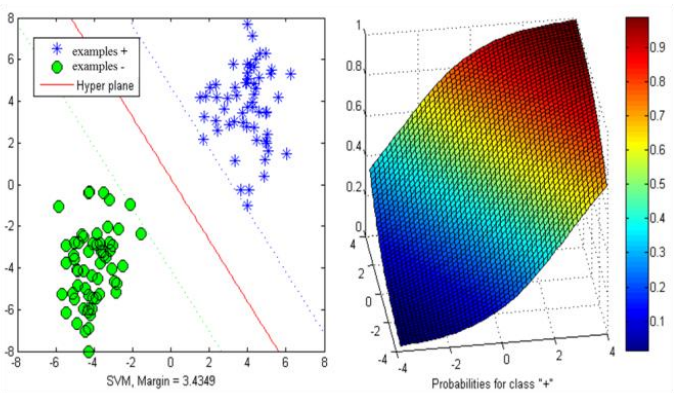


Fig. 1: Example of SVM classification and probability estimation for a 2D binary problem

2.3 From bi-class to multi-class problems

SVM were originally designed for bi-class classification problems and the passage to multi-class problems is still an on-going research area [39, 40, 41, 42]. There are two major approaches to solve this type of problems. The first and intuitive one named "All-together", consists in formulating a cost function with Q (the number of classes in our problem) hyperplanes [43, 44]:

$$\begin{cases} \min_{\boldsymbol{w},\xi} \frac{1}{2}\sum_{m=1}^{Q}(\boldsymbol{w_m}.\boldsymbol{w_m}) + C\sum_{i=1}^{l}\sum_{m\neq y_i}\xi_i^m \\ \left(\boldsymbol{w_{y_i}}.\boldsymbol{x_i}\right) + b_{y_i} \geq (\boldsymbol{w_m}.\boldsymbol{x_i}) + b_m + 2 - \xi_i^m \\ \xi_i^m \geq 0, \quad i = 1,\dots,l \quad m \in \{1,\dots,Q\} \setminus y_i \end{cases} \tag{8}$$

This optimization problem, with so many variables, is in general difficult to solve [39, 44]. Therefore, an alternate approach consists in dividing the multi-class problem in several binary sub-problems. There are numerous methods that perform this subdivision. The most popular are *One against one* [24, 25, 45], *One against all* [23, 26], and *Error-correcting output codes* (ECOC) [27, 46].

Yet another approach for tackling the multiclass classification problem utilizes a hierarchical division of the output space *i.e.* the classes are arranged into a hierarchical structure: *Directed Acyclic Graph* (DAG) [33], or *Binary decision Trees* (BDT) [31]. In the tree structure, The tree is created such that the classes at each parent node are divided into a number of clusters, one for each child node. The process continues until the leaf nodes contain only a single class. For sake of shortness, we will only focus on the BDT method, because it is the one we will use in the sequel. It is important to note that the BDT processing time is smaller than the other methods because the depth of the decision tree is of order $\log_2(Q)$ since at every level the tree eliminates approximately half of the remaining classes.

In [31], it is proposed to build a binary decision tree in which at every node the remaining classes are separated in two subgroups g_1 and g_2. A SVM classifier decides to which subgroup the new point belongs to, and hence in which direction to move. In order to build the tree (the first step in the classification procedure) a clustering algorithm divides all the Q classes into sub-groups g_1 and g_2. The algorithm calculates the gravity centers of all classes. The two classes with the biggest Euclidean distance from each other are assigned to each of the sub-groups. Then the algorithm checks the closest class to one of the sub-groups, this class is assigned to that sub-group, their gravity center is recalculated with the new points that have just been added. This is repeated until all classes have been assigned to one of the groups. For each sub-group the clustering algorithm is repeated until there is no more than one class per sub-group.

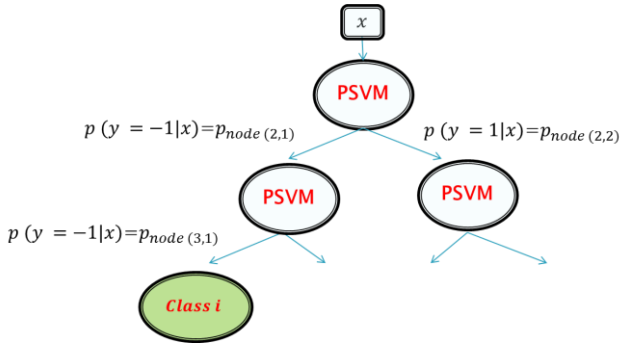


Fig. 2: Example of a probability decision tree

2.4 Probabilistic decision trees (PDT)

In this paper, we propose to use a binary decision tree following the idea introduced by Madzarov [31] presented in the previous section. But instead of using a simple SVM classifier at each node, we use a SVM classifier associated with a sigmoid function to estimate the probability of membership to each sub-group in the node, as shown in Fig. 2.

We can then build a probability function for each leaf, knowing the path that we have to follow to reach it. Note that there is only one way to get to a leaf. It follows that the probability functions are unique for a trained tree:

$$p(y = i|x) = \prod_{h=1}^{leaf} p_{node\,(h,l)} \tag{9}$$

The parameter h represents the level of the tree ($h = 1$ corresponds to the root node). The previous expression states that the probability of membership of an element in the class i is calculated as the product of the probabilities of the decisions taken in all the nodes visited until reaching the leaf. By $node(h, l)$ we mean the lth node in the hth level. For a given tree, we know the probability functions of the Q classes. For future unknown cases, we just have to evaluate the Q analytical functions and then choose the class with the highest score. However, one of the most interesting issues of the PDT is that we can have more than one prediction for one example. We can have a list of all the possible classes, ordered following their plausibility, which is measured with the posterior probabilities estimation.

2.5 A geometry based probabilistic decision trees (GPDT)

The probabilistic decision tree, described previously, can be built using many different criteria [31]. For example using the Euclidian distance between the gravity centers, the margin obtained by pairwise SVM, or some physical or functionalities criteria (mechanical, geometrical, etc.). This degree of freedom could be very helpful, because in this way we can introduce knowledge on the system that might help the classification task. In the case of damage location and due to restrictions upon actually damaging structures, it is sufficient in several applications only to locate the most likely damaged area instead of having the exact damage location. We thus propose here to build the probabilistic decision tree by partitioning the structure in subparts and to generate decisions associated with each of these geometrical subparts.

Let's consider that the structure has a geometrical topology that can be partitioned iteratively along affine hyperplanes. Then, one can constrain the decision tree to generate decision boundaries following these specific axes. The method that we propose is based on a recursive division, at each node of the decision tree, of the upstream classes in two disjoint groups with respect to their spatial location. The structure will thus be subdivided into small geometric zones to be monitored and the groups are simply determined according to their zone membership with respect to the symmetry axis. This geometric tree will furthermore facilitate the training step by giving an automatic and systematic procedure to perform it. In summary, the decision tree is deduced here from the specifications of the geometric boundaries of the structures and each of its leaves will represent a subpart of the structure.

2.6 Illustration of the SVM-GPDT algorithm

The whole SVM-GPDT algorithm is illustrated within a simple example. Let's consider a rectangular structure that is divided in 8 subzones to be monitored (Fig. 3(a)). In addition to the localization step, we also add a detection step but no quantification step in this geometric based probabilistic decision tree. Suppose that a database containing examples for each zone in the damaged and undamaged states has been constructed. 9 classes are considered: the 8 zones in presence of a damage (classes 1 to 8) and the undamaged class with healthy examples of all the zones (class 0). The first step of the tree is the detection step, where the classifier is trained to generate a decision with respect to the presence or not of damage. If a damage is detected, then decisions are taken recursively by following the symmetry axes of the structures to localize it. Upstream classes are divided in two groups by clustering symmetrical zones as shown in Fig. 3(a). The logical and geometric depiction of this tree is illustrated in Fig. 3(b). As we are dealing with PSVM classifier, the posterior probability of a damage to belong to each zone is also available at the end of the process (see Fig. 3(e) and Fig. 3(d)).

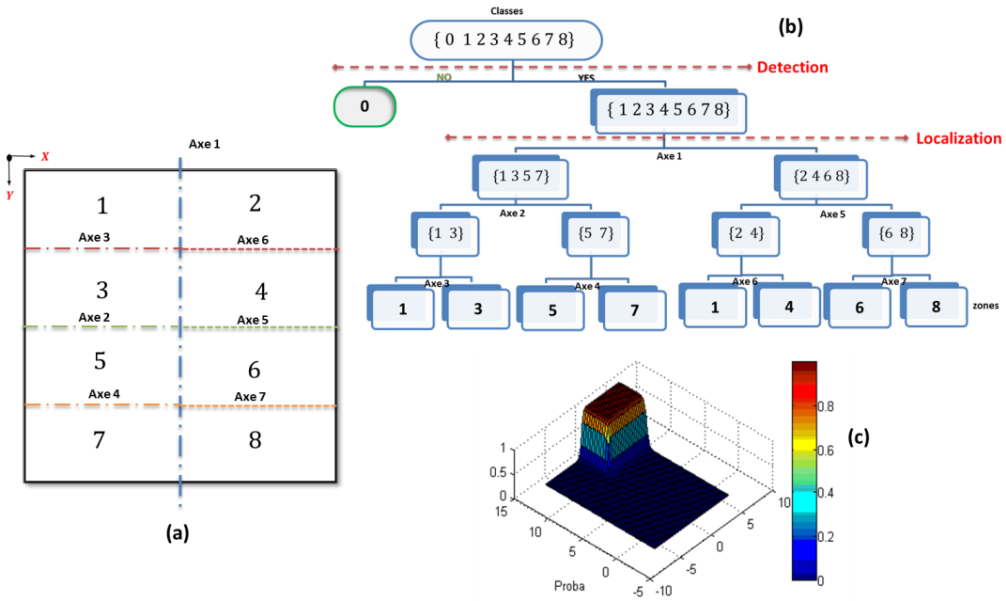


Fig. 3: SVM-GPDT example: $(\boldsymbol{a})$ rectangular structure with 8 zones and their associated axes of symmetry. $(\boldsymbol{b})$ GPDT construction. $(\boldsymbol{c})$ spatial damage probability distribution for zone 2.

3 Damage detection, localization, and quantification on an active composite plate

3.1 Experimental setup

The proposed damage detection, localization, and quantification method is applied to a free-free composite plate ($300 \times 400 \times 2.4\,mm$) with three circular piezoelectric buzzers of diameter $20\,mm$ bonded on its surface (Fig. 4.). These active elements can work either as actuators or sensors [47]. The structure is made up of 8 layers carbon epoxy material all oriented along the length of the plate (*i.e.* at 0°). The mechanical properties of the composite material are given in **Table 1**.

Property	$E_{11} = E_{22}$ (GPa)	E_{33} (GPa)	$G_{12} = G_{13} = G_{23}$ (GPa)	ν_{21}	ρ (Kg/m^3)
Value	65.7	8.1	5.2	0.02	1554

Table 1: Mechanical properties of the composite material

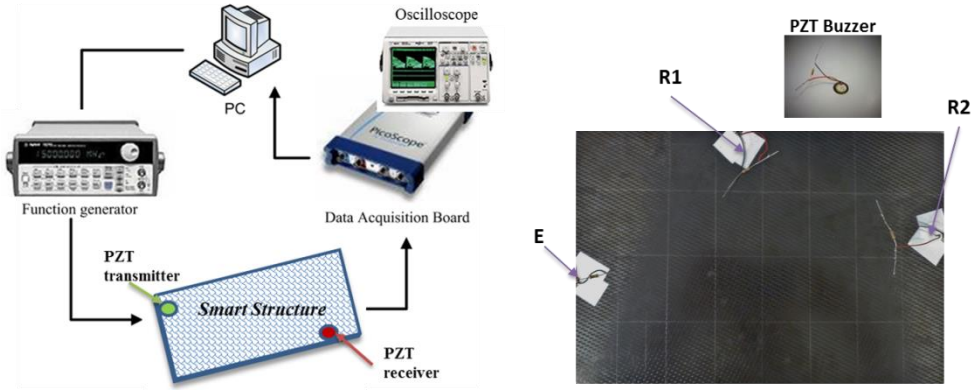


Fig. 4: The smart structure and the test bench.

The experimental setup and the actual smart plate are presented in Fig. 4. The piezoelectric elements are arbitrarily positioned (Fig. 4) but care is taken to avoid putting them on the axis of symmetry, which may introduce confusions in the localization process. For the present application, only one emitter is used.

As we cannot impose real damage in a reversible manner, we have experimentally simulated its effect. In this work, local masses were added on predefined area of the structure as a mean of simulating damage. The justification behind that approach was that local stiffness reductions will reduce the natural frequencies and an alternative way of reducing the frequencies might be to locally increase mass. Therefore, to simulate damage at different locations and with different severities S_1 and S_2, the plate was divided into damaged zones (Fig. 5) and two calibrated masses (S_1: $m_1 = 20g$ and S_2: $m_2 = 10g$) have been positioned on each zone. As each damage scenario is reversible, this also allows monitoring the repeatability of the measurements. The number of damaged zones has been first fixed to 16 zones. Then to test the spatial selectivity of the approach we have increased the number of monitored zones to 32 (Fig. 5).

3.2 Features extraction

To extract damage-sensitive features, we use an active SHM approach based on a correlation technique that relies on wave diffraction pattern recognition. This approach, developed by the authors for tactile sensing purposes [48] is diverted here to be used to generate damage-sensitive features.

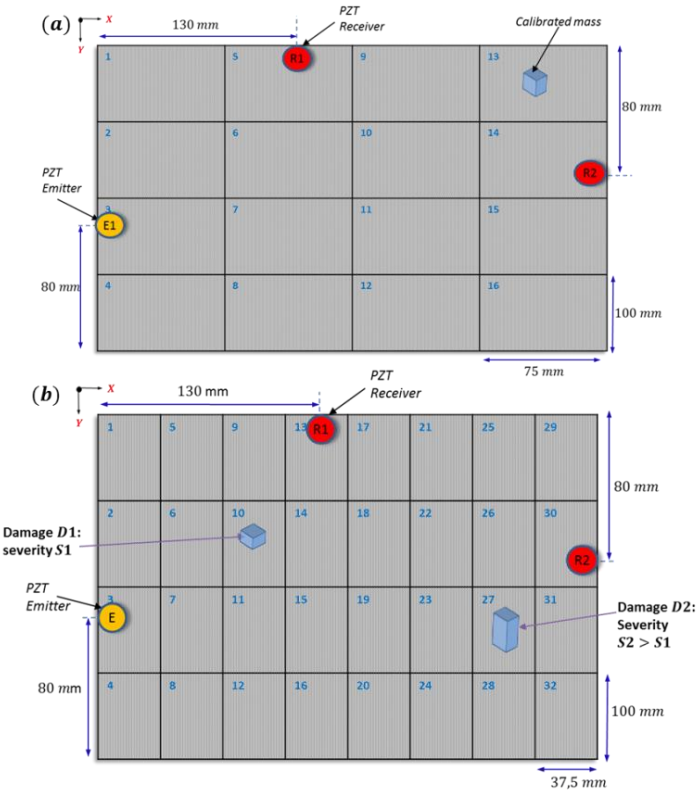
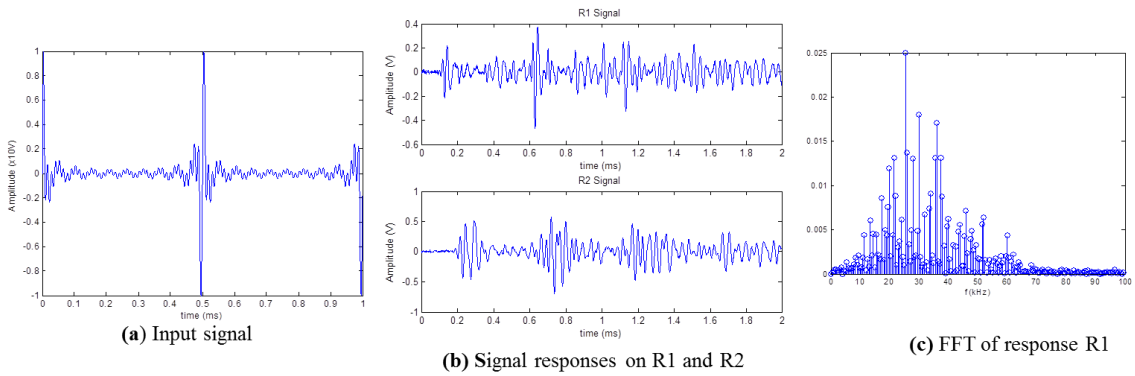


Fig. 5: The predefined damage zones: (***a***) 16 damaged zones and (***b***) 32 damaged zones with two levels of severity.



(a) Input signal

(b) Signal responses on R1 and R2

(c) FFT of response R1

Fig. 6: (***a***) Excitation signal (***b***) Time response **(c)** FFT response

The strategy retained here consists in monitoring the acoustic wave propagation (or the energy carried out by the Lamb wave) that can be affected by impact damages. Considering that the energy propagation is disturbed by the presence of damage, the diffraction signal can then provide the information needed to locate and quantify it. Following this observation, we use an active damage monitoring process based on the Amplitude Disturbed Diffraction Pattern (ADDP) phenomenon observed using the permanent emission of selected non-resonant Lamb waves (> 20). Lamb waves at different frequencies have different propagating velocities and ratios between the normal and the shear components. Therefore, they provide distinct diffraction behaviors when damage occurs on the plate. With an appropriate analysis and calibration procedure, we are able to locate the damage position and to quantify it.

For our application, an excitation signal composed of 31 frequency components distributed from $20\ \text{kHz}$ to $100\ \text{kHz}$ has been constructed with a sampling frequency of $2\ \text{MHz}$ and duration of $1\ \text{ms}$ as depicted in Fig. 6. The number of the chosen frequencies is a tuning parameter selected to discriminate between features at numerous predefined damage positions. The reliability of this process depends highly on this choice. Following the approach developed in [48], we use the contrast value of the ADDP image as a criterion to validate a particular frequency excitation. In the sequel, we propose to use as damage features, the FFT (Fast Fourier Transform) of the signal received by each sensor $R1$ and $R2$.

Given that each sensor perceives different parts of the information, we have concatenated the signals coming from both sensors, obtaining a vector containing 400 attributes.

$$x_i = [\, f_1(R_1), f_2(R_1), \dots, f_{200}(R_1), f_1(R_2), \dots, f_{200}(R_2)] \quad i = \{1, \dots, n\}$$

In terms of SVM (see Section. 2.1), the labels y_j associated with the examples x_i will be the Q zones in which the plate is divided ($Q = 16$ or 32), as shown in Fig. 5.

3.3 SHM framework and SVM-GPDT classification architecture

The damage detection, localization and quantification algorithm proposed here is a sequential process that can be described by three main steps: calibration, learning and identification (Fig. 7). In the proposed framework, the first step consists in training the system. Then, when damage is detected, we first quantify its severity and then locate it. The general architecture of the classifier is illustrated in Fig. 8.

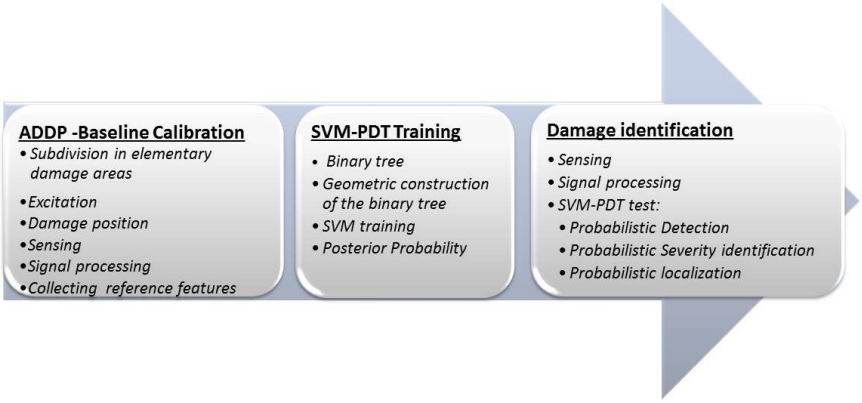


Fig. 7: The SHM classification based framework

The detection test is a simple probabilistic SVM, where we have to choose between two classes damaged and undamaged. As the number of severity classes are smaller than the number of damaged location (here only 2 classes), we have decided to perform an SVM-PDT to select which severity class we are concerned with and then to locate the damage. Knowing that SVM performances are heavily dependent on the hyper-parameters (C, σ^2) being used, it is recommended to perform a model selection procedure on each database to be learned (severity, location). This has been done according to Lin's algorithm [37] designed to choose the optimal hyper-parameters. The procedure for training is described in Fig. 8(b).

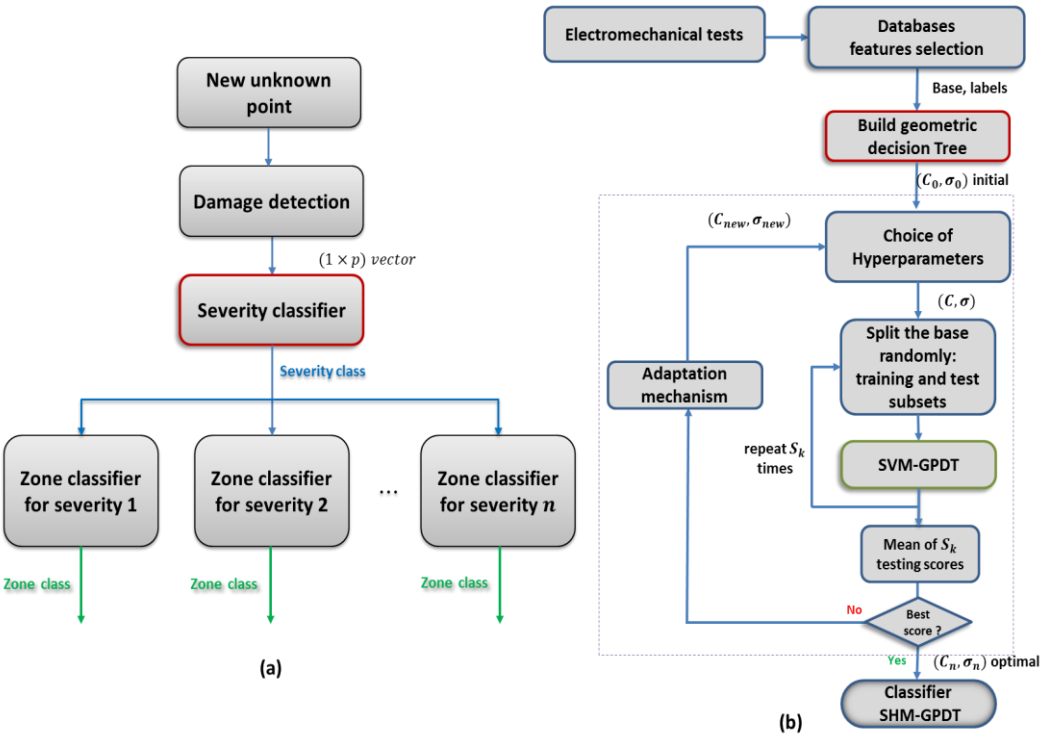
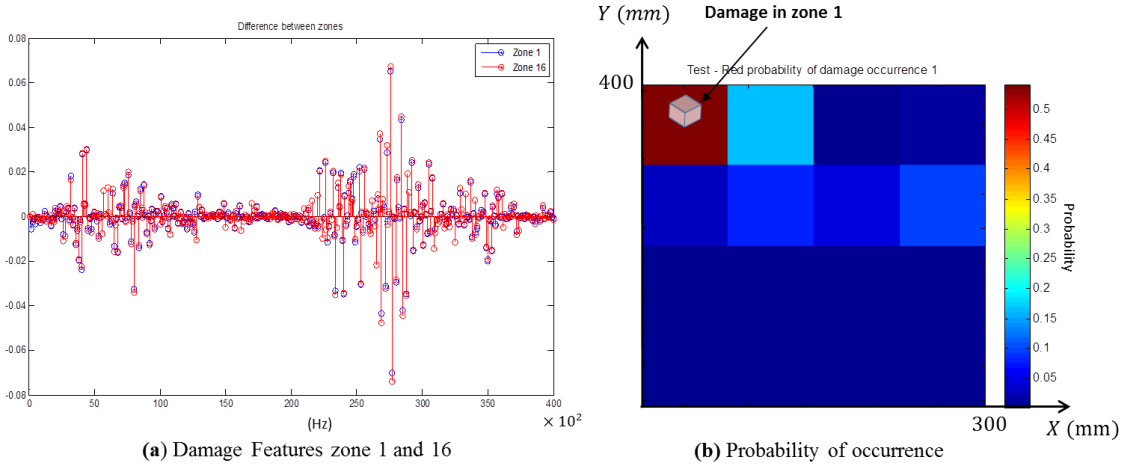


Fig. 8: General Architecture. $(\boldsymbol{a})$ Severity-zone classification. $(\boldsymbol{b})$ Adaptation strategy

3.4 Probabilistic damage localization

To assess the performances of the proposed geometric classification approach, we have applied it first to locate damages on the plate partitioned into 16 zones. The idea in this subsection is to show how the posterior probability of a damage location is constructed. In these tests, only one type of damage ($m_1 = 20$ g) has been considered. The database is composed of 96 normalized measures made on the plate (6 repetitions for each position, $n = 6$). In Fig. 9(a), an example of damaged features (from zone 1 and 16) used to train the classifier is shown. One can notice from these signals, that the difference is not obvious and thus that it is hard to distinguish between them.



(a) Damage Features zone 1 and 16 **(b)** Probability of occurrence

Fig. 9: Damage localization in zone 1: (***a***) damage features, (***b***) spatial probability distribution

The SVM-PDT classifier has been trained using a *k-fold* Cross-Validation method [49], with $k = 4$. So the classifier is trained 4 times, using 75% of the data to train the classifier and the remaining 25% of the points for testing. Once trained, the probability associated with each damage position can be shown on a figure with a color amplitude scale. The algorithm had a performance of 88% of examples well classified (*i.e.* well localized here) using a Gaussian kernel and the optimal hyper-parameters found after the optimization mechanism, which are $C = 10^6$ and $\sigma = 10^3$.

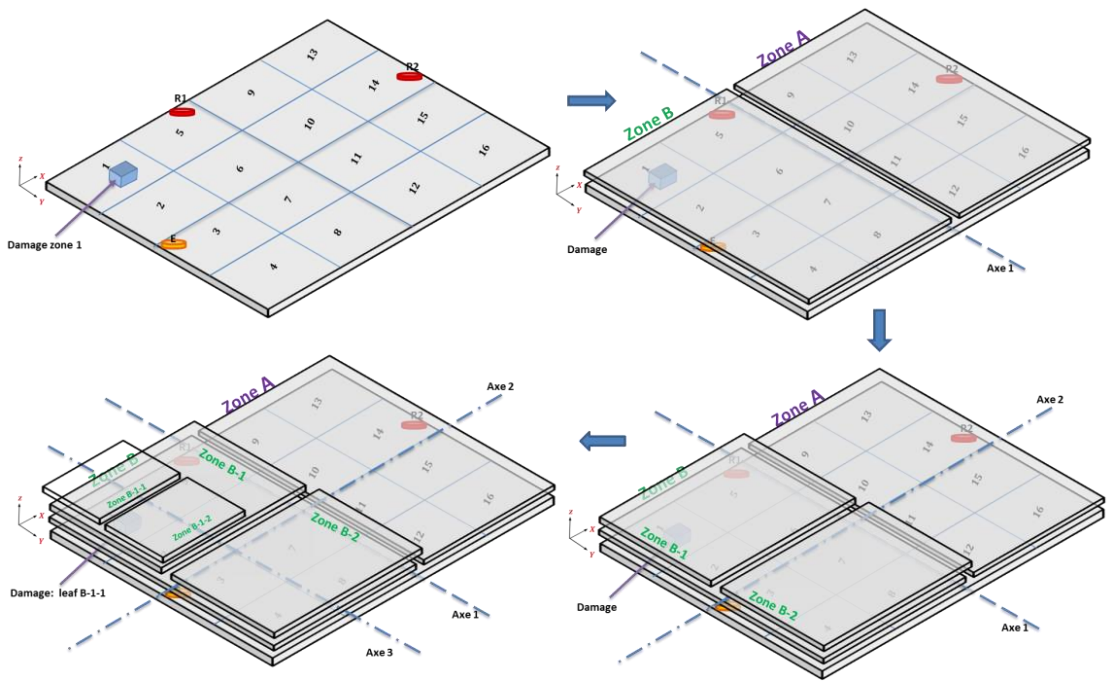


Fig. 10: GPDT decision boundaries for damage locate in zone 1

As presented in Section 3, a geometric approach based on the recursive generation of decision boundaries following symmetries along affine hyperplanes is used to generate the GPDT. As results, we depict in Fig. 9(b) the damage posterior spatial distribution provided by our algorithm. For this damage location test, snapshots at different nodes of the tree are illustrated in Fig. 10. The upstream classes are divided in two groups by clustering symmetrical zones. If we analyze the decision boundaries generated to locate the damage in zone 1, one can see from Fig. 10 that the symmetry axes are alternatively chosen along Y axis and X axis. This automatic way of subdividing the structure and constructing the tree is not mandatory and alternate splitting strategies can be chosen.

In Fig. 11(a), the probability distribution over the plate is shown for a test that was made with a damage located in zone 13. This test shows one of the advantage of the proposed pattern recognition approach, which is that not only the most likely damaged zone is given but also zones with small damage probability. Indeed, we can notice from Fig. 11(a), that the neighbor's zones, have small probabilities of being the damaged zone. This information can be useful in the decision and prognostic steps as it can help improving the specification of robust bounds. The tree construction with the respective posterior probability at each node is illustrated in Fig. 11(b). To evaluate the posterior probability of a damaged zone, we use equation (9) by following the path generated to reach it. Note that there is only one way to get to a leaf (or equivalently a damaged zone). As depicted in Fig. 11, the algorithm detects and locates the damage in zone 13. The posterior probability associated with this diagnostic is $p(zone\ 13|u) = 0.5948$ for zone 13 and $p(zone\ 9|u) = 0.1301$ for damage in zone 9.

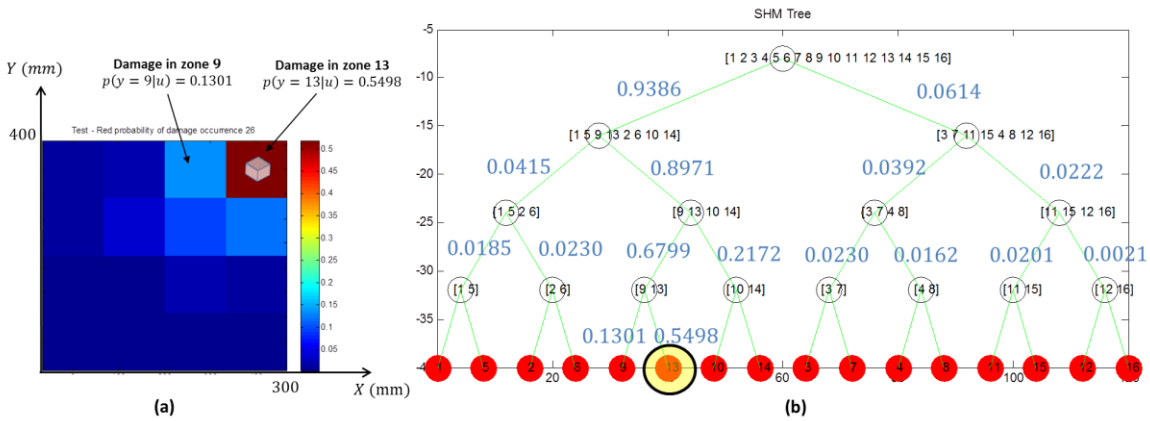


Fig. 11: **(a)** Probability of damage location. **(b)** Probability path for damage in zone 13.

3.5 Probabilistic damage localization and quantification

We present now the results of the SVM-GPDT for damage localization and severity quantification. The plate is now divided in 32 zones (Fig. 5(b)) and electromechanical tests with added masses, simulating damage, have been conducted. We performed 8 tests ($n = 8$) for each zone, using the two masses $m_1 = 20g$ and $m_2 = 10g$. and hence two severity (S_1 and S_2) Therefore, our database has now 512 entries ($32\ zones\ \times\ 2\ masses\ \times\ 8\ repetitions$).

Following the architecture presented in Figure 12, our first concern was to determine the severity of the damage. To do so, all the examples from the tests performed with the mass m_1 have been assigned to a class, and the tests performed with the mass m_2 to another class. This way, we have a simple bi-class probabilistic SVM classifier to quantify severity. Again, we selected the Gaussian kernel and used Lin's algorithm [50] to determine the optimal hyper-parameters. The algorithm had a performance of 99.42% of examples well classified in terms of severity with $C = 10^6$ and $\sigma = 10^3$.

After the determination of the severity of the damage, we proceed to its localization. A second classifier is trained, using a procedure similar to the one shown in Fig. 8(b), the only difference being that we will only present the data for one type of severity to this classifier. This makes the task easier for the classifier, because different severities have different features and it will be very hard to the classifier to learn different signatures pointing to the same zone. So, it is better to only have a single signature for each zone, knowing the severity of the damage.

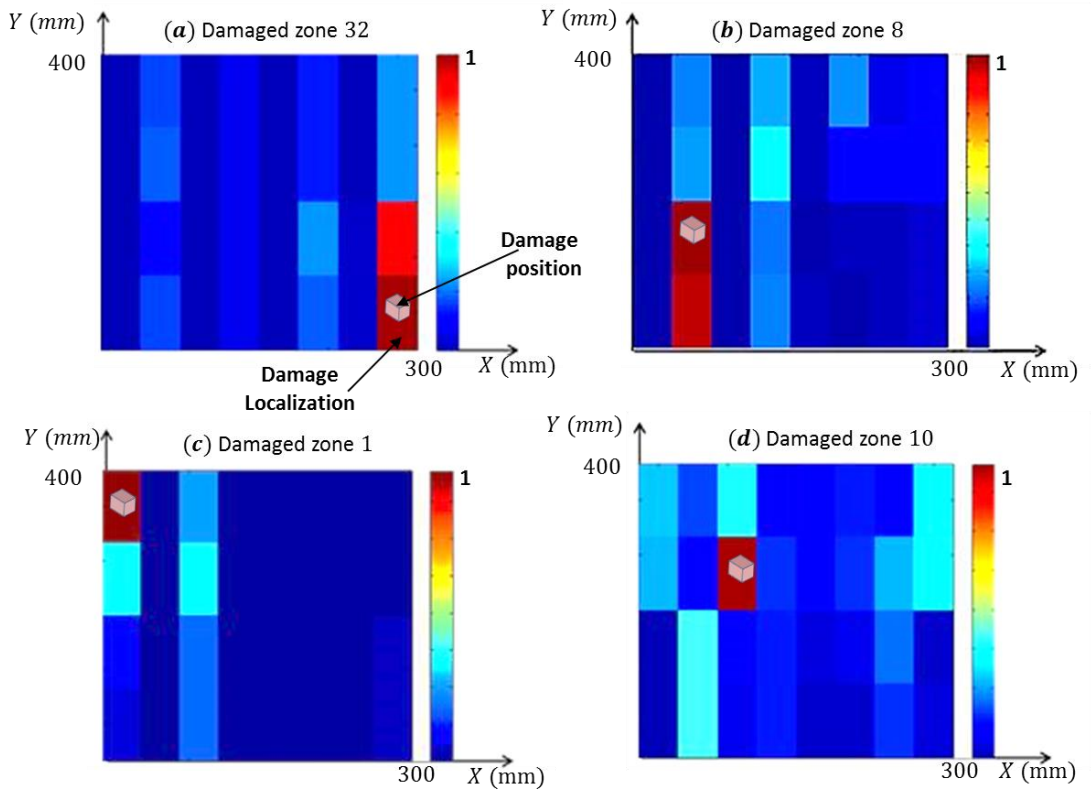


Fig. 13: Normalized location probability for a damage with severity S_1

For severity S_1, we had a localization performance of 65.62%, using $C = 10^6$ and $\sigma = 10^2$. Using the test subset, normalized probability of several damage locations are depicted in Fig. 13 showing the effectiveness of the proposed approach.

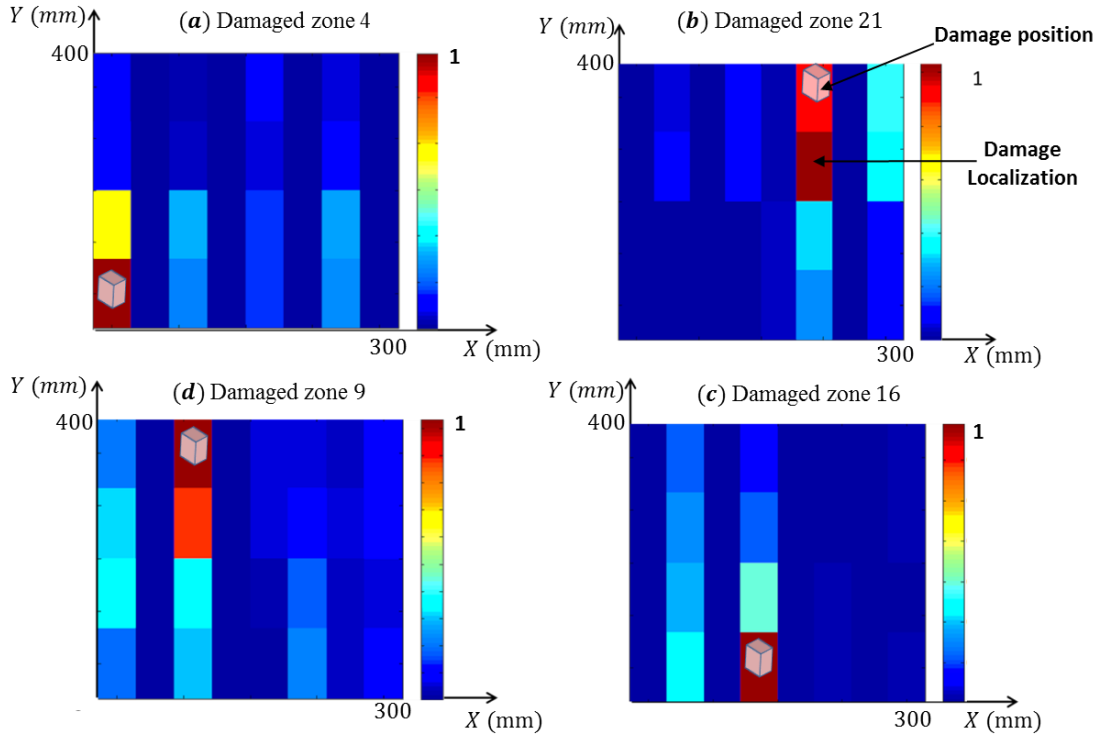


Fig. 14: Normalized location probability for a damage with severity S_2

For severity S_2 we obtained a performance of 62.5%, using $C = 10^5$ and $\sigma = 10^3$. Results for some damage location are shown in Fig. 14. As expected, results for severity S_1 are better than those for S_2. This was expected as the damage effects on measured signals are less significant with a smaller mass. Moreover, for some damaged zones as shown in Fig. 14(b), the location results could be misclassified in a neighboring zone, which is not such a problem. This classification error is related to the accuracy of the ADPP process used to generate damage features. Better results can be obtained with more frequencies in the excitation signal in order to discriminate some spurious features that can appear with wave reflections.

4 Discussion

4.1 Availability of a learning database

In the proposed approach, monitoring small zones facilitate the diagnostic but assume the availability of a damaged database for each zone. Moreover, good spatial resolutions suppose that several damaged structures are used in order to construct the precise database needed for the learning process. This is generally unrealistic unless the structures are cheap and easily available. To overcome this problem, one can simulate experimentally the damage effect instead of imposing a real one on the structure. For example, by changing locally in a reversible manner some of the characteristics of the structure (stiffness, natural frequencies …). The problem of acquiring data for each damage location can also be alleviated by using first, a high-fidelity physical model of the structure, usually by means of finite element analysis, that simulates the entire physical

phenomenon: non linearity, degradation, and environmental operational conditions. And then, a surrogate model could be derived from the precise one to avoid a large number of heavy computations. This lower fidelity model will approximate the response of the complete one in an efficient manner and generate sufficient data for the learning process [51].

4.2 Hierarchical or "flat" classification for damage monitoring?

Notionally, one can substitute hierarchical classification with "flat" one (standard binary or multiclass classification). Indeed, if we overlook our geometric class structure and replace the hierarchy with a set of classes, we can perform a multi-class classification problem. This means that all the relevant information that we have introduced through our knowledge about the damage monitoring problem and the structure geometry are ignored. In order, to show how this geometric hierarchical class structure boosts the performance of the damage localization, we have performed new experiments by comparing the performance of our classifier against other known classifiers. A multiclass SVM, using the one against all (SVM-OAA) method was chosen. This flat classifier was trained with all the samples from the training base (16 zones and S_1 severity), doing a *k-fold* cross-validation method with $k = 4$. The classifier has good results when testing with training data (91%) but difficulty for generalization as shown in Fig. 15. In this figure the two classifiers had to classify new test points generated with slightly different environmental conditions. The damages have been introduced by the small weight that has been positioned at: (a) the center of zone 9, (b) the center of zone 11 and (c) at the frontier between zones 12 and 16. As it can be seen, even the SVM-GPDT misclassified damage location, one could give credit to these partially correct results by checking the neighbor and the top-level zones given by the geometric hierarchy structure. Therefore, the maintenance procedure is facilitated and enhanced.

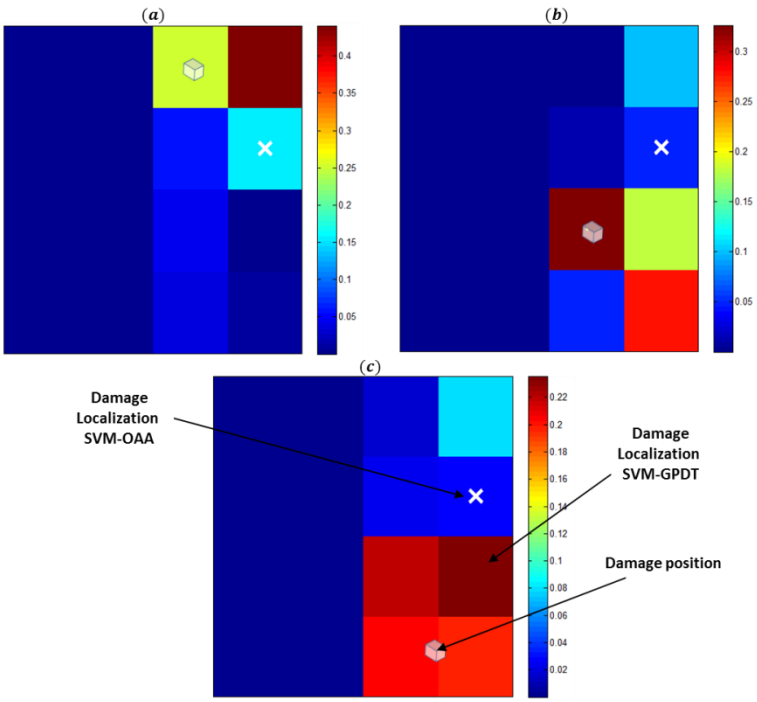


Fig. 15: Predicted damaged zones for the SVM-OAA (white cross) and SVM-GPDT.

4.3 Multi class or multi-label classification problem?

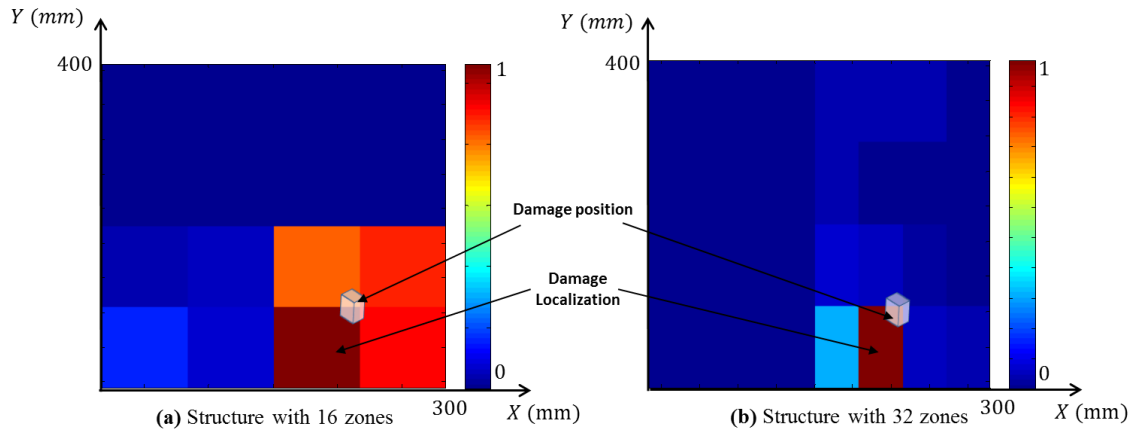
The purpose of the present paper is to derive an original answer to the problem of damage detection, localization and quantification. A question that can be raised is whether a multi-class formulation of this problem is more appropriate than a multi-label formulation. The difference between the notions of class and label lies in the fact that an example can belong to only class, but can have multiple labels. Multi-label classification is thus an extension of multi-class classification where data can have one or more labels [52]. Due to its great potential applications, multi-label learning has now been receiving more and more attention from many fields [53, 54, 44]. According to [55, 30], in the context of hierarchical classification, most approaches could be called multi-label. Indeed, if we recall the tree class structure presented in Fig. 3b, and suppose that the output of the classifier is the class {1}. Then it's natural to say also that the classifier have 3 outputs because class {1} belongs to classes {1 3} and {1 3 5 7} too. Therefore, with this simple definition, any hierarchical classification could be considered multi-label in this sense. Hierarchical multi-label classification algorithm is still a challenging problem and most approaches consider the case where it can be assigned more than one class at any level of the tree [56].

Going back to the SHM problem, the output of SHM systems is a vector with 3 features: (*i*) damage zone, (*ii*) damage severity, (*iii*) damage type. The definition of each features is straightforward: several damage locations, several severity levels (BVID , VID) and different damage types (impact, debonding, delamination, …). With that in mind, it comes out that a class can be defined in a unique manner: one zone, one severity level and one type. With this class definition, the only overlapping examples are those lying along the frontier between two classes for a given feature. For example, damage could be located at the frontier between two zones or having a severity level between two pre-defined levels. These overlapping examples are thus a direct consequence of the a priori discretization of the features location and severity that are, in reality, continuous and not an intrinsic property of the problem under study. From the authors point of view, the SHM problem can therefore be naturally cast as a multiclass problem and not as a multi-label one. From that point of view, the proposed classification algorithm can be seen as an implicit probabilistic multi-label method where labels exist at the frontiers between classes. However, another point of view would be to define classes in relation to the different features. Following that idea, there will be one class for each damage location, one class for each damage type, and one class for each damage severity. Knowing this, there is a natural overlap between the different classes: for example, damages located in a given zone can be of distinct types and of severities. So with this definition of class, classes are indeed labels and the problem to face is a multi-label one. In Section 4.1, the consequence of spatial discretization is studied. It is shown that if the localization result has an ambiguity between neighborhood zones then by looking to the node just above in the decision tree, a macro-damaged zone can be defined. This is still a helpful guidance for the inspection task as afterwards the damage location can be refined by using traditional inspection techniques.

4.4 Spatial resolution versus computational cost

The performances of the classifier in term of spatial selectivity depend heavily on the method used to generate features and the availability of the damaged learning database. Increasing the spatial resolution will increase the training and testing times. The training time depends on the

number of classes Q and the number of training samples. However, increasing the number of zones will enhance the detection process by reducing the rate of false damage localization. To illustrate this point, we have given to the two classifier (16 and 32 classes or zones), "unknown" damages to detect and locate. These damages are new experiences performed by introducing the mass $m_1 = 20g$ on the plate with 16 classes at positions located on the edge of the separation zones and then this damage is monitored with 32 zones. In Fig. 16, a result is presented. We can notice that as expected the ambiguity of localization has decreased with the increase of the number of zones and consequently an increase of computational cost. We can also notice that, in general, the output of both classifiers is inline with the actual damage location. Therefore, this should be a tradeoff that one have to deal with when the applying such approaches in practice.



(a) Structure with 16 zones **(b)** Structure with 32 zones

Fig. 16: Spatial selectivity, number of zones and damage localization

4.5 Localization before quantification, or the opposite?

In the literature, it is very often stated that the localization step should occur before the quantification phase [1]. However, in the present work the quantification step has been performed before the localization one and gave excellent discrimination results between the two damages of different severities (added mass of $20\ g$ and the added mass of $10\ g$). Moreover, getting one and only one signature for each zone makes easier the classification task. If the localization is done first, different signatures will be assigned for the same localization. Those signatures could be very different if the gap between the severities becomes higher. It may even happen that the classifier could not find similar attributes in these groups of mixed severities, which would lead to poor classification performances. Another solution could have been to perform first the localization, and then, at the end of the GPDT to add a bi-class classifier to each leaf to assess the severity of the damage. However, this approach suffers from the fact that instead of one classifier needed for the present approach, one needs now 32 classifiers. So, from a computational point of view, this approach does not seem to be justified here. It can nevertheless be interesting, as a perspective, to implement a *"localization then quantification"* version of the present algorithm to compare its results with the present results.

5 Conclusion

In this paper, an original process for damage detection, localization and quantification on smart mechanical structures instrumented with piezoelectric active elements has been presented and validated experimentally. The SVM Geometric Probabilistic Decision Trees (SVM-GPDT) constitutes an original approach to the multi-class probabilistic damage classification problem. This algorithm takes advantage of the decision tree architecture and of the posterior probability provided by PSVM. At each node of the PDT, SVM classification associated with a sigmoid function is performed to estimate the probability of membership to each sub-group. A probability function is then built for each leaf by following the path that the PDT has generated for it. For each branch the outputs of all the nodes composing the branch are combined to lead to a complete evaluation of the probability when reaching the final leaf (representing the class associated with the branch). Formally, we thus presented and validated an interesting solution to solve a multi-class data classification problem in a manner that produces confidence probabilities associated with each damage feature. Moreover, the availability of the damage posterior probability will facilitate embedded damage monitoring. The SHM process will be trained offline and assessed online by calculating in real time the probabilities of having damages.

The approach has been tested experimentally on a composite plate, giving reliable predictions in terms of detection, localization and quantification. For real, applications, this approach is interesting as the chosen damage-sensitive feature is not very sensitive to border conditions, and the excitation with non-resonance frequencies will not accentuate the damage progression and impact the durability of the monitored structure.

Regarding the classification performances of the prosed method, it first advantage lies in the fact that it allows not just for classification, but for classification with an associated posterior probability. The second one is that the classification procedure proposed here is relatively fast as it is of logarithmic complexity. The third one is that the notion of geometry is introduced to the classifier. Because SHM is a problem in the real space, we can help the classifier by giving it some a-priori information that might discard data that is not relevant in a particular classification process, reducing the data to be treated, the time for treating it and the possibilities of mistakes. The last advantage is that as it relies on SVM, relatively few examples are needed for training. Regarding drawbacks, the first one is that the proposed method is based on discrete classes and thus imposes quantification of the different data features. The second one concerns the fact that in order to train the method example of damaged states are needed, and such examples are in practice not widely available. Actually, it is well known, that the main issue when applying pattern recognition approaches to damage monitoring is to construct a database of damage conditions. Indeed, the damage diagnostic system will not perform better than the data used to build it. As we need to avoid damaging the structure, two approaches can be used: multi-physical modeling and/or simulation of damage effects. It's this last approach that was used in this work. Another drawback is related to training set of data and the classifier generalization performances. Every classifier has its limits regarding the generalization capacity. Further tests should be performed in order to have a more precise idea about its capacities. We think that this drawback might be mitigated thanks to the probabilistic output. When the classifier is completely lost, he should provide a low probability associated to the uncertain prediction. The last drawback is linked to the training stage. Indeed, there is in reality, two training stage: the training of the SVMs of each node, and the training to get the probabilities out of each node. Thus, the previously mentioned advantage of the method (which is the need of few training points) has to be evaluated by considering that drawback.

Suppose now that actual data related to damaging locations are available to train the machine learning diagnostics. Then, another problem rise which is how to elaborate features that could reliably discriminate between damage and undamaged states. The ADDP process has been used within this paper but maybe some other damage-sensitive features will perform better.

Another useful issue in a data mining is to have access to posterior probability associated with each damage classification. The approach presented throughout this work proposes a solution to the multi-class data classification problem in a manner that produces confidence probabilities associated with each damage feature. Each damage feature is classified from the most probable one to the least probable one. This is very interesting because this information can be valuable in the decision and prognostic steps and can help in improving the specification of robust bounds. Moreover, this knowledge could help the maintenance procedure by crossing with expert data.

References

[1] K. Worden, C. R. Farrar, G. Manson and G. Park, "The fundamental axioms of structural health monitoring.," *Proceeding of the Royal Society,* vol. 463, pp. 1639-1664, 2007.

[2] F.-K. Chang, "Structural Health Monitoring: Condition-based Maintenance," in *8th International Workshop on Structural Health Monitoring (IWSHM),* Stanford, 2011.

[3] D. Balageas, C. P. Fritzen and A. Güemes, Structural Health Monitoring, London : ISTE Ed., 2006.

[4] E. P. Carden and P. Fanning, "Vibration based Based Condition Monitoring: A Review," *Structural Health Monitoring Journal,* no. 3, 2004.

[5] S. Chesné and A. Deraemaeker, "Damage localization using transmissibility functions: A critical review," *Mechanical Systems and Signal Processing,* vol. 38, p. 569–584, 2013.

[6] C. Fritzen, D. Jennewein and T. Kiefer, "Damage detection based on model updating methods," *Mechanical Systems and Signal Processing,* vol. 12, no. 1, p. 163–186, January 1998.

[7] Y. Zou, L. Tong and G. P. Steven, "Vibration-based lmodeldependent damage (delamination) identification and health monitoring for composite structures - a review," *Journal of Sound and Vibration,* vol. 230, no. 2, pp. 357-378,, 2000.

[8] M. Basseville, L. Mevel and M. Goursat, "Statistical model-based damage detection and localization: subspace-based residuals and damage-to-noise sensitivity ratios," *Journal of Sound and Vibration,* vol. 275, pp. 769-794, 2004.

[9] L. Mevel, I. Gueguen and D. Tcherniak, "LPTV subspace analysis of wind turbines Data," in *7th European Workshop on Structural Health Monitoring,* Nantes, 2014.

[10] R. Hajrya and N. Mechbal, "Principal Component Analysis and Perturbation Theory Based Robust Damage Detection of Multifunctional Aircraft Structure," *Structural Health Monitoring an International Journal,* 2013.

[11] R. Hajrya and N. Mechbal, "Perturbation analysis for robust damage detection with application to multifunctional aircract structures," *Smart Structures and Systems, An international Journal,* p. to appear, 2015.

[12] M. Rebillat, R. Hajrya and N. Mechbal, "Nonlinear structural damage detection based on cascade of hammerstein models," *Mechanical Systems and Signal Processing,* vol. 48, no. 12, pp. 247-259, 2014.

[13] C. R. Farrar, K. Worden, M. D. Todd, G. Park, J. Nichols, D. E. Adams, M. T. Bement and K. Farinholt, "Nonlinear system identification for damage detection," USA, 2007.

[14] Z. Su and L. Ye, Identification of Damage Using Lamb Waves, Springer, 2009.

[15] J.-B. Ihn and F.-K. Chang, "Pitch-catch active sensing methods in structural health monitoring for

aircraft structures.," *Structural Health Monitoring,* vol. 7, no. 1, pp. 5-19, 2008.

[16] S. Zhongqing, Y. Lin and L. Ye, "Guided Lamb waves for identification of damage in composite structures: A review," *Journal of Sound and Vibration,* vol. 295, p. 753–780, 2006.

[17] Y. Liu, J. Nikolovski, N. Mechbal, M. Hafez and M. Vergé, "An acoustic multi-touch sensing method using amplitude disturbed ultrasonic wave diffraction patterns," *Sensors and Actuators A,* vol. 162, pp. 394-399, 2010.

[18] Y. Liu, N. Mechbal and M. Vergé, "Damage monitoring based on wave illumination of structures," in *International Workshop on Structural Health Monitoring (IWSHM)*, Stanford, USA, 2011.

[19] C. R. Farrar and K. Worden, Structural Health Monitoring: A Machine Learning Perspective, John Wiley & Sons, 2012, p. 654.

[20] L. Roseiro, U. Ramos and R. Leal, "Neural networks in damage detection of composite laminated," *WSEAS Transactions on Systems,* vol. 4, no. 4, p. 430–434, 2005.

[21] K. Saeed, N. Mechbal, G. Coffignal and M. Vergé, "Artificial Neural Network Based Structural Damage Diagnosis Using Nonparametric Subspace Residual," in *7th International Workshop on Structural Health Monitoring, IWSHM,* Stanford (USA), 2009.

[22] Y. Liu, H. H. Zhang and Y. Wu, "Hard or soft classification? large-margin unified machines," *Journal of the American Statistical Association,* vol. 106, no. 493, pp. 166-177, 2011.

[23] V. Vapnik, Statistical learning theory, Wiley, New York, 1998.

[24] J. Friedman, *Another approach to polychotomous classifcation,* Stanford: Stanford University, Department of Statistics, 1996.

[25] R. K. Eichelberger and V. S. Sheng , "Does One-Against-All or One-Against-One Improve the Performance of Multiclass Classifications?," in *the Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.

[26] R. Rifkin and A. Klautau, "In Defense of One-Vs-All Classification," *Journal of Machine Learning Research,* vol. 5, pp. 101-141, 2004.

[27] T. Dietterich and G. Bakiri, "Solving multiclass learning problems via error correcting output codes," *Journal of Artificial Intelligence Research,* vol. 2, pp. 263-286, 1995.

[28] J. Platt, "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods.," in *Advances in Large Margin Classifiers*, Cambridge, MIT Press, 2000.

[29] N. García-Pedrajas and D. Ortiz-Boyer, "An empirical study of binary classifier fusion methods for multiclass classification," *Information Fusion,* vol. 12, pp. 11-130, 2011.

[30] C. N. J. Silla and A. Freitas, "A survey of hierarchical classification across different application domains," *Data Mining and Knowledge Discovery,* vol. 22, no. 1-2, pp. 31-72, 2011.

[31] G. Madzarov, D. Gjorgjevikj, I. Chorbev and others, "A multi-class svm classifier utilizing binary decision tree," *Informatica,* vol. 33, no. 2, pp. 233-241, 2009.

[32] W. David, J. Hosmer and S. Lemeshow, Applied Logistic Regression, Wiley, 2004.

[33] J. Platt, N. Cristianini and J. Shawe-Taylor, "Large margin DAGs for multiclass classification," *Advances in neural information processing systems,* vol. 12, no. 3, pp. 547-553, 2000.

[34] J. Uribe, N. Mechba, M. R. K. Bouamama and M. Pengov, "Probabilistic Decision Trees using SVM for Multi-class Classification," in *International Conference on Control and Fault-Tolerant Systems, SysTol'13*, Nice, 2013.

[35] Y. Ma and G. Guo, Support Vector Machines Applications, Springer, 2014.

[36] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology,* vol. 2, pp. 27:1--27:27, 2011.

[37] S. Keerthi and C.-J. Lin, "Asymptotic behaviors of support vector machines with Gaussian kernel,"

Neural computation, vol. 15, no. 7, pp. 1667-1689, 2003.

[38] H.-T. Lin, C.-J. Lin and R. C. Weng, "A note on Platt's probabilistic outputs for support vector machines," *Mach. Learn.,* vol. 68, no. 3, pp. 267-276, oct 2007.

[39] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *Neural Networks, IEEE Transactions on,* vol. 13, no. 2, pp. 415-425, 2002.

[40] M. Aly, "Survey on Multiclass Classification Methods,," USA, 2005.

[41] S. Khan and M. Madden, "A survey of recent trends in one class classification," in *Artificial Intelligence and Cognitive Science: 20th Irish Conference, AICS*, Berlin, Springer-Verlag, 2010, pp. 188-197.

[42] N. Mehra and S. Gupta, "Survey on multiclass classification methods," *International Journal of Computer Science and Information Technologies,* vol. 4, no. 4, pp. 572-576, 2013.

[43] J. Weston and C. Watkins, *Multi-class support vector machines Weston, J., & Watkins, C. (1998). Multi-class support vector machines. Technical Report CSD-TR-98-04,* London: Department of Computer Science, Royal Holloway, University of London, 1998.

[44] T. Ganu, S. Shevade and S. S., "Sparse Max-Margin Multiclass and Multi-label Classifier Design for Fast Inference," in *SIAM International Conference on Data Mining (SDM),* 2013.

[45] D. Tsujinishi, T. Koshiba and S. Abe, "Why Pairwise is Better Than One-Against-All or All-at-Once," 2007.

[46] M. A. Bagheri, G. A. Montazer and E. Kabi, "A subspace approach to error correcting output codes," *Pattern Recognition Letters,* vol. 34, pp. 176-184, 2013.

[47] A. Benjeddou, "Piezoelectric smart structures and systems," *Smart Structures and Systems,* vol. 5, no. 6, p. Preface, 2009.

[48] Y. Liu, J. Nikolovski, N. Mechbal, M. Hafez and M. Vergé, "Tactile objects based on an amplitude disturbed diffraction pattern method," *Applied Physics Letters,* vol. 95, no. 25, 2009.

[49] J. Rodriguez, A. Perez and J. Lozano, "Sensitivity Analysis of k-Fold Cross Validation in Prediction Error Estimation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 32, no. 3, pp. 569-575, March 2010.

[50] S. S. Keerthi and C.-J. Lin, *Asymptotic behaviors of support vector machines with Gaussian kernel,* vol. 15, MIT Press, 2003, pp. 1667-1689.

[51] B. Echard, N. Gayton, M. Lemaire and N. Relun, *Reliability Engineering and System Safety,* vol. 111, p. 232–240, 2013.

[52] W. Gao and Z.-H. Zhou, "On the consistency of multi-label learning," *Artificial Intelligence,* Vols. 199-200, pp. 22-44, 2013.

[53] G. Tsoumakas and I. Katakis, "Multi-Label Classification: An Overview," *International Journal of Data Warehousing & Mining,* vol. 3, no. 3, 2007.

[54] M. R. Boutell, J. Luo, X. Shen and C. M. Brown, "Learning multi-label scene classification," *Pattern Recognition,* vol. 37, pp. 1757-1771, 2004.

[55] D. Tikk, G. Biró and J. Yang, "A hirarchical text categorization approach and its application to FRT expansion," *Aust. T. Intell. Inform. Process Syst.,* vol. 8, no. 3, pp. 123-131, 2004.

[56] F. De Comité, R. Gilleron and M. Tommasi, "Learning Multi-label Alternating Decision Trees from Texts and Data," in *Machine Learning and Data Mining in Pattern Recognition*, Berlin, Springer-Verlag, 2003, pp. 35-49.

[57] S. Fassois and J. Sakellariou, Statistical Time Series Methods for SHM, C. Boller and F. a. F. Y. Chang, Eds., Chichester: John Wiley and Sons Ltd, 2009, pp. 443-472.