A INTELLIGENT SYSTEM FOR FAILURE PREDICTION IN *LCMS* USING BAYESIAN NETWORKS

A. O. Addin[1], S. M. Sapuan[2], E. Mahdi[3] and M. Othman[4]

[1]Laboratory of Advanced Technology, Institute of Advanced Technology
[2]Department of Mechanical and Manufacturing Engineering
[3]Department of Aerospace Engineering
[4]Department of Communication Technology and Networks
Universiti Putra Malaysia, 43400 Serdang,
Selangor.

Abstract

This paper intended to introduce the Bayesian network in general and the Naïve-bayes classifier in particular as one of the most successful intelligent systems to classify damages in composite materials. A method for feature subset selection has also been introduced. The method is based on mean and maximum values of the amplitudes of waves after dividing them into folds then grouping them by a clustering algorithm (e.g. k-means algorithm). The Naïve bayes classifier and the feature subset selection method were analyzed and tested on sets of data. The data sets were conducted based on artificial damages created in quasi-isotopic laminates of the AS4/3501-6 graphite/epoxy system and ball bearing of the type 6204 with a steel cage. The Naïve bayes classifier and the proposed feature subset selection algorithm have been shown as very effective techniques for damage detection in composite materials.

Introduction

Recently, there has been a tremendous growth in the usage of laminated composite materials (*LCMs*) in all types of engineering structures (e.g. aerospace, automotive, and sports). *LCMs* are fabricated by stacking plates or plies of composite materials together to acquire unique properties (e.g high strength and stiffness, and light weight) that cannot be guaranteed by individual constituents of the laminate. However, in practical situations, material failure or damage may occur during manufacturing processes or in-service. The manufacturing related damages are like foreign object inclusion, porosity, and resin rich areas. In-service damages can happen in the case of aeronautical materials because a tool is dropped during maintenance, there is a bird or hail strike in plain flight, perhaps runway debris striking the aircraft during takeoff or landing. The damages have the potential of growing and leading to catastrophic loss of human life, and decrease in economy. Examples of real-life damages can be shown as airline crashes, space shuttle explosions, and building and bridge collapses. The early detection and characterization of in-situ damages in composite materials are very significant to ensure their structural health and integrity, prevent them from catastrophic failures, and prolong their service life. [1, 3].

One of the potential solutions used for damage detection is the structural health monitoring (SHM). The literature defines the *SHM* as the acquisition, validation, and analysis of technical data to facilitate the life-cycle management decisions [2]. Kessler et al. [3] stated that *SHM* denotes a reliable system with the ability to detect and interpret adverse changes in a structure due to damage or normal operation. There are several advantages to using a *SHM* system over traditional inspection cycles, such as reduced down-time, elimination of component tear-down inspections, and the potential prevention of failure during operation. Aerospace structures have one of the highest payoffs for *SHM* applications since damage can lead to catastrophic and expensive failures, and the vehicles involved undergo regular costly inspections [3].

There are several components required to design a successful and robust *SHM* system for dam-age detection. It essentially involves implementation of a nondestructive evaluation (*NDE*) technique (e.g. ultrasonic, eddy-current, acoustic emission, and radiography) to a structure to acquire data for the damage detection. The other components are sensing systems, communications, and algorithms to quantitatively assess the damage detection by interpreting the large amounts of data collected by the sensors. The need for quantitative damage detection methods that can be applied to complex structures has encouraged the *SHM* community to borrow and implement many techniques from artificial intelligent (*AI*) and machine learning (*ML*). In some *NDE* techniques like ultrasonic, actuators and sensors are mounted on the surface of the testing material. The actuators are used to propagate waves into the material. The waveforms reflected by damages and the surfaces of the material are captured and digitized by the sensors for comparison to waveforms captured during the calibration of the damage detection system. The results of this test are then fed into an *AI* or *ML* technique to quantitatively specify the characteristics of the damages found on the material.

There are several components required to design a successful and robust *SHM* system for dam-age detection. It essentially involves implementation of a

nondestructive evaluation (*NDE*) technique (e.g. ultrasonic, eddy-current, acoustic emission, and radiography) to a structure to acquire data for the damage detection. The other components are sensing systems, communications, and algorithms to quantitatively assess the damage detection by interpreting the large amounts of data collected by the sensors. The need for quantitative damage detection methods that can be applied to complex structures has encouraged the *SHM* community to borrow and implement many techniques from artificial intelligent (*AI*) and machine learning (*ML*). In some *NDE* techniques like ultrasonic, actuators and sensors are mounted on the surface of the testing material. The actuators are used to propagate waves into the material. The waveforms reflected by damages and the surfaces of the material are captured and digitized by the sensors for comparison to waveforms captured during the calibration of the damage detection system. The results of this test are then fed into an *AI* or *ML* technique to quantitatively specify the characteristics of the damages found on the material.

Neural network (*NN*) is one of the *ML* techniques which has been widely adopted by many researchers in this area[4]. Chakraborty[5] introduces an approach that predicts the presence of embedded delamination (in terms of location, shape, and size) in fiber reinforced plastic composite laminates using back propagation (*BP*) *NN* with 3 layers (input, hidden, and output). The network has been tested to predict the presence of delamination along with its size, shape, and location. Su and Ye [6] have demonstrated a lamb wave (*LW*) propagation-based quantitative identification scheme for delamination in carbon-fiber reinforced polymer (*CFRP*) composite structures using a multi-layer *BP NN*. Other methods like rule-based, fuzzy logic, and genetic algorithms also have been adopted for the damage detection and identification.

Recently, Bayesian network (*BN*) has emerged as a *ML* technique and a generalizing graph-based framework for creating statistical models of domains with inherent uncertainty. *BN* has attracted a great deal of attention in research institutions as well as in industry as a good modeling tool for medical systems, risk prediction, forecasting, robotics, computer games, and etc.[7, 8]. Nevertheless, the *BN* has not been well recognized by the *SHM* community.

The objective of this paper is to introduce *BN* in general and Naïve bayes in particular as a classifier to simulate damage detection in *LCM*s. The paper also aims to present a novel method for feature subset selection of wave amplitudes of damage detection.

The paper is organized as follows, section two gives a preliminary overview to BN based on composite materials. Section three shows the Naïve bayes in SHM systems and how it can be utilized for damage detection employing. Section four introduces the novel f-fold feature subset selections. Section five shows the experiments and the data sets used to analyze and test the *f*-fold algorithm. Section six shows the results and evaluation of the classifier and the *f*-folds algorithm.

Bayesian Networks

BNs are defined as graphical models that allow users to encode relationships between variables of interest and reason about uncertain domains. They consist of a qualitative part, where features from graph theory are used, and a quantitative part consisting of potentials, which are real-valued functions over a set of variables from the graph. They consist of the following:

- A network structure $G = V, E,$ where $V = V_1, V_2, \ldots ,V_n$ represents a set of variables and E represents a set of directed arcs between the variables.
- Each variable has a finite set of mutually exclusive states.
- A set of conditional probability tables (*CPTs*) associated with each variable.

The directions of the arcs in *BNs* often represent causal dependency between variables. In *BNs*, a variable is a parent of a child, if there is an arc from the former to the later. *BNs* model the quantitative strength of the connections between them, allowing their probabilistic beliefs to be updated automatically as new information arrive. The arcs in any *BNs* are not permitted to be directed cycles, one cannot start from a variable and simply come back to it by following the direction of the arcs in the network. For this reason the networks are known as directed acyclic graphs (*DAGs*) [7, 8].

BNs can be built by an expert on the domain of study, a structure learning algorithm that automatically extract the structure from a data set, or a combination of both.

The values of each variable should be mutually exclusive and exhaustive, that means the variable must take on exactly one of these values at a time. For example, if one considers building a model to predict the presence of a damage in a composite material, many factors might be taken into account, e.g. the age of the material (*Age*) and whether a tool dropped on the material (*Tool-Drop*). These factors can be represented as variables in the model connected by directed links according to the direction of impacts (see Figure 1). In the figure, the variables *ToolDrop* and *Age* have an impact on the variable *Damage*. That means the presence of the damage can be determined

by the states of *ToolDrop* and Age. It cannot be agreed that the damage on the material has caused the dropping of the tool on the material or has an impact on the age of the material. Every variable can take one of a different type of discrete values (the states of the variable). The variables Damage and *ToolDrop* might be represented by states, which take boolean values yes and no. The variable *Age* might be represented by states, which take boolean values *yes* and *no*. The variable *Age* might be represented by states that take ordered values, *new*, *medium*, and *old*.

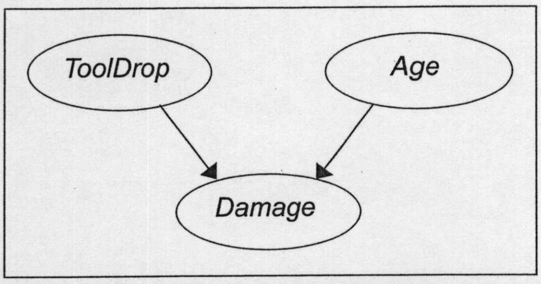


Fig. 1: A small BN structure for damage detection in *LCM*

If *A* is assumed to be a variable with *n* states, $a_1, a_2, ..., a_n$, then *P(A)* denotes a probability distribution over these states:

$$P(A) = (x_1, x_2, ..., x_n; \qquad x_i \leq 0; \qquad \sum_{i=1} x_i = 1 \qquad (1)$$

where x_i is the probability of *A* being in state a_i. This can be written as $P(A=a_i) = x_i$ or $P(a_i) = x_i$, e.g. P(*Age* = *new*) = 0.8.

The basic concept in the *BN* treatment of certainties in causal networks is conditional probabilities. If the variable *B* has *m* states b_i, b_2, b_n, the conditional probability statement can be shown as follows:

"The probability of the event *a* given the event *b* is *x*."

which can be written as $P(a|\ b) = x$. The probability $P(A|\ B)$ implies an n m table including the probabilities $P(a_i|b_j)$.

The fundamental rule for probability calculus is:

$$P(a|b)P(b) = P(a,b) \qquad (2)$$

where $P(a, b)$ is the probability of the joint event *a* and *b*. From this, it can be said that $P(a\ b)\ P(b) = P(b\ a)\ P(a)$, and this yields the well known *Bayes'* rule:

$$P(b \mid a) = \frac{P(a \mid b)P(b)}{P(a)} \qquad (3)$$

In Figure 1, the variable Damage has two parents and the variables *ToolDrop* and *Age* have no any parents. The joint probability distributions for the variables are shown as *P(Damage| Age, ToolDrop) P(ToolDrop)*, and *P(Age)*. These probabilities are determined by an expert or automatically extracted from a data set. Since the variables *ToolDrop* and *Age* have no parents, their prior probabilities can be specified as follows:

- *P(ToolDrop = yes) = 0.8 and P(ToolDrop = no) = 0.2*
- *P(Age = new) = 0.2, P(Age = medium) = 0.7, and P(Age = old) = 0.1*

The variable Damage has *3* states and *2* parents, each parent with 2 states. The conditional probability distribution of this variable can be shown as on Table 1. The table has *12* probability values *(3 2 2)*.

Table 1: *CPT* for *P (Damage| Age, ToolDrop)*. The *yes* and *no* in the first column represent the states of *Damage*.

ToolDrop	*yes*			*no*		
Age	***new***	***medium***	***old***	***new***	***medium***	***old***
yes	*0.2*	*0.4*	*0.9*	*0.01*	*0.5*	*0.4*
no	*0.8*	*0.6*	*0.1*	*0.99*	*0.5*	*0.6*

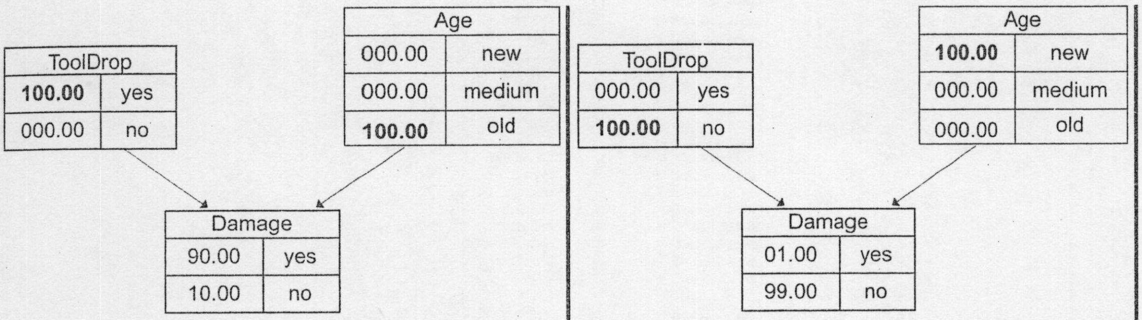
BNs give full representation of probability distributions over their variables. They can be conditioned on any subset of their variables, supporting any direction of reasoning. That means any variables may be query variables and any may be evidence variables. Whenever new information have arrived new beliefs can be calculated. We have shown that *P(ToolDrop = yes) = 0.8* and *P(Age = old) = 0.1*. Suppose it has been discovered that a tool is dropped on the material and the material is very old, then *P(ToolDrop = yes) = 1.0* and *P(Age = old) = 1.0*. These probabilities are shown in Figure 2 as percentages (*100.00* and *00.00*) on bold fonts. This kind of probabilities are sometimes referred as evidence or instantiation. In *BNs*, when new evidence arrives to some variables, the beliefs on other variables may be changed. This can be shown by carefully studying Figure 2. This process of conditioning on some variables, when observing the value of other variables, is known as probability propagation, inference, or belief updating.

BNs are powerful tools for knowledge representation and inference under uncertainties. Nevertheless, they are not considered very well as classifiers in *SHM* systems. Naïve-bayes is one of the *BNs* classifiers (e.g. C4.5) that surprisingly can outperform many sophisticated classifiers when working on data sets where the features are not strongly correlated.

Naïve-bayes Classifier in *SHM*

Naïve-bayes has a strong assumption that all variables in the network are independent of the classification variable (Figure 3). It is very easy to build a Naïve-bayes network structure, and it does not require a structure learning algorithm.

The amplitudes shown in Figure 4 represent voltage amplitudes of Lamb-waves produced and collected by *PZT* sensors and actuators mounted on the surface of quasi-isotropic graphite/epoxy laminates. The first specimen is a control unit (laminate without damage), and the rest of the specimen contain artificial damages. These damages are delimination, crack, and hole. The figure shows that sound waves behave differently when passing through the laminate without and with damage, and every damage produces different amplitudes. Amplitudes with many cases and different kind of damages can be used to learn the conditional probability tables of variables *(P(Amplitude | Damage))* in the network. Ultimately, the model can be used to predict the damages in laminated



(a) The evidence that the tool is (b) The evidence that the tool is not dropped *(ToolDrop = yes)* and the ma dropped *(ToolDrop = no)* and the material is old *(Age = old)*, increased our rial is new (Age = new), decreased our belief on the damage to 90%

(b) The evidence that the tool is not dropped *(ToolDrop= no)* and the material is new *(Age= new)*, decreased our belief on the damage to 1%.

Fig. 2: Changing of believes on BNs, when some evidence are entered

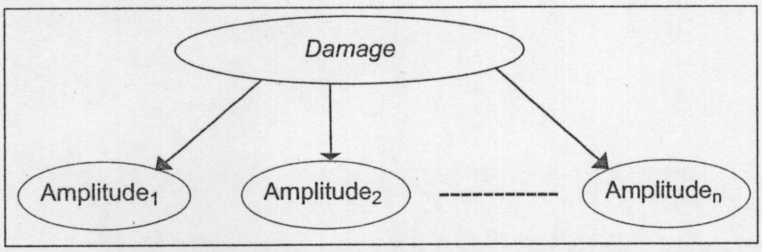


Fig. 3: A Naïve-bayes for damage detection by using some amplitudes of wave

composite materials with the highest posterior probability. The probabilities of the damages are determined by entering the new evidence obtained from the amplitudes of the new case to the network.

The amplitudes shown in Figure 4 were generated by using a constant interval of time (microseconds). For every laminate a set of *600* amplitudes were collected. If all of these amplitudes were used as variables on the damage detection model, the model would be overwhelmed, complicated, and its accuracy might slightly be decreased. Different techniques have been adopted for feature subset selections to decrease the size of the data and increase the accuracy. Some of these techniques extract the peaks of the amplitudes as feature subsets, but it is very difficult to be sure whether these peaks can be representative to the whole wave. The rest of the techniques have different kinds of limitations and disadvantages. So as to overcome some of these limitations and tackle some of these disadvantages, the *f*-folds feature subset selection algorithm has been developed.

f-folds Feature Subset Selection Algorithm

In Figure 4, the amplitudes formed using a constant interval of time (microseconds). A different data set might be acquired, if the interval value had been changed. If it had been assumed that the interval was increased *10* times more than the original one, then the original amplitudes would be divided into *60* folds (*10* amplitudes in each fold). In this case *10* different data sets would be formed each with *60* amplitudes. The amplitudes included in each set depend on the first amplitude selected from the first fold, if the first amplitude was the first to be included, then the first amplitudes in other folds would be included to the data set, if the second one was the first one to be included, then the seconds in all other folds would be included in the data set, and etc. This has been used as a base to formalize the k-folds feature subset selection algorithm shown below.

Algorithm 1 (k-folds feature subset selection algorithm)

Input:
Amps = amp_1, amp_2, , amp_n (*Amplitudes to be clustered*).
k (*number of clusters*), f (*number offolds*).

Outputs:
Means = m(c1), m(c2), , m(ck)
Maxs = max(c1), max(c2), , max(ck)
Mins = min(c1), min(c2), , min(ck)

procedure Clustering

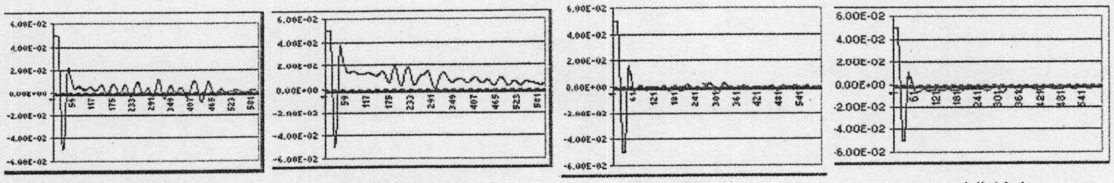
1. Divide Amps into ffolds (fold(1), fold(2), , fold(f)), = fold(f), fold(i) = $fold(j)_1$, $fold(j)_2$, $fold(j)_m$,

2. Create a new data set NewAmp = nAmp(1), nAmp(2), where A = fold(k)i A nAmp(i)1 i mand 1 k (the number of elements in each fold is m = n f).

3. Implement a clustering algorithm (e.g. k-means) on NewAmp, to return k clusters.

4. Return the mean, maximum, and minumum values of the clusters.

The input to thef-folds feature subset selection algorithm (Algorithm 1) is a set of n amplitudes (Amps = amp_1, amp_2, , amp_n). In step *1* the algorithm divides the data set intof folds. All folds contain the same number of m amplitudes, where $m = n f$. In step 2 the algorithm forms a new set of data containing m records by assigning the amplitudes with the same index in all



(a) Without damage (control) (b) Delimination (c) Crack (d) Hole

Fig. 4: Time trace of voltage amplitudes collected from graphite/epoxy laminates

folds to the data set as one record (e.g. the first amplitudes in all folds form the first record and so on). This creates the data set *NewAmp (nAmp(1), nAmp(2), , nAmp(m))*. The number of variables in each record is f (the number of folds). In step 3 the algorithm implements a clustering algorithm (e.g. k-means algorithm or *EM* algorithm) on *NewAmp* to divide their instances into k clusters. Since each record has f variables, the algorithm returns f mean values, f maximum values, and f minimum values of each cluster. These values would be considered as representatives to the clusters and when combined together they can replace the original data set. For example, if there are *100* instances in the cluster, only *3* instances are used (means, maximums, and minimums). The total number of the variables (t) in each damage type would be reduced to $3 \, x f x \, k$, when the means, maximums, and minimums of the clusters are considered. Finally, it will be reduced to $f x \, k$, if only the means are considered. The values of f and k must be determined by the user such that $t << n$, which believed to decrease the number of variables to a minimum that highly increase the accuracy of the model and simplify it.

Specimens and Data Sets

Two data sets collected and used in this paper. The data sets were collected using two sets of specimens. The first set used to test the f-folds feature sub-set selection algorithm and the second one was used to test the Naïve bayes classifier. The two sets of the specimens are described below.

Quasi-isotropic Laminates Data

The specimens used to test the f-folds feature sub-set selection algorithm were collected from Kessler [3]. The specimens were *25cm x 5cm* rectangular $[90/\pm45/0]_S$ quasi-isotropic laminates of the AS4/3501-6 graphite/epoxy system. Three *PZT* piezoceramic patches mounted on the surface of each specimen. The *PZT* cut into *2cm x 0.5cm* patches so that the longitudinal wave would be favored over the transverse one, and three patches used on each specimen to actuate and accurately measure the transmitted and reflected waves. The first channel, which served as the trigger for all of the channels, connected to the output channel and actuating *PZT*, two others connected to the sensing piezoceramic patches to the specimen to serve as a control channel in order to zero out drift. A few shapes of piezoceramic patches used to produce Lamb waves, and as expected waves propagated parallel to each edge, i.e. longitudinally and transversely for a rectangular patch and circumferentially from a circular piezo. Various types of damages were introduced to the specimens including, holes, fiber fracture, matrix cracking, and delamination. Lamb waves were propagated to the specimens by using *15* and *50KHz* frequencies.

Every one of the data sets was divided into different number of f folds ($3 \leq f \leq 10$) and a subsets of data were created from these folds for every data set. When the graphs of the subsets of every data set were plotted, there were many subsets showed similar shape of graphs as shown in Figure 5. This gives an indication that the subsets of the data set can be divided into clusters, where the means of these clusters can be used as representatives to these clusters for damage detection.

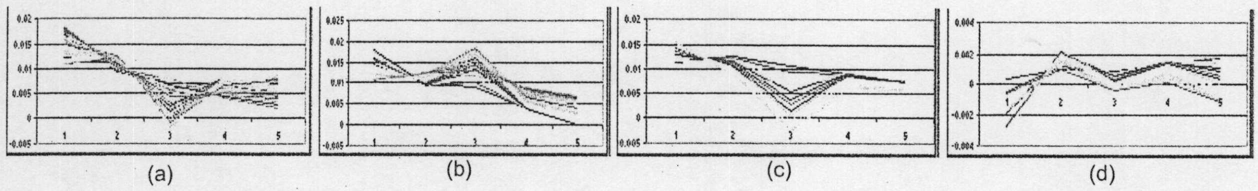
The Ball-bearing Data

The data set considered to test the Naïve bayes classifier is a set of vibration data from a type of ball bearing operating under different fault conditions. The ball bearing is of the type *6204* with a steel cage. The raw measurement data took the form of an acceleration signal recorded on the outer casing for the bearing in five states:

1. New ball bearing *(N)*.
2. Outer race completely broken *(O)*.
3. Broken cage with one loose element *(B)*.
4. Damaged cage, four loose elements *(D)*.
5. No evident damage, badly worn ball bearing *(ND)*.

The rotational frequency was *24.5 625Hz* and a tacho-signal was used for the measurement. The sampling frequency for the time data was *16384Hz* and the acquisition system was a *Bruuel* and *Kjaer* Spectrum analyzer. The points were recorded in 56 instances of 2048 samples, where 11 instances for case 1, 9 for case 2, 12 for each case of 3, 4 and 5.

Each signal was divided into overlapping 64-point intervals each offset by eight points from its predecessor. Each set was Fourier transformed and the magnitude of each spectral line was recorded. This yielded a sequence of 32-component vectors for classification [10].



(a) (b) (c) (d)

Fig. 5: The similarity of wave shapes of the data sets created by dividing wave of 600 amplitudes into 10 folds.

Results and Evaluations

The Naïve bayes classifier used in this study was implemented in the open-source machine learning package Weka [11]. The aim of the Naïve bayes classifier was to assign the data to five class labels. Before implementing the classification algorithm the data were preprocessed. Firstly, the data were prepared for the *f*-fold feature sub-set selection algorithm. Special softwares were developed for the feature subset selection algorithm. The programming language used was Java. Next the data were formatted using the Weka format. The clustering algorithm used for the *f*-fold algorithm was the *k*-means algorithm. Since the number of clusters must be specified for the *k*-means algorithm, different number of clusters were tested.

In this paper 8 folds with different number of clusters ranging from 2 to 8 were used. The features extracted for the classification were the maximum and mean values of the clusters amplitudes. It has been assumed that the maximum values represent the peaks of the amplitudes. Table 2 shows the confusion matrix of the classification result using 8folders and 4clusters. In the table, the number of correctly classified instances is 52 out of 56 (the accuracy is 946429%) and incorrectly classified instances is 4 (714%).

Table 2: The classification results when the number of clusters was 4

N	O	B	D	N	classifiedas
1	0	0	0	0	N
		0			O
0	0	11	1	0	B
0	0	0	1	1	D
0	0	0	0	12	N D

The classification algorithm has been implemented many times for the same number of clusters. Most of the time similar results were obtained; the best results were obtained when the number of clusters were 3 and 4. The best classification assignment obtained was 94.65% when the the number of clusters was 4. It can be seen from the table that 2 number of clusters slightly decreases the classification accuracy but using more than 4 clusters decreases the accuracy but does not affect it so much.

Figure 6 shows the classification accuracies obtained when using the mean values together with the peak values, the mean values alone, and the peak values alone. Obviously, the best accuracies were obtained when the mean values used together with the peak values for the classification. The mean values showed better results than the peak values. Generally, in all cases the best accuracies were obtained when the number of clusters was 3 and 4. The accuracies have not shown any change when using more than 4 clusters.

Number of Clusters	*Correctly #(%)*	*Incorrectly #(%)*
2	*49 (87.5)*	*7(12.5)*
3	*52 (92.86)*	*4 (7.15)*
4	*53 (94.65)*	*3 (5.35)*
5	*52 (92.86)*	*4 (7.14)*
6	*52 (92.86)*	*4 (7.14)*
7	*52 (92.86)*	*4 (7.14)*
8	*52 (92.86)*	*4 (7.14)*

Table 3: The tables compare the classification results when different number of clusters ranging from 2 to 8 were used (the number of folders was 8)

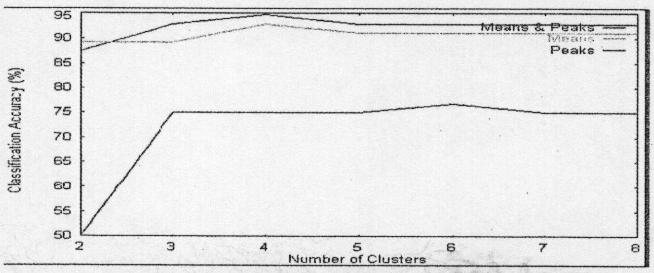


Fig. 6: The figure shows the improvement in the classification accuracy when using the mean and peak values together rather than using them separately

Conclusions

Bayesian networks in general and Naïve-bayes in particular are powerful formalisms for reasoning under uncertainty that can be employed as classification techniques for damage detection in composite materials. The *f*-fold feature subset selection algorithm shows the best classification results when the mean values used together with and the peak values rather than using them separately.

In this paper, the *f*-fold feature subset selection algorithm has been tested only for 8 folds; it is planned in the future that the algorithm will be tested for different number of folds. It is planned also to improve the algorithm so as to be capable to specify the number of folds and clusters. The results Naïve bayes classifier will also be compared with other classification algorithms (e.g Neural networks).

References

1. R. Miller and P. McIntire, "Nondestructive Testing Handbook", American Society for Nondestructive Testing, 5 (1987).

2. S. Hall, "The effective management and use of structural health data", Proceedings of the 2nd Inter-national Workshop on Structural Health Monitoring, (1999) 265-275.

3. S. Kessler, S. Spearing, M. Atalla, E. Cesnika, and C. Soutisb, "Structural health monitoring in composite materials using frequency response methods", Composites Part B, 33 (2002) 87-95.

4. Y. Shenfang, W. Lei, and P. Ge, "Neural network method based on a new damage signature for structural health monitoring", Thin-Walled Structures, 43 (2005) 553-563.

5. D. Chakraborty, "Artificial neural network based delamination prediction in laminated composites", Material and Design, 26 (2004) 1-7.

6. Z. Su and L. Ye, "Lamb wave-based quantitative identification of delamination in CF/EP composite structures using artificial neural algorithm", Composite Structures, 66 (2004) 627637.

7. F. V. Jensen, "Bayesian Networks and Decision Graphs", (Springer-Verlag New York, Inc., 2001). K. [9] B. Korb and A. E. Nickolson, "Bayesian Artificial Intelligence", (Chaman & HallCRC, 2004).

8. D. Heckerman, "A Tutorial on Learning With Bayesian Networks", (Microsoft Research Technical Report, 1995).

9. K. Worden and A. Lane, Damage identification using support vector machines, Smart Mater. Struct. 10 (2001) 540547.

10. I. Witten and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, (1999) 416, Morgan Kaufmann, San Francisco.