

EVIDENCE-BASED DAMAGE CLASSIFICATION FOR AN AIRCRAFT STRUCTURE

Worden, K¹, Manson, G¹ & Dencœux, T²

¹Dynamics Research Group, Department of Mechanical Engineering,
University of Sheffield, Mappin Street, Sheffield S1 3JD, UK.

²U.M.R. CNRS 6599 Heudiasyc, Université de Technologie de Compiègne,
BP 20529-F-60205 Compiègne Cedex, France.

k.worden@sheffield.ac.uk

ABSTRACT.

KEYWORDS: Dempster-Shafer Theory, Damage Location, Structural Health Monitoring.

1 INTRODUCTION

The recent past has seen considerable use of machine learning techniques for Structural Health Monitoring (SHM). The basic idea of the approach is to use data measured from undamaged and damaged structures in order to train a learning machine to assign a condition label to previously unseen data. The simplest problem of SHM is arguably *damage detection*. This is most easily carried out in the machine learning context by using a novelty detector [1]. Novelty detection involves the construction of a model of the normal condition of a system or structure, which can then be used in a hypothesis test on unseen data to establish whether the new data corresponds to normal condition or not. The advantage of the novelty detection approach is that it can be carried out using unsupervised learning, i.e. with only samples of undamaged data. If a more detailed diagnosis of a system is required, e.g. it is necessary to specify the type or location of damage in a structure, this can still be done using machine learning methods. For higher levels of diagnostics, algorithms based on classification or regression are applicable; however, these must be applied in a supervised learning context and examples of data from both the undamaged and damaged conditions can be used [2].

The most popular classifiers for damage location and quantification so far have been those based on Multi-Layer Perceptron (MLP) neural networks [3] (although there is growing popularity for classifiers based on the concepts of statistical learning theory - like support vector machines [2]). Training of MLP networks as classifiers is usually accomplished by using the *1 of M* strategy [3] which implicitly assumes a Bayesian probabilistic basis for the classification. While probability theory is only one (but arguably the most important) of a group of theories which can quantify and propagate uncertainty, other theories of uncertainty, perhaps with the exception of fuzzy set theory, have been largely unexplored in the context of damage identification. The object of this paper is to design a classifier for damage location based on the Dempster-Shafer theory of evidence [4]. To the knowledge of the authors, DS theory was first used in the context of SHM by researchers at El Paso as in [5]. The theory has also been used in the context of machine condition monitoring [6]. The reason for exploring the possibilities of DS theory is that it *extends* probability in a number of ways which are potentially exploitable in an SHM context. For the moment though, the current paper is looking only to demonstrate that the method is competitive with the probability-based MLP approach on an experimental case study of an aircraft wing. The DS classifier here is also implemented using a neural network structure [7].

The layout of the paper is as follows. Section Two provides a pedagogical introduction to DS theory and places its use in the context of SHM. Section Three briefly describes the neural network implementation of the DS Classifier. Section Four describes the case study discussed in this paper and the results of the analysis are given in Section Five. The paper ends with some discussion and conclusions.

2 DEMPSTER-SHAFER REASONING

Theory Dempster-Shafer Theory is a means of decision-fusion which is formulated in terms of probability-like measures but extends probability theory in a number of important respects. The basic idea, that of *belief* was introduced by Dempster in [8] and extended in Shafer's treatise [4].

The basic model is formulated in similar terms to probability. In the place of the sample space is the *frame of discernment* Θ , which is the set spanning the possible events for observation $A_i, i = 1, \dots, N$. On the basis of sensor evidence, each event or union of events is assigned a degree of probability mass or *Basic Belief Assignment* (BBA) m such that,

$$0 \leq m(A_i) \leq 1 \quad \forall A_i \subseteq \Theta \quad (1)$$

$$m(\phi) = 0 \quad (2)$$

$$\sum_{A_i \subseteq \Theta} m(A_i) = 1 \quad (3)$$

where ϕ is the empty set. (Note that normalisation, as in equation (3), is not always required in belief function theory. In particular, it is not the case in the Transferable Belief Model (TBM), an important variant developed by Smets [9].) The difference between this *evidential* theory and probability theory is that the total probability mass need not be exhausted in the assignments to individual events. There is allowed to be a degree of *uncertainty* or *ignorance*. This is sometimes denoted by a probability mass assignment to the *whole* frame of discernment $m(\Theta)$ or $m(A_1 \cup \dots \cup A_N)$.

The *belief* in an event B is denoted $\text{Bel}(B)$ and is defined by,

$$\text{Bel}(B) = \sum_{A_i \subseteq B} m(A_i) \quad (4)$$

and this is the total probability which is committed to the support of the proposition that B has occurred.

The *doubt* in the proposition B is denoted by $\text{Dou}(B)$ and is defined by

$$\text{Dou}(B) = \text{Bel}(\underline{\underline{B}}) \quad (5)$$

i.e. the doubt is the total support for the negation of the proposition B (negation is denoted by a double underline).

One of the fundamental differences between Dempster-Shafer and probability theory is that the belief and doubt *do not necessarily sum to unity* i.e. it is not certain that $B \cup \underline{\underline{B}}$ is true. This can be illustrated diagrammatically as follows:

The uncertainty Un in the proposition B is that portion of the probability mass which does not support B or its negation. If further evidence were provided, some of the uncertainty could move in support of B but the mass assigned to the doubt *cannot* move. This means that the possible belief in B is bounded above by the quantity $\text{Bel}(B) + \text{Un}(B) = 1 - \text{Dou}(B)$ and this quantity is termed the *plausibility* of B and denoted $\text{Pl}(B)$. The plausibility can also be defined by,

$$\text{Pl}(B) = \sum_{A_i \cap B \neq \phi} m(A_i) \quad (6)$$

The interpretation of some common instances of the uncertainty interval for a proposition B is as follows.

[0,0] B is impossible.

[1,1] B is certain.

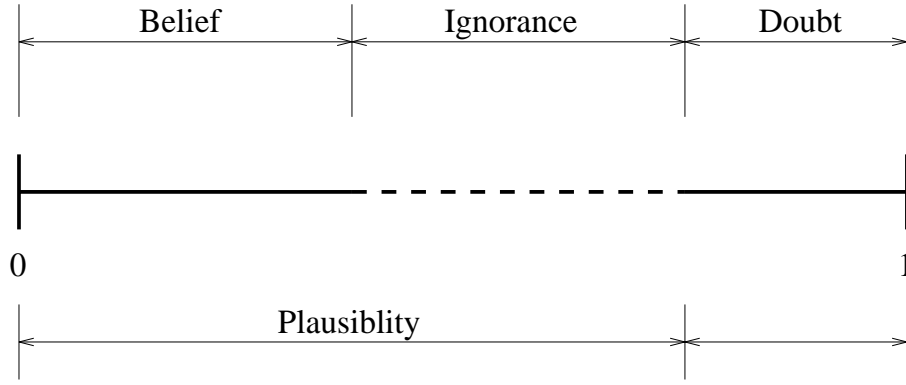


Figure 1: The Dempster-Shafer uncertainty interval.

[0.75,0.75] There is no uncertainty, B has a true probability of 0.75.

[0,1] There is total ignorance regarding B .

[0.25,1] B is plausible, there is no support for \underline{B} .

[0,0.75] \underline{B} is plausible, there is no support for B .

[0.25,0.75]. Both B and \underline{B} are plausible.

All this suffices to establish terminology, to explain how to compute belief functions and how to interpret the results. It does not provide a means of data fusion - that requires the use of *Dempster's combination rule*.

Suppose that one has two sensors 1 and 2. Basic probability assignments are possible on the basis of either sensor alone, denoted m_1 and m_2 . Belief functions Bel_1 and Bel_2 can be computed. Dempster's rule allows the calculation of an overall probability assignment m_+ and a corresponding overall belief function Bel_+ , where this *direct sum*,

$$\text{Bel}_+ = \text{Bel}_1 \oplus \text{Bel}_2 \quad (7)$$

is induced by m_+ . Suppose that sensor 1 makes assignments $m_1(A_i)$ to the proposition A_i (which can, and usually does, include the frame of discernment), and sensor 2 makes assignments $m_2(B_j)$, then Dempster's rule makes assignments as follows.

Consider a matrix with row entries labelled by i and column entries j , then the $(i, j)^{\text{th}}$ element of the matrix is an assignment of probability mass $m_1(A_i) \times m_2(B_j)$ to the proposition $A_i \cup B_j$.

In mathematical terms, Dempster's combination rule is expressed as

$$m_+(C) = \sum_{A_i \cap B_j = C} m_1(A_i)m_2(B_j) \quad (8)$$

and,

$$\text{Bel}_+(C) = \sum_{B \subseteq C} m_+(B) \quad (9)$$

Unfortunately things are not quite as straightforward as this. Problems arise in using Dempster's rule if the intersection between supported propositions A_i and B_j is empty. In this circumstance a non-zero mass assignment will be made to the empty set ϕ and this contradicts the basic definition of the mass assignment which demands $m_+(\phi) = 0$. In order to preserve this rule, Dempster's rule *must*

assign zero mass to non-overlapping propositions. However, if this is the case, probability mass is lost and the total mass assignment for m_+ will be less than unity, contradicting another rule for probability numbers. A valid mass assignment is obtained by *re-scaling* m_+ to take account of the lost mass. If the mass lost on non-overlapping propositions totals k , the remaining mass assignments should be re-scaled by a factor $K = 1/(1 - k)$. The combination rule (8) is modified to,

$$m_+(C) = K \sum_{A_i \cap B_j = C} m_1(A_i)m_2(B_j) \quad (10)$$

The differences between the Dempster-Shafer approach and the probabilistic are manifest. First of all, probabilistic - or rather Bayesian - approaches are unable to accommodate ignorance. All probability must be assigned to the set of propositions under consideration. Secondly, the Bayesian approach is unable to meaningfully assign probabilities to the union of propositions. If the uncertainty for all propositions is zero and the mass assigned to unions is zero, Dempster-Shafer is reduced to Bayesian probability reasoning.

There are other frameworks which seek to extend Bayesian methods in a similar manner to Dempster-Shafer such as the Generalised Evidence Processing (GEP) approach of [11] and those proposed in [12] and [13].

3 THE DS NEURAL NETWORK

The object of this section is to briefly describe the neural network implementation of the DS-based classifier. Much more detail can be found in the original reference [7].

The basic idea will be to assign one of M classes C_1, \dots, C_M (these form the frame of discernment), to a feature vector \underline{x} on the basis of a set of N training examples $\underline{x}_1, \dots, \underline{x}_N$. Suppose the vector \underline{x} is close to a training example \underline{x}_i with respect to an appropriate distance measure d ($d_i = \|\underline{x} - \underline{x}_i\|$). It is then appropriate that the class of the vector \underline{x}_i influences ones beliefs about the class of \underline{x} . One has evidence about the class of \underline{x}_i . The approach to the classification taken in [14] is to allocate belief to the event C_q (the class of \underline{x}_i , according to the distance $d_i(x)$).

$$m^i(C_q) = \alpha \phi_q(d_i) \quad (11)$$

where $0 < \alpha < 1$ is a constant and ϕ_q is an appropriate monotonically decreasing function. Each training vector close to \underline{x} will contribute some degree of belief. For each training vector, a degree of belief is also assigned to the whole frame of discernment Θ as follows,

$$m^i(\Theta) = 1 - \alpha \phi_q(d_i) \quad (12)$$

The function ϕ_q used here is the basic Gaussian,

$$\phi_q(d_i) = \exp(-\gamma_q(d_i)^2) \quad (13)$$

where γ_q is a positive constant associated with class q . To simplify matters, one confines the construction of the belief assignment for the vector \underline{x} to a sum of the beliefs induced by its nearest neighbours. The sum is computed using Dempster's combination rule as described in Section Two. Actually, a further simplification is made to speed up the processing. Rather than summing over the nearest neighbours from the whole training set in order to assign the belief, one sums over a set of *prototypes* constructed from the training set by a clustering algorithm. Each prototype \underline{p}_i is assigned a degree of membership to the class q denoted by u_q^i with the constraint $\sum_{q=1}^M u_q^i = 1$. These are used to compute the belief in the class q for \underline{x} given the distances d_i from the prototypes.

Although it is a gross simplification, the algorithm can be summed up as follows:

1. Construct the prototypes \underline{p}_i from the training data using a clustering algorithm.

2. Given a vector \underline{x} , compute the distances from the vector to the prototypes. Using the parameters d_i and u_q^i assign a degree of belief for each class q .
3. Use Dempster's combination rule to compute the total belief in each class from all the contributing prototypes.

The algorithm extends the probabilistic classifier by also making an assignment to the frame of discernment and this quantifies the degree of uncertainty of the classification. The reference [7] explains how the algorithm can be implemented in terms of a four-layer neural network. Unlike an MLP, the network is not a simple feed-forward structure.

In order to assign a class to the vector \underline{x} , one selects that with the largest overall belief assignment induced by the training data (There are other decision strategies as in [15]).

4 A DAMAGE LOCATION EXAMPLE

In the two papers [16, 17], methods of novelty detection were applied to the damage detection problem for experimental structures. In [16], the structure of interest was an idealised laboratory model of an aircraft wingbox, while in [17], the problem was to detect damage in an inspection panel of a Gnat aircraft wing. In both cases, a novelty detection approach was adopted based on the statistical method of outlier analysis. The next stage of the programme was to investigate the possibilities for damage location in the Gnat wing. Due to restrictions on actually damaging the structure, it was decided to investigate if a method could be developed to see which of a number of inspection panels had been removed. As there were a small number of distinct panels, the problem of damage location was cast as one of classification. Only a short summary of the study will be given here, the interested reader can refer to [18] for more details.

Due to the success of using novelty detectors for the damage detection problem, it was decided to attempt to extend this approach to see whether it could be used for the location problem. A network of sensors was used to establish a set of novelty detectors, the assumption being that each would be sensitive to different regions of the wing. Once the relevant features for each detector had been identified and extracted, a neural network was used to interpret the resulting set of novelty indices.

Test Set-up and Data Capture As described above, damage was simulated by the sequential removal of nine inspection panels on the starboard wing. Figure 2 shows a schematic of the wing and panels.

The area of the panels varied from about 0.008 m to 0.08 m with panels P3 and P6 the smallest. Measured transmissibilities were used as the basic features for construction of the novelty indices and were recorded in three groups, A, B and C as shown in Figure 2. Each group consisted of four sensors (a centrally placed reference transducer and three others). In each case, the transmissibility was the ratio of the acceleration spectrum at a receiver transducer divided by the acceleration spectrum at the reference transducer for the group. Only the transmissibilities directly across the plates were measured in this study. One 16-average transmissibility and 100 one-shot measurements were recorded across each of the nine panels for seven undamaged conditions (to increase robustness against variability) and the 18 damaged conditions (two repetitions for the removal of each of the nine panels).

Feature Selection and Novelty Detection The feature selection process for the novelty detectors was conducted by inspecting the transmissibility functions to find small regions of the frequency range of each which distinguished between damage conditions. An exhaustive visual classification of potential features as weak, fair or strong was made with the intention of only selecting fair or strong features, the details can be found in [18]. In order to simplify matters, only the group A transmissibilities were considered to construct features for detecting the removal of one of the group A panels; similarly for groups B and C.

Initially 44 candidate features were evaluated using outlier analysis. The best features were chosen according to their ability to correctly identify the 200 (per panel) damage condition features as outliers

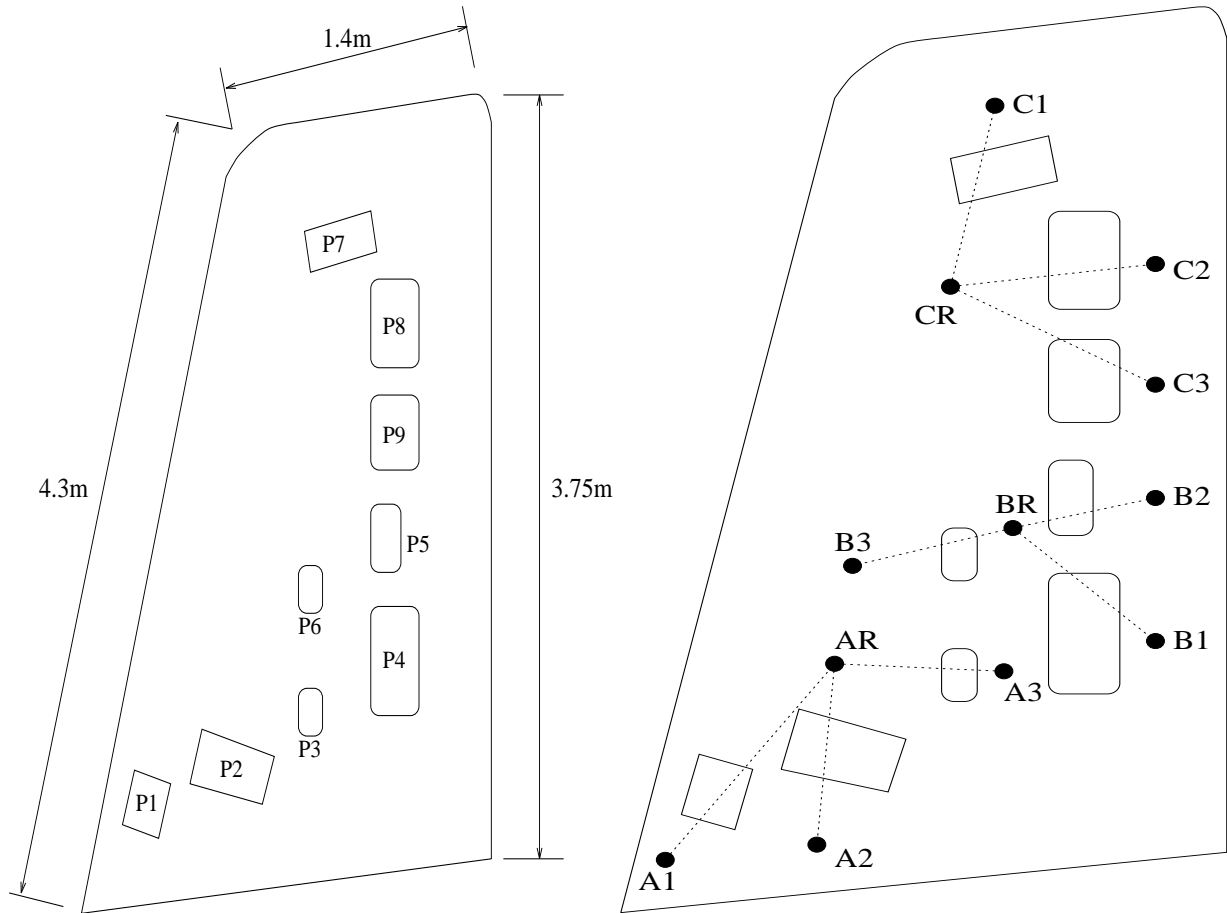


Figure 2: Schematic of the starboard wing inspection panels and transducer locations.

while correctly classifying as inliers, those features corresponding to the undamaged condition. Figure 3 shows the results of the outlier analysis for a feature that was clearly able to recognise removal of inspection panel 4. Once the 44 features had been selected by the empirical approach, a Genetic Algorithm was used to select the best 4 location features by optimising the classification error using an MLP as the classifier [19]. The reduction to 4 features was made to ensure that the MLP network used for comparison with the DS network later was unlikely to suffer from overtraining.

The data was divided into training, validation and testing sets in anticipation of presentation to the classifier. As there were 200 patterns for each damage class, the total number of patterns was 1800. These were divided evenly between the training, validation and testing sets, so (with a little wastage) each set received 594 patterns, comprising 66 representatives of each damage class. The plot in Figure 3 shows the discordancy (novelty index) values returned by the novelty detector over the whole set of damage states. The horizontal dashed lines in the figures are the thresholds for 99% confidence in identifying an outlier, they are calculated according to the Monte Carlo scheme described in [20]. The novelty detector substantially fires only for the removal of panel for which it has been trained. This was the case for most panels but there were exceptions (e.g. there were low sub-threshold discordancies for the smaller panels and some novelty detectors were sensitive to more than one damage type).

Note that there are now two layers of feature extraction. At the first level, certain ranges of the transmissibilities were selected for sensitivity to the various damage classes. These were used to construct novelty detectors for the classes. At the second level of extraction, the 9 indices themselves were used as features for the damage localisation problem. This depends critically on the fact that the various damage detectors are local in some sense, i.e., they do not all fire over all damage classes.

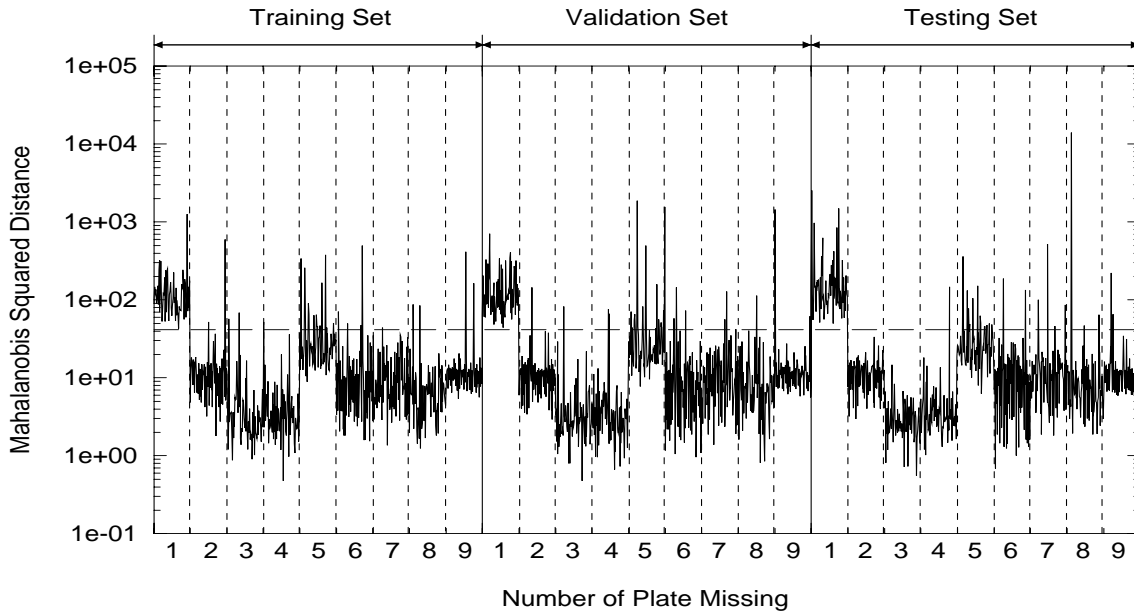


Figure 3: Outlier statistic for all damage states for the novelty detector trained to recognise panel 4 removal.

This was found to be true in this case.

5 NETWORKS FOR DAMAGE LOCATION

The final stage of the analysis was to produce a classifier based on the DS neural network algorithm which could serve as a damage location system. As with a standard MLP network, the specification of the DS network structure requires hyperparameters; in this case, the number of prototypes (analogous to the number of hidden units in the first layer of the network) and the starting values of the weights before training. These were computed by a cross-validation procedure as for the MLP [21]. Many neural networks were trained with the same training data but with differing numbers of prototypes and initial weights. Up to 30 prototypes were considered, and in each case 10 randomly chosen initial conditions were used. The best network was selected by observing which produced the minimum misclassification error on the validation set. The final judgement of the network capability was made by using the independent testing set.

The results for the presentation to the classifier are summarised in the confusion matrix given in Table 1. The best DS network used 29 prototypes. The probability of correct classification was 89.7%. There were 4 events associated with the frame of discernment, corresponding to probability mass of 0.007. (Note that there are other ways to implement rejection, [15].) This means that allowing for the fact that the network indicates when it has insufficient evidence to make a classification, the classification error is 9.6%.

The main source of confusion is in locating damage to the two smallest panels, 3 and 6, and of course this was anticipated.

In order to make a comparison with the standard approach, the algorithm chosen was a standard Multi-Layer Perceptron (MLP) neural network. The neural network was presented with 9 novelty indices at the input layer and required to predict the damage class at the output layer.

The procedure for training the neural network again followed the guidelines in [21]. The training set was used to establish weights, whilst the network structure and training time etc. were optimised using the validation set. The testing set was then presented to this optimised network to arrive at a final classification error. For the network structure, the input layer necessarily had four neurons, one for each novelty index, and the output layer had nine nodes, one for each class.

Prediction	1	2	3	4	5	6	7	8	9	Θ
True Class 1	54	5	5	0	0	0	2	0	0	0
True Class 2	0	63	0	0	2	0	0	0	0	1
True Class 3	6	1	56	2	0	0	0	0	0	1
True Class 4	5	0	1	55	0	3	0	2	0	0
True Class 5	0	0	0	0	65	0	0	1	0	0
True Class 6	2	2	2	4	0	54	1	0	0	1
True Class 7	0	1	1	0	0	0	61	2	1	0
True Class 8	0	0	1	0	1	0	0	62	1	1
True Class 9	0	0	0	0	0	0	0	3	63	0

Table 1: Confusion matrix for best DS network using 4 log features - testing set.

The training phase used the 1 of M strategy [3]. This approach is simple, each pattern class was associated with a unique network output; on presentation of a pattern during training, the network was required to produce a value of 1.0 at the output corresponding to the desired class and 0.0 at all other outputs.

It is known that MLP networks trained using a squared-error cost function with the 1 of M strategy for the desired outputs, actually estimate Bayesian posterior probabilities for the classes with which the outputs are associated [3]. This means that such a network actually implements a Bayesian decision rule if each pattern vector is identified with the class associated with the highest output.

The best neural network had 19 hidden units and resulted in a testing classification error of 0.118 i.e. 89.2% of the patterns were classified correctly. This means that the misclassification probability is of course 10.8%. The confusion matrix is given in Table 2. Again, the main errors were associated with the two small panels P3 and P6.

Prediction	1	2	3	4	5	6	7	8	9
True Class 1	61	1	0	0	0	0	1	0	0
True Class 2	0	63	0	0	3	0	0	0	0
True Class 3	1	0	48	8	0	5	7	2	0
True Class 4	0	1	3	56	0	2	0	4	0
True Class 5	0	0	0	0	66	0	0	0	0
True Class 6	4	1	0	9	0	52	0	0	0
True Class 7	1	0	0	0	0	0	59	5	1
True Class 8	1	0	0	0	1	0	1	59	4
True Class 9	0	0	0	0	0	0	0	5	61

Table 2: Confusion matrix for best MLP neural network using 4 log features - testing set.

6 CONCLUSIONS

The main conclusion here is that the Dempster-Shafer approach to classification implemented as a neural network gives comparable results to the standard analysis using a MLP neural network. The data used is from a full-scale experimental test on an aircraft wing and is therefore a stringent test of algorithms from the point of view of SHM. In fact the DS network shows a slight improvement, giving a classification probability of 89.7% compared to 89.2% for the MLP. One possible advantage of the DS approach is the fact that it can assign patterns to the frame of discernment and thus indicate to the analyst that there is insufficient evidence to make a classification. The effect is small here, with

only 0.7% of the probability mass assigned to Θ . Taking this effect into account, one can say that the probability of misclassification of the DS network is 9.6% compared to 10.8% for the MLP network. It is conceivable that situations could arise where this scale of difference could be important. Another advantage of using the DS approach might be to provide methodological diversity in order to use data fusion. In this case, the errors made by the DS network and the MLP network are similar; nonetheless there may be some advantage to fusion.

REFERENCES

- [1] Hayton P, Utete S, King D, King S, Anuzis P, Tarassenko L. Static and dynamic novelty detection methods for jet engine health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*. 2007;365:593-514.
- [2] Worden K, Manson G. The application of machine learning to structural health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*. 2007;365:515-537.
- [3] Bishop CM. *Neural Networks for Pattern Recognition*. Oxford University Press; 1998.
- [4] Shafer G. *A Mathematical Theory of Evidence*. Princeton University Press; 1976.
- [5] Osegueda RA, Seelam SR, Mulupuru B, Kreinovich V. Statistical and Dempster-Shafer techniques in testing structural integrity of aerospace structures. In *NDE for Health Monitoring and Diagnostics*, SPIE, San Diego, CA. Paper No.5047-18; 2003.
- [6] Yang B-S, Kim KJ. Application of Dempster-Shafer theory in fault diagnosis of induction motors using vibration and current signals. *Mechanical Systems and Signal Processing*. 2006;20:403-420.
- [7] Deneux T. A neural network classifier based on Dempster-Shafer theory. *IEEE Transactions on Systems, Man and Cybernetics*. 2000;30:131-150.
- [8] Dempster AP. Upper and lower probabilities induced by a multi-valued mapping. *Annals of Mathematical Statistics*. 1967;38:325-339.
- [9] Smets Ph. The Combination of Evidence in the Transferable Belief Model. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1990;12:447-458.
- [10] Shafer G, Logan R. Implementing Dempster's rule for hierarchical evidence. *Artificial Intelligence*. 1987;33:271-298.
- [11] Thomopoulos SCA. Theories in distributed decision fusion: comparison and generalisation. In: *Sensor Fusion III: 3D Perception and Recognition*. Proceedings of SPIE 1383; 1990.
- [12] Barnett JA. Computational methods for a mathematical theory of evidence. In: *Proceedings of the 7th International Joint Conference on Artificial Intelligence*; 1981. pp.868-875.
- [13] Yen J. A reasoning model based on an extended Dempster-Shafer theory. In: *Proceedings of the 5th AAAI-86 national Conference on Artificial Intelligence*; 1986. pp.125-131.
- [14] Deneux T. A k-nearest neighbor classification rule based on Dempster-Shafer theory. *IEEE Transactions on Systems, Man and Cybernetics* 1995;25:804-813.
- [15] Deneux T. Analysis of evidence-theoretic decision rules for pattern classification. *Pattern Recognition* 1997;30:1095-1107.

- [16] Worden K, Manson G, Allman DJ. Experimental validation of structural health monitoring methodology I: novelty detection on a laboratory structure. *Journal of Sound and Vibration* 2003;259:323-343.
- [17] Manson G, Worden K, Allman DJ. Experimental validation of structural health monitoring methodology II: novelty detection on an aircraft wing. *Journal of Sound and Vibration* 2003;259:345-363.
- [18] Manson G, Worden K, Allman DJ. Experimental validation of structural health monitoring methodology III: Damage location on an aircraft wing. *Journal of Sound and Vibration* 2003;259:365-385.
- [19] Worden K, Manson G, Hilson G, Pierce SG. Genetic optimisation of a neural damage locator. *Journal of Sound and Vibration*. 2007.
- [20] Worden K, Manson G, Fieller NR. Damage detection using outlier analysis. *Journal of Sound and Vibration*. 2000;229:647-667.
- [21] Tarassenko L. *A Guide to Neural Computing Applications*. Arnold; 1998.