On the Use of Hidden Markov Modeling and Time-frequency Features for Damage Classification in Composite Structures

WENFAN ZHOU,[1] NARAYAN KOVVALI,[1] WHITNEY REYNOLDS,[2]
ANTONIA PAPANDREOU-SUPPAPPOLA,[1,*] ADITI CHATTOPADHYAY[2] AND DOUGLAS COCHRAN[1]

[1]*Department of Electrical Engineering, Arizona State University, Tempe, Arizona, USA*

[2]*Department of Mechanical and Aerospace Engineering, Arizona State University, Tempe, Arizona, USA*

ABSTRACT: A novel approach based on hidden Markov models (HMMs) is proposed for damage classification in composite structures. Time-frequency damage features are first extracted from the measured signals using the matching pursuit decomposition algorithm. The features are then incorporated as observation sequences to be modeled statistically by the HMMs. Once built, the HMMs are integrated very efficiently into a Bayesian framework for the classification of structural damage. Both discrete and continuous observation density HMMs are considered; continuous HMMs are shown to yield better accuracy, but at the cost of added computational complexity. A decision fusion procedure is employed to combine the local classification results at each sensor, significantly enhancing the overall classification performance. The utility of the proposed technique is demonstrated by its application to the classification of delamination damage, impact damage, and progressive tensile damage in laminated composites.

Key Words: Integrated vehicle health management, composite structures, damage detection, damage classification, matching pursuit decomposition, hidden Markov model, sensor fusion.

INTRODUCTION

THE development of structural health monitoring (SHM) and integrated vehicle health management (IVHM) systems is currently drawing much attention and research interest in the aerospace industry. The goal is to provide real-time and life-cycle structural or vehicle health information for reducing maintenance costs and increasing in-service capacity. A successful system for health monitoring necessitates integration of solutions to multiple tasks from different areas including multi-scale analysis, sensor placement optimization, damage detection, and classification (diagnosis), damage prognosis, and structural remaining life prediction (Staszewski et al., 2003; Farrar and Lieven, 2007; Farrar and Worden, 2007). Among these tasks, the development of effective damage detection and classification schemes is a major component of the health management framework.

The process of damage detection and identification in complex materials and structures strives to provide effective indicators on the presence, type, location, size or severity of damage for a structure of interest. In recent years, a broad range of methodologies have been developed for the detection, analysis, and classification of damage in metals and composites (Staszewski et al., 2003). One example is the Lamb wave method (Lee and Staszewski, 2003a; Lee and Staszewski, 2003b; Giurgiutiu, 2005), which utilizes guided Lamb waves for damage detection. Practical application of the Lamb wave method is challenging due to its requirement of a large number of transducers (for large structures), complex data interpretation, and sensitivity to parameters other than structural damage (Aircraft Health Monitoring, 2006). This detection method performs well, however, when coupled with appropriate signal processing techniques (Staszewski et al., 1997; Niethammer et al., 2001; Paget et al., 2003). The signal processing techniques effectively extract and interpret maximum discriminatory information about the damage wave physics from available data. The general procedure comprises of data preprocessing, feature extraction and information condensation, pattern recognition and classification, data fusion, and decision making (Staszewski, et al., 2003; Farrar anol Worden, 2007).

The purpose of feature extraction and selection is to generate a compact representation of the measured data, which captures the prominently distinguishable characteristics of data from different damage classes. The data collected using sensors comprises measurement of Lamb waves propagating through the structure.

*Author to whom correspondence should be addressed.
E-mail: papandreou@asu.edu
Figures 1–6 and 11 appear in color online: http://jim.sagepub.com

1045-389X/09/11 1271–18 $10.00/0 DOI: 10.1177/1045389X08099968

Recent developments in signal processing techniques allow for more powerful structural monitoring systems by providing sophisticated feature extraction methods and their efficient incorporation into the monitoring procedure. Popular feature extraction and data analysis tools include Fourier analysis (Gelman et al., 2004a; Abdel-Galil et al., 2005; Loewke et al., 2005), autoregressive (AR) models (Sohn and Farrar, 2000; Sohn et al., 2001), and wavelet transforms (Jeong and Jang, 2000; Wang and Gao, 2003; Sun and Chang, 2004). However, these techniques are not well-matched to the time-varying spectral nature of structural data (Victorov, 1967). Material wave-physics can be characterized by dispersive or time-varying phenomena, with structures often behaving akin to wave-guides. An appropriate model, therefore, is one which can allow analysis of signals with spectral content that varies with time, unlike the Fourier transform and AR models. The multi-resolution wavelet transform is capable of revealing temporal and spectral patterns in signals. However, it has the drawback of low resolution in high-frequency regions (Papandreou-Suppappola, 2002), limiting its applicability for detecting small-scale damage where high-frequency modes are required. The wavelet packet decomposition used in (Sun and Chang, 2002; Eren and Devaney, 2004; Sun and Chang, 2004), on the other hand, adopts redundant basis functions to achieve arbitrary but nonadaptive time-frequency (TF) resolution.

The natural tool to employ for fully exploiting the time-varying spectral signal structure is TF processing (Mallat, 1998; Papandreo-Suppappola, 2002). The matching pursuit decomposition (MPD) (Mallat and Zhang, 1993; Mallat, 1998) is a technique that yields a compact representation for signals in terms of basis functions selected adaptively so as to best match signal components of interest. When the basis set (dictionary) comprises of TF atoms, the MPD can be used to obtain adaptive TF representations (TFRs) (Mallat and Zhang, 1993; Papandreou-Suppappola, 2002). The MPD has been used as a feature extraction tool in acoustic signal classification (Ebenezer et al., 2004) and target identification (Bharadwaj et al., 1999; Runkle et al., 1999b). Recently, the MPD has been employed with dictionaries composed of either highly localized TF shifted and scaled Gaussian atoms or TF shifted real measured data for the classification of damage in metallic structures (Das et al., 2005; Kovvali et al., 2007; Zhou et al., 2007a; Zhou et al., 2007b; Chakraborty et al., 2008; Channels et al., 2008).

In addition to their use for analysis, the extracted TF features can also be incorporated in pattern recognition algorithms for damage classification. Although deterministic classification methods have been investigated in the literature (e.g., Jeong and Jang, 2000; Eren and Devaney, 2004; Loewke et al., 2005), their properties do not allow exploration of the uncertainty prevalent in the structural characterization problem. Uncertainty is an important feature of the damage classification problem in real-world health management systems due to variability in structure geometry, material properties, temperature and other environmental effects, sensor characteristics and measurement inaccuracy, and insufficient knowledge about the process of damage nucleation and evolution. In order to deal with the magnitude and complexity of such variations, a stochastic classification approach must be considered which can capture the statistical properties characterizing the underlying physical process, accounting for uncertainties and thereby achieving robustness to inherent variability. Statistical models also facilitate the integration of both available and unavailable information in a consistent and effective manner.

Some statistical techniques used in the literature include Bayesian probabilistic inference methods (Beck et al., 1999; Sohn and Law, 2000), statistical process control (Sohn and Farrar, 2000; Sohn et al., 2000; Fugate et al., 2001; Sun and Chang, 2004), outlier analysis (Sohn et al., 2001; Park et al., 2005; Sohn et al., 2005; Kim et al., 2006), wavelet packets (Sun and Chang, 2004), Markov chain Monte Carlo (MCMC) techniques (Yuen et al., 2004), artificial neural networks (Lee et al., 2006), and support vector machines (SVMs) (Das et al., 2007). For example, outlier analysis is used for novelty detection to provide indicators of the presence of unseen damage, with the algorithm trained using undamaged data only. In (Park et al., 2005; Sohn et al., 2005), extreme value statistics (EVS) were incorporated in outlier analysis to model the tails or outliers of data distributions. In Yuen et al. (2004), MCMC techniques were used to tackle numerical integration for obtaining Bayesian updates of dynamic models for the estimation of damage probabilities. In Das et al. (2007), a one-class SVM technique combined with statistical processing was adopted to automatically detect and classify four types of damages. Another prominent stochastic classification technique is based on hidden Markov models (HMMs) (Rabiner and Juang, 1986; Rabiner, 1989). The HMM is a powerful tool that has been successfully applied in many real-world problems, including speech recognition (Rabiner, 1989), handwritten word recognition (Mohamed and Gader, 2000), image classification (Li et al., 2000), target classification (Bharadwaj et al., 1999; Runkle et al., 1999a; Runkle et al., 1999b), and damage prognosis (Rammohan and Taha, 2005). For example, the HMM was used to model structural damage prognosis and predict remaining service life based on simulated data, with the performance shown for a pre-stressed concrete bridge (Rammohan and Taha, 2005). In Gupta et al. (in press), HMMs were constructed on the corresponding symbolic sequences of the time series data for anomaly detection in mechanical vibration systems.

In this article, the use of HMMs for modeling TF features extracted from composite data using the MPD is proposed for the classification of structural and material damage. The data collected using sensors comprises measurement of Lamb waves propagating through the composite structures. The HMM defines underlying nonobservable physical states in the structure and works by modeling the temporal transitions between these underlying states with a Markov random process in conjunction with the observation statistics. Two types of HMMs are considered: discrete and continuous. The discrete HMM treats the MPD features of the damage data as discrete codes, while continuous HMM models them using a Gaussian mixture model (GMM). Data from each structural condition (damage class) is modeled with a separate HMM. Experimentally collected data is used to construct and verify the HMMs. Once built, the HMMs are integrated very efficiently into a Bayesian framework for the classification of structural damage. Often, multiple sensing devices are deployed (for example, of different types, or at different physical locations) with the goal of improving detection performance by providing additional information. In this article, a sensor fusion procedure is described that can be employed to efficiently integrate the information gathered from all available sensors. The performance of the proposed classification algorithm is demonstrated for delamination, impact damage, and progressive tensile damage in laminated composites.

The remainder of this article is organized as follows. 'Theoretical Framework' reviews the algorithms used by the HMM classifier. The section 'HMM-Based Damage Classification Algorithm' discusses the implementation to real damage classification environments, and the section 'Sensor Fusion' develops the sensor fusion procedure for the HMM. 'Damage Classification in Composite Structures' presents the classifier performance for three types of damage in laminated composites.

THEORETICAL FRAMEWORK

In this section, we describe in brief the core theory behind the proposed classifier. For more details about these techniques the reader is referred to the literature (Rabiner and Juang, 1986; Rabiner 1989; Mallat and Zhang, 1993; Mallat, 1998; Duda et al., 2001; MacKay, 2003).

Matching Pursuit Decomposition

The MPD (Mallat and Zhang, 1993; Mallat, 1998) yields a representation of signals in terms of basis functions chosen from a custom-built dictionary. The algorithm iteratively decomposes a given signal $s(t) \in \mathbf{L}^2(\mathbb{R})$ as

$$s(t) = \sum_{l=0}^{L-1} \rho_l g_{\eta_l}(t) + r_L(t),$$

where $r_L(t)$ denotes the residue after L MPD iterations (with $r_0(t) \equiv s(t)$). The expansion coefficient ρ_l, given by

$$\rho_l = \int_{-\infty}^{\infty} r_l(t)\, g_{\eta_l}^*(t)\, \mathrm{d}t, \quad l = 0, \ldots, L-1, \tag{2.1}$$

is the inner-product between the residue $r_l(t)$ and the dictionary atom $g_{\eta_l}(t) \in \mathbf{L}^2(\mathbb{R})$ that depends on the parameter set η_l. This dictionary atom is selected from a dictionary $\mathcal{D} = \{g_\eta(t)\}_{\eta \in \Xi}$ so as to maximize the magnitude of the inner-product in (2.1) at the lth iteration:

$$g_{\eta_l}(t) = \underset{g_\eta \in \mathcal{D}}{\operatorname{argmax}} \left| \int_{-\infty}^{\infty} r_l(t)\, g_\eta^*(t)\, \mathrm{d}t \right|. \tag{2.2}$$

When truncated to L terms, the MPD yields a compact and unique representation

$$s_L(t) = \sum_{l=0}^{L-1} \rho_l g_{\eta_l}(t) \approx s(t)$$

in terms of the selected family of basis functions (Mallat and Zhang, 1993; Mallat, 1998). The truncation limit L is usually chosen such that the energy of the residue after L iterations is smaller than some predefined value. When L is sufficiently large, the MPD extracts the most important signal components of interest while effectively filtering out unwanted signal components such as noise.

The MPD dictionary $\mathcal{D}$ does not need to be orthonormal but is required to be complete (Mallat and Zhang, 1993; Mallat, 1998). It consists of highly localized Gaussian atoms, which are TF shifted and scaled versions of a basic Gaussian atom $g(t) = Ce^{-t^2/2}$. The dictionary atoms are given by

$$g_\eta(t) = C_\eta\, e^{-\kappa^2 (t-\tau)^2} \cos(2\pi\nu t), \tag{2.3}$$

where τ is the time-shift, ν is the frequency-shift, κ is the (positive) scaling parameter, and C_η is a normalizing constant for unit energy. Each Gaussian atom $g_\eta(t) \in \mathcal{D}$ is thus fully characterized by the parameter set $\eta = \{\tau, \nu, \kappa\}$ from $\eta \in \Xi = \mathbb{R}^2 \times \mathbb{R}^+$.

The Gaussian-windowed harmonics used as atoms have the advantage of good TF localization properties, resulting in decompositions with good resolution in both time and frequency. This is because, as can be shown by the uncertainty principle (Papandreou-Suppappola, 2002),

they are the functions of choice when the time-bandwidth product is to be minimized. In addition, these Gaussian atoms have computational benefits derived from the availability of closed-form analytical expressions for some of their transformations such as the Fourier transform and the Wigner distribution TF representation (TFR) (Papandreou-Suppappola, 2002). The TF structure of the dictionary atoms also makes possible the efficient evaluation of the inner products in (2.2) using the fast Fourier transform (FFT), greatly reducing the computational complexity of the algorithm (see Mallat and Zhang, 1993 for details).

Each dictionary atom is a Gaussian-windowed harmonic of the form shown in Equation (2.3), and it is defined uniquely by the combination of time shift τ, frequency shift ν, and scaling factor κ. The size of the dictionary is then determined by the range and increment of the parameters τ, ν, and κ. These are chosen such that these atoms cover the entire region of interest in the TF plane, and the computational effort involved in the MPD remains reasonable (the larger the dictionary, the better the approximation but the slower the decomposition). Since the TF structure of the received signals can be different in different applications, the dictionary design is in general different in each case.

Figure 1 shows an example of the MPD of the sum of three Gaussian signals in the TF plane. Figure 1(a) shows the original signal in the time-domain, and Figure 1(b) shows MPD-TFR of the signals that corresponds to the weighted sum of the Wigner distribution of each extracted Gaussian atom in the TF plane (Mallat and Zhang, 1993).

The extracted atom at the lth iteration provides a vector of four parameters, consisting of a time shift τ_l, frequency shift ν_l, scale κ_l, and expansion coefficient ρ_l of the MPD representation. These vectors will be used as the feature vectors needed by the HMM algorithm described in the next section.

Hidden Markov Modeling

Hidden Markov modeling (Rabiner and Juang, 1986; Rabiner, 1989) is a stochastic approach used to model sequential data as a Markov process (Rabiner, 1989) with unknown parameters that are determined from observed data. The estimated model parameters act as unique identifiers for that model and can be used, for example, in pattern recognition applications.

The hidden Markov model (HMM) defines a probability distribution over a sequence $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ of observation vectors[1] $\mathbf{y}_n$, $n = 1, \ldots, N$, by defining the unobserved (hidden) discrete state vector $\mathbf{x} = [x_1\, x_2 \ldots x_N]^T$. The model imposes Markov dynamics on the hidden states x_n, as well as independence of the observation vector $\mathbf{y}_n$ from all other variables given x_n, $n = 1, \ldots, N$. Considering J distinct states, the state variable x_n assumes values from the alphabet $\{1, \ldots, J\}$. The parameters of the model are: (a) the $J \times 1$ initial state distribution vector $\boldsymbol{\pi}$ whose jth element π_j is the probability (Pr) that the initial state x_1 is equal to j, i.e., $\pi_j = \Pr(x_1 = j)$; (b) the $J \times J$ state-transition matrix $\mathbf{A}$ whose (i,j)th element is $a_{ij} = \Pr(x_{n+1} = j | x_n = i)$, $i, j = 1, \ldots, J$; and (c) the state-dependent observation density function or likelihood $\mathcal{B}$ whose jth element is $b_j(\mathbf{y}_n) = p(\mathbf{y}_n | x_n = j)$, $j = 1, \ldots, J$. Together, we denote the parameters for the model as $\Theta = \{\boldsymbol{\pi}, \mathbf{A}, \mathcal{B}\}$. Since π_j and a_{ij} are probabilities, they are positive and lie within the range $[0, 1]$. Moreover, by definition, they must satisfy the stochastic constraints $\sum_{j=1}^{J} \pi_j = 1$ and $\sum_{j=1}^{J} a_{ij} = 1$, $i = 1, \ldots, J$. An example of a $J = 3$ state HMM is depicted in Figure 2.

In a *discrete* HMM, the observation vectors $\mathbf{y}_n$, $n = 1, \ldots, N$ are discrete (denoted using scalar y_n) and can only take one out of K possible values from a finite set $\{q_1, \ldots, q_K\}$. In this case, the state-dependent observation density $\mathcal{B}$ reduces to a $J \times K$ matrix $\mathbf{B}$ whose (j,k)th element is $b_{jk} = \Pr(y_n = q_k | x_n = j)$, $j = 1, \ldots, J$ and $k = 1, \ldots, K$, and $\sum_{k=1}^{K} b_{jk} = 1$, $j = 1, \ldots, J$. In a *continuous* HMM, the observation vectors $\mathbf{y}_n$ are continuous and $\mathcal{B}$ is often modeled using a GMM:

$$b_j(\mathbf{y}_n) = p(\mathbf{y}_n | x_n = j) = \sum_{m=1}^{M} c_{jm} \mathcal{N}(\mathbf{y}_n; \boldsymbol{\mu}_{jm}, \boldsymbol{\Sigma}_{jm}), \quad j = 1, \ldots, J, \tag{2.4}$$

where M is the number of mixture components, c_{jm} is the weighting coefficient, and $\mathcal{N}$ represents a Gaussian distribution with mean vector $\boldsymbol{\mu}_{jm}$ and covariance matrix $\boldsymbol{\Sigma}_{jm}$ of the mth mixture component in state j. The stochastic constraints impose $c_{jm} \in [0, 1]$ and $\sum_{m=1}^{M} c_{jm} = 1$, $j = 1, \ldots, J$.

With the model parameters defined, the HMM algorithm consists of two major steps. The first is *model fitting*, which is used to learn model parameters given some training data. The second is *model comparison*, which is used to determine which model is more plausible given some test data. For the model fitting level of inference, given a training observation sequence[2] Y, a maximum-likelihood (ML) estimate of the parameters

$$\Theta_{\mathrm{ML}} = \mathrm{argmax}_{\Theta}\{\log p(Y|\Theta)\} \tag{2.5}$$

[1]For notational consistency throughout the article, a sequence $Y = \{\ldots\}$ is denoted by a capital letter and its elements (scalar or vector) are within curly brackets; a (column) vector $\mathbf{y} = [\ldots]^T$ (where T denotes vector transpose) is denoted by a lower case bolded letter and its elements (scalars) are within square brackets; a matrix $\mathbf{A}$ is denoted by a capital bolded letter; and a probability density function $\mathcal{B}$ is denoted by a capital calligraphic letter.

[2]For simplicity, we only consider a single observation sequence; the treatment is easily extended to a data set comprising of multiple independent and identically distributed observation sequences.

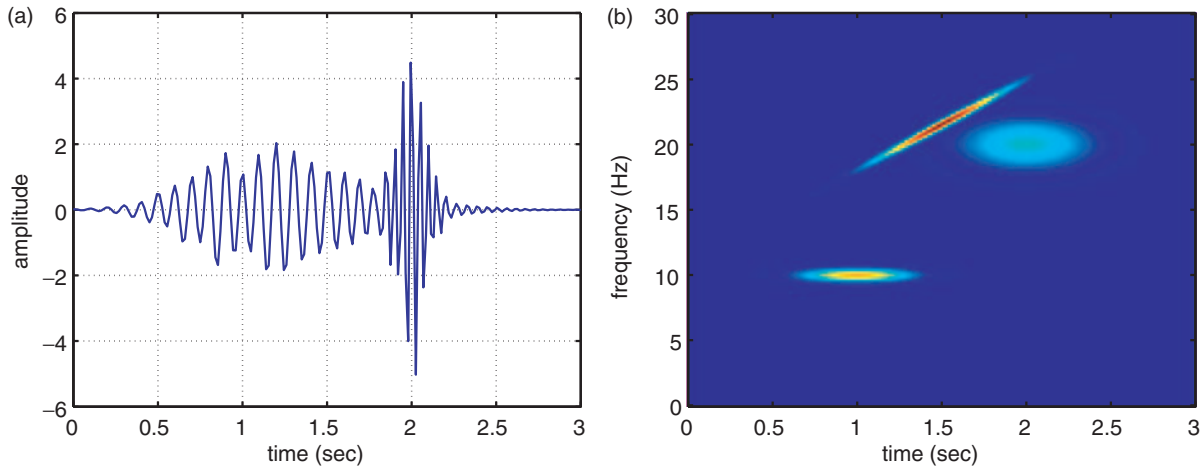

Figure 1. *Sum of three Gaussian functions: (a) time-domain representation, and (b) TF domain MPD-TFR.*

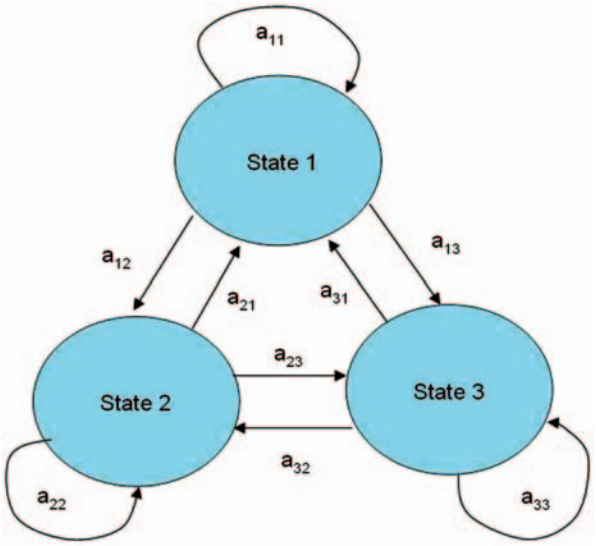

Figure 2. *Depiction of a 3-state HMM.*

is computed using the Baum-Welch algorithm (Rabiner and Juang, 1986; Rabiner, 1989). Here $p(Y|\Theta)$ is the likelihood of the observation sequence Y given the HMM with the parameter set Θ. This is a special case of the expectation-maximization (EM) algorithm (Dempster et al., 1977) which iteratively maximizes the likelihood of the training data. At the lth iteration, the E-step of the re-estimation procedure infers the posterior distribution over the hidden states given a current parameter setting Θ^l, and the M-step computes a new estimate $\Theta^{(l+1)}$ by maximizing the log-likelihood with respect to Θ using the statistics learned in the E-step. The combined EM step can be written as

$$\Theta^{(l+1)} = \underset{\Theta}{\operatorname{argmax}} \left\{ \sum_{\mathbf{x}} p(\mathbf{x}|Y, \Theta^{(l)}) \log p(\mathbf{x}, Y|\Theta) \right\}, \quad (2.6)$$

where the sum is over all possible state vectors $\mathbf{x}$. The process is repeated in an iterative fashion, and the algorithm is guaranteed to converge to a local maximum of the log-likelihood function (Dempster et al., 1977; Duda et al., 2001; MacKay, 2003). In practice, the iterations are terminated after the relative change in log-likelihood drops below a certain threshold.

In the model comparison, the predictive likelihood of a test observation sequence Y^{test} is computed as

$$\begin{aligned} p(Y^{\text{test}}|\Theta_{\text{ML}}) &= \sum_{\mathbf{x}} \mathrm{p}(\mathrm{Y}^{\text{test}}, \mathbf{x}|_{\text{ML}}) \\ &= \sum_{\mathbf{x}} \left[\pi_{x_1} \left(\prod_{n=1}^{N-1} a_{x_n x_{n+1}} \right) \left(\prod_{n=1}^{N} b_{x_n}(\mathbf{y}_n^{\text{test}}) \right) \right], \end{aligned} \quad (2.7)$$

where the summations are over all possible state vectors $\mathbf{x}$.

Note that a direct evaluation of the summations in (2.6) and (2.7) is very intensive. For example, the calculation of the predictive likelihood directly from (2.7) requires $2N \times J^N$ operations. Note, however, that the computations can be carried out in a very efficient manner using the forward-backward procedure (Rabiner, 1989), reducing the complexity to J^2N. In order to use this procedure, the forward variable is defined as $\alpha_n(j) = p(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n, x_n = j|\Theta)$, which corresponds to the probability of the (forward) partial observation sequence $\{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ (until time n) and state $x_n = j$, given the model Θ. The backward variable defined as $\beta_n(j) = p(\mathbf{y}_{n+1}, \mathbf{y}_{n+2}, \ldots, \mathbf{y}_N|x_n = j, \Theta)$ is the probability of the (backward) partial observation sequence $\{\mathbf{y}_{n+1}, \ldots, \mathbf{y}_N\}$ (from time $n+1$ to N), given state $x_n = j$ and model Θ. The Markov assumption on the state dynamics and the independence of the observations allows these variables to be iteratively and efficiently computed as:

$$\alpha_1(j) = \pi_j b_j(\mathbf{y}_1), \quad \alpha_{n+1}(j) = \left[\sum_{i=1}^{J} \alpha_n(i) a_{ji} \right] b_j(\mathbf{y}_{n+1}),$$

$$\beta_N(j) = 1, \quad \beta_n(j) = \sum_{i=1}^{J} a_{ji} b_i(\mathbf{y}_{n+1}) \beta_{n+1}(i).$$

The re-estimation formulas in the lth iteration for the HMM parameters are (Rabiner, 1989):

$$\pi_j^{(l+1)} = \frac{\alpha_1^{(l)}(j)\beta_1^{(l)}(j)}{\sum_{i=1}^{J}\alpha_1^{(l)}(i)\beta_1^{(l)}(i)}, \tag{2.8}$$

$$a_{ij}^{(l+1)} = \frac{\sum_{n=1}^{N-1}\alpha_n^{(l)}(i)a_{ij}^{(l)}b_j^{(l)}(\mathbf{y}_{n+1})\beta_{n+1}^{(l)}(j)}{\sum_{n=1}^{N-1}\alpha_n^{(l)}(i)\beta_n^{(l)}(i)}. \tag{2.9}$$

In the discrete HMM,

$$b_{jk}^{(l+1)} = \frac{\sum_{n=1_{s.t.\mathbf{y}_n=q_k}}^{N}\alpha_n^{(l)}(j)\beta_n^{(l)}(j)}{\sum_{n=1}^{N}\alpha_n^{(l)}(j)\beta_n^{(l)}(j)}, \tag{2.10}$$

where $\alpha_n^{(l)}(j)$ and $\beta_n^{(l)}(j)$ are calculated given the lth re-estimated model $\Theta^{(l)}$. Note that the summation in the numerator of (2.10) is over those n for which $\mathbf{y}_n = q_k$. As the observation density $\mathcal{B}$ is modeled using a GMM as in (2.4) for continuous HMMs, the update equations for the observation density parameters assume the form (Rabiner, 1989):

$$c_{jm}^{(l+1)} = \frac{\sum_{n=1}^{N}\gamma_n^{(l)}(j,m)}{\sum_{n=1}^{N}\sum_{\hat{m}=1}^{M}\gamma_n^{(l)}(j,\hat{m})}, \tag{2.11a}$$

$$\boldsymbol{\mu}_{jm}^{(l+1)} = \frac{\sum_{n=1}^{N}\gamma_n^{(l)}(j,m)\mathbf{y}_n}{\sum_{n=1}^{N}\gamma_n^{(l)}(j,m)}, \tag{2.11b}$$

$$\boldsymbol{\Sigma}_{jm}^{(l+1)} = \frac{\sum_{n=1}^{N}\gamma_n^{(l)}(j,m)(\mathbf{y}_n-\boldsymbol{\mu}_{jm}^{(l)})(\mathbf{y}_n-\boldsymbol{\mu}_{jm}^{(l)})^T}{\sum_{n=1}^{N}\gamma_n^{(l)}(j,m)}, \tag{2.11c}$$

where $\gamma_n^{(l)}(j,m)$ is defined as the probability that state $x_n = j$ and that the mth Gaussian mixture accounts for the observation $\mathbf{y}_n$ given the lth re-estimated model $\Theta^{(l)}$. It is given by

$$\gamma_n^{(l)}(j,m) = \frac{\alpha_n^{(l)}(j)\beta_n^{(l)}(j)}{\sum_{\hat{j}=1}^{J}\alpha_n^{(l)}(\hat{j})\beta_n^{(l)}(\hat{j})}\ \frac{c_{jm}^{(l)}\mathcal{N}(\mathbf{y}_n;\boldsymbol{\mu}_{jm}^{(l)},\boldsymbol{\Sigma}_{jm}^{(l)})}{\sum_{\hat{m}=1}^{M}c_{j\hat{m}}^{(l)}\mathcal{N}(\mathbf{y}_n;\boldsymbol{\mu}_{j\hat{m}}^{(l)},\boldsymbol{\Sigma}_{j\hat{m}}^{(l)})}. \tag{2.12}$$

The predictive likelihood $p(Y^{\text{test}}|\Theta_{\text{ML}})$ in (2.7) is computed efficiently by noting that

$$p(Y^{\text{test}}|\Theta_{\text{ML}}) = \sum_{j=1}^{J}\alpha_N(j). \tag{2.13}$$

In practice, a scaling procedure (Rabiner, 1989) is applied to the algorithm to ensure that the implementation is numerically stable in finite-precision arithmetic. This issue arises because of the manipulation of a large number of products involving extremely small probabilities, which can exceed the dynamic range of a computer's precision.

HMM-BASED DAMAGE CLASSIFICATION ALGORITHM

The HMM based structural damage classification algorithm is discussed next. It involves three major steps: (i) extracting TF features from the measured data using the MPD; (ii) discretizing the features (for the discrete HMM case); and (iii) using the features from Λ damage classes as observations in the HMM algorithm to estimate the respective model parameters of the Λ HMMs. Once the model parameters are obtained, they are used in a classification scheme to decide which of the Λ classes a test signal belongs to.

MPD Time-frequency Feature Extraction

The effective extraction of discriminatory features from observed data is the critical first step of a successful classification system. The MPD is thus utilized as the feature extraction method as it can provide unique decompositions of the measured signals. Every decomposition has L extracted atoms, each of which has an associated parameter vector that includes the amplitude, time-shift, frequency-shift, and scale change features of the atom. The features provide the TF information that represents the wave physics necessary for distinguishing signals from different structural damage classes. As an example, Figure 3 shows the time-domain plots and MPD-TFR TF plots of signals obtained from healthy and progressive tensile damaged structures. Although the time-domain plots look similar, the MPD-TFRs show more differences because of the discriminatory features extracted by the MPD. Note that the truncation limit L can be determined according to the fraction of signal energy that is sufficient to provide distinguishing information. The amount of signal energy extracted increases with L as the energy of the residue decreases. An example of the residual signal energy diminishing with increasing MPD iterations is shown in Figure 4. In this work, L is chosen such that the energy of the residue is within 10–20%.

In order to use them as observation sequences for HMM classification, the MPD features are sorted in increasing order of time shift τ_l value and cast into a sequence of four-dimensional vectors $\mathbf{y}_l = [\rho_l\ \tau_l\ \nu_l\ \kappa_l]^T$, $l = 0, \ldots, L-1$. The resulting observation sequence Y is obtained as $Y = \{\mathbf{y}_0, \ldots, \mathbf{y}_{L-1}\}$. This observation sequence will be modeled statistically using HMMs for classification where $N = L$ in section 'Hidden Markov Modeling'.

Vector Quantization

The MPD feature observation sequence extracted from the damage waveforms can be directly used

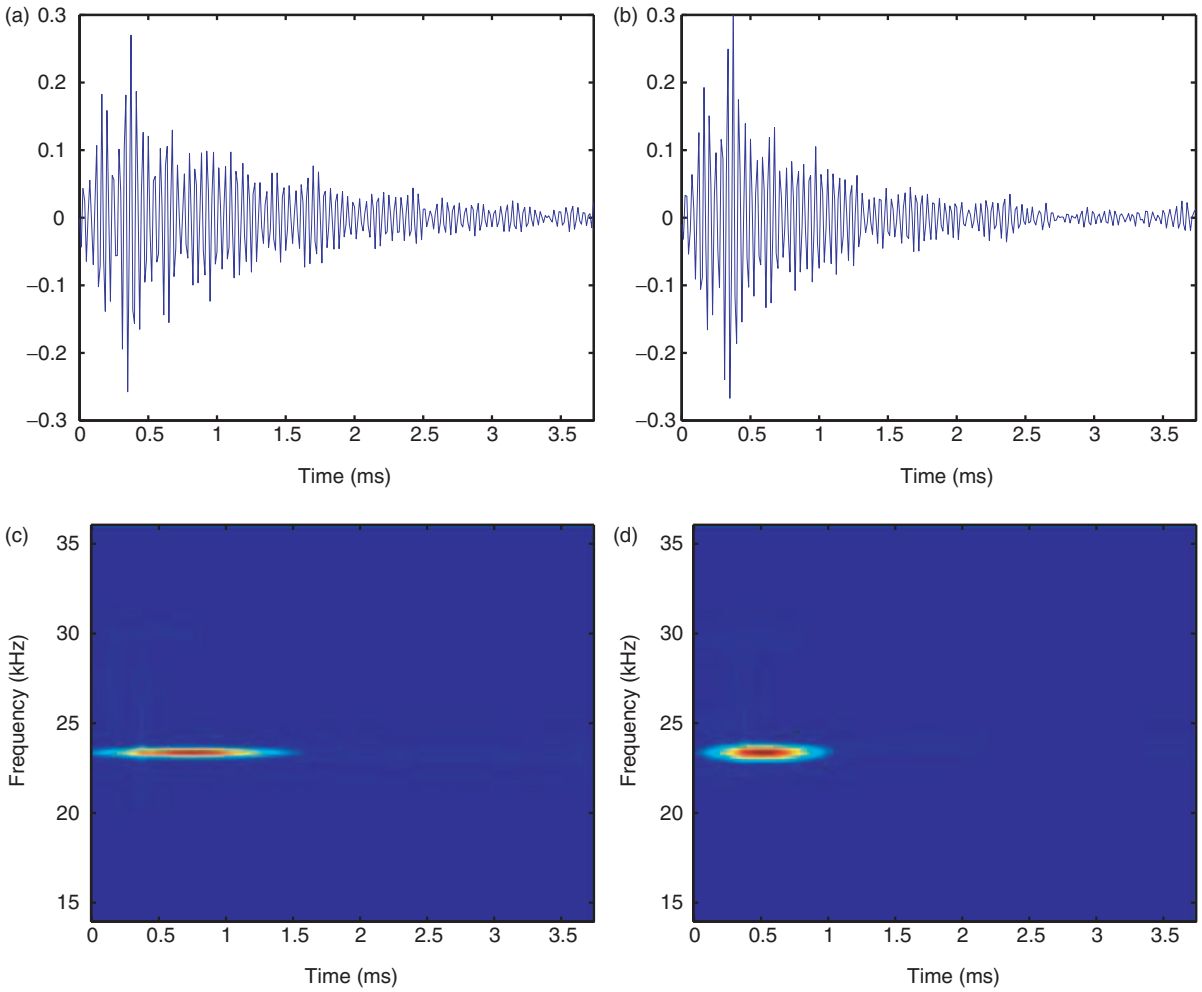

Figure 3. *Example time-plots and MPD-TFRs of signals from healthy and progressive tensile damaged structures: (a) Signal from healthy structure; (b) Signal from damaged structure; (c) MPD-TFR of Signal from healthy structure; (d) MPD-TFR of signal from damaged structure.*

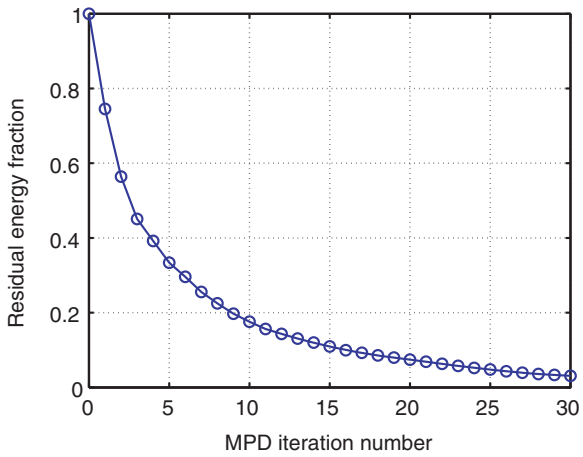

Figure 4. *Residual signal energy for increasing number of MPD iterations.*

by continuous HMMs for modeling, but if discrete HMMs are to be used, an additional discretization step is mandatory. This is performed using vector quantization (VQ) (MacKay, 2003), a nearest-neighbor classification process that maps each feature vector $\mathbf{y}_l$ onto one of a finite set of codebook symbols (codes) $\{q_1, q_2, \ldots, q_K\}$. The codebook is determined using the k-means algorithm (MacKay, 2003) which models the features with a K-component GMM. The parameters of the GMM are estimated using an EM algorithm, and the codes are taken as the estimated means of the Gaussians, with the code assignment performed in a ML fashion. The EM algorithm is initialized with a codebook obtained using the Linde-Buzo-Gray method (Linde et al., 1980). The distortion or quantization error is computed by utilizing the shift and scale invariant Mahalanobis squared distance:

$$d^2(\mathbf{y}_i, \mathbf{y}_j) = (\mathbf{y}_i - \mathbf{y}_j)^T \Omega^{-1} (\mathbf{y}_i - \mathbf{y}_j),$$

where $\Omega = E\left[(\mathbf{y}_i - E[\mathbf{y}_i])(\mathbf{y}_j - E[\mathbf{y}_j])^T\right]$ and $E[\cdot]$ denotes statistical expectation. In general, the quantization error decreases as the number of codes K is increased.

HMM Unobservable States

As discussed in section 'MPD Time-Frequancey Feature Extraction', the HMM observations Y are not the actual measured data but the TF features (discretized in the discrete HMM case) extracted from the measured data using the MPD, i.e., $Y = \{\mathbf{y}_0, \ldots, \mathbf{y}_{L-1}\}$ with $N = L$. Given the MPD features of the time-varying received signals, the HMM defines the nonobservable states as the regions of stationarity corresponding to the underlying physical states induced in the structure in response to the transmitted excitation signal. By time-varying signals, we mean signals whose frequency content changes with time. Thus, by regions of stationarity, we mean regions in time where the frequency content remains close to constant. The HMM works by modeling the temporal transitions between the underlying states with a Markov random process in conjunction with the observation statistics. The features from each structural condition or damage class are modeled with a separate HMM as the associated wave physics is different in each case. The training data available from each damage class is used to learn the parameters of the corresponding HMM. As a result, the number of states J to use in the HMMs can be estimated empirically by examination of the TF characteristics of the features obtained from the training data. For this, we make use of the MPD-TFR, which can be computed efficiently from the signal MPD (refer to the section 'Matching Pursuit Decomposition'). States are defined as regions in the TF plane where the signal's TF characteristics are relatively unchanging. This is illustrated in Figure 5 with the MPD-TFR of a signal with three transition states.

Initial Model Parameters

The initial guess for the λth model parameters $\Theta^{\lambda} = \{\boldsymbol{\pi}^{\lambda}, \mathbf{A}^{\lambda}, \mathcal{B}^{\lambda}\}$ required in the EM algorithm used to train the HMMs is determined from the state estimates in the section 'HMM Unobservable States'. Specifically, the initial guess for the state transition matrix $\mathbf{A}$ is a $J \times J$ matrix whose elements are constrained to be

$$a_{ij} = \begin{cases} 0, & j < i \text{ or } j > i+2 \\ 1, & i = j = J, \\ 0 \le a_{ij} \le 1, & \text{otherwise} \end{cases}$$

where $i, j = 1, \ldots, J$. This definition allows only left-to-right transitions between adjacent states. The free entries of $\mathbf{A}$ are then initialized based on the frequencies of the state transitions. The initial guess for the state-dependent observation density matrix $\mathbf{B}$ for discrete HMMs is determined by counting the number of times a discrete code is observed in each state. Specifically, the element b_{jk}, $j = 1, \ldots, J$, $k = 1, \ldots, K$, of $\mathbf{B}$ is set equal to the fraction of times q_k occurs in state j. The first element of the initial state probability vector $\boldsymbol{\pi}$ is chosen to be 1; the remaining elements are set to zero. The resulting HMM is a special left-right HMM (Rabiner, 1989) in which the state indices can only increase or stay the same as time progresses, and only transitions between adjacent states are allowed. Note that the parameters initially set to zero will remain zero during the re-estimation procedure.

For the continuous HMM case, the number of components M in the GMM must also be selected and the respective parameters initialized. Specifically, the value of M is initially estimated by examining the empirical distributions of the feature vectors and later adjusted according to the classification performance on a validation data set. After M has been chosen, the initial guess of the parameters of the state-dependent Gaussian mixtures is obtained using vector quantization with M codes. The Gaussian mixture coefficients, means and covariances are selected according to the proportions, means, and covariances of the data associated with the codes. Experience shows that this approach yields a good initial guess, one that works much better than a random starting point. Note that, in this work, the GMMs use uncorrelated Gaussian distributions so that the covariance matrices are restricted to be diagonal.

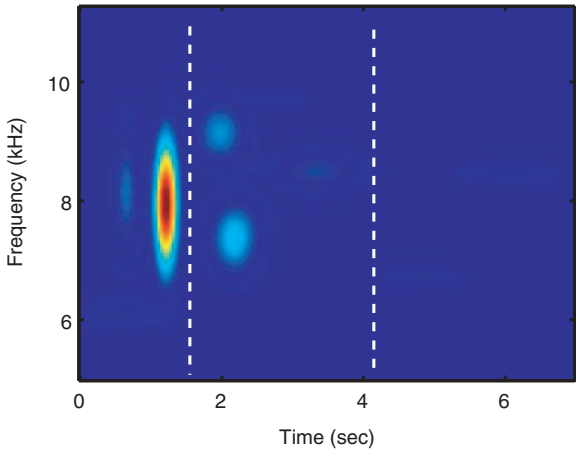

Figure 5. *MPD-TFR of a training signal from a damage class. The dotted vertical lines correspond to TF separators, resulting in $J = 3$ HMM states.*

HMM Classifier

For the purpose of classification, the second level of inference, *model comparison*, is considered. The task is to determine which model is most plausible given some test data. In the present framework, once the model parameters have been estimated, classification of the test data can be performed based on its predictive likelihood as computed by the HMM associated with each damage class. Specifically, a given test

observation sequence Y^{test} is assigned to damage class $C_{\hat{\lambda}}$ where

$$\hat{\lambda} = \underset{\lambda}{\text{argmax}}\{\log p(Y^{\text{test}}|\Theta_{\text{ML}}^{\lambda})\}, \tag{3.1}$$

and $\Theta_{\text{ML}}^{\lambda}$ denotes the parameters of the λth HMM (that has been trained on data from damage class C_{λ}). The computation of $\log p(Y^{\text{test}}|\Theta_{\text{ML}}^{\lambda})$ is carried out efficiently using the forward–backward algorithm discussed in 'Hidden Markov Modeling'. In this manner, the HMMs are integrated very efficiently into the Bayesian framework for the task of damage classification.

An Illustrative Example

As an illustration of the proposed classifier, a damage detection problem is considered where the goal is to determine whether a structure is healthy or damaged. With data provided from both structural conditions (classes), this can be viewed as a simple two-class damage classification problem. Specifically, based on the available training data, a new test signal needs to be classified to one of the two classes as healthy or damaged.

The algorithm models the data from each of the two classes with an HMM. Specifically, if there are 100 independent and identically distributed experimentally collected signals from each class, healthy and damaged, then these signals constitute the training data used to learn the parameters of $\Lambda = 2$ HMMs. The training procedure is summarized in the following steps:

1. TF feature extraction is performed on the training signals as discussed in section 'MPD Time-frequency Feature Extraction' to obtain the observation sequences $\{Y_1^{\lambda}, \ldots, Y_{100}^{\lambda}\}$, $\lambda = 1, 2$.
2. If discrete HMMs are used, then each of the continuous feature vector $\mathbf{y}_n$ in Y is quantized to a corresponding discrete code from the codebook $\{q_1, q_2, \ldots, q_K\}$.
3. For each class, the MPD-TFRs of the training data are examined in order to choose the number of states J as discussed in the section 'HMM Unobservable States'. They are also used to obtain the initial guess for the model parameters according to the strategy discussed in 'Initial Model Parameters'.
4. The two classes correspond to two HMM models, $\lambda = 1$ (healthy) and $\lambda = 2$ (damaged). For each class, the parameters of the associated HMM are learned using the re-estimation Equations (2.9)–(2.12) and the training data from that class. The learnt parameters Θ_{ML}^1 and Θ_{ML}^2 achieve the local maxima of $\log p(Y_1^1, \ldots, Y_{100}^1|\ \Theta)$ and $\log p(Y_1^2, \ldots, Y_{100}^2|\Theta)$, respectively (Equation (2.5)).

Once the training is completed, an HMM with parameters Θ_{ML}^1 for the healthy class and an HMM with parameters Θ_{ML}^2 for the damaged class are available. A test observation sequence Y^{test} is then classified to a Class C_1 (corresponding to the healthy structure) if its predictive likelihood satisfies

$$\log p(Y^{\text{test}}|\Theta_{\text{ML}}^1) > \log p(Y^{\text{test}}|\Theta_{\text{ML}}^2),$$

where the likelihoods are computed as described in the section Hidden Markov Modeling (Equations (2.7) and (2.13)).

SENSOR FUSION

In the structural damage detection and classification problem, data is usually collected by many distributed sensing devices with the goal of exploiting the added information from multiple sensors to produce more accurate and robust (e.g., to sensor malfunction or debonding) classification performance. Optimal Bayesian sensor fusion can however be difficult to realize because it requires modeling of the full joint density of the combined data (Gelman et al., 2004b). As an alternative, the Bayesian *decision* fusion procedure is employed in which (local) classification is first performed independently using the data collected at each sensor, and the resulting decisions are then integrated at a fusion center to arrive at a global decision. This approach of fusion, while sub-optimal, is simpler to implement and still leads to significant improvements in the performance of the classifier.

The sensor fusion algorithm is formulated by considering R distributed sensors, Λ damage classes C_{λ}, $\lambda = 1, \ldots, \Lambda$, and measured signal $s^r(t)$ from the rth sensor, $r = 1, \ldots, R$. The sensor fusion works by first making a class assignment decision u^r for each signal $s^r(t)$ based on the HMM algorithm discussed in the sections 'Hidden Markov Modeling' and 'MPD Time-frequancy Feature Extraction', 'Vector Quantization', 'HMM Unobservable States', 'Initial Model Parameters' and 'HMM Classifier'. In particular, $u^r = \lambda$, if $s^r(t)$ is assigned to damage class C_{λ}. This yields a class assignment vector $\mathbf{u} = [u^1 \ldots u^R]^T$ from which the fusion center declares damage class $C_{\hat{\lambda}}$ if

$$\hat{\lambda} = \underset{\lambda=1,\ldots,\Lambda}{\text{argmax}}\{\Pr(\mathbf{u}|\mathrm{C}_{\lambda})\}, \tag{4.1}$$

where $\Pr(\mathbf{u}|\mathrm{C}_{\lambda})$ is the probability distribution of the class assignment vector $\mathbf{u}$ conditioned on the data actually belonging to damage class C_{λ}. Note that the $\Pr(\mathbf{u}|\mathrm{C}_{\lambda})$ has to be estimated from available training or validation data. The validation data is a set of known signals, which is not used to train the HMMs but to verify the performance of the trained classifier. Assuming statistical

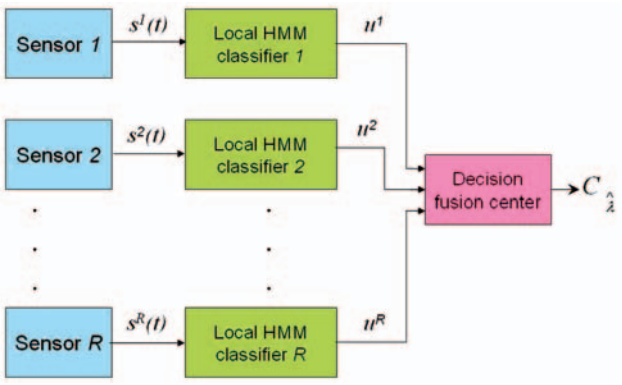

Figure 6. *Block diagram of the sensor fusion algorithm.*

independence of the distributed sensors, the probability distribution $\Pr(\mathbf{u}|C_\lambda)$ can be written as $\Pr(\mathbf{u}|C_\lambda) = \prod_{r=1}^{R} \Pr(u^r|C_\lambda)$. Using this in (4.1), the sensor fusion classification result then becomes

$$\hat{\lambda} = \underset{\lambda=1,\ldots,\Lambda}{\operatorname{argmax}} \left\{ \prod_{r=1}^{R} \Pr(u^r|C_\lambda) \right\}. \tag{4.2}$$

Note that the quantity $\Pr(u^r|C_\lambda)$ in (4.2) can be obtained directly during the implementation of the local classifiers. Details of this are given in the section 'Delamination Damage Classification Results'. The block diagram of the sensor fusion algorithm is shown in Figure 6.

DAMAGE CLASSIFICATION IN COMPOSITE STRUCTURES

An application of the proposed algorithm to the classification of damage in composite structures is presented next. Both discrete and continuous HMMs are utilized. The classifier is first tested using data from only one sensor. This is then followed by the integration of local decisions applying the sensor fusion procedure.

Experimental Setup and Data Collection

The composite structures considered in this article are carbon-fiber symmetric cross-ply laminates, each consisting of 16 plies (Figure 7). Three sets of experiments were performed, introducing three common types of damage in laminates.

DELAMINATION IN LAMINATED COMPOSITES

Seeded delaminations at different locations are considered in the laminates. The damage class definition is summarized in Table 1. These delaminations are located at the 4th interface on the edge and corner of the sample. Data was collected using piezoelectric transducers (PZT) mounted on the structure as shown in Figure 8. A burst signal (with 8 kHz center frequency) was used as the excitation from the sensor in the center, and the responses were measured at the four PZTs located at the corners. Data was sampled at 500 kHz. Each PZT provides 360 signals from each class to be used for classification.

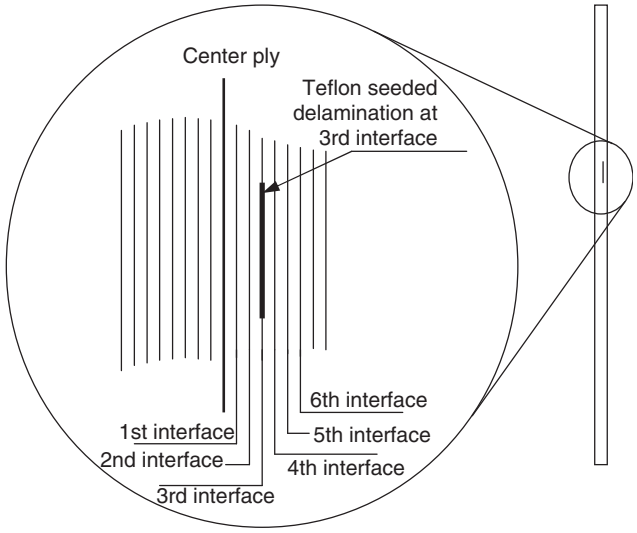

Figure 7. *Schematic showing the composite laminate structure.*

Table 1. Class definitions for seeded delamination.

Class 1	Healthy state
Class 2	5% delamination at the 1st interface of the laminate
Class 3	5% delamination at the 2nd interface of the laminate
Class 4	5% delamination at the 3rd interface of the laminate
Class 5	5% delamination at the 4th interface of the laminate
Class 6	5% delamination at the 5th interface of the laminate
Class 7	5% delamination at the 6th interface of the laminate
Class 8	Corner delamination at the 4th interface
Class 9	Edge delamination at the 4th interface

IMPACT TESTING ON LAMINATED COMPOSITES

Figure 9(a) shows the experimental setup of the impact testing performed on a laminated composite by an impact machine. Impact velocities of 2.53, 2.11, and 1.71 m/s were used to damage samples with a 35 lb impacting head. The four damage classes are defined as in Table 2. A 5-cycle burst of 50 kHz center frequency was used as the excitation signal from a PZT as shown in Figure 9. Data was collected from the PZTs placed on the two sides of the impact zone and sampled at 2 MHz. From each class, and from each PZT, there are 150 signals available for classification, a total of 1200 signals.

PROGRESSIVE TENSILE DAMAGE ON A LAMINATED COMPOSITE

A laminate with a pre-cut notch on the edge was placed under a tensile testing machine to produce progressive tensile damage as shown in Figure 10(a). The 10 damage classes corresponding to 10 damage stages are defined in Table 3. The load was incremented

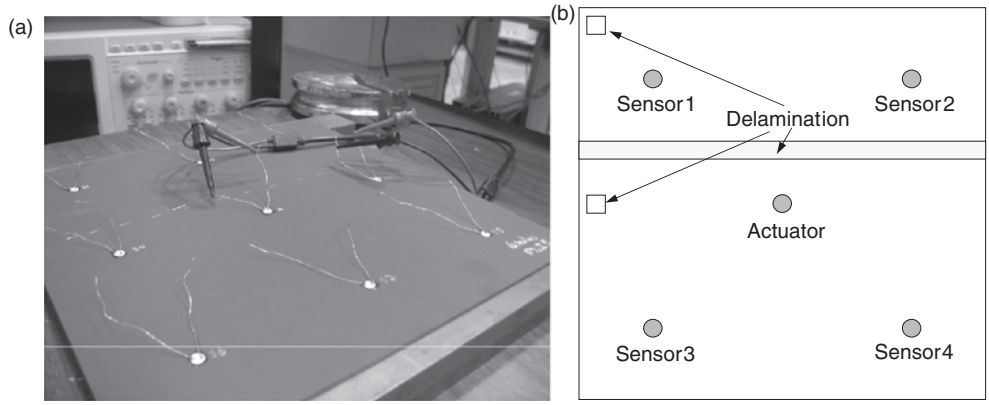

Figure 8. *(a) Experimental setup for delamination damage in a 292.1 mm×292.1 mm laminated composite with 5% delamination, and (b) schematic showing the delamination and sensor layout.*

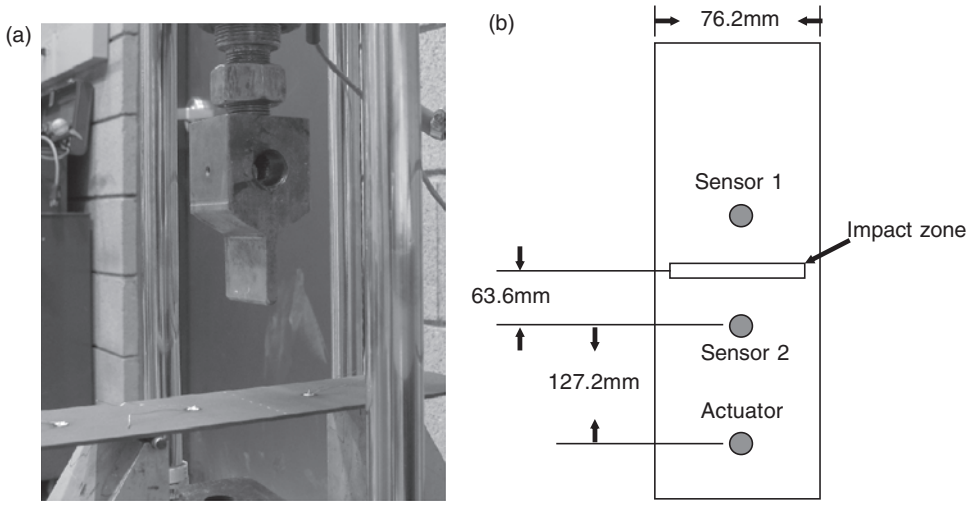

Figure 9. *(a) Experimental setup for impact testing in a laminated composite, and (b) schematic showing the impact zone and sensor layout.*

Table 2. Class definitions for impact damage.

Class 1	Healthy state (before impact testing)
Class 2	Impact damage with velocity 2.53 m/s
Class 3	Impact damage with velocity 2.11 m/s
Class 4	Impact damage with velocity 1.71 m/s

until ultimate failure at 477 MPa. A 5-cycle burst at a 25 kHz frequency was used as the excitation to the laminate from the actuator as shown in Figure 10. Responses were recorded from another PZT at each stage of the incremental loading. From each damage class, 150 signals are available for classification.

Classification Results

As a preprocessing step, the measured signals are first filtered and downsampled (if needed), mean-centered, normalized, and time-aligned (aligned to begin at the same time point). The data is divided into three sets: one set is used for training the HMMs (training set); one set is used to verify the choice of model parameters (such as the number of GMM components M in the continuous HMM) and to obtain the discrete distributions $\Pr(u^r|C_\lambda)$ for implementation of the sensor fusion (see section 'Sensor Fusion') (validation set); and the last set is used to test the individual and global classifier performance (testing set). The purpose of the validation step is to verify the choice of model complexity (the number of states, J, and the number of Gaussian components, M) and parameter initialization. This is achieved by examining the classification performance of the trained HMMs on the validation set, based on which the parameter choice is adjusted. The resulting HMMs are then used to classify the testing data. Results are presented demonstrating damage classification in the composite examples described above using both discrete and continuous HMMs.

The number of HMM states, J, was estimated using the approach described in the section 'HMM Unobservable States'. In all the results reported, J was usually chosen to be 2 or 3, depending on the damage class in question. The simulations indicated that the performance of the classifier was not very sensitive to this choice.

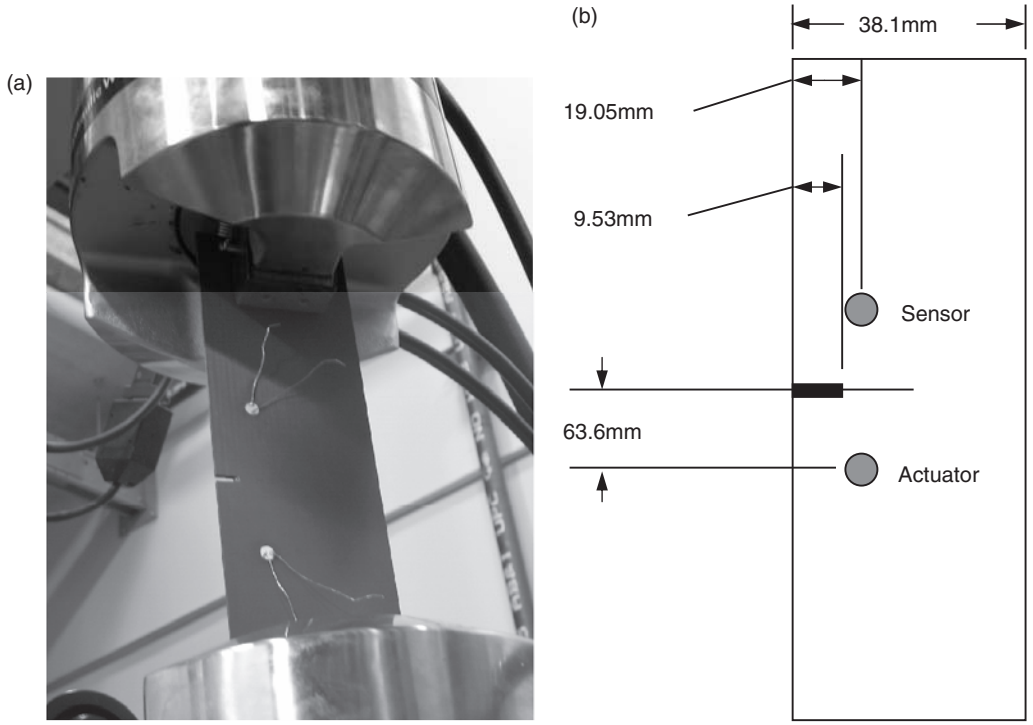

Figure 10. *(a) Experimental setup for progressive tensile damage in a laminated composite, and (b) schematic showing the notch and sensor layout.*

Table 3. Class definitions for progressive tensile damage.

Class 1	Healthy state
Class 2	Damage under 5350 lb tensile loading
Class 3	Damage under 5700 lb tensile loading
Class 4	Damage under 6500 lb tensile loading
Class 5	Damage under 6900 lb tensile loading
Class 6	Damage under 7300 lb tensile loading
Class 7	Damage under 7700 lb tensile loading
Class 8	Damage under 8200 lb tensile loading
Class 9	Damage under 8800 lb tensile loading
Class 10	Damage under 9000 lb tensile loading

The number of iterations of the EM algorithm used for training the HMMs was determined based on the convergence of the log-likelihood. Figure 11 shows example plots of the training data log-likelihood versus training iteration number using the discrete and continuous HMM, respectively, for delamination damage data. The log-likelihood increases monotonically with the training iterations. In the simulations, 25 iterations were usually sufficient for the likelihood to converge.

The classification performance is quantified using $\Lambda \times \Lambda$ confusion matrices, where Λ is the number of classes defined. The (i, j)th entry of the confusion matrix indicates the probability that a signal actually from class C_i is classified to class C_j. In the ideal case of perfect classification, the confusion matrix is an identity matrix.

DELAMINATION DAMAGE CLASSIFICATION RESULTS

TF damage features were extracted from each signal using $L = 10$ MPD iterations with a dictionary composed of about 2 million normalized TF Gaussian atoms. This choice of truncation limit corresponded to a residual signal energy of about 20%. Of the 360 signals available for each damage class, 150 signals were used for training, 105 were used for validation set, and the remaining 105 were used for testing.

Tables 4 and 5 give the confusion matrices for the delamination damage classification using discrete and continuous HMMs, respectively, when only the data from PZT 3 was utilized. The corresponding classifier parameters are: $L = 10$ MPD iterations, $K = 64$ codes (for the discrete HMM), and $M = 9$ Gaussian mixture components (for the continuous HMM). An average correct classification rate $\xi = 80.3\%$ was obtained from the discrete HMM and $\xi = 95.8\%$ from the continuous HMM. Note that ξ is a percentage indicator for classification performance that is defined as the mean of the diagonal values of the confusion matrix. The continuous HMM was seen to perform significantly better than the discrete HMM; it provided a 16% increase in the correct classification ξ.

Tables 6 and 7 show the confusion matrices of the delamination damage classification when sensor fusion was incorporated using the same codes and Gaussian mixtures. Compared with the results of Tables 4 and 5,

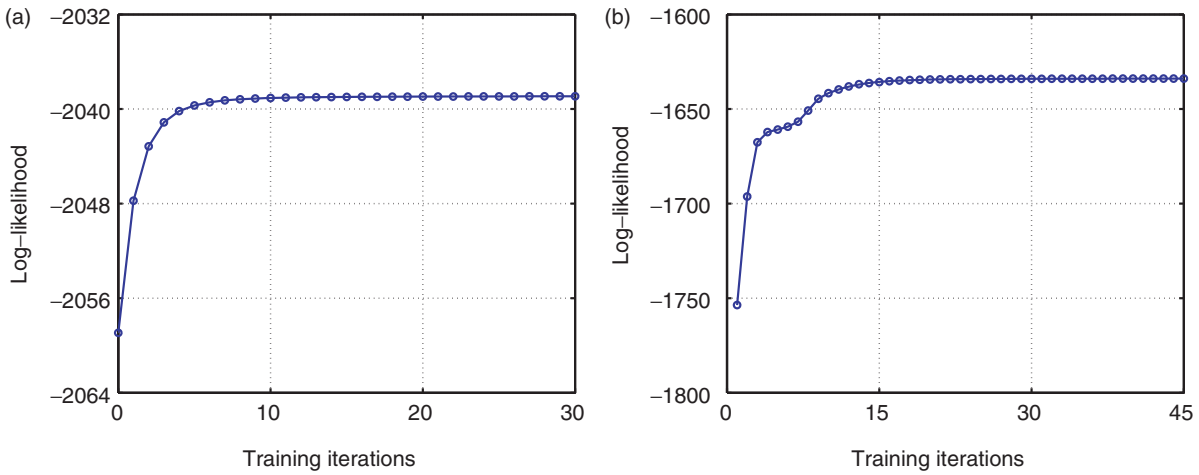

Figure 11. *Log-likelihood vs (a) discrete and (b) continuous HMM training iteration number for the delamination damage example.*

Table 4. Confusion matrix for delamination damage classification using discrete HMMs and data from PZT 3.

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
Class 1	**0.8476**	0.0095	0	0	0.0095	0.0572	0.0571	0.0191	0
Class 2	0	**0.7810**	0.0571	0.0381	0.0762	0.0095	0.0286	0	0.0095
Class 3	0	0.0762	**0.8762**	0.0095	0	0	0	0	0.0381
Class 4	0	0.0095	0.0381	**0.7905**	0.0095	0.0095	0.0857	0.0191	0.0381
Class 5	0.0476	0.0762	0	0	**0.6381**	0.0667	0.1143	0.0476	0.0095
Class 6	0.0571	0.0857	0.0095	0.0476	0.0191	**0.7524**	0.0191	0.0095	0
Class 7	0	0.0667	0	0.0286	0.1523	0.0286	**0.7143**	0	0.0095
Class 8	0	0.0191	0	0	0.0667	0.0095	0.0190	**0.8762**	0.0095
Class 9	0	0.0667	0.0476	0	0.0095	0	0.0095	0.0191	**0.8476**

Table 5. Confusion matrix for delamination damage classification using continuous HMMs and data from PZT 3.

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
Class 1	**0.9810**	0	0	0	0.0095	0	0.0095	0	0
Class 2	0	**0.9143**	0.0667	0	0.0095	0	0	0	0.0095
Class 3	0	0.0381	**0.9524**	0	0.0095	0	0	0	0
Class 4	0	0	0	**1.0000**	0	0	0	0	0
Class 5	0	0	0	0	**0.9524**	0.0190	0.0286	0	0
Class 6	0	0.0381	0	0	0.0286	**0.9333**	0	0	0
Class 7	0	0.0095	0	0	0.0572	0	**0.9333**	0	0
Class 8	0	0	0	0	0.0095	0	0	**0.9905**	0
Class 9	0	0.0381	0	0	0	0	0	0	**0.9619**

Table 6. Confusion matrix for delamination damage classification using discrete HMMs combined with sensor fusion.

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
Class 1	**0.9810**	0	0	0	0	0.0095	0.0095	0	0
Class 2	0.0095	**0.8571**	0.0476	0	0.0286	0.0286	0	0	0.0286
Class 3	0	0	**0.9714**	0	0	0.0095	0	0	0.0191
Class 4	0	0	0	**0.9238**	0.0191	0.0095	0	0.0095	0.0381
Class 5	0.0191	0.0095	0	0.0476	**0.7714**	0.0095	0.1143	0.0286	0
Class 6	0.0095	0.0191	0	0.0095	0.0191	**0.9333**	0	0.0095	0
Class 7	0	0.0191	0	0	0.1333	0.0095	**0.8286**	0	0.0095
Class 8	0	0.0095	0	0	0.0095	0	0	**0.9524**	0.0286
Class 9	0.0095	0.0095	0.0191	0	0	0	0.0095	0.0191	**0.9333**

Table 7. Confusion matrix for delamination damage classification using continuous HMMs with sensor fusion.

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
Class 1	**1.0000**	0	0	0	0	0	0	0	0
Class 2	0.0095	**0.9714**	0	0.0191	0	0	0	0	0
Class 3	0.0286	0	**0.9714**	0	0	0	0	0	0
Class 4	0.0286	0.0190	0	**0.9524**	0	0	0	0	0
Class 5	0.0381	0.0286	0	0.0476	**0.8667**	0	0.0190	0	0
Class 6	0.0286	0	0	0	0	**0.9714**	0	0	0
Class 7	0.0286	0	0	0	0	0	**0.9714**	0	0
Class 8	0.0190	0	0	0	0	0	0	**0.9810**	0
Class 9	0.0381	0	0	0	0	0	0	0	**0.9619**

it is found that the sensor fusion procedure dramatically increases the performance of the classifier (the average correct classification rate ξ is 14% higher than that when only the data from PZT 3 and discrete HMM is used). Note that $\Pr(u^r|C_\lambda)$ in Equation (4.2), that gives the probability of assigning a signal from Class C_λ to Class u^r at Sensor r, is actually the (λ, u^r)th element of the confusion matrix Γ^r_Λ. These probabilities are learned using the validation data set.

To show the effect of different choices of model parameters on the efficiency of the proposed classifier for the delamination classification, Table 8 summarizes the values ξ from the discrete HMM with different number of codes, K, the continuous HMM, and the sensor fusion procedure. The achieved correct classification rates ξ are tabulated for the discrete HMM and continuous HMMs, both with and without sensor fusion. As K increases, the performance of the discrete HMM classifier improves and approaches that of the continuous HMM classifier. This is expected, because the quantization error decreases with increasing K. Meanwhile, the fusion procedure lead to a significant improvement in overall classification performance. Even when the results from the individual sensors were poor, the improvement using fusion was remarkable.

The most accurate classification ($\xi = 96\%$) is observed when the continuous HMM and sensor fusion are both utilized. The improvements shown, however, are afforded at an added computational cost because the more complex models have more parameters to be estimated and the fusion step requires additional effort as the algorithm needs to be performed at each sensor.

Table 8. Average correct classification rates ξ for delamination damage classification using discrete and continuous HMMs, with and without sensor fusion.

	PZT 1(%)	PZT 2(%)	PZT 3(%)	PZT 4(%)	Fusion
Discrete HMM, K=32 codes	61.3	50.3	70.9	57.8	85.40
Discrete HMM, K=64 codes	66	58.4	79.2	70.4	90.6
Discrete HMM, K=128 codes	71.9	70.2	80.3	72.9	91.2
Continuous HMM	91.5	91.5	95.8	92.6	96.1

Table 9. Confusion matrix for impact damage classification using discrete HMMs and data from PZT 1.

	Class1	Class 2	Class 3	Class 4
Class 1	**0.8000**	0	0.1000	0.1000
Class 2	0.0250	**0.9000**	0	0.0750
Class 3	0.1250	0	**0.8750**	0
Class 4	0.1000	0.0500	0	**0.8500**

IMPACT DAMAGE CLASSIFICATION RESULTS

It could be seen that during impact testing when the impact velocity was 1.71 m/s, the sample did not have any visible damage. With a velocity of 2.11 m/s, little visible damage was observed, such as breakage of the first ply fiber and delamination at the 7th interface. As the velocity increased to 2.53 m/s, visible damage occurred in the form of fiber breakage at several plies and multiple delaminations in the laminate. For the impact damage classification, 70 signals from each class were used for training, 40 were used for validation and 40 were used for testing. Feature extraction was performed using $L = 20$ MPD iterations with a dictionary composed of about 7 million normalized TF Gaussian atoms. This choice of truncation limit corresponded to a residual signal energy of about 15%.

Table 9 provides the confusion matrices for impact damage classification in the laminated composite using discrete HMMs, when only the data from PZT 1 was utilized. For modeling with the discrete HMM, the features were quantized using $K = 128$ codes. In the continuous HMMs, $M = 6$ components were used with the GMM. It can be seen that Class 3 and Class 4 are not separated clearly from Class 1. This is in accordance with the measurements because physically the damages of these two classes (especially Class 4) are quite mild so that they appear similar to the healthy case. However, this confusion in the classification was cleared up

Table 10. Confusion matrix for impact damage classification using continuous HMMs and data from PZT 1.

	Class1	Class 2	Class 3	Class 4
Class 1	**0.9750**	0.0250	0	0
Class 2	0	**1**	0	0
Class 3	0	0.0750	**0.9250**	0
Class 4	0	0.0500	0	**0.9500**

Table 11. Confusion matrix for impact damage classification using discrete HMMs with sensor fusion.

	Class1	Class 2	Class 3	Class 4
Class 1	**0.9500**	0.0250	0	0.0250
Class 2	0	**0.9750**	0.025	0
Class 3	0.0500	0	**0.9500**	0
Class 4	0.0250	0	0	**0.9750**

Table 12. Confusion matrix for impact damage classification using continuous HMMs with sensor fusion.

	Class1	Class 2	Class 3	Class 4
Class 1	**0.9750**	0	0	0.0250
Class 2	0	**1**	0	0
Class 3	0.0250	0	**0.9750**	0
Class 4	0	0.0500	0	**0.9500**

Table 13. Average correct classification rates ξ for impact damage classification using discrete and continuous HMMs, with and without sensor fusion.

	PZT 1(%)	PZT 2(%)	Fusion (%)
Discrete HMM, K=32 codes	75.6	76.9	85
Discrete HMM, K=64 codes	78.1	85	85.6
Discrete HMM, K=128 codes	85.6	95.6	96.3
Continuous HMM	96.25	96.3	97.5

with the application of continuous HMMs. Table 10 shows the confusion matrix for the continuous HMM case, and the improvement in the classification is clear. The average correct classification rate was $\xi = 85.2\%$ for discrete HMMs and $\xi = 96.3\%$ for continuous HMMs.

The confusion matrices with sensor fusion are presented in Tables 11 and 12, and the summary of ξ values can be found in Table 13. As before, it can be seen that ξ increased with increasing K for the discrete HMMs, or by transitioning from discrete HMMs to continuous HMMs, or by the use of sensor fusion. From the results of Table 13, very accurate classification was obtained ($\xi = 97.5\%$) when the fusion process integrated local continuous HMM classification results.

PROGRESSIVE TENSILE DAMAGE CLASSIFICATION RESULTS

For progressive tensile damage, the crack level is hard to classify when increasing the load does not lead to significant changes in damage level. Here, 70 signals were used for training, 40 signals for validation, and 40 for testing. The number of MPD iterations was $L = 20$ (corresponding to a residual signal energy of about 10%) with the dictionary composed of about 7 million Gaussian atoms. $K = 512$ codes were used for the discrete HMM.

Tables 14 and 15 show the confusion matrices for the progressive tensile damage classification using discrete and continuous HMMs, respectively (the corresponding average correct classification rates are shown in Table 16). Even with $K = 512$, the discrete HMMs confused the damage levels between 5700 lb and 7700 lb (Class 3 to Class 7) to some extent. However, classification performance increased when continuous HMMs were employed. In view of the complexity of the data, the number of Gaussian mixture model components M was taken considerably larger: $M = 20$ for the five HMMs for Class 3–Class 7, and $M = 6$ for the other HMMs. Significant improvement was observed from the discrete to the continuous HMM case.

Table 14. Confusion matrix for progressive tensile damage classification using discrete HMMs.

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10
Class 1	**0.9250**	0.0250	0.0250	0.0250	0	0	0	0	0	0
Class 2	0	**0.9000**	0.0500	0	0.0250	0.0250	0	0	0	0
Class 3	0	0.0250	**0.6000**	0.2750	0.0750	0	0.0250	0	0	0
Class 4	0	0	0.1750	**0.6000**	0.2000	0.0250	0	0	0	0
Class 5	0	0	0.1250	0.2250	**0.6500**	0	0	0	0	0
Class 6	0.0250	0.0250	0.0250	0	0	**0.6750**	0.2500	0	0	0
Class 7	0	0	0	0	0	0.3750	**0.6250**	0	0	0
Class 8	0	0	0	0	0	0	0.0250	**0.9750**	0	0
Class 9	0	0	0	0	0	0	0.0250	0	**0.9000**	0.0750
Class 10	0	0	0	0	0	0	0	0	0	**1**

Table 15. Confusion matrix for progressive tensile damage classification using continuous HMMs.

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10
Class 1	**0.9500**	0	0	0.0250	0	0	0	0	0.0250	0
Class 2	0	**0.9750**	0.0250	0	0	0	0	0	0	0
Class 3	0	0.0250	**0.9500**	0.0250	0	0	0	0	0	0
Class 4	0	0	0	**0.9750**	0.0250	0	0	0	0	0
Class 5	0	0	0	0.0250	**0.9750**	0	0	0	0	0
Class 6	0	0	0	0	0	**0.8750**	0.1250	0	0	0
Class 7	0	0	0	0	0	0.1500	**0.8250**	0.0250	0	0
Class 8	0	0	0	0	0	0	0	**0.9750**	0.0250	0
Class 9	0	0	0	0	0	0	0	0	**1**	0
Class 10	0	0	0	0	0	0	0	0	0	**1**

Table 16. Average correct classification rates ξ for progressive tensile damage classification using discrete and continuous HMMs.

Discrete HMM K=128 codes	Discrete HMM K=256 codes	Discrete HMM K=512 codes	Continuous HMM
66.8%	73.5%	78.5%	95%

CONCLUSIONS

A novel HMM-based algorithm has been presented for the classification of damage in composite structures. Feature extraction is performed using the MPD with highly localized TF Gaussian atoms, and the extracted 4D feature vectors are statistically modeled with HMMs. Two types of HMMs, discrete and continuous, are employed. The discrete HMM requires the observed data to be quantized, while the continuous HMM models the state-dependent observation densities using GMMs. Experimentally collected data is used to train the models (learn model parameters), and damage classification is efficiently performed in a Bayesian framework. A decision fusion procedure is implemented that integrates the local classification results from individual sensors.

The application of the proposed algorithm to the classification of delamination damage, impact damage, and progressive tensile damage in laminated composites yields very good results, with percentage correct classification rates near 90%. The performance of the discrete HMM classifier is a function of the number of codes used, and results are seen to improve as the number of codes is increased. The accuracy of the continuous HMM classifier is found to be superior to that of the discrete HMM classifier. When available data from different damage classes have similar features, then more complex models are needed, such as those obtained by increasing K for discrete HMMs or increasing M for continuous HMMs.

The algorithm's efficiency in terms of computational effort involved is governed in part by the amount of data required for training the model, which in turn is dictated by the complexity of the model. In the discrete HMM, the complexity increases with the number of codes used, while for the continuous HMM, the number of parameters is related to the number of mixture components in the GMM. In both cases, the number of parameters to be estimated increases with the number of states. In the continuous HMM, the number of GMM components is an additional parameter that has to be estimated at extra cost using techniques such as cross-validation. Also, learning and inference in the continuous HMM is generally more computationally intensive than that for the discrete HMM. Note that the training data need not be experimentally collected; it can be physically modeled and numerically simulated.

The sensor fusion process makes use of the information available from all distributed sensors and leads to large improvements in classification performance. Even when the results from the individual classifiers at each sensor are poor (only 50–70% correct classification), those from the overall decision fusion scheme remain accurate (correct classification rates are near 85%). The fusion procedure is expected to increase the robustness of the classification algorithm to the problem of sensor malfunction. As a drawback, additional data and computational effort is required in order to learn the statistics of the individual classifiers. Also note that the fusion technique used is a Bayesian decision fusion scheme, which is sub-optimal but computationally less expensive than an optimal sensor fusion procedure where the data from all available sensors is modeled jointly. The best classification performance is observed when using the continuous HMM classifier together with sensor fusion. In real applications, the desired accuracy of the results and the time and computational resources available ultimately dictate the choice of features, feature extraction method, model type and complexity, and sensor fusion procedure.

It should be noted that while the application examples provided are for the discrimination of different levels and locations of damage, the proposed damage classifier is based on a general framework and can also be used to

differentiate between different types of damage. Importantly, the classification method is based on the Bayesian framework, which requires knowledge of the distribution of the data conditioned on each class of interest. When these distributions are available (e.g., modeled using HMMs), this approach is known to be optimal. Training data is assumed to be available from each class and is used to estimate the parameters of the HMM used to model that class. The variational Bayesian learning method (Beal, 2003; MacKay, 2003) is currently under investigation for automatically estimating the model complexity from the given data. The number of HMM states can then be determined automatically, thus avoiding the need for cumbersome and expensive empirical examination of the data. This approach also yields approximations to the full posterior distribution over the model parameters, which is a more powerful and robust tool for inference tasks than the present ML point estimates.

ACKNOWLEDGMENTS

This research was supported in part by the Department of Defense Air Force Office of Scientific Research Grant FA95550-06-1-0309 (Program manager: Dr. Victor Giurgiutiu) and by the Department of National Aeronautics and Space Administration (NASA) IVHM Grant NNX07AD70A (Program manager: Dr. Steven Arnold). The authors would like to acknowledge the reviewers for their comments that helped improve the document.

REFERENCES

Abdel-Galil, T.K., El-Saadany, E.F., Youssef, A.M. and Salama, M.M.A. 2005. "Disturbance Classication Using Hidden Markov Models and Vector Quantization," *IEEE Transactions on Power Delivery*, 20:2129–2135.

Aircraft Health Monitoring, 2006. Polytec GmbH Laser Measurement Systems Application Note VIB-A-01, http://www.polytec.com/jpn/_files/LM_AN_VIB-A-01_2006_07_US.pdf

Beal, M.J. 2003. Variational Algorithms for Approximate Bayesian Inference. PhD Thesis, Gatsby Computational Neuroscience Unit, University College, London.

Beck, J.L., Au, S.K. and Vanik, M.W. 1999. "A Bayesian Probabilistic Approach to Structural Health Monitoring," In: *Proceedings of the American Control Conference,* San Diego, CA.

Bharadwaj, P.K., Runkle, P. and Carin, L. 1999. "Target Identication with Wave-based Matched Pursuits and Hidden Markov Models," *IEEE Transactions on Antennas and Propagation*, 47:1543–1554.

Chakraborty, D., Soni, S., Wei, J., Kovvali, N., Papandreou-Suppappola, A., Cochran, D. and Chattopadhyay, A. 2008. "Physics Based Modeling for Time-frequency Damage Classication," In: *International Symposium on Smart structures and Materials & Nondestructive Evaluation and Health Monitoring*, Vol. 6926, p. 69260M.

Channels, L., Chakraborty, D., Simon, D., Kovvali, N., Spicer, J., Papandreou-Suppappola, A., Cochran, D., Peralta, P. and Chattopadhyay, A. 2008. "Ultrasonic Sensing and Time-frequency Analysis for Detecting Plastic Deformation in an Aluminum Plate," In: *International Symposium on Smart structures and Materials & Nondestructive Evaluation and Health Monitoring*, Vol. 6926, p. 69260P.

Das, S., Srivastava, A.N. and Chattopadhyay, A. 2007. "Classification of Damage Signatures in Composite Plates Using One-class SVMs," In: *IEEE Aerospace Conference*, 1–19, March.

Das, S., Papandreou-Suppappola, A., Zhou, X. and Chattopadhyay, A. 2005. "On the use of the Matching Pursuit Decomposition Signal Processing Technique for Structural Health Monitoring," *Proc. of SPIE*, 5764:583–594.

Dempster, A., Laird, N. and Rubin, D. 1977. "Maximum Likelihood from Incomplete Data Via the EM Algorithm," *Journal of the Royal Statistical Society, Series B*, 39:1–38.

Duda, R.O., Hart, P.E. and Stork, D.G. 2001. Pattern Classication, 2nd edn, Wiley Interscience.

Ebenezer, S.P., Papandreou-Suppappola, A. and Suppappola, S.B. 2004. "Classification of Acoustic Emissions using Modified Matching Pursuit," *EURASIP Journal on Applied Signal Processing*, 3:347–357.

Eren, L. and Devaney, M.J. 2004. "Bearing Damage Detection Via Wavelet Packet Decomposition of the Stator Current," *IEEE Transactions on Instrumentation and Measurement*, 53:431–436.

Farrar, C.R. and Lieven, N.A.J. 2007. "Damage Prognosis: the future of Structural Health Monitoring," *Philosophical Transactions of the Royal Society, Series A*, 365:623–632.

Farrar, C.R. and Worden, K. 2007. "An Introduction to Structural Health Monitoring," *Philosophical Transactions of the Royal Society, Series A*, 365:303–315.

Fugate, M.L., Sohn, H. and Farrar, C.R. 2001. "Vibration-based Damage Detection using Statistical Process Control," *Mechanical Systems and Signal Processing*, 15:707–721.

Gelman, L., Giurgiutiu, V. and Petrunin, I. 2004a. "Advantage of using the Fourier Components Pair Instead of Power Spectral Density for Fatigue Crack Diagnostics," *International Journal of COMADEM*, 7:18–22.

Gelman, A., Carlin, J.B., Stern, H.S. and Rubin, D.B. 2004b. Bayesian Data Analysis, 2nd ed, CRC Press.

Giurgiutiu, V. 2005. "Tuned Lamb Wave Excitation and Detection with Piezoelectric Wafer Active Sensors for Structural Health Monitoring," *Journal of Intelligent Material Systems and Structures*, 16:291–305.

Gupta, S., Singh, Srivastav, D.A. and Ray, A. "Measurement of Behavioral Uncertainties in Mechanical Vibration Systems: A Symbolic Dynamics Approach." *Mechanical Vibrations: Measurement, Effects, and Control*, in press.

Jeong H. and Jang, Y. 2000. "Fracture Source Location in thin Plates using the Wavelet Transform of Dispersive Waves," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 47:612–619.

Kim, S.D., In, C.W., Cronin, K.E., Sohn, H. and Harries, K. 2006. "Application of Outlier Analysis for Baseline-free Damage Diagnosis," In: *SPIE International Symposium on Smart Structures and Materials*, Vol. 6174, p. 6174H.

Kovvali, N., Das, S., Chakraborty, D., Cochran, D., Papandreou-Suppapola, A. and Chattopadhyay, A. 2007. "Time-frequency Based Classication of Structural Damage," *in 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference,* Honolulu, Hawaii, pp. 2047–2055.

Lee, B.C. and Staszewski, W.J. 2003a. "Modelling of Lamb Waves for Damage Detection in Metallic Structures, Part I: Wave propagation," *Smart Materials and Structures*, 12:804–814.

Lee, B.C. and Staszewski, W.J. 2003b. "Modeling of Lamb Waves for Damage Detection in Metallic Structures, Part II: Wave Interactions with Damage," *Smart Materials and Structures*, 12:815–824.

Lee, J.W., Kirikera, G.R., Kang, I., Schulz, M.J. and Shanov, V.N. 2006. "Structural Health Monitoring using Continuous Sensors

and Neural Network Analysis," *Smart Materials and Structures*, 15:1266–1274.

Li, J., Najmi, A. and Gray, R.M. 2000. "Image Classification by a two Dimensional Hidden Markov Model," *IEEE Transactions on Signal Processing*, 48:517–533.

Linde, D., Buzo, A. and Gray, R.M. 1980. "An Algorithm for Vector Quantizer Design," *IEEE Transactions on Communications*, 28:84–95.

Loewke, K., Meyer, D., Starr, A. and Nemat-Nasser, S. 2005. "Signal Identication in Smart Composite Materials using the Two-dimensional Fast Fourier Transform," *Smart Materials and Structures*, 14:895–903.

MacKay, D.J.C. 2003. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

Mallat, S.G. 1998. Second ed. A Wavelet Tour of Signal Processing. Academic Press.

Mallat, S.G. and Zhang, Z. 1993. "Matching Pursuits with Time-frequency Dictionaries," *IEEE Transactions on Signal Processing*, 41:3397–3415.

Mohamed, M.A. and Gader, P. 2000. "Generalized Hidden Markov Models, Part II: Application to Handwritten Word Recognition," *IEEE Transactions on Fuzzy Systems*, 8:82–94.

Niethammer, M., Jacobs, L.J., Qu, J. and Jarzynski, J. 2001. "Time-frequency Representations of Lamb Waves," *Journal of the Acoustical Society of America*, 109:1841–1847.

Paget, C.A., Grondel, S., Levin, K. and Delebarre, C. 2003. "Damage Assessment in Composites by Lamb Waves and Wavelet Coefficients," *Smart Materials and Structures*, 12:393–402.

Papandreou-Suppappola, A. ed., 2002. Applications in Time-Frequency Signal Processing. Florida: CRC Press.

Park, G., Rutherford, A.C., Sohn, H. and Farrar, C.R. 2005. "An Outlier Analysis Framework for Impedance-based Structural Health Monitoring," *Journal of Sound and Vibration*, 286:229–250.

Rabiner, L.R. and Juang, B.H. 1986. "An Introduction to Hidden Markov Models," *IEEE ASSP Magazine*, 4–15.

Rabiner, L.R. 1989. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," In: *Proceedings of the IEEE*, 77:257–286.

Rammohan, R. and Taha, M.R. 2005. "Exploratory Investigations for Intelligent Damage Prognosis using Hidden Markov Models," In: *IEEE International Conferences on Systems, Man and Cybernetics*, 2:1524–1529.

Runkle, P.R., Bharadwaj, P.K., Couchman, L. and Carin, L. 1999a. "Hidden Markov Models for Multiaspect Target Classification," *IEEE Transactions on Signal Processing*, 47:2035–2040.

Runkle, P.R., Carin, L., Couchman, L., Yoder, T.J. and Bucaro, J.A. 1999b. "Multiaspect Target Identication with Wave-based Matched Pursuits and Continuous Hidden Markov Models," In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21:1371–1378.

Sohn, H. and Farrar, C.R. 2000. "Statistical Process Control and Projection Techniques for Structural Health Monitoring," *in Conference on System Identication and Structural Health Monitoring*, Madrid, Spain.

Sohn, H. and Law, K.H. 2000. "Application of Load-dependent Ritz Vectors to Bayesian Probabilistic Damage Detection," *Probabilistic Engineering Mechanics*, 15: 139–153.

Sohn, H., Allen, D.W., Worden, K. and Farrar, C.R. 2005. "Structural Damage Classification Using Extreme Value Statistics," *Journal of Dynamic Systems, Measurement, and Control*, 127:125–132.

Sohn, H., Czarnecki, J.A. and Farrar, C.R. 2000. "Structural Health Monitoring using Statistical Process Control," *Journal of Structural Engineering*, 1356–1363.

Sohn, H., Farrar, C.R., Hunter, N.F. and Worden, K. 2001. "Structural Health Monitoring using Statistical Pattern Recognition Techniques," *Transactions of the ASME*, 123:706–711.

Staszewski, W.J., Boller, C. and Tomlinson, G.R. 2003. *Health Monitoring of Aerospace Structures*, England, Wiley.

Staszewski, W.J., Pierce, S.G., Worden, K., Philp, W.R., Tomlinson, G.R. and Culshaw, B. 1997. "Wavelet Signal Processing for Enhanced Lambwave Defect Detection in Composite Plates Using Optical Fiber Detection," *Optical Engineering*, 36:1877–1888.

Sun, A. and Chang, C.C. 2004. "Statistical Wavelet-based Method for Structural Health Monitoring," *Journal of Structural Engineering*, 1354–1361.

Sun, Z. and Chang, C.C. 2002. "Structural Damage Assessment Based on Wavelet Packet Transform," *Journal of Structural Engineering*, 1055–1062.

Victorov, I.A. 1967. Rayleigh and Lamb Waves: Physical Theory and Applications. Plenum Press, New York.

Wang, C. and Gao, R.X. 2003. "Wavelet Transform with Spectral Post-processing for Enhanced Feature Extraction," *Transactions on Instrumentation and Measurement*, 52:1296–1301.

Yuen, K.-V., Beck, J.L. and Au, S.K. 2004. "Structural Damage Detection and Assessment by Adaptive Markov Chain Monte Carlo Simulation," *Structrual Control and Health Monitoring*, 11:327–347.

Zhou, W., Chakraborty, D., Kovvali, N., Papandreou-Suppappola, A., Cochran, D. and Chattopadhyay, A. 2007a. "Damage Classification for Structural Health Monitoring using Time-frequency Feature Extraction and Continuous Hidden Markov Models," In: *Asilomar Conference on Signals, Systems, and Computers,* (Pacific Grove, California):848–852.

Zhou, W., Kovvali, N., Papandreou-Suppappola, A., Cochran, D. and Chattopadhyay, A. 2007b. "Hidden Markov Model Based Classification of Structural Damage," *Proc. of SPIE*, Vol. 6523, p. 652311.