Artificial Bee Colony for Classification of Acoustic Emission Signal Source

S.N. Omkar[a] * and Senthilnath. J[a]
[a]Department of Aerospace Engineering, Indian Institute of Science, Bangalore – 560012, INDIA.

Abstract
In this paper the approach for classification of acoustic emission signals to their respective sources is employed using a swarm intelligence technique called artificial bee colony. In this work, artificial bee colony is employed to train a multilayer perceptron neural network which is used for the classification of the acoustic emission signal to their respective source. Acoustic emission is carried out using pulse, pencil and spark signal source on the surface of solid steel block. The signal parameters are measured using AET 5000 system. To begin with, the complexity for acoustic emission data set is verified using conventional statistical technique like principal component analysis and traditional training algorithm like multilayer perceptron neural network trained using the backpropagation algorithm. The experiment shows in both the case the classification is not accurate. For this complex acoustic emission data set multilayer perceptron neural network trained using the artificial bee colony algorithm is applied resulting improved classification. The multilayer perceptron neural network trained using the artificial bee colony based technique has an advantage over conventional statistical techniques and traditional training algorithm because they are distribution free, i.e., no knowledge is required about the distribution of data and also they do not get stuck in local minima. To overcome error in learning and also the risk involved in misclassification, we include risk-sensitive loss function. The modified multilayer perceptron neural network trained using the artificial bee colony based on risk sensitive loss function has impressive classification performance in terms of the overall accuracy as well as per class accuracy.

Keywords: Acoustic Emission; Artificial Bee Colony; Risk-sensitive loss function

1. INTRODUCTION

Acoustic Emission (AE) signal refers to the stress (or) pressure wave produced due to transient energy release caused by the process of irreversible deformation in material. Material damages in aerospace components like fiber composites can be detected as well as interpreted qualitatively and quantitatively with the help of acoustic emission tests [1]. AE tests can unravel vital information regarding the structural health of the components being tested [2, 3]. Manufacturing defects, if any, can also be monitored using AE signals [4]. The failure of the component due to fatigue can also be predicted based on the source of acoustic emissions [5].

Superficial similarities between signals produced by different sources may belie difference in features hidden in them. Hence, the fundamental step in tackling this problem is to extract these features to uniquely characterize signal and noise. Because of statistical variation in source characteristics (or) time varying source characteristics, classification of these signals from noise is difficult. Some researchers have reported the use of pattern recognition in AE signal classification [6 to 10]. Most of these studies are based on simple problems and they are not generalized to all situations of signal analysis.

Different parameters of AE signal are measured to classify the signal to corresponding source. Hence, the problem of classification boils down to multi-category pattern classification. The problems associated with AE signal classification are so varied in nature because of the complexity and diversity of the phenomena. Hence, we have to use some intelligent mechanisms to classify such complex AE signals. Traditionally, conventional statistical techniques such as Principal Component Analysis (PCA)

[11] are used to classify the multi-category pattern classification problems. In addition, soft intelligent techniques like Genetic Programming [12], Fuzzy C-means [13], Ant Colony Optimization (ACO) [14] and neural networks (NN) [15, 16] have been used.

In Principal Component Analysis (PCA) approach, the mean vector and covariance matrix are calculated and then the mean value is subtracted from each of the data dimensions. The mean subtracted value is the average across each dimension. Using covariance matrices; eigenvectors and eigenvalues are calculated. They provide us with information about the patterns in the data [17]. However, the PCA sometimes does not provide a satisfactory performance in classification, because the PCA averages not only the characteristics of between-class but also the characteristics of within-class [18].

In the Genetic Programming (GP) approach [19] arithmetic function sets are used to evolve Genetic Programming Classifier Expressions (GPCE). Each GPCE is trained to recognize the samples that belong to its own class and reject the samples that belong to other classes [20]. The experimental result shows that GP is able to classify the AE data set accurately. But the major drawback of GP approach is the complexity of the GPCEs evolved [12].

Ant Colony Optimization [21] is used to derive knowledge from the training data and is also used to generate simple expert system like rule base to classify the data set. These rules are applied sequentially to the testing data in accordance with their quality, the rule with highest quality being applied first. The attributes that constitute the rules of a particular class are limited, but the number of possible sets of rules, which can be built considering the range of the values of the attributes, is very large. Ant-Miner [22] selects only such attributes that are significant in characterizing the class and their respective attribute values to provide with simple rules [14].

Neural Networks (NN) have also proven to be suitable for classification using multidimensional data [23] and have been reported to perform better when compared with the statistical classifiers [24]. Multilayer perceptron neural network (MLPNN) is most widely used for host of NN applications. Training a network is an essential factor for the success of the neural networks. Among the several learning algorithms available, back-propagation has been the most popular and most widely implemented learning algorithm of MLPNN. Back propagation typically consists of presenting the vectors of input features to the network along with the desired output values. The objective is to reduce some measure of error between actual and target outputs by adjusting the weights and biases [25]. The back-propagation algorithm uses gradient information and hence is plagued by the problem of local minima [26].

To avoid this problem, a host of other algorithms have been explored for MLPNN training. Swarm intelligence algorithms inspired from nature are one such alternative [27]. These are a new range of computational algorithms that have emerged from the behaviour of social insects. Social insects are usually characterized by their self-organization and the absence of central control. Still, complex group behaviour emerges from the interactions of individuals who exhibit simple behaviours by themselves. In social insects, every individual is self-autonomous. They can only obtain local information, and interact with their geographical neighbours. All these features characterize swarm intelligence. Examples of systems like this can be found in nature, including bee colonies, ant colonies, bird flocking, animal herding, fish schooling etc. Inspired by the bee behaviour, Artificial Bee Colony (ABC) [28] is one of the generally applicable techniques used for optimizing numerical functions and real-world problems. ABC is one of the swarm intelligence algorithms which have been used for training Artificial Neural Network (ANN) [29]. ABC is employed to train a multilayer perceptron neural network which is used for the classification problem [29].

In this work, the approach for classification of acoustic emission (AE) signals to their respective source is done using multilayer perceptron neural network (MLPNN) trained using the ABC algorithm (MLPNN-ABC) and the modified MLPNN-ABC using risk sensitive loss function have been presented. Initially, for the entire AE signal data set the complexity have been verified using the conventional statistical technique such as Principal Component Analysis (PCA). Further the complexity have been verified based on traditional training algorithm like MLPNN trained using the backpropagation algorithm (MLPNN-BP); here the AE signal data set have been divided into training and testing sample. For this complex AE signal data set MLPNN-ABC is applied, which is used to derive knowledge from the training data and also used to generate optimum weight for the given network configuration to classify the training sample. Likewise, these optimum weight matrices are applied sequentially for the classification of the testing data. For further improvement of classification, the modified MLPNN-ABC using risk sensitive loss function technique is applied for the AE signal data set. This technique is intriguing because of its ability to classify the data efficiently by generating the optimum weights for the given network configuration. In this paper, we demonstrate the application of ABC for generating optimum weight and its subsequent use for acoustic emission signal classification.

In this paper we discuss the acoustic emission signal in section 2, followed by swarm intelligence for pattern classification in section 3. The simulation and results obtained are described in section 4. Conclusions are drawn in Section 5.

2. ACOUSTIC EMISSION SIGNAL

Acoustic Emission signal (AE) refers to the stress or pressure waves produced due to transient energy release caused by the process of irreversible deformation in the material [6, 7, 8]. AE signals are electrical version of these waves perceived by sensitive transducers. These signals can be analyzed to characterize the source of their emission. Various sources of AE signal are

1. Sounds produced by cracking of timber and tin cry.
2. Earthquakes and mine collapses, which are the largest naturally occurring emission sources.
3. Local dynamic movements in metal, such as propagation of crack, twinning, slip, dislocation movements.

In real life, because of the presence of ambient noise and pseudo AE signals, it is very difficult to classify the sources of AE signals. We can eliminate the spurious signals by frequency filtering and amplitude threshold [8]. Even in noise free condition, signals from more than one source can complicate the issue. The experiment is carried out under the assumption of noise free condition and the respective parameters of acoustic signals are measured. Superficial similarities between the AE signals produced by different sources may belie differences in the features hidden in them. Thus, we have to extract the fundamental features to classify AE signal.

Basically, there are two types of AE signals: (1) *burst type* - amplitude of the signal rises sharply and decays gradually, and (2) c*ontinuous* - a continuous, sustained signal. AE signals are rich in information about their source mechanisms. This information is measured using appropriate waveform parameters. A typical acoustic emission signal with its parameters is shown in figure 1.

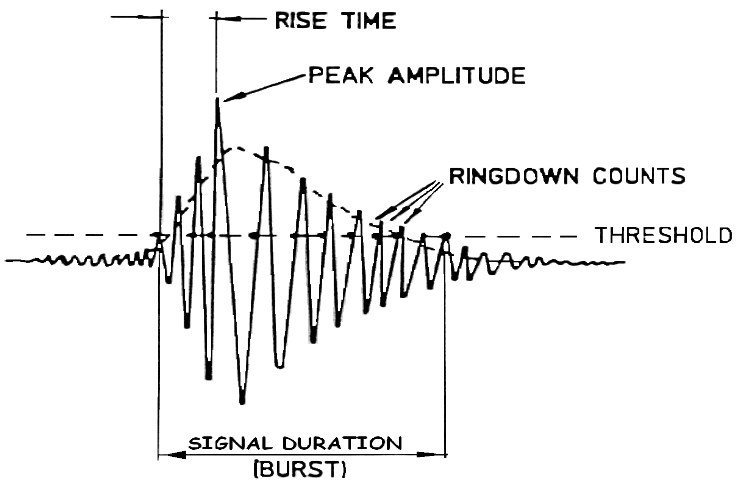


Figure 1. Acoustic Emission Parameters of a Typical Signal

In our study, we assume noise free burst type AE signal from metal surfaces. The following are the typical burst type AE signal parameters:

- **Signal Duration (E_d)**: The beginning of signals are marked when the envelope of the signals crosses the threshold value V_T and end is marked when the threshold value falls below $V_{T.}$ E_d is the time (microseconds) distance between beginning and end of the mark.
- **Peak Amplitude (P_a)**: Highest peak obtained (decibels) by a signal in an event.
- **Rise Time (R_t)**: Time (microseconds) taken to reach its peak after it first crosses the threshold value (V_T)$_.$

- **Ring down count (R_d)**: Number of time the signal crosses the threshold value (V_T).
- **Event gap**: Time (microseconds) interval between two successive signals

Full time energy, RMS value, and dominant spectral frequency are also parameters of burst type acoustic emission signal. The signal parameter like, E_d, P_a, R_t, and R_d is mainly measured for classification purpose. At few instance some of these parametric data of different source tend to resemble resulting complex data set. Here, the challenge arises how efficiently the signal to corresponding source can be classified. Hence, the problem for classifying AE signal boils down to multi-category pattern classification.

3. SWARM INTELLIGENCE FOR PATTERN CLASSIFICATION

In social insect colonies, each individual seems to have its own agenda; and yet the group as a whole appears to be highly organized. Apparently, algorithms based on swarm intelligence and social insects begin to show their effectiveness and efficiency to solve difficult problems. A swarm is a group of multi-agent system such as bees, in which simple agents coordinate their activities to solve the complex problem of the allocation of labor to multiple forage sites in dynamic environments. An important and interesting behavior of bee colonies is their foraging behavior, and in particular, how bees find a food source based on the amount of nectar and successfully bring back to the hive. In a real bee colony the bees are grouped as scout bees, employed bees and onlookers. Initially, the foraging process begins in a colony by scout bees (unemployed bees) which explore food source by moving randomly. At the entrance of the hive is an area called the dance-floor, where dancing takes place. Upon their return to the hive from a foraging trip, it communicates by performing the so-called waggle dance [31] so as to recruit other bees to go to the food source. A bee waiting on the dance area for making decision to choose a food source is called an onlooker, which seem to learn information from the dance regarding the food source: its nectar amount, the direction in which it will be found and its distance [31,32]. If the scouts discover rich food source then the scout bees are selected and classified as the forager bee (employed bee). After waggle dancing the forager bee leaves the hive to get nectar with their fellow bees that were waiting inside the hive. The number of follower bees assigned to nectar depends on the overall quality of the nectar. Upon arrival, the bees take a load of nectar and return to the hive relinquishing the nectar to a food-storer (onlooker) bee. In this way a good food source is exploited, and the number of foragers at this site is reinforced.

In a robust search process, exploration and exploitations process must be carried out together. In the ABC algorithm [28, 30], the scout bees control the exploration process, while the employed bees and onlookers' carryout the exploitation process in the search space. The number of employed bees and the onlookers is equal to the number of solution in the population. The employed bee whose food source has been exhausted becomes a scout bee. The position of an enhanced nectar amount of a food represents a possible solution to the optimization problem.

At the first step, create a population of n artificial (scout) bees placed randomly in the search space representing the food source position, where n denotes the size of population. After initialization, the population of the positions (solutions) is subjected to repeated iteration of the search processes of the employed bees, the onlooker bees and scout bees. This search process can be divided into two phases:

i) Exploration phase: For each solution x_{ij}, where i = 1,2..n and j is dimensional vector. The scout bees explore a new food source with x_i. This operation can be defined as in (1)

$$x_i^j = x_{\min}^j + (x_{\max}^j - x_{\min}^j) * rand(0,1) \tag{1}$$

Here the value of each component in every x_i vector should be clamped to the range [x_{min}, x_{max}] to reduce the likelihood of scout bees leaving the search space (S). The population spread is restricted within the search space S i.e $x_{ij} \in S$ and in the equation (1) x_{min} and x_{max} is the lower and upper limit respectively of the search scope on each dimension.

ii) Exploitation phase: In this phase, assuming the scout bees which has explored food source are selected as an employed bees which produces a modification on the position (solution) in her memory depending on the local information (visual information) and tests the nectar amount (fitness value) of the new source (new solution). If the nectar amount of the new one is higher than that of the previous one, the bee memorizes the new position and forgets the old one. Otherwise it memorizes the position of the previous one. After all employed bees complete the search process; they communicate the nectar information of the food sources and their position information with the onlooker bees on the dance area. An onlooker bee evaluates the nectar information taken from all employed bees and chooses a food source with better nectar amount. As in the case of the employed bee, onlooker bee also produces a modification on the position in her memory and checks the nectar amount of the candidate source. Providing that its nectar is higher than that of the previous one, the bee memorizes the new position and forgets the old one.

An artificial onlooker bee chooses a food source depending on the new positions, using the equation 2.

$$P_i = \begin{cases} v_i, \mathit{if}\ (f(x_i) \geq f(v_i)) \\ x_i, \mathit{if}\ (f(x_i) \leq f(v_i)) \end{cases} \tag{2}$$

In order to select the better nectar position found by an onlooker, O_b is defined as

$$O_b = \arg\min_{P_i} f(P_i),\ 1 \leq i \leq n \tag{3}$$

where P_i is the best fitness value of the solution i which is proportional to the nectar amount of the food source in the position i and n is the number of food sources which is equal to the number of employed bees.

In order to produce a candidate food position from the old one in memory, the ABC uses the following equation (4):

$$v_{ij} = x_{ij} + \alpha(x_{ij} - x_{kj}) \tag{4}$$

where k=1, 2,..., n and j = 1, 2,...,D are randomly chosen indexes. Although k is determined randomly, it has to be different from i. a is a adaptively generated random number. It controls the production of neighbour food sources around x_{ij} and represents the comparison of two food positions visually by a bee. As can be seen from (3), as the difference between the parameters of the x_{ij} and x_{kj} decreases, the perturbation on the position x_{ij} gets decrease, too. Thus, as the search approaches to the optimum solution in the search space, the step length is adaptively reduced. The food source of which the nectar is abandoned by the bees is replaced with a new food source by the scout bees.

3.1 Multi-Layer Perceptron Neural Network

Multilayer perceptron (MLP) are a class of feed forward neural networks trained with the standard back-propagation algorithm or other techniques like Artificial Bee Colony (ABC). In traditional training algorithm, the MLPNN are usually trained using back propagation (MLPNN-BP) algorithm which is a gradient based optimization algorithm. The back propagation algorithm has difficulties in handling local optima and cannot yield optimal adjustable weights for MLP networks [26]. The output layer has four neurons, representing the four AE classes. For this study, we consider only a single hidden layer perceptron network based classifier, with eight neurons in the hidden layer.

Training basically involves presenting the training samples as input vectors through a neural network, calculating the error of the output layer, and then adjusting the weights of the network to minimize the error. Each “training epoch” involves one exposure of the network to a training sample from the training set, and adjustment of each of the weights of the network each layer by layer. The “outside” samples make up the “validation” set.

3.2 Artificial Bee Colony for Multi-Layer Perceptron Neural Network

In this model, single hidden layer perceptron network based architecture is used for the AE signal classification. This is coupled with a ABC based learning algorithm to train the network. ABC has been extensively used for training artificial neural network and proved to be more efficient than many other gradient based training algorithms [33]. This can be mainly attributed to stochastic nature of the algorithm which makes it very robust and flexible.

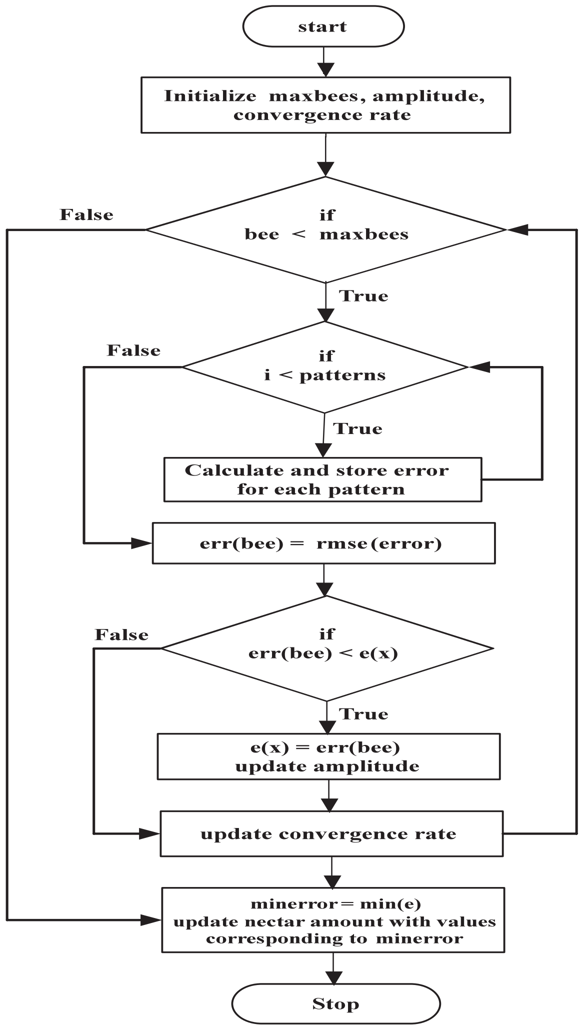
In ABC every bee explores a possible solution - in the current case the optimum weight matrix for the given network configuration. The training error err(x) is used as the fitness value; this indicates the extent of conformance of the network output with actual output. Minimising the error (fitness value) will lead to the best set of weights for the given network configuration and the network is said to be trained. For any classifier, its performance is dependent on the chosen loss function. Selection of proper loss function err(x) for a given problem is often difficult [34,35]. Different classification techniques in machine learning employ different loss functions to get better classification accuracy. Commonly neural network classifiers employ mean square error minimization [36,37] or cross entropy [38,39]. In our study, we use the loss function such as root mean sum of squared residuals (error) in the training data as the fitness values of the ABC. This serves as a qualitative performance measure of the network learning and is given in equation (5).

$$rmse = \sqrt{\frac{1}{N_1} \sum_{\in N_1} (\bar{y}(k) - \hat{y}(k))^2} \tag{5}$$

where $\bar{y}(k)$ is the time value of the output, and $\hat{y}(k)$ is the estimated output of the neural network. N_1 is the number of data points used in the training set.

Our objective is to minimize this fitness value. At each time step the randomness amplitude and speed of convergence of each bees is changed towards its food source. The random factor prevents the swarm getting stuck in the wrong place and speed of convergence is used to identify the rate at which bees converge to a solution.

The flow chart of the procedure:



In general most of the classification algorithms pursue the goal of minimizing the expected number of errors in classification (errors in learning process). Further it has been shown that minimization of this error alone is not an adequate criterion for multi-class classification problems [40]. In general, there is an action behind every classifier decision and different kinds of errors result in different risk [41]. The amount of risk involved in the false decision is critical in such applications. To handle such situation, minimizing risk measures other than the probability of misclassification should be preferred, i.e., if the effects of learning and decision are considered together for model development, the classifier performance can be improved considerably. Inclusion of the risk in the loss function becomes crucial for problems where there is a severe imbalance in the number of training samples per class and also the total number of training samples is small. The small number of training samples and the presence of imbalance in training set result in not knowing the input distribution completely. This results in higher approximation and estimation errors. To overcome these difficulties in multi-category classification problem, we have employed the risk-sensitive hinge loss function introduced by Suresh et. al. [40]. In the proposed loss functions, risk factor is integrated with the existing modified hinge loss function.

3.3 Risk sensitive hinge loss function

The neural classifier weights are adapted using the Risk Sensitive Hinge Loss (RSHL) [40]. For a two-class problem, it has been proven [42] that the truncated output of the classifier developed using hinge loss function can approximate the posterior probability accurately and the bound on approximation error is less than that of other loss functions. The RSHL is similar to the class of hinge loss functions, but incorporates the risk factor explicitly in the formulation. The defined RSHL function is

$$RSHL = \begin{cases} -\left(m_{kj}+1\right)^2 y_{ij}\hat{y}_{ij} & \text{if } y_{ij}\hat{y}_{ij} \le -1 \\ \left(m_{kj} y_{ij}\hat{y}_{ij} - 1\right)^2 & \text{if } -1 \le y_{ij}\hat{y}_{ij} \le 0 \\ \left(y_{ij}\hat{y}_{ij} - 1\right)^2 & \text{if } 0 \le y_{ij}\hat{y}_{ij} \le 1 \\ 0 & \text{if } y_{ij}\hat{y}_{ij} \ge 1 \end{cases} \tag{6}$$

where m_{kj} is the risk factor and index k is the true class label of observation feature vector X_i. The RSHL has different effects for different regions of error. In our formulation, the risk factor is different for different classes and the loss function penalizes the misclassified samples heavily. In RSHL, the risk factor is included only when the sample is misclassified. Also, the adaptation of risk factor in RSHL minimizes the estimation error due to imbalance in training data. Thus, RSHL minimizes both approximation and estimation errors [40].

3.4 Estimation of risk factor m_{kj}

The risk factor plays a vital role in minimization of error for classification problems by penalizing the misclassification heavily. In the current work we have used an adaptively allocated the risk factor, the adaptation of risk factor in frequent intervals helps the classifier in minimizing the effect of imbalance in the training set and substantially increases the accuracy of classification [40]. The risk factor is adaptively allocation based on the method proposed by Suresh et.al. [40]. Here the risk factor is decided based on the number of training samples in each class and the number of epochs for developing the classifier model. The risk matrix is also proportional to the number of misclassified patterns and cost of misclassification.

4. SIMULATION AND RESULTS

4.1 Data Collection

Three methods are commonly used for simulating acoustic emission on a test specimen:

a. Breaking a fixed length of 2H pencil lead.
b. Feeding a pulse by a pulse generator to AE transducer mounted on the surface (the pulse is generated by a piezoelectric crystal transducer excited by a 5V electrical pulse input).
v. Generating spark by a spark gun close to the surface. (A commercial gas lighter generates the spark signal).

All these sources give rise to burst-type AE signals. However, due to inherent differences in the mechanism, the signals from three sources differ in one or more of the signal parameters. AE signals are simulated by the three sources on the surface of a 0.1524m-long, 0.1016m-wide and 0.0508m-thick steel block. The receiving transducer is kept at a fixed location on one side of the steel block. Events are simulated at random and signal parameter like, E_d, P_a, R_t, and R_d is measured for classification purpose. These parameters are also affected by a noise source. The transducer outputs are measured using AET5000 [43] system. We have simulated 199 events, with four field of data item on each record corresponding to measured parameters. Out of 199 records, 40 belong to the pencil source, 45 to the pulse source, 39 to the spark source and the rest to noise (unknown source). A subset signals generated using the three source and unknown sources are given in Table 1. The complete data is considered for classification using PCA. Incase of MLPNN-BP, ABC trained MLPNN classifier with and without the risk loss function; the complete data is segregated as training data and testing data.

Table 1. Subset of experimental dataset

Sl. No.	Event Duration (E_d)	Peak Amplitude (P_a)	Rise Time (R_t)	Ring down count (R_d)	Class
1	52	481	93	11	A
2	69	382	95	11	A
3	59	536	93	11	A
4	1	1	81	1	B
5	1	1	79	1	B
6	2	6	83	5	B
7	73	409	92	136	C
8	75	408	92	136	C
9	105	362	97	255	C
10	2	2	83	1	D
11	2	2	84	1	D
12	5	11	90	2	D

4.1.1 Selection of Training Patterns

The patterns are selected for training and testing the MLPNN-BP and MLPNN-ABC with and without the risk loss function based techniques. The training pattern from the population of patterns is selected using the well-known method of selection, based on the largest deviation of patterns from their mean. The training and testing data set used here is the same as that used by Suresh S. et al [12] in their implementation of Genetic programming. There are 4 classes in the data sets, namely *A* - pencil source, *B* - noise (unknown source), *C* - pulse and *D* - spark. There is lot of interference in features between the noise and spark signals. In our study, 199 data points are collected in which 40(samples) data points belong to class-A, 75 data points belong to class-B, 45 data points belong to class-C and 39 data points belong to class-D. The training set data points are used for generating the optimum weight matrix and the testing set data points are used for obtaining classification. In our experiment, 14 data points belonging to class-A, 17 data points belonging to class-B, 15 data points belonging to class-C and 16 data points belonging to class-D are used for creating clusters. The rest of the data points 26 (belonging to the class-A), 58 (belonging to the class-B), 30(belonging to the class-C), and 23 (belonging to the class-D) are used for testing.

4.2 Evaluation of the classification accuracy

To evaluate the performance, the data set is used to arrive at the classification matrix which is of size n x n, where n is the number of classes. A typical entry $\mathbf{q_{ij}}$ in the classification matrix shows how many samples belonging to class *i* have been classified into class *j*. For a perfect classifier, the classification matrix is diagonal. However due to misclassification we get off-diagonal elements. The individual efficiency of class *i* is defined (for all j) as

$$q_{ii} / \Sigma\, q_{ji} \tag{7}$$

for all j. The overall efficiency is defined as

$$(\Sigma\, \mathbf{q_{ii}}) / N \tag{8}$$

where N is the total number of elements in the data set.

4.3 Principal Component Analysis Simulation and Classification

Principal Component Analysis (PCA) evolves the statistical measures for classification. Here the complete AE signal samples (199 data set) is used for classification. The PCA algorithm creates a data which includes average across each dimension, covariance matrix, eigenvectors and eigenvalues. This data is then used to generate clusters by K-means clustering technique. The clustering is used to group the data set to their respective classes and generate the classification matrix.

From the classification matrix for the complete data set as depicted in Table 2, we can observe that samples belonging to noise (unknown source) are getting classified without any misclassification and hence have individual efficiencies of 100%. But, between pencil source, pulse and spark, there is misclassification. Eleven samples belonging to pencil source are misclassified - out of which four samples are misclassified as noise and seven samples belonging to pencil source are misclassified as spark. Three samples belonging to pulse are misclassified as spark. Hence, pencil source has an individual efficiency of 72.5%, while pulse has an efficiency of 93.3%. But the entire spark samples are misclassified as noise (38 samples) and pulse (1 sample) and hence have individual efficiency of 0%. Also, the overall classification is not so good with an efficiency of 73.37%. Thus, conventional method like PCA fails to classify spark samples efficiently next we explore classification using MLPNN-BP by segregating the entire sample as training data and testing data.

Table 2. Classification Matrix of Acoustic Emission signal data set created by PCA Classifier

	Class A	Class B	Class C	Class D	Individual Efficiency
Class A	29	4	0	7	72.5 %
Class B	0	75	0	0	100 %
Class C	0	0	42	3	93.3 %
Class D	0	38	1	0	0 %

A - Pencil source, *B* - noise (unknown source), *C* - pulse and *D* - spark
Overall Efficiency = 73.37 %

4.4 Multi-Layer Perceptron Neural Network Simulation and Classification

Multilayer Perceptron Neural Network trained using Back Propagation algorithm (MLPNN-BP) with a single hidden layer, 4 inputs, 8 neurons in the hidden layer and 4 neurons in the output layer is trained with 62 samples. From the classification matrix for the training data as depicted in Table 3, we can observe that in the training set itself, samples belonging to pencil source and pulse are getting without any misclassification and hence have individual efficiencies of 100%. One sample belonging to noise is misclassified as pulse, with individual efficiency 94.12%. In case of spark the entire sample has been trained to be noise (14 samples) and pulse (2 samples) with 0% individual efficiency. MLPNN-BP also fails considerably to train the spark sample; to overcome this problem we have successfully used MLPNN-ABC with and without the risk loss function method to efficiently classify the training and testing data.

Table 3. Classification Matrix of Acoustic Emission signal training data set created by MLPNN Classifier

	Class A	Class B	Class C	Class D	Individual Efficiency
Class A	14	0	0	0	100 %
Class B	0	16	1	0	94.12 %
Class C	0	0	15	0	100 %
Class D	0	14	2	0	0 %

A - Pencil source, *B* - noise (unknown source), *C* - pulse and *D* - spark
Overall Efficiency = 72.58 %

4.5 Artificial Bee Colony Simulation and Classification

The ABC trained MLPNN classifier with and without the risk loss function has been employed for AE signal classification. The ABC parameters which includes *number of bees (n), Randomness amplitude of bees* (α), *learning rate* (β), *Speed of convergence* (γ) *and maximum iteration (it)* are varied until they produce most favorable classification result. The optimum values for the above parameters for the most favorable results are as follows:

Number of bees, n = 1000,
Randomness amplitude of bees, $\alpha = 1.55$
Learning rate, $\beta = 1.55$
Speed of convergence is adaptively generated for each iteration, $\gamma = [0.5,\ldots,1]$
Maximum number of iterations, it=3000

The classification matrices obtained for the Training and Testing data using MLPNN-ABC with and without risk sensitive loss function are shown in tables 4 to 7. Figure 2 illustrates graphically the individual efficiencies of each class on classification of training and testing data using MLPNN-ABC with and without risk sensitive loss function.

Table 4: Classification Matrix of Acoustic Emission signal training data set created by MLPNN-ABC Classifier

	Class A	Class B	Class C	Class D	Individual Efficiency
Class A	14	0	0	0	100 %
Class B	0	15	0	2	88.24 %
Class C	0	0	15	0	100 %
Class D	0	6	0	10	62.5 %

A - Pencil source, *B* - noise (unknown source), *C* - pulse and *D* - spark
Overall Efficiency = 87.1 %

Table 5: Classification Matrix of Acoustic Emission signal testing data set created by MLPNN-ABC Classifier

	Class A	Class B	Class C	Class D	Individual Efficiency
Class A	24	0	1	1	92.31 %
Class B	0	55	0	3	94.83 %
Class C	0	0	30	0	100 %
Class D	0	10	1	12	52.17 %

A - Pencil source, *B* - noise (unknown source), *C* - pulse and *D* - spark
Overall Efficiency = 88.32 %

Table 6: Classification Matrix of Acoustic Emission signal training data set created by MLPNN-ABC Classifier with risk loss function

	Class A	Class B	Class C	Class D	Individual Efficiency
Class A	14	0	0	0	100 %
Class B	0	16	0	1	94.12 %
Class C	0	0	15	0	100 %
Class D	0	4	0	12	75 %

A - Pencil source, *B* - noise (unknown source), *C* - pulse and *D* - spark
Overall Efficiency = 91.94 %

Table 7: Classification Matrix of Acoustic Emission signal testing data set created by MLPNN-ABC Classifier with risk loss function

	Class A	Class B	Class C	Class D	Individual Efficiency
Class A	24	0	0	2	92.31 %
Class B	0	55	0	3	94.83 %
Class C	0	0	30	0	100 %
Class D	0	3	1	19	82.61 %

A - Pencil source, *B* - noise (unknown source), *C* - pulse and *D* - spark
Overall Efficiency = 93.43 %

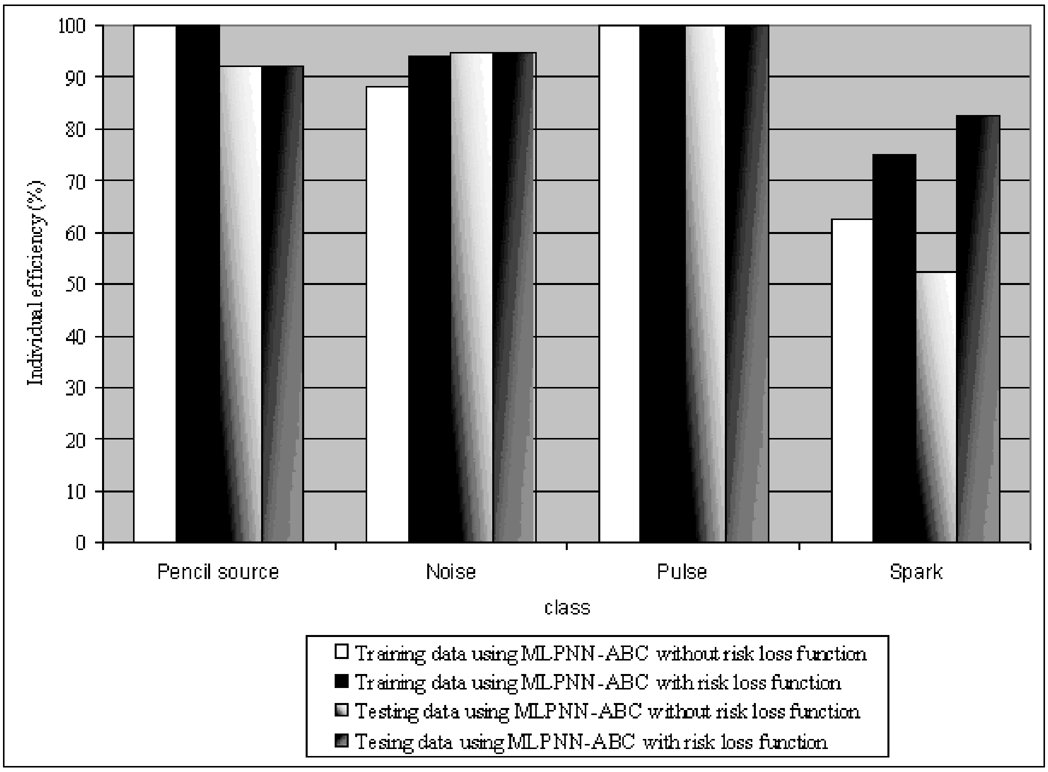


Figure 2. Individual classification efficiencies of training and testing data

4.5.1 MLPNN-ABC without risk sensitive loss function

From the classification matrix for the training data without the risk loss function as depicted in Table 4, we can observe that in the training set, samples belonging to pencil source and pulse are getting classified without any misclassifications and hence have individual efficiencies of 100%. But, between noise and spark, there are some misclassifications. Two samples belonging to noise are misclassified as spark and six samples belonging to spark are misclassified as noise. Hence, noise has an individual efficiency of 88.24%, while spark has an efficiency of 62.5%. However, the overall classification is with an efficiency of 87.1%.

Next, the optimum weight matrix for the given network configuration are applied to the testing data sets and the efficiencies are evaluated. As we can notice from Table 5, the classification matrix for the testing, samples belonging to Pulse are getting classified without any misclassifications and hence have an individual efficiency of 100%. Two samples belonging to pencil source is misclassified one as pulse and the other as spark and hence have an individual efficiency of 92.31%. Samples belonging to Noise and Spark have misclassifications between themselves with three samples of Noise misclassified as Spark and ten samples of Spark as Noise and a single spark sample is misclassified as pulse and hence has an individual efficiency 94.83% and 52.17% respectively. An overall efficiency of 88.32% is recorded which is slightly higher than that obtained for the training data set.

4.5.2 MLPNN-ABC with risk sensitive loss function

For the same training data, the classification matrix with the risk loss function is depicted in Table 6. Here we can observe that in comparison with Table 4, individual efficiency for samples belonging to pencil source, noise and pulse remains the same where as for the samples belonging to spark there is a improvement with individual efficiency of 75%. Also, the overall classification is impressive with an efficiency of 91.94%. It can be clearly observed form the results that there is a marked improvement in the accuracy of the training data sets using risk sensitive loss function.

Next, the weight matrix which is obtained during the training phase is applied to the testing data sets. The classification matrix with the risk loss function is depicted in Table 7. Here we can observe that in comparison with Table 5, individual efficiency for samples belonging to pencil source, noise and pulse remains the same where as for samples belonging to spark there is a remarkable improvement with individual efficiency of 82.61%. However, the overall classification is impressive with an efficiency of 93.43%. Here also we can observe form the results that there is a marked improvement in the individual and overall accuracy of the classification.

The overall accuracy of MLPNN-ABC without risk sensitive loss function is 88.32% whereas MLPNN-ABC with risk loss function the accuracy obtained is 93.43%. This implies that in comparison with MLPNN-ABC without risk sensitive loss function, the modified MLPNN-ABC using risk sensitive loss function for classification has impressive individual and the overall efficiency.

5. CONCLUSIONS

In this paper, we have demonstrated the complexity of acoustic emission signal data using principal component analysis algorithm to evolve the statistical measures for classification and the multilayer perceptron neural network trained using the backpropagation algorithm for classification. For this complex data set, the multilayer perceptron neural network trained using the artificial bee colony algorithm for classification and the modified multilayer perceptron neural network trained using the artificial bee colony based on risk sensitive loss function for multi category pattern classification has been applied. The experiment shows that multilayer perceptron neural network trained using the artificial bee colony not only overcomes the problem of local minima but also able to classify the acoustic emission data set accurately. Further the inclusion of the risk loss function along with multilayer perceptron neural network trained using the artificial bee colony, greatly improves the efficiency of the individual and overall classification. This optimum weight matrix can be used to determine the class of the acoustic emission as soon as the test is conducted. This is very useful in on-the-field non-destructive evaluation. The detection and characterization of the source of the acoustic emission and hence the hidden defects in advanced structures before they grow to a critical size would be a probable use of such methodology.

6. REFERENCES

[1] R. D. Finlayson, M. Freisel, M. Carlos, J.C. Lenain, Health monitoring of aerospace structures with acoustic emission and acousto ultrasonics, Insight - *Journal of the British Institute of Non-destructive testing,* 43 (3), 2001, 155 – 158.

[2] T. Krietsch, J. Bohse, Selection of acoustic emissions and classification of damage mechanisms in fiber composite materials, in: Proceedings of Acoustic Emission World Meeting (AEWM), Kamuela, Hawaii , vol.16, 1998, pp. S233-S242

[3] A. Rabiei, M. Enoki, T. Kishi, A study on fracture behavior of particle reinforced metal matrix composites by using acoustic emission source characterization, *Materials Science and Engineering* A293, 2000, 81–87.

[4] X. Liu, E. Kannatey-Asibu Jr, Classification of AE Signals for Monitoring Martensite Formation From Welding, *Welding Journal*, 69 (10), 1990, 389s-394s.

[5] V. Emamian, M. Kaveh, A. H. Tewfik, Acoustic emission classification for failure prediction due to mechanical fatigue, Smart Structures and Materials 2000: Sensory Phenomena and Measurement Instrumentation for Smart Structures and Materials, vol 3986, 2000, pp.78-84

[6] R. K. Elsley, L. J. Graham, Pattern Recognition Technique Applied to Sorting Acoustic Emission signals. Prod., IEEE Ultrasonic Symp., 1976, pp. 147-150.

[7] R. K. Elsley, L. J. Graham, AE Source Identification by Frequency Spectral analysis for Aircraft Monitoring Applications, *Journal of Acoustic Emission,* 4 (4), 1987, 115-123.

[8] C. R. L. Murthy, B. Dattaguru, A. K. Rao, Application of Pattern Recognition Technique in Acoustic Emission Signal analysis, *Journal of Acoustic Emission*, 6 (1), 1987, 19-28.

[9] C. Roy, A. Maslouhi, D. Gaucher, Z. Piasta, Classification of acoustic emission sources in CFRP assisted by pattern recognition analysis, *Canadian Aeronautics and Space Journal* 34, 1988, 224-232.

[10] J.Z. Sikorska, P.J. Kellyb, J. Pan, Development of an AE data management and analysis system, Mechanical Systems and Signal Processing 20, 2006, 2321–2339.

[11] J. T. Anderson, R. S. Gregory, William T. Collins, Acoustic classification of marine habitats in coastal Newfoundland, *ICES Journal of Marine Science*, 59, 2002, 156–167.

[12] S. Suresh, S. N. Omkar, V. Mani, C. Menaka, Classification of Acoustic Emission Signal Sources using Genetic Programming, *Journal of Aerospace Sciences and Technologies*, 56, 2004, 26-40.

[13] S. N. Omkar, S. Suresh, T. R. Raghavendra, V. Mani, Acoustic Emission Signal Classification using Fuzzy C-means Clustering, Proceedings of the 9^{th} International Conference on Neural Information Processing (ICONIP '02), Singapore, vol. 4, No. 18-22, 2002, pp. 1827 – 1831.

[14] S. N. Omkar, T. R. Raghavendra, Rule extraction for classification of acoustic emission signals using Ant Colony Optimization, *Engineering Applications of Artificial Intelligence*, In Press, Corrected Proof, Available online 26 March 2008.

[15] Tinghu Yan, K. Holford, D. Carter, J. Brandon, Classification of acoustic emission signatures using a self-organization neural network, *Journal of acoustic emission,* 17, 1999, 49-59.

[16] P. Wilkinson, R. L. Reuben, J. D. C. Jones, J. S. Barton, D. P. Hand, T. A. Carolan, S. R. Kidd, Tool wear prediction from acoustic emission and surface characteristics via an artificial neural network, *Mechanical systems and signal processing* 13 (6), 1999, 955-966.

[17] L. I. Smith, A tutorial on Principal Components Analysis (2002).

[18] G. Xuan, P. Chai, X. Zhu, Y. Q. Shi, D. Fu, A Novel Pattern Classification Scheme: Classwise Non-Principal Component Analysis (CNPCA), International Conference on Pattern Recognition (ICPR), vol. 3, 2006, pp. 320 – 323

[19] Koza, J.R, Genetic Programming: On the Programming of Computers by Means of Natural Selection (MIT Press, Cambridge, 1992).

[20] Kishore J.K, Patnaik L.M, Mani V, Agarwal V K, Application of Genetic Programming for multicategory pattern classification, *IEEE Transactions on Evolutionary Computation*, 4, 2000, 242–258.

[21] Dorigo M, Blum C, Ant colony optimization theory: a survey, Theoretical Computer Science, 344, 2005, 243–278.

[22] Parpinelli R.S, Lopes H.S, Freitas A.A, Data mining with an ant colony optimization algorithm, *IEEE Transactions on Evolutionary Computation*, 6, 2002, 321–332.

[23] Manish Mangal, Manu Pratap Singh, Analysis of pattern classificationfor the multidimensional parity-bit-checking problem with hybrid evolutionary feed-forward neural network, *Neurocomputing* 70 (7-9), 2007, 1511-1524.

[24] Niles L, Silverman H, Tajchman G, Bush M, How limited training data can allow a neural network to outperform an 'optimal' statistical classifier, in: Proceedings of ICASSP-89, 1989, pp. 17–20.

[25] Marchant, J. A, Onyango C. M, Comparison of a Bayesian classifier with a multilayer feed-forward neural network using the example of plant/weed/soil discrimination, Computers and Electronics in Agriculture, 39, 2003, 3-22.

[26] S. Haykin, Neural Networks – A Comprehensive Foundation, Second Edition (Macmillan College, New York, 1994).

[27] Eric Bonabeau, Macro Dorigo, Guy Theraulaz, Swarm intelligence-from natural to artificial system (Oxford University Press, New York, 1999).

[28] D. Karaboga, B. Basturk, On the performance of artificial bee colony (ABC) algorithm, *Applied Soft Computing* 8 (1), 2008, 687-697.

[29] D. Karaboga, B. Akay, C. Ozturk, Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks, *Springer Berlin 4617*, 2007, 318-329.

[30] X.S. Yang, Engineering Optimizations via Nature-Inspired Virtual Bee Algorithms, Lecture Notes in Computer Science, vol. 3562 (Springer-Verlag GmbH, 2005) 317-323.

[31] Frisch K.V, Bees: Their Vision, Chemical Senses and Language, Revised Edition (Ithaca, N.Y.: Cornell University Press, 1976).

[32] Von Frisch K, The Dance Language and Orientation of HoneyBees (Harvard University Press, 1967).

[33] D.T. Pham, E. Koç, A. Ghanbarzadeh, S. Otri, Optimisation of the Weights of Multi-Layered Perceptrons Using the Bees Algorithm, in: Proceedings of 5th International Symposium on Intelligent Manufacturing Systems, 2006, pp. 38-46.

[34] V. Vapnik, Statistical Learning Theory (Wiley, New York, 1998).

[35] N. Cristianini, J. S. Taylor, An Introduction to Support Vector Machines (Cambridge University Press, Cambridge, UK, 2000).

[36] M. D. Richard, R. P. Lippmann, Neural network classifiers estimate Bayesian a posteriori probabilities, *Neural Comput*., 3, 1991, 461–483.

[37] J. Cid-Sueiro, J. Arribas, S. Urban-Munoz, A. Figueiras-Vidal, Cost functions to estimate a posteriori probabilities in multi-class problems, *IEEE Trans. Neural Networks* 10, 1999, 645–656.

[38] R. Rojas, A short proof of the posterior probability property of classifier neural networks, *Neural Comput*. 8, 1996, 41–43.

[39] A. Cichocki, R. Unbenhauen, Neural Networks for Optimization and Control (Baffins Lane, U.K.: Wiley, 1993).

[40] S. Suresh, N. Sundararajan, P. Saratchandran, Risk Sensitive Loss Functions for Sparse Multi-Category Classification Problems, *Information Science* 178 (12), 2008, 2621-2638.

[41] M. Kubat, R. C. Holte, S. Matwin, Machine learning for the detection of oil spills in satellite radar images, *Machine Learn*. 30 (2/3), 1998, 195–215.

[42] T. Zhang, Statistical behavior and consistency of classification methods based on convex risk minimization, *Annals of Statistics* 32 (1), 2003, 56-85.

[43] M. Abdul Majeed, *Acoustic Emission Signal Analysis: An Unsupervised Pattern Recognition approach*, Ph.D. Thesis, Indian Institute of Science, Bangalore, 1991.