ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Engineering Applications of Artificial Intelligence 16 (2003) 657–665

Engineering Applications of ARTIFICIAL INTELLIGENCE

www.elsevier.com/locate/engappai

Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection

B. Samanta*, K.R. Al-Balushi, S.A. Al-Araimi

Department of Mechanical and Industrial Engineering, College of Engineering, Sultan Qaboos University, P.O. Box 33, PC 123, Muscat, Sultanate of Oman

Received 28 August 2002; received in revised form 14 May 2003; accepted 18 September 2003

Abstract

A study is presented to compare the performance of bearing fault detection using two different classifiers, namely, artificial neural networks (ANNs) and support vector machines (SMVs). The time-domain vibration signals of a rotating machine with normal and defective bearings are processed for feature extraction. The extracted features from original and preprocessed signals are used as inputs to the classifiers for two-class (normal or fault) recognition. The classifier parameters, e.g., the number of nodes in the hidden layer in case of ANNs and the radial basis function kernel parameter (width) in case of SVMs along with the selection of input features are optimized using genetic algorithms. The classifiers are trained with a subset of the experimental data for known machine conditions and are tested using the remaining set of data. The procedure is illustrated using the experimental vibration data of a rotating machine. The roles of different vibration signals and signal preprocessing techniques are investigated. The results show the effectiveness of the features and the classifiers in detection of machine condition.

Keywords: Condition monitoring; Feature selection; Genetic algorithm; Bearing faults; Neural network; Rotating machines; Signal processing; Support vector machines

1. Introduction

Condition monitoring of machines is gaining importance in industry because of the need to increase reliability and to decrease possible loss of production due to machine breakdown. The use of vibration and acoustic emission (AE) signals is quite common in the field of condition monitoring of rotating machinery. By comparing the signals of a machine running in normal and faulty conditions, detection of faults like mass unbalance, rotor rub, shaft misalignment, gear failures and bearing defects is possible. These signals can also be used to detect the incipient failures of the machine components, through the on-line monitoring system, reducing the possibility of catastrophic damage and the machine down time. Some of the recent works in the area are listed in Shiroishi et al. (1997), McFadden (2000), Nandi (2000), Randall (2001), Al-Balushi and Samanta (2002), Antoni and Randall (2002), McCormick and Nandi (1997) and Dellomo (1999). Although often the visual inspection of the frequency-domain features of the measured signals is adequate to identify the faults, there is a need for a reliable, fast and automated procedure of diagnostics.

Artificial neural networks (ANNs) have been applied in automated detection and diagnosis of machine conditions (Nandi, 2000; McCormick and Nandi, 1997; Dellomo, 1999; Samanta and Al-Balushi, 2001; Samanta and Al-Balushi, 2003) treating these as generalization/classification problems based on learning pattern from examples or empirical data modeling. However, the traditional neural network approaches have limitations on generalization giving rise to models that can overfit the data. This deficiency is due to the optimization algorithms used in ANNs for selection of parameters and the statistical measures used to select the model. Support vector machines (SVMs), based on statistical learning theory, are gaining applications in the areas of machine learning, computer vision and pattern recognition because of the high accuracy and good generalization capability (Burges, 1998; Guyon and Christianini, 1999; Scholkopf, 1998; Gunn, 1998; Vapnik, 1999). The main difference between the ANNs

*Corresponding author. Tel.: +968-515-355; fax: +968-513-315.
E-mail address: samantab@squ.edu.om (B. Samanta).

0952-1976/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/j.engappai.2003.09.006

and SVMs is in the principle of risk minimization (RM). In case of SVMs, structural risk minimization (SRM) principle is used minimizing an upper bound on the expected risk whereas in ANNs, traditional empirical risk minimization (ERM) is used minimizing the error on the training data. The difference in RM is to improve the generalization performance of SVMs compared to ANNs. The possibilities of using SVMs in machine condition monitoring applications are being considered only recently (Nandi, 2000; Jack and Nandi, 2000a, 2001). In Jack and Nandi (2001), a procedure was presented for condition monitoring of rolling element bearings comparing the performance of two classifiers, ANNs and SVMs, with all calculated signal features and fixed parameters for the classifiers. In this, vibration signals were acquired under different operating speeds and bearing conditions. The statistical features of the signals, both original and with some preprocessing like differentiation and integration, low- and high-pass filtering and spectral data of the signals were used for classification of bearing conditions.

However, there is a need to make the classification process faster and accurate using the minimum number of features which primarily characterize the system conditions with an optimized structure of ANNs and SVMs (Nandi, 2000; Jack and Nandi, 2000a). Genetic algorithms (GAs) were used for automatic feature selection in machine condition monitoring (Nandi, 2000; Jack and Nandi, 2000a, b; Jack and Nandi, 2002; Samanta et al., 2001). In Jack and Nandi (2002), the procedure of Jack and Nandi (2001) was extended to introduce a GA-based approach for selection of input features and classifier parameters, like number of neurons in the hidden layer in case of ANNs and the RBF kernel parameter, width, in case of SVMs. The features were extracted from the entire signal under each condition and operating speed. In Samanta et al. (2001), some preliminary results of ANNs and GAs were presented for fault detection of gears using only the time-domain features of vibration signals. In this approach, the features were extracted from finite segments of two signals: one with normal condition and the other with defective gears.

In the present work, the procedure of Samanta et al. (2001) is extended to the diagnosis of bearing condition using vibration signals of longer duration and larger feature space. Comparisons are made between the performance of ANNs and SVMs, both without and with automatic selection of features and classifier parameters. The main difference between the present work and Jack and Nandi (2002) is in the process of feature extraction from the time-domain signal. Fig. 1 shows flow diagram of the proposed procedure. The selection of input features and the classifier parameters are optimized using a GA-based approach. These features, namely, mean, root mean square (rms), variance, skewness, kurtosis and normalized higher order (up to ninth) central moments are used to distinguish between normal and defective bearings. Moments of order higher than nine are not considered in the present work to keep the input vector within a reasonable size without sacrificing the accuracy of diagnosis. The roles of different vibration signals and signal preprocessing techniques are investigated. The results show the effectiveness of the extracted features from the acquired and preprocessed signals in diagnosis of the machine condition. The procedure is illustrated using the vibration data of an experimental setup with normal and defective bearings.

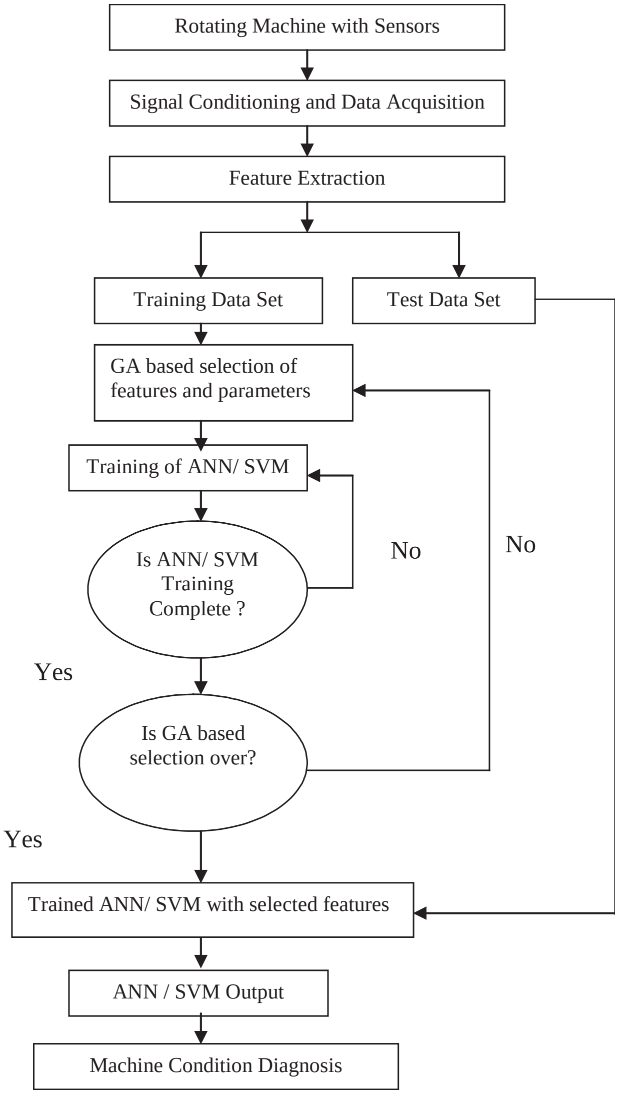
Rotating Machine with Sensors
Signal Conditioning and Data Acquisition
Feature Extraction
Training Data Set
Test Data Set
GA based selection of features and parameters
Training of ANN/ SVM
Is ANN/ SVM Training Complete ?
No
No
Yes
Is GA based selection over?
Yes
Trained ANN/ SVM with selected features
ANN / SVM Output
Machine Condition Diagnosis

Fig. 1. Flow chart of diagnostic procedure.

2. Vibration data

Fig. 2 shows the schematic diagram of the experimental test rig. The rotor is supported on two ball

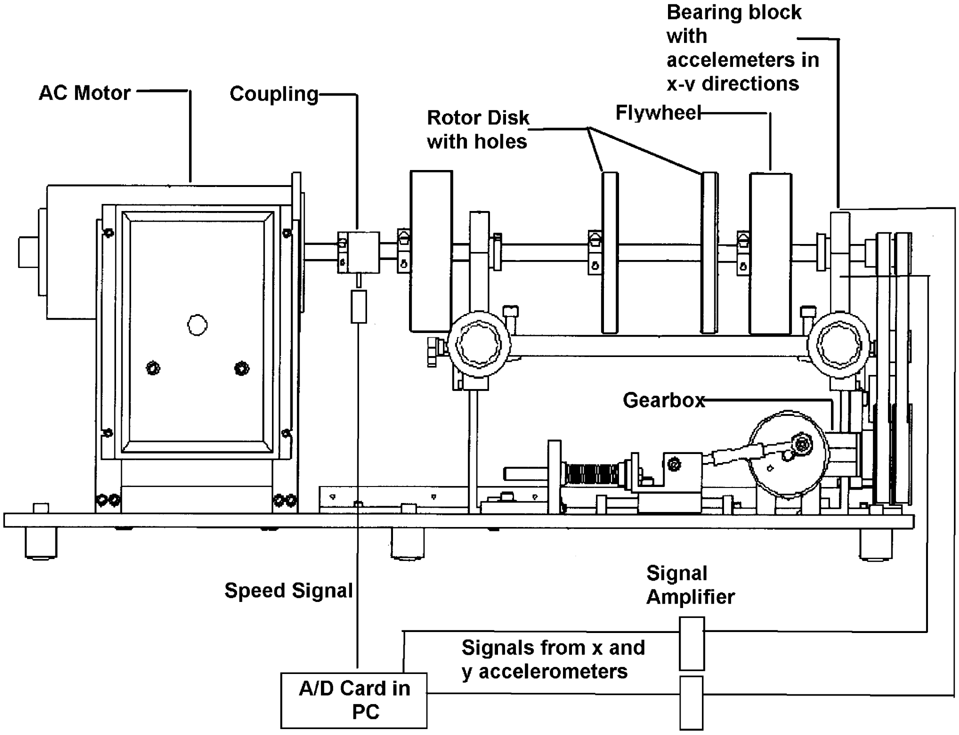

Fig. 2. Experimental test rig.

bearings. The rotor was driven by an electrical AC motor through a flexible coupling. Two accelerometers were mounted on the right-hand side (RHS) bearing support, with an angle of 90° to measure vibrations in vertical and horizontal directions (x and y). Separate measurements were obtained for two conditions, one with normal bearings and the other with a fault on the outer race of the RHS bearing. The accelerometers were connected through charge amplifiers to two channels of a PC-based data acquisition system. The one pulse per revolution of the shaft was sensed by a proximator and the signal was connected to the data acquisition system. Measurements were obtained at a sampling rate of 49152 samples/s. The accelerometer signals were processed through charge amplifiers with lower and higher cut-off frequencies of 2 Hz and 100 kHz, respectively. The number of samples collected for each channel was 49,152. In the present work, these time-domain data were preprocessed to extract the features for using as inputs to the classifiers (ANNs and SVMs).

3. Feature extraction

3.1. Signal statistical characteristics

One set of experimental data each with normal and defective bearings was considered. For each set, 2 vibration signals consisting of 49,152 samples (q_i) were obtained using accelerometers in vertical and horizontal directions to monitor the machine condition. The magnitude of vibration was constructed from the two component signals, $z = \sqrt{(x^2 + y^2)}$. In the present work, these samples were divided into 48 bins of 1024 (n) samples each. Each of these bins was further processed to extract the following features (1–9): mean (μ), rms, variance (σ^2), skewness (normalized 3rd central moment, γ_3), kurtosis (normalized 4th central moment, γ_4), normalized fifth to ninth central moments ($\gamma_5 - \gamma_9$) as follows:

$$\gamma_n = \frac{E\{[q_i - \mu]^n\}}{\sigma^n} \quad n = 3-9, \tag{1}$$

where $E\{\ \}$ represents the expected value of the function. Fig. 3 shows plots of some of these features extracted from the vibration signals (q_i) x, y and z, with each row representing the features for one signal. Only a few of the features are shown as representatives of the full feature set.

3.2. Time derivative and integral of signals

The high- and low-frequency content of the raw signals can be obtained from the corresponding time derivatives and the integrals. In this work, the first time derivative (dq) and the integral (iq) have been defined, using sampling time as a factor, as follows:

$$\mathrm{d}q(k) = q(k) - q(k-1), \tag{2}$$

$$\mathrm{i}q(k) = q(k) + q(k-1). \tag{3}$$

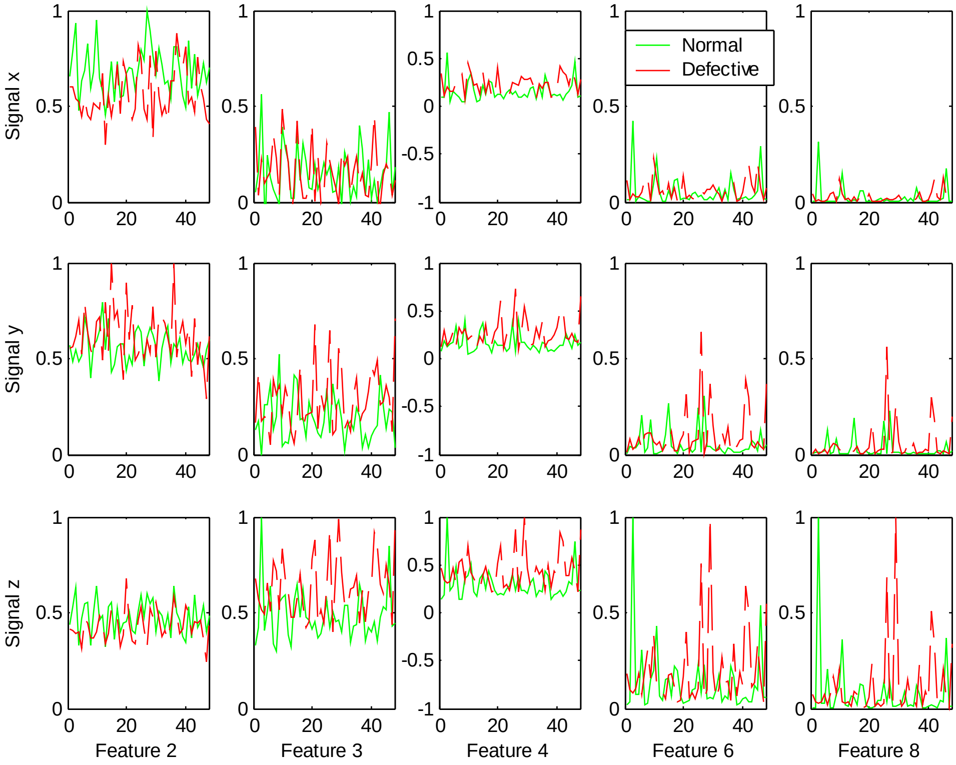

Fig. 3. Time-domain features of acquired signals: —— normal, ------ defective.

These derivative and the integral of each signal were processed to extract additional set of 18 features (10–27).

3.3. High- and low-pass filtering

The raw signals were also processed through low- and high-pass filters with a cut-off frequency as one-tenth ($f/10$) of the sampling rate ($f = 49152$ Hz). These filtered signals were processed to obtain a set of 18 features (28–45).

3.4. Normalization

Each of the features was normalized by dividing each row by its absolute maximum value for better speed and success of the network training. The total set of normalized features consists of 45 × 144 × 2 array where each row represents a feature and the columns represent the number of bins (48) per signal multiplied by the total number of signals (3) and two (2) bearing conditions—normal and defective.

4. Artificial neural network

The feed forward neural network, used in this work, consists of input layer, hidden layer and output layer. The input layer has nodes representing the normalized features extracted from the measured vibration signals. There are various methods, both heuristic and systematic, to select the neural network structure and activation functions (Haykin, 1999). The number of input nodes was varied from 1 to 45 and that of the output nodes was 2. The target values of two output nodes can have only binary levels representing ‘normal’ (N) and ‘failed’ (F) bearings. The inputs were normalized in the range of 0.0–1.0. In the ANN, the activation functions of sigmoid were used in the hidden layers and in the output layer, respectively. The ANN was created, trained and implemented using Matlab neural network toolbox with backpropagation (BPN) and the training algorithm of Levenberg-Marquardt. The ANN was trained iteratively to minimize the performance function of mean square error (MSE) between the network outputs and the corresponding target values. At each iteration, the gradient of the performance function (MSE) was used to adjust the network weights and biases. In this work, a mean square error of 10^{-6}, a minimum gradient of 10^{-10} and maximum iteration number (epoch) of 500 were used. The training process would stop if any of these conditions were met. The initial weights and biases of the network were generated automatically by the program.

5. Support vector machines

SVMs are introduced by Vapnik in the late 1960s on the foundation of statistical learning theory. However, since the middle of 1990s, the algorithms used for SVMs started emerging with greater availability of computing power, paving the way for numerous practical applications (Burges, 1998; Guyon and Christianini, 1999; Scholkopf, 1998; Gunn, 1998; Vapnik, 1999; Weston and Watkins, 1999). The basic SVM deals with two-class problems—in which the data are separated by a hyperplane defined by a number of *support vectors*. A simple introduction of SVM is presented here for completeness. Readers are referred to the tutorials on SVMs (Burges, 1998; Scholkopf, 1998) for details.

The SVM can be considered to create a line or hyperplane between two sets of data for classification. In case of two-dimensional situation, the action of the SVM can be explained easily without any loss of generality. In Fig. 4, a series of points for two different classes of data are shown, circles (class A) and squares (class B). The SVM attempts to place a linear boundary (solid line) between the two different classes, and orient it in such a way that the margin (represented by dotted lines) is maximized. In other words, the SVM tries to orient the boundary such that the distance between the boundary and the nearest data point in each class is maximal. The boundary is then placed in the middle of this margin between the two points. The nearest data points are used to define the margins and are known as *support vectors* (SV, represented by gray circle and square). Once the support vectors are selected, the rest of the feature set can be discarded, since the SVs contain all the necessary information for the classifier.

The boundary can be expressed in terms of

$$(\mathbf{w}\cdot\mathbf{x})+b=0,\quad \mathbf{w}\in R^N,\quad b\in R, \tag{4}$$

where the vector $\mathbf{w}$ defines the boundary, $\mathbf{x}$ is the input vector of dimension N and b is a scalar threshold. At the margins, where the SVs are located, the equations for class A and B, respectively, are

$$(\mathbf{w}\cdot\mathbf{x})+b=1 \tag{5}$$

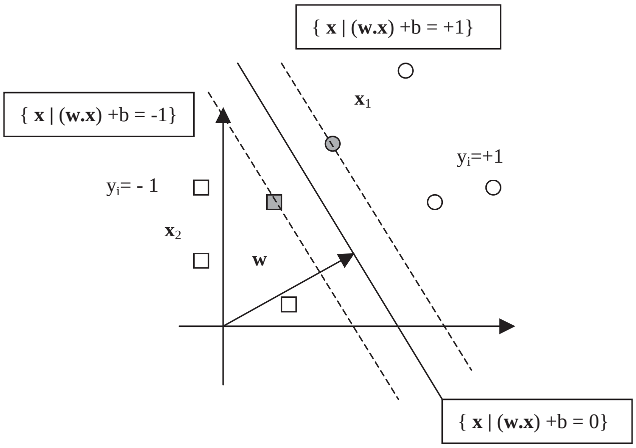

Fig. 4. Classification of data by SVM.

and

$$(\mathbf{w}\cdot\mathbf{x})+b=-1. \tag{6}$$

As SVs correspond to the extremities of the data for a given class, the following decision function holds good for all data points belonging to either A or B:

$$f(x)=\text{sign}((\mathbf{w}\cdot\mathbf{x})+b). \tag{7}$$

The optimal hyperplane can be obtained as a solution to the optimization problem:Minimize

$$\tau(w)=\tfrac{1}{2}\|w\|^2$$

subject to

$$y_i((\mathbf{w}\cdot x_i)+b)\geqslant 1,\quad i=1,\ldots,l, \tag{9}$$

where l is the number of training sets. The solution of the constrained quadratic programming (QP) optimization problem can be obtained as

$$\mathbf{w}=\Sigma v_i x_i, \tag{10}$$

where x_i are SVs obtained from training. Putting (10) in (7), the decision function is obtained as follows:

$$f(x)=\text{sign}\left(\sum_{i=1}^{l} v_i(x\cdot x_i)+b\right). \tag{11}$$

In cases where the linear boundary in input spaces will not be enough to separate two classes properly, it is possible to create a hyperplane that allows linear separation in the higher dimension (corresponding to curved surface in lower dimensional input space). In SVMs, this is achieved through the use of a transformation $\phi(\mathbf{x})$ that converts the data from an N-dimensional input space to Q-dimensional feature space:

$$\mathbf{s}=\phi(\mathbf{x}), \tag{12}$$

where $\mathbf{x}\in R^N$ and $\mathbf{s}\in R^Q$. Fig. 5 shows the transformation from input space to feature space where the nonlinear boundary has been transformed into a linear boundary in feature space. Substituting the transformation in (11) gives

$$f(x)=\text{sign}\left(\sum_{i=1}^{l} v_i(\phi(x)\cdot\phi(x_i))+b\right). \tag{13}$$

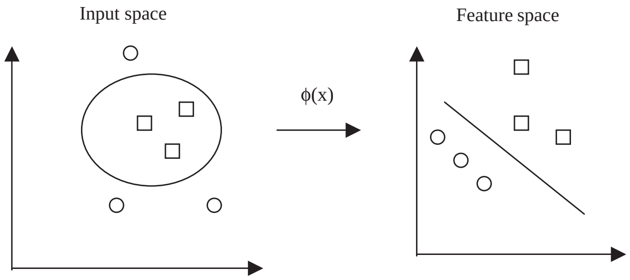

Fig. 5. Nonlinear separation of input and feature space.

The transformation into higher-dimensional feature space is relatively computation-intensive. A kernel can be used to perform this transformation and the dot product in a single step provided the transformation can be replaced by an equivalent kernel function. This helps in reducing the computational load and at the same time retaining the effect of higher-dimensional transformation. The kernel function $K(\mathbf{x}\cdot\mathbf{y})$ is defined as

$$K(\mathbf{x}\cdot\mathbf{y}) = \phi(\mathbf{x})\cdot\phi(\mathbf{y}). \tag{14}$$

The decision function is accordingly modified as

$$f(x) = \operatorname{sign}\left(\sum_{i=1}^{l} v_i K(x\cdot x_i) + b\right). \tag{15}$$

The parameters v_i are used as weighting factors to determine which of the input vectors are actually support vectors ($0<v_i<\infty$). There are different kernel functions like polynomial, sigmoid and radial basis function (RBF) used in SVM. In the present work, RBF kernel, given by Eq. (16) is used.

$$K(x,y) = \exp(-|x-y|^2/2\sigma^2). \tag{16}$$

The width of the RBF kernel parameter (σ) can be determined in general by an iterative process selecting an optimum value based on the full feature set.

In case there is an overlap between the classes with nonseparable data, the range of parameters v_i can be limited to reduce the effect of outliers on the boundary defined by SVs. For nonseparable case, the constraint is modified ($0<v_i<C$). For separable case, C is infinity while for nonseparable case, it may be varied, depending on the number of allowable errors in the trained solution: few errors are permitted for high C while low C allows a higher proportion of errors in the solution.

To control generalization capability of SVM, there are a few free parameters like limiting term C and the kernel parameters like RBF width σ. In the present work, C is chosen as 100 and the kernel parameter σ and features are selected using GA-based approach. The numerical implementation of SVM is mainly based on QP with options of decomposing a large-scale QP problems into a series of smaller-size QP problems (Platt, 1998; Campbell, 2000; Joachims, 1999). In the present work, the SVMs are trained using an adapted version of decomposition methods and working set selection strategies similar to that of Joachims (1999).

6. Genetic algorithms

Genetic algorithms (GAs) have been considered with increasing interest in a wide variety of applications (Goldberg, 1989; Michalewicz, 1999; Tang et al., 1996). These algorithms are used to search the solution space through simulated evolution of 'survival of the fittest'. These are used to solve linear and nonlinear problems by exploring all regions of state space and exploiting potential areas through mutation, crossover and selection operations applied to individuals in the population (Michalewicz, 1999). The use of genetic algorithm needs consideration of six basic issues: chromosome (genome) representation, selection function, genetic operators like mutation and crossover for reproduction function, creation of initial population, termination criteria, and the evaluation function (Goldberg, 1989; Michalewicz, 1999; Tang et al., 1996). These issues, in the context of the present work, are briefly discussed in this section.

A population size of ten individuals was used starting with randomly generated genomes. This size was chosen to ensure relatively high interchange among different genomes and to reduce the likelihood of convergence within the population. GA was used to select the most suitable features and one variable parameter related to the particular classifier: the number of neurons in the hidden layer for ANN and the RBF kernel width (σ) for SVM. For a training run needing N different inputs to be selected from a set of Q possible inputs, the genome string would consist of $N+1$ real numbers. The first N numbers ($x_i, i=1,N$) in the genome are constrained to be in the range $1\leqslant x_i\leqslant Q$ for both ANNs and SVMs

$$X = \{x_1 x_2 \ldots x_N x_{N+1}\}^{\mathrm{T}}. \tag{17}$$

The last number x_{N+1} has to be within the range $S_{\min}\leqslant x_{N+1}\leqslant S_{\max}$. The parameters $S_{\min}$ and $S_{\max}$ represent, respectively, the lower and the upper bounds on the classifier parameter.

A probabilistic selection function, namely, normalized geometric ranking (Houk et al., 1995) was used such that the better individuals, based on the fitness criterion in the evaluation function, have higher chance of being selected. Nonuniform-mutation function (Michalewicz, 1999) using a random number for mutation based on current generation and the maximum generation number, among other parameters was adopted. Heuristic crossover (Michalewicz, 1999) producing a linear extrapolation of two individuals based on the fitness information was chosen. The maximum number of generations was adopted as the termination criterion for the solution process. The classification success for the test data was used as the fitness criterion in the evaluation function.

7. Simulation results

The dataset ($45\times144\times2$) consisting of 45 normalized features for each of the three signals split in form of 48 bins of 1024 samples each with two bearing conditions were divided into two subsets. The first 24 bins of each signal was used for training the ANNs and SVMs giving a training set of $45\times72\times2$ and the rest ($45\times72\times2$) was used for validation. For ANNs, the target value of the first output node of was set 1 and 0 for normal and failed bearings, respectively, and the values were interchanged (0 and 1) for the second output node. For SVMs, the target values were specified as 1 and −1, respectively, representing normal and faulty conditions. Results are presented to see the effects of sensor location and signal processing for diagnosis of machine condition using ANNs and SVMs without and with feature selection based on GA. The training success for most cases was 100%.

7.1. *Performance comparison of ANNs and SVMs without feature selection*

In this section classification results are presented for straight ANNs and SVMs without feature selection. For each case of straight ANNs, number of neurons in the hidden layer was kept at 24 and for straight SVMs, a constant width (σ) of 0.5 was used. These values were chosen on the basis of training trials.

7.1.1. *Effect of sensor location*

Table 1 shows the classification results for each of the signals using all the input features (1–45). The test success of SVMs was slightly higher than with ANNs for signals x and z. The test success was in the range of 81.25–93.75% for ANNs and 75.00–95.83% for SVMs.

Table 1
Performance comparison of classifiers without feature selection for different sensor locations

Dataset	Input features	Test success (%)	
		ANN ($N = 24$)	SVM ($\sigma = 0.50$)
Signal x	1–45	93.75	95.83
Signal y	1–45	81.25	75.00
Signal z	1–45	85.42	95.83

7.1.2. *Effect of signal pre-processing*

Table 2 shows the effects of signal processing on the classification results for straight ANNs and SVMs with all three signals ($x-z$). In each case, all the features from the signals without and with signal processing were used. The test success of straight ANNs was in the range of 84.03–98.61% whereas it was 88.89–97.92% for SVMs. For both types of classifiers, test success was worst with the features from original signals (without preprocessing), whereas performance was best with the features from derivative and integral of the signals.

Table 2
Performance comparison of classifiers without feature selection for different signal preprocessing

Dataset	Input features	Test success (%)	
		ANN ($N = 24$)	SVM ($\sigma = 0.50$)
Signals $x-z$	1–9	84.03	88.89
Derivative/integral	10–27	98.61	97.92
High-/low-pass filtering	28–45	96.88	95.14

7.2. *Performance comparison of ANNs and SVMs with feature selection*

In this section, classification results are presented for ANNs and SVMs with feature selection based on GA. In each case, only three features were selected from the corresponding range of input features. In case of ANNs, the number of neurons in the hidden layer was selected in the range of 10–30 whereas for SVMs, the RBF kernel width (σ) was selected in the range of 0.2–2.0 with a step size of 0.2. The ranges and increments were chosen based on trial runs keeping the computation at reasonable level.

7.2.1. *Effect of sensor location*

Table 3 shows the classification results along with the selected parameters for each of the signals. In all cases, the input features were selected by GA from the entire range (1–45). The improvement in test success was substantial for signals x and z with feature selection approach for both ANNs and SVMs. Test success was 100% for signal z for both types of classifiers. This success may be explained as signal z combines the characteristics of signals in both horizontal (x) and vertical (y) directions. Though there were some common input features selected by both classifiers, the selected feature sets were not identical. This may be explained as the difference in roles of the feature sets in two classifiers.

7.2.2. *Effect of signal pre-processing*

Table 4 shows the effects of signal processing on the classification results for all signals ($x-z$) with GA. In all cases, only three features from the signals without and with signal pre-processing were used from each of these ranges. The test success was in the range of 82.64–99.31% for ANNs and 87.50–97.22% for SVMs. However, the performance with features selection was still not satisfactory. This may be due to the small number (3) of features selected.

7.3. *Effect of number of features*

Table 5 shows the results with different numbers of features for both types of classifiers. The classification performance of SVM (98.61%) was substantially better than the ANN (85.06%) without any feature selection. For each case, features were selected from the entire range (1–45). The performance of ANN was slightly better than SVM with three features selected. However, for both classifiers, test success was 100% with six

Table 3
Performance comparison of classifiers with feature selection for different sensor locations

Dataset	GA with ANN			GA with SVM (3 features)		
	Input features	No. of hidden neurons	Test success (%)	Input features	Width (σ)	Test success (%)
Signal x	13, 19, 42	10	97.92	20, 41, 42	0.20	97.92
Signal y	27, 33, 41	26	93.75	3, 11, 39	0.20	81.25
Signal z	21, 40, 41	18	100	11, 21, 44	0.20	100

Table 4
Performance comparison of classifiers with feature selection for different signal preprocessing

Dataset	GA with ANN			GA with SVM (3 features)		
	Input features	No. of hidden neurons	Test success (%)	Input features	Width	Test success (%)
Signals 1–4	4, 5, 6	27	82.64	1, 5, 6	1.40	87.50
Derivative/integral	14, 15, 22	25	98.61	14, 16, 20	0.60	97.22
High-/low-pass filtering	39, 41, 42	10	99.31	30, 31, 39	0.20	97.22

Table 5
Performance comparison of classifiers with different number of features

Classifier	Number of features	Features	Classifier parameter (N/σ)	Test success (%)
Straight ANN	45	1–45	24	85.06
GA with ANN	3	4, 14, 18	21	99.31
GA with ANN	6	5, 13, 23, 30, 32, 39	25	100
Straight SVM	45	1–45	0.50	98.61
GA with SVM	3	19, 37, 38	0.80	98.61
GA with SVM	6	5, 23, 24, 30, 33, 39	0.20	100

features selected. Four of the six selected features were same in both classifiers. Computation time (on a PC with Pentium III processor of 533 MHz and 64 MB RAM) for SVM training was in the range of 0.581–1.622 s which was relatively much lower than the ANNs. However, direct comparison in computation time is difficult due to the difference in code efficiency. The training time for SMVs was also in the similar range for results in Tables 3 and 4.

8. Conclusions

A procedure is presented for diagnosis of bearing condition using two classifiers, namely, ANNs and SVMs with GA-based feature selection from time-domain vibration signals. The selection of input features and the appropriate classifier parameters have been optimized using a GA-based approach. The roles of different vibration signals and signal preprocessing techniques have been investigated. The performance of SVM have been found to be substantially better than ANN with the entire feature set. The use of GAs with only six features gave 100% classification for both ANNs and SVMs. The training time was substantially less for SVMs than ANNs. The results show the potential application of GAs for selection of features and classifier parameters in machine condition detection.

Acknowledgements

The financial support from Sultan Qaboos University grant IG/ENG/MIED/01/01 to carry out the research is gratefully acknowledged.

References

Al-Balushi, K.R., Samanta, B., 2002. Gear fault diagnosis using energy-based features of acoustic emission signals. Proceedings of IMechE, Part I: Journal of Systems and Control Engineering 216, 249–263.

Antoni, J., Randall, R.B., 2002. Differential diagnosis of gear and bearing faults. Transactions of the ASME, Journal of Vibration and Acoustics 124, 165–171.

Burges, C.J.C., 1998. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 2, 955–974.

Campbell, C., 2000. Algorithmic approaches to training support vector machines: a survey. Proceedings of ESANN2000 Bruges, Belgium, Paper #ESANN2000-355.

Dellomo, M.R., 1999. Helicopter gearbox fault detection: a neural network-based approach. Transactions of the ASME, Journal of Vibration and Acoustics 121, 265–272.

Goldberg, G.E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, NY, USA.

Gunn, S.R., 1998. Support vector machines for classification and regression. Technical report, University of Southampton, Department of Electrical and Computer Science.

Guyon, I., Christianini, N., 1999. Survey of support vector machine applications. http://www.clopinet.com/isabelle/Projects/SVM/applist.html, Proceedings of NIPS'99 Special Workshop on Learning with Support Vector.

Haykin, S., 1999. Neural Networks: A Comprehensive Foundation, second ed. Prentice-Hall, NJ, USA.

Houk, C.R., Joines, J., Kay, M., 1995. A genetic algorithm for function optimization: a matlab implementation. North Carolina State University, Report no: NCSU_IE_TR_95_09.

Jack, L.B., Nandi, A.K., 2000a. Comparison of neural networks and support vector machines in condition monitoring applications. Proceedings of COMADEM 2000, Houston, TX, USA, pp. 721–730.

Jack, L.B., Nandi, A.K., 2000b. Genetic algorithms for feature extraction in machine condition monitoring with vibration signals. IEE Proceedings of Vision & Image Signal Processing 147, 205–212.

Jack, L.B., Nandi, A.K., 2001. Support vector machines for detection and characterisation of rolling element bearing faults. Proceedings of Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 215, 1065–1074.

Jack, L.B., Nandi, A.K., 2002. Fault detection using support vector machines and artificial neural networks, augmented by genetic algorithms. Mechanical Systems and Signal Processing 16, 373–390.

Joachims, T., 1999. Making large-scale SVM learning practical. In: Scholkopf, B., Burges, C.J., Simola, A. (Eds.), Advances in Kernel Methods-Support Vector Learning. MIT Press, Cambridge, MA, USA, pp. 169–184.

McCormick, A.C., Nandi, A.K., 1997. Classification of the rotating machine condition using artificial neural networks. Proceedings of IMechE, Part C: Journal of Mechanical Engineering Science 211, 439–450.

McFadden, P.D., 2000. Detection of gear faults by decomposition of matched differences of vibration signals. Mechanical Systems and Signal Processing 14, 805–817.

Michalewicz, Z., 1999. Genetic Algorithms + Data Structures = Evolution Programs, third ed. Springer, New York, USA.

Nandi, A.K., 2000. Advanced digital vibration signal processing for condition monitoring. Proceedings of COMADEM 2000, Houston, TX, USA, pp. 129–143.

Platt, J., 1998. How to implement SVMs. IEEE Intelligent Systems 13, 26–28.

Randall, R.B. (guest ed.), 2001. Special issue on gear and bearing diagnostics. Mechanical Systems and Signal Processing 15(5), 827–993.

Samanta, B., Al-Balushi, K.R., 2001. Use of time domain features for the neural network based fault diagnosis of a machine tool coolant system. Proceedings of IMechE, Part I: Journal of Systems and Control Engineering 215, 199–207.

Samanta, B., Al-Balushi, K.R., 2003. Artificial neural network based fault diagnostics of rolling element bearings using time-domain features. Mechanical Systems and Signal Processing 17 (2), 317–328.

Samanta, B., Al-Balushi, K.R., Al-Araimi, S.A., 2001. Use of genetic algorithm and artificial neural network for gear condition diagnostics. Proceedings of COMADEM 2001, University of Manchester, UK, pp. 449–456.

Scholkopf, B., 1998. SVMs—a practical consequence of learning theory. IEEE Intelligent Systems 13, 18–19.

Shiroishi, J., Li, Y., Liang, S., Kurfess, T., Danyluk, S., 1997. Bearing condition diagnostics via vibration and acoustic emission measurements. Mechanical Systems and Signal Processing 11, 693–705.

Tang, K.S., Man, K.F., Kwong, S., He, Q., 1996. Genetic algorithms and their applications. IEEE Signal Processing Magazine 13, 22–37.

Vapnik, V.N., 1999. An overview of statistical learning theory. IEEE transactions on Neural Networks 10, 988–1000.

Weston, J., Watkins, C., 1999. Support vector machines for multi-class recognition. Proceedings of ESANN'99, Bruges, Belgium, pp. 219–224.