Condition Diagnosis of Bearing System Using Multiple Classifiers of ANNs and Adaptive Probabilities in Genetic Algorithms

[1]LILI A. WULANDHARI, [2]ANTONI WIBOWO, [3]MOHAMMAD I. DESA

[1]School of Computer Science, Bina Nusantara University, 11480 Jakarta, INDONESIA,
[2]School of Quantitative Sciences, UUM College of Arts and Sciences, Universiti Utara Malaysia, 06010 Sintok Kedah, MALAYSIA,
[3]Advanced Informatics School (AIS), Universiti Teknologi Malaysia, 54100 Kuala Lumpur, MALAYSIA
[1]lwulandhari@binus.edu, [2]antoni@uum.edu.my , [3]mishak@utm.my

Abstract: - Condition diagnosis in bearing systems needs an effective and precise method to avoid unacceptable consequences from total system failure. Artificial Neural Networks (ANNs) are one of the most popular methods for classification in condition diagnosis of bearing systems. Regarding to ANNs performance, ANNs parameters have important role especially connectivity weights. In several running of learning processes with the same structure of ANNs, we can obtain different accuracy significantly since initial weights are selected randomly. Therefore, finding the best weights in learning process is an important task for obtaining good performance of ANNs. Previous researchers have proposed some methods to get the best weights such as simple average and majority voting. However, these methods have some limitations in providing the best weights especially in condition diagnosis of bearing systems. In this paper, we propose a hybrid technique of multiple classifier-ANNs (mANNs) and adaptive probabilities in genetic algorithms (APGAs) to obtain the best weights of ANNs in order to increase the classification performance of ANNs in condition diagnosis of bearing systems. The mANNs are used to provide several best initial weights which are optimized by APGAs. The set optimized weights from APGAs, afterward, are used as the best weights for condition diagnosis. Our experiment shows mANNs-APGAs give better results than of the simple average and majority voting in condition diagnosis of bearing systems. This experiment also shows the distinction of maximum and minimum accuracy in mANNs-APGAs is lower than the two existing methods.

Keywords: - Adaptive Probabilities Genetic Algorithms, Bearing Systems, Condition Diagnosis, Majority Voting, Multiple Artificial Neural Networks, Simple Average.

1 Introduction

A fault is a condition where an unexpected distinction occur in minimal one of component or parameter characteristic from the acceptable, usual or standard condition[1]. In modern industrial plant, this unexpected condition can lead the total failure of the whole system. Therefore, an effective condition diagnosis is needed to detect faults much earlier and unacceptable consequences from total system failure can be avoided. Currently, the condition diagnosis system applies two main directions in its development, they are hardware redundancy and analytical redundancy [2]. Hardware redudancy applies reduplication of physical devices and usually a voting system to detect the occurence of a fault and its location in the system. The disadvantages of this approach is the significant cost for the necessary extra equipment. Analytical redudancy uses redundant functional relationship between variables of the system. The advantage of this method is it does not need extra equipment to detect the occurence of a fault.

In analytical redudancy approach of the condition diagnosis system, faults are identified through a comparison between measured signal with estimated values [3].This estimated values are the result of mathematical model of the monitored systems. Therefore in order to obtain a good performance of the system it requires very accurate mathematical model. An error model can affect to the performance directly, especially when the monitored system is nonlinear. In other hand, modelling a nonlinear system of the critical system is not easy, it must be deal with the complexity of the system. Due to the complexity of analytical redudancy, some researchers develop the artificial intellegence approach for condition diagnosis system, i.e., neural networks, fuzzy logic, evolutionary algorithm such as genetic algorithm and Particle Swarm Optimization (PSO**)**.

Artificial neural networks (ANNs) are one of the most popular methods which are used many researchers and still used recently. They are applied in several fields, for the instances in electrical field to diagnose the condition of circuit [4-7], plant fault diagnosis [8-10] and mechanical fault diagnosis application [11-13]. ANNs are used to model the behaviours of the system and classify the classes of those behaviours. ANNs are suitable tool for modelling the behaviours of a system due to they have these three important characteristics: generalization ability, noise tolerance and fast response once trained [14]. Even if the training data are affected by noise, ANNs is still able to generalize the system behaviour with the level of accuracy is being proportional to the level of noise [2].

Generally, ANNs performance is influenced by one of parameters namely connectivity weights [15]. Connectivity weights has important role in providing a good performance of ANNs in condition diagnosis. If we run ten times a certain ANN with the same parameters, says ANN[1] 30-30-30-30-16, we can obtain different accuracy significantly since we use initial weights randomly. It is indicated by the range of the minimum and maximum point of accuracy which is significantly different as shown in Figure 1.

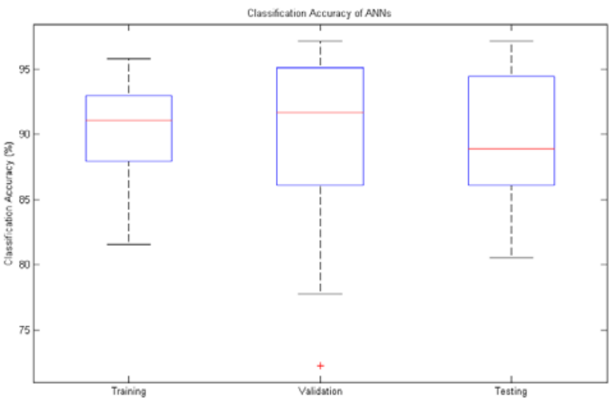


Fig. 1 Classification accuracy of ANNs learning process experiments

From Figure 1, we can see that the minimum accuracy of training is around 82% and the maximum is around 96%, it has distinction around 14%. The validation and test accuracy as well has high distinction between minimum and maximum values that is around 25% and 16% repectively.This implies the ANN 30-30-30-30-16 performances are unstable due to the final weights of ten running ANN 30-30-30-30-16 are different and do not converge to an optimum weights.The final weights of ten ANN 30-30-30-30-16 for the classification are different since initial weights are selected randomly.

To overcome the above limitation, some researchers proposed multiple-ANNs approach. Hashem and Schimeiser [16] proposed linear combination method called mean square error-optimal linear combination (MSE-OLC). They proposed a technique to find optimal combination weights which satisfies minimum MSE. However this method needs the additional computational effort to estimate the optimal combination weights [16].The other methods are simple average and majority voting which are two of the common combination rules. Both of the methods can decrease the error of the classification from 27.1% into 25.8% for majority voting and 25.5 % for simple averaging respectively [17].

Simple average[15, 18, 19] is a method which combine the weights from each classifier of mANNs by finding their averages. This method is straightforward techniques, but we have to assumes that all the weights of classifiers involved are equally good [15]. Unfortunately, we can find that the weights of classifier in mANNs are not equally good. Hence, the average of combination weights is not guaranted as the best one.Therefore, we have possibility to obtain the classification accuracy of simple average is less than the accuracy of the best classifier obtained in the mANNs.While majority voting is a method in which the set of weights of maximum accuracy classifier is assigned as the best weights [18, 20-23]. However this method has limitations in determining which the best accuracy, since ANNs learning process involves training, validation and testing process. ANNs do not guarantee if the training accuracy is good then the validation and testing accuracy are also good. It means that it is difficult to determine the best structure of ANNs for our classification as shown in Figure 1. Therefore, it is needed an alternative method to overcome the limitation of the simple average and majority voting.

In this paper, we propose a hybrid technique of mANNs and APGAs to find the best weights of mANNs in order to improve the classification performance in condition diagnosis of bearing system. The mANNs are used to provide several best initial weights which are optimized by APGAs.

[1]Refer to Section 3 for topology 30-30-30-30-16.

These optimized weights from APGAs, afterward, are used as the best weights for condition diagnosis. Our proposed method will give the better set of weights compared to the simple average and majority voting. Since the average of combination weights is not guaranted as the best one implying that the accuracy of ANNs using simple average can be lower than the accuracy of ANNs using majority voting. Whereas our proposed method use the set of ANNs' weights from majority voting as one candidate solution in APGAs. In other words, the set of weights from majority voting is assigned as one chromosome in inital population of APGAs. Afterward, APGAs try to find the better solution through several generations implying that our proposed method gives at least the same result with majority voting.The rest of this paper section is arranged as follows: Section 2 is explained the theory and methodology of mANNs-APGAs algorithm. In Section 3 we present the results and discussion of simple average, majority voting and mANNs-APGAs in condition diagnosis of bearing systems. Finally, the conclusion will be presented in Section 4.

2 Theory and Methodology

In this paper we propose a hybrid of mANNs-APGAs to diagnose the condition of bearing systems. The framework of mANNs-APGAs is given in Figure 2. From this figure, the *n* ANNs classifiers are trained to obtain the *n* best candidate set of weights. The *n* set of weights are assigned as chromosomes in initial population of APGAs. Afterward, APGAs try to find a set of best weights through their operators. We use the adaptive probabilities technique in GAs to maintain the diversity of the population by varying the probability of crossover (p_c) and mutation (p_m) [24-28]. Maintaining diversity of the population are useful to prevent GAs converge prematurely. Premature convergence leads GAs stuck in local optima which might be not the best solution of GAs [25]. After a certain generation of APGAs, we obtain the better set of weights compared to weights in initial population. It means that the obtained set of weights of APGAs is better than the *n* set of weight from *n* ANNs classifiers.

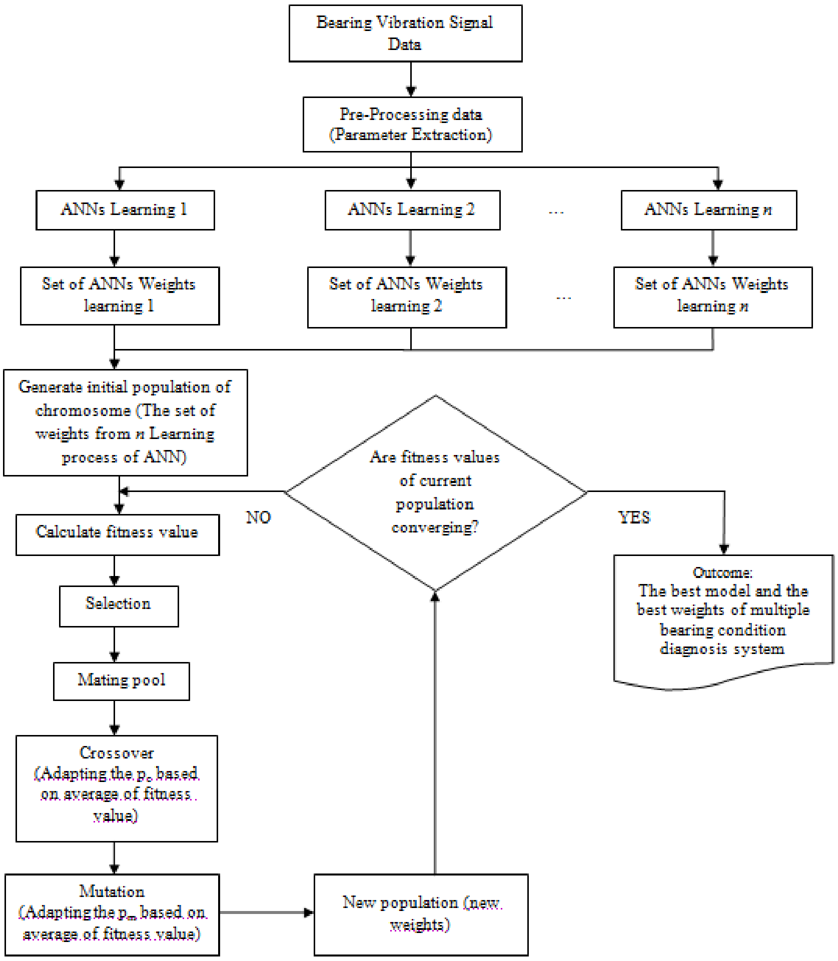


Fig. 2 The scheme of ANNs-APGAs in condition diagnosis system

2.1 Pre-processing of bearing vibration data

Bearings are parts in machine that are used to support rotating shaft. Appropriate bearing design can minimize the friction and its failure may cause expensive loss of production [29]. Unfortunately, bearing is one of machine parts which has a high percentage of defect compared to the other component such as stator winding and rotor [30]. Therefore, an early and effective fault diagnosis of bearing is an essential task.

In this paper, the vibration signal data used were obtained from the Case Western Reserve University Bearing Data Center [31]. The vibration data was recorded from ball bearing of Drive End (DE) and Fan End (FE) motors [31], the specification of the bearings are given in Table 1. Three accelerometers were attached to the housing with magnetic base; the structure of the bearings and accelerometers is presented in Figure 3. These accelerometers record seven conditions of vibration data. They are both bearings in normal condition, Fan End Bearing Inner Race Fault (FE-IRF), Drive End Bearing Inner Race Fault (DE-IRF), Fan End Bearing Outer Race fault (FE-ORF), Drive End Bearing Outer Race fault (DE-ORF), Fan End Bearing Ball Fault (FE-BF) and Drive End Bearing Ball Fault (DE-BF). Table 2 shows the example of bearing vibration data from FE, DE and base accelerometer (BA).

Table 1 The specification of the bearings [31]

Bearing	Inside Diameter (inches)	Outside diameter (inches)	Thickness (inches)	Ball diameter (inches)	Pitch diameter (inches)
DE bearing	0.9843	2.0472	0.5906	0.3126	1.537
FE bearing	0.6693	1.5748	0.4724	0.2656	1.122

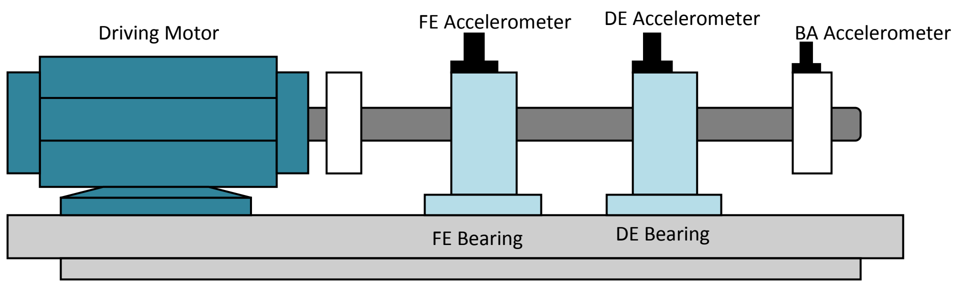


Fig. 3 The Structure of bearings and accelerometers

These vibration data are extracted into ten features namely standard deviation, skewness, kurtosis, the maximum peak value, absolute mean value, root mean square value, crest factor, shape factor, impulse factor and clearance factor [32]. A shown by Figure 3, the data is recorded from three accelerometers and these data are extracted into ten features. Based on this scheme we have thirty parameters for the input of the algorithm. It means that we have thirty neurons for the input layer of the ANNs.

The input parameters are classified into sixteen condition classes (see Table 3). Refer to Table 2; we can see that each condition has three kinds of data from three accelerometers. So that normally, we only have seven condition classes for this kind of data. However in this paper we expand into sixteen classes by mixing and combining the available data. For the instance the classes of FE-IRF and DE IRF, for this classes we use the data from FE-IRF accelerometer for FE column and DE-IRF accelerometer for DE column. While for BA data we use the average between BA data of FE-IRF and DE IRF. This expansion of classes aim to yield specific condition from the bearings system, therefore the precise and effective diagnosis can be achieved.

Table 2 Example of bearing vibration data

No	Normal Bearing		Drive End Bearing Fault Data (DE)									Fan End Bearing Fault Data (FE)								
			Inner race fault (IRF)			Ball Fault (BF)			Outer race Fault (ORF)			Inner race fault (IRF)			Ball Fault (BF)			Outer race Fault (ORF)		
	DE	FE	BA	DE	FE	BA	DE	FE	BA	DE	FE	BA	DE	FE	BA	DE	FE	BA	DE	FE
1	0.053	0.146	0.065	-0.083	-0.402	0.016	-0.003	-0.247	0.000	0.009	-0.407	0.098	-0.025	-0.051	0.017	-0.168	0.320	-0.031	-0.134	0.127
2	0.089	0.098	-0.023	-0.196	-0.005	0.017	-0.096	0.143	0.069	0.424	0.263	0.042	-0.029	-0.192	-0.004	0.181	0.326	-0.120	0.003	-0.259
3	0.100	0.055	-0.089	0.233	-0.107	-0.036	0.114	0.003	0.031	0.013	0.495	-0.042	-0.046	0.051	-0.169	0.044	-0.260	-0.006	-0.027	-0.060
4	0.059	0.037	-0.094	0.104	-0.074	-0.045	0.257	-0.107	-0.037	-0.265	-0.423	0.081	0.001	0.151	-0.069	-0.270	0.031	0.060	-0.184	0.454
5	-0.005	0.054	-0.076	-0.181	0.209	0.008	-0.058	0.136	-0.116	0.237	-0.307	0.059	-0.037	-0.095	0.090	-0.138	0.447	-0.131	-0.203	0.075

Table 3 Sixteen classes of bearing conditions

No	Condition	No	Condition
C1	FE and DE Normal	C9	FE-IRF and DE-ORF
C2	FE Normal and DE-IRF	C10	FE-IRF and DE-BF
C3	FE Normal and DE-ORF	C11	FE-ORF and DE-IRF
C4	FE Normal and DE-BF	C12	FE-ORF and DE-ORF
C5	FE-IRF and DE Normal	C13	FE-ORF and DE-BF
C6	FE-ORF and DE Normal	C14	FE-BF and DE-IRF
C7	FE-BF and DE Normal	C15	FE-BF and DE-ORF
C8	FE-IRF and DE-IRF	C16	FE-BE and DE-BF

2.2 Hybrid Multiple ANNs-APGAs

This section describes the hybrid of mANNs-APGAs algorithm as follows:

1. Let (I_k, T_k) be the kth input and target pair of the problem to be solved by ANN, with $k=1,2,\ldots,N_{in}$ and N_{in} is the number of paired data.
2. Let N_{pop}, N_{chro}, p_{c0}, p_{m0}, and N_{iter} be the number of populations, number of chromosomes, initial crossover probability, initial mutation probability, and the maximum number of iterations, respectively. Initialize p_{c0}, p_{m0}, R_{pc} and R_{pm} where R_{pc} are random vector of numbers which generated in range [0, 1] with size $1 \times N_{chro}/4$ and R_{pm} are random vector of numbers which generated in range [0, 1] with size $1 \times N_{chro}/2$. Set $i=0$.
3. Determine the ANN architecture in term of the number of input neuron, hidden layer, hidden neuron and output neuron, and the activation functions.
4. Assume the ANN learning 1, ANN learning 2,…,ANN learning n have the same structure. Then execute the ANN learning 1, ANN learning 2,…,ANN learning n where n is number of mANNs classifiers.
5. Extract the set of weights from each learning process and assign them as initial population Q_0 in APGAs.
6. Calculate the fitness value $F(i,j)$ of the jth chromosome in population Q_i using

$$F(i,j)=\frac{1}{E(i,j)} \qquad j=1,2,\ldots,N_{chro} \tag{1}$$

where $E(i,j)$ is Mean Square Error (MSE) of the jth chromosome in the population Q_i. It is calculated based on the selected BPPN architecture as follows

$$E(i,j)=\frac{1}{2}\sum_{k=1}^{N_{in}}\left(T_{kj}-O_{kj}^{i}\right)^2 \tag{2}$$

where

T_{kj} = Target of the kth input in the jth chromosome

O_{kj}^{i} = Output of the kth input in the jth chromosome of the population Q_i based on the selected ANN architecture

7. Generate the mating pool by selecting the best chromosomes using roulette selection methods.
8. Select parent pairs of population Q_i, say $\left(\phi_{1s}^{i},\phi_{2s}^{i}\right)$ from the mating pool for crossover mechanism where $s=1,2,\ldots,S$; and $S=\left\lceil \frac{N_{chro}}{4}\right\rceil$.
9. Calculate the crossover probability of the sth

$$p_c(i,\phi_{1s}^{i},\phi_{2s}^{i})=\begin{cases} p_{c0}\dfrac{\left(F_{\max}(i)-F'(i,s)\right)}{\left(F_{\max}(i)-\overline{F}(i)\right)}, & if\ F'(i,s)>\overline{F}(i) \\ p_{c0}, & otherwise \end{cases} \tag{3}$$

parents pairs in the population Q_i [26].
where,

$$F'(i,s)=\begin{cases} F(\phi_{1s}^{i}) & if\ F(\phi_{1s}^{i})>F(\phi_{2s}^{i}) \\ F(\phi_{2s}^{i}) & otherwise \end{cases}$$

$F(\phi_{1s}^{i}), F(\phi_{2s}^{i})$: Fitness value of parent 1 and parent 2 respectively

$F_{\max}(i)$: Maximum fitness value of the population Q_i

$\overline{F}(i)$: Average fitness value of the population Q_i

10. Calculate mutation probability of the kth chromosome of offspring in the population Q_i [26], $k=1,2,\ldots,\left\lceil \frac{N_{chro}}{2}\right\rceil$

$$p_m(i,j)=\begin{cases} p_{m0}\dfrac{\left(F_{\max}(i)-F(i,j)\right)}{\left(F_{\max}(i)-\overline{F}(i)\right)} & if\ F(i,j)>\overline{F}(i) \\ p_{m0} & otherwise \end{cases} \tag{4}$$

where $F(i,j)$ is the fitness value of the jth chromosome in the population Q_i

11. Set $i=i+1$

Generate Q_i by applying crossover and mutation mechanism based on the following rules:

a. for $s=1:S$

if $p_c(i,\phi_{1s}^i,\phi_{2s}^i) \geq R_{pc}(s)$ do crossover between ϕ_{1s}^i and ϕ_{2s}^i. Otherwise, copy ϕ_{1s}^i and ϕ_{2s}^i as offsprings.

b. for $j=1: N_{chro}$

if $p_m(i,j) \geq R_{pm}(j)$ do mutation of the jth chromosome. Otherwise, the jth chromosome is kept unchanged.

12. If Q_i converge or i is equal to N_{iter} then the best chromosome is obtained and assigned as the optimal weights from mANNs learning. Else, go to step 6

3 Result Analysis and Discussion

In this paper, our experiment used 240 samples data with 30 input parameters and 16 output classes. These 240 samples data are divided into 80%, 10% and 10% for training, validation and testing, respectively. We try three architecture of ANN to find the best architecture as follows (1) 30 neurons of input, 30 neurons of the first hidden layer and 16 neurons of output layers, (2) 30 neurons of input, 30 neurons of the first hidden layer, 30 neurons of the second hidden layer and 16 neurons of output layer and (3) 30 neurons of input, 30 neurons of the first hidden layer, 30 neurons of the second hidden layer, 30 neurons of the third hidden layers and 16 neurons of output layer. We refer m-l_1-l_2-l_3-n as ANNs with m neurons input, l_1 is the number of neurons in the first hidden layers, l_2 is the number of neurons in the second hidden layers, l_3 is the number of neurons in the third hidden layers and n neurons output.

The performance of this algorithm is assessed by the classification accuracy which is calculated using the following equation [33]

$$\text{class. Accuracy} = \frac{\text{total true output class}}{\text{Total output}} \times 100\% \qquad (5)$$

The performance mANNs-APGAs are compared to simple average and majority voting method as shown in Table 4. The results of simple average, majority voting and mANNs-APGAs are obtained from twelve running with five different epochs. In Table 4, we can see that mANNs-APGAs have better performance in training, validation and testing process. For mANNs-APGAs with topology 30-30-30-30-16, it can achieve 99.5%, 100% and 100% for training, validation and testing, respectively. The classification accuracies of mANNs-APGAs are significantly better than the results of simple average and majority voting as shown in Figure 4. The mANNs-APGAs with topology 30-30-30-30-16 increase the accuracy about 16.1%, 16.1% and 24.0% for training, validation and testing of simple average respectively, and 12.4%, 13.5% and 11.9% for majority voting.

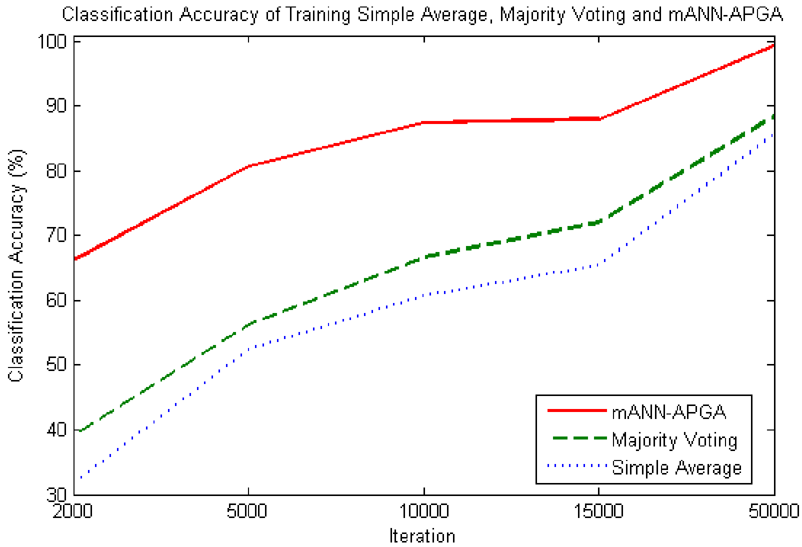
The mANNs-APGAs with topology 30-30-30-30-16 are also capable to reduce the range between maximum and minimum value of classification accuracy in each learning process as shown in Figure 5. In Figure 5, the distinction between maximum and minimum accuracy are 11%, 8% and 1% for training, validation and testing respectively. It decreases around 21%, 68% and 94% compared with the ANN 30-30-30-30-16 in the training, validation and testing process. It means that mANNs-APGAs with topology 30-30-30-30-16 are more capable for maintaining the consistency of the accuracy in learning process.

Table 4 The classification accuracy of simple average, majority voting and mANNs-APGAs

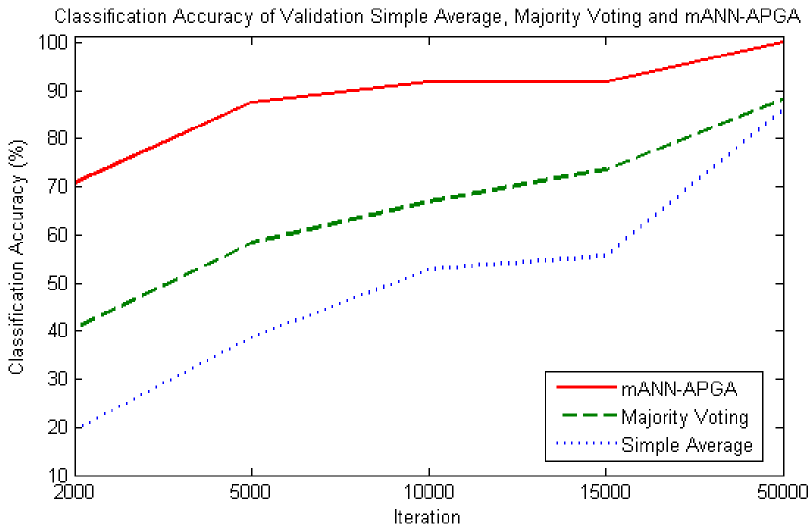
Method	Iteration	Training (%)	Validation (%)	Testing (%)
Simple Average ANN 30-30-16	2000	56.6	44.4	38.9
	5000	63.1	52.8	50
	10000	60.8	33.3	41.7
	15000	64.9	41.7	44.4
	50000	79.2	63.9	44.4

Table 4 The classification accuracy of simple average, majority voting and mANNs-APGAs (continued)

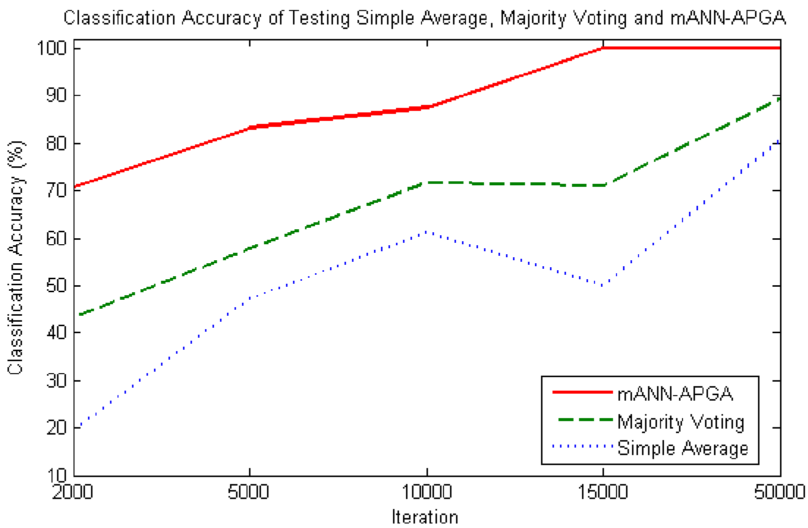
Method	Iteration	Training (%)	Validation (%)	Testing (%)
Majority Voting ANN 30-30-16	2000	47.8	42.2	35.8
	5000	58.7	47.2	48.6
	10000	69.1	59.2	58.3
	15000	73.6	63.1	67.8
	50000	84.9	70.3	74.2
Multiple ANN-APGA 30-30-16	2000	76.6	77.8	50.0
	5000	77.1	77.8	75.0
	10000	80.7	83.333	91.7
	15000	81.8	75.0	91.7
	50000	90.1	86.1	83.3
Simple Average ANN 30-30-30-16	2000	39.2	30.6	19.4
	5000	47.6	44.4	30.6
	10000	54.2	50.0	44.4
	15000	65.5	61.1	58.3
	50000	81.6	58.3	72.2
Majority Voting ANN 30-30-30-16	2000	43.7	43.1	42.2
	5000	61.6	58.9	59.4
	10000	69.4	69.4	67.5
	15000	72.4	74.2	70.6
	50000	88.0	88.9	89.7
Multiple ANN-APGA 30-30-30-16	2000	66.7	66.7	62.5
	5000	78.7	79.2	79.2
	10000	85.4	83.3	91.7
	15000	85.4	87.5	87.5
	50000	95.8	95.8	95.8
Simple Average ANN 30-30-30-30-16	2000	31.6	19.4	19.4
	5000	52.4	38.8	47.2
	10000	60.7	52.8	61.1
	15000	65.5	55.6	50
	50000	85.7	86.1	80.6
Majority Voting ANN 30-30-30-30-16	2000	39.1	40.6	43.3
	5000	56.3	58.3	57.8
	10000	66.7	66.9	71.7
	15000	72.1	73.6	71.1
	50000	88.5	88.1	89.4
Multiple ANN-APGA 30-30-30-30-16	2000	66.2	70.8	70.8
	5000	80.7	87.5	83.3
	10000	87.5	91.7	87.5
	15000	88.02	91.7	100.0
	50000	99.5	100.0	100.0



(a)



(b)



(c)

Fig. 4 The classification accuracy of training (a), validation (b) and testing (c) task from simple average, majority voting and mANNs-APGAs for topology 30-30-30-30-16

4 Conclusion

This paper presented hybrid technique of mANNs-APGAs to improve the classification accuracy of ANNs in condition diagnosis of bearing systems. The mANNs were used to provide several best initial weights which were optimized by APGAs. These optimized weights from APGAs, afterward, were used as the best weights for condition diagnosis. This proposed method requires more learning time process compare to simple average and majority voting, however, the mANNs-APGAs gave the better set of weights. Consequently, the mANNs-APGAs give the better classification accuracy. The issue of learning process is not sufficient important in real applications since we only use the best weights in ANNs to diagnose the condition of new vibration signal data directly without extensive learning process again. Our experiment has shown that the mANNs-APGAs with topology 30-30-30-30-16 increase the accuracy about 16.1%, 16.1% and 24.0% for training, validation and testing of simple average respectively, and 12.4%, 13.5% and 11.9% for majority voting.

The mANNs-APGAs with topology 30-30-30-30-16 are also proven able to reduce the distinction between maximum and minimum accuracy of learning process. It means that the mANNs-APGAs performances are more stable compared with ANN 30-30-30-30-16 due to the best weights of mANNs-APGAs is better than ANN 30-30-30-30-16. It shows that ANNs-APGAs with topology 30-30-30-30-16 is more capable to maintain the consistency of the accuracy in learning process. In future works, we want to try to reduce the number of parameters involved in order to reduce running time in learning process and diagnosis of new vibration signal data.

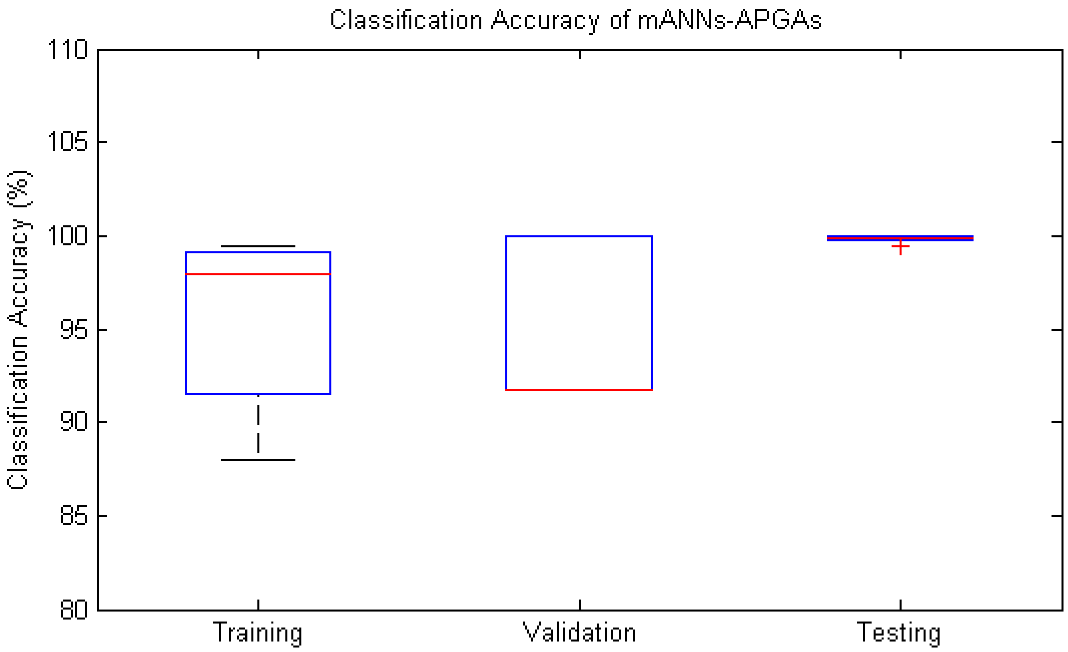


Fig. 5 Classification accuracy of mANNs-APGAs learning process.

Acknowledgment

The authors thank to Universiti Teknologi Malaysia (UTM) and Ministry of High Education (MOHE) for Exploratory Research Grant Scheme (ERGS) Vote No. R.J130000.7828.4L146. The first author sincerely thank to UTM for awarding International Doctoral Fellowship (IDF) and Bina Nusantara University. In addition, the second author would like to express a sincere gratitude to School of Quantitative Sciences - Universiti Utara Malaysia (UUM) for support and encouragement.

References

[1] *RAM*, in *Reliability, Availability and Maintainability Dictionary*. 1988, ASQC Quality press. Milwaukee.

[2] Bocaniala, C.D. and V. Palade, *Computational Intelligence Methodologies in Fault Diagnosis: Review and State of the Art.* Computational Intelligence in Fault Diagnosis, Advanced Information and Knowledge Processing, 2006: p. 1-36.

[3] S. Simani, C. Fantuzzi, and R.J. Patton, *Model-based Fault Diagnosis in Dynamic System Using Identification Techniques*. Advances in Industrial Control. 2003: Springer-Verlag London Limited 2003.

[4] Kagle, B.J., et al. *Multi-Fault Diagnosis of Electronic Circuit Boards Using Neural Networks*. in *1990 International Joint Conference on Neural Networks*. 1990.

[5] Al-Jumah, A.A. and T. Arslan. *Artificial Neural Network Based Multiple Fault Diagnosis in Digital Circuits* in *IEEE International Symposium on Circuits and Systems (ISCAS) 1998*. 1998.

[6] Wang, C., Y. Xie, and G. Chen. *Fault Diagnosis Based on Radial Basis Function Neural Network in Analog Circuit*. in *International Conference on Communications, Circuit, and Systems (ICCCAS) 2004*. 2004.

[7] Lei, W., S. Rong-Ping, and C. Jian-Hua. *Application of RBF Neural Network in Fault Diagnosis of FOG SINS*. in *International Conference on Control, Automation and System*. 2008. Seoul, Korea.

[8] Hoskins, J.C., K.M. Kaliyur, and D.M. Himmelblau, *Fault Diagnosis in Complex Chemical Plants Using Artificial Neural Networks.* AIChe Journal, 1991. **37**(1): p. 137 - 141.

[9] Khanmohammadi, S., I. Hassanzadeh, and H.R.Z. Poor, *Fault Diagnosis Competitive Neural Networks with Prioritized Modification Rule of Connection Weights.* Artificial Intelligence in Engineering 2000. **14**: p. 127-132.

[10] Mitoma, T., H. Wang, and P. Chen, *Fault diagnosis and condition surveillance for plant rotating machinery using partially-linearized neural network.* Computer and Industrial Engineering, 2008. **55**: p. 783-794.

[11] Ogaji, S. and R. Singh, *Artificial neural networks in fault diagnosis: A gas turbine scenario.* Computational Intelligence in Fault Diagnosis, Advanced Information and Knowledge Processing 2006: p. 179-207.

[12] Bakhary, N., H. Hao, and A.J. Deeks, *Damage detection using artificial neural network with consideration of uncertainties.* Engineering Structure 29, 2007: p. 2806-2815.

[13] Payganeh, G., et al., *Machine Fault Diagnosis Using MLPs and RBF Neural Networks.* Applied Mechanics and Materials, 2012. **110-116**: p. 5021-5028.

[14] Puscasu, G., et al., *Sisteme de Conducere Clasice si Inteligente a Proceselor.* . MATRIX ROM, Bucharest, Romania, 2000.

[15] Hashem, S., *Optimal Linear Combinations of Neural Networks.* Neural Networks, 1997. **10**(4): p. 599-614.

[16] Hashem, S. and B. Schmeiser, *Improving model accuracy using optimal linear combinations of trained neural networks.* Neural Networks, IEEE Transactions on, 1995. **6**(3): p. 792-794.

[17] Ueda, N., *Optimal linear combination of neural networks for improving classification performance.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2000. **22**(2): p. 207-215.

[18] Daren, Y., H. Qinghua, and B. Wen. *Combining multiple neural networks for classification based on rough set reduction*. in *Neural Networks and Signal Processing, 2003. Proceedings of the 2003 International Conference on*. 2003.

[19] Alexandre, L.A., A.C. Campilho, and M. Kamel, *On combining classifiers using sum and product rules.* Pattern Recognition Letters, 2001. **22**(12): p. 1283-1289.

[20] Toman, H., et al., *Generalized Weighted Majority Voting with an Application to Algorithms Having Spatial Output*, in *Hybrid Artificial Intelligent Systems,* E. Corchado, et al., Editors. 2012, Springer Berlin Heidelberg. p. 56-67.

[21] Cordella, L., et al., *A Weighted Majority Vote Strategy Using Bayesian Networks*, in *Image Analysis and Processing – ICIAP 2013,* A. Petrosino, Editor. 2013, Springer Berlin Heidelberg. p. 219-228.

[22] Dong, H. and S. Chen, *Target Recognition Methods Based on Multi-neural Network Classifiers Fusion*, in *Advances in Neural Networks – ISNN 2013,* C. Guo, Z.-G. Hou, and Z. Zeng, Editors. 2013, Springer Berlin Heidelberg. p. 638-647.

[23] Alpaydin, E. *Multiple neural networks and weighted voting*. in *Pattern Recognition, 1992. Vol.II. Conference B: Pattern Recognition Methodology and Systems, Proceedings., 11th IAPR International Conference on*. 1992.
[24] Mak, K.L., Y.S. Wong, and X.X. Wang, *An adaptive genetic algorithm for manufacturing cell formation.* Int. J. Adv. Manuf. Technol 2000. **16** p. 491-497.
[25] Aihong, J. and Y. Lizhe, *Fault diagnosis based on adaptive genetic algorithm and BP neural network.* 2010 2nd International Conference on Computer Engineering and Technology, 2010. **6**: p. 427-430.
[26] Srinivas, M. and L.M. Patnaik, *Adaptive probabilities of crossover and mutation in genetic algorithms.* IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS, 1994. **24**(4): p. 656-667.
[27] Blum, S., et al., *Adaptive mutation strategies for evolutionary algorithms.* The annual conference: EVEN at Weimarer Optimierungsund Stochastiktage 2.0,, 2001.
[28] Snasel, V., P. Kromer, and J. Platos, *Benchmarking hybrid selection and adaptive genetic operators.* Václav Snášel (Ed.): Znalosti 2008, ISBN 978-80-227-2827-0. FIIT STU Bratislava, Ústav informatiky a softvérového inzinierstva, 2008: p. 224-233.
[29] Harnoy, A., *Bearing Design in Machinery : Engineering Tribology and Lubrication*. 2003: Marcel Dekker, Inc.
[30] Rodriguez, P.V.J. and A. Arkkio, *Detection of Stator Winding Fault in Induction Motor Using Fuzzy Logic.* Applied Soft Computing 2008. **8**: p. 1112-1120.
[31] Loparo, K.A. http://csegroups.case.edu/bearingdatacenter/pages/welcome-case-western-reserve-university-bearing-data-center-website.
[32] Li, W., et al., *Feature Extraction and Classification of Gear Faults Using Principal Component Analysis.* Journal of Quality in Maintenance Engineering 2003. **9**(2): p. 132-143.
[33] Rossiter, D.G., *Technical Note: Statistical methods for accuracy assesment of classified thematic maps*. 2004.