BEARING FAULTS DETECTION AND IDENTIFICATION USING RELATIONAL DATA CLUSTERING WITH COMPOSITE DIFFERENTIAL EVOLUTION OPTIMIZATION

Issam Abu-Mahfouz and Amit Banerjee
Penn State University Harrisburg
777 West Harrisburg Pike
Middletown, PA 17111
USA

Abstract: Bearing faults in machinery are among the most critical faults that require attention by maintenance personnel at early stages of fault initiation. In many cases it is difficult to directly and accurately identify the fault type and its extent under varying operating conditions. This work demonstrates a novel procedure for bearing fault detection and identification in an experimental set-up. Three seeded faults, in the rotating machinery supported by the test ball bearing, include inner race fault, outer race fault and one roller fault. The rotor is run at different speeds and with small level of rotating mass unbalance. Accelerometer based vibration signals are analyzed for the different bearing faults' signatures using statistical features, frequency spectra and wavelet coefficients. The composite differential evolution technique is proposed for parameter estimation when the system response is known *a-priori*. The algorithm is compared to five other differential evolution algorithms using conventional crossover and mutation operators. The objective is to correlate bearing faults to the extracted vibration features. The results of this analysis will be extended for applications in real time bearing condition monitoring system.

Key words: Bearing fault; Condition monitoring; diagnostics; differential evolution; evolutionary algorithms; parameter identification; vibration; wavelets.

1. INTRODUCTION

Vibration monitoring in combination with signal processing techniques are effective indicators of wear and damage for bearings used in rotating machinery [1]. The capability of successfully and accurately detecting the faults and their types at early stages is of great importance for the safety and economical aspects in any industrial setup. Therefore, predictive condition monitoring procedures are required to extract the features related to the faulty element bearing. Several well-established methods have been used for bearing fault diagnosis such as spectral analysis, envelop detection, autoregressive modeling, cyclostationary analysis, hidden Markov models (HMM), Helbert-Huang transform, and wavelet decomposition. Recent research in the field for machine condition and predictive health monitoring have concentrated on the development of advanced signal processing and machine learning techniques using artificial Intelligence (AI) and fuzzy logic algorithms. See for example [2-5] as a sample. El-Thalji and Jantunen [6] provided a critical review of the Predictive Health Monitoring methods of the entire defect evolution process i.e. over the whole life time and suggests enhancements for rolling element bearing monitoring.

The purpose of this study is to explore the potentials and capabilities of an artificial intelligence algorithm, namely the differential evolution (DE) in developing machine condition monitoring (MCM) methods for ball bearings supporting a rotor-disk system under periodic unbalance excitation. Evolutionary algorithms (EA) are metaheuristic optimization algorithms that uses mechanisms inspired by processes of biological evolution such as reproduction, mutation, recombination and survival of the fittest. They are used to successfully solve problems that cannot be efficiently solved by classical mathematical polynomial methods. Differential Evolution (DE) is a stochastic population-based evolutionary algorithm (EA) in which current information at each iteration is used to guide the search process. In this work, the vibration signals obtained from an experimental setup with faulty bearings, as described in the following section, are processed to extract features that form the search landscape for the DE technique.

2. EXPERIMENTAL SET-UP AND SIGNAL PROCESSING

A rotor dynamic test rig is used for collecting vibration signals from bearings with seeded artificial faults. The schematic of the set-up is shown in Fig. 1. The bearing used for this test is a SKF single row radial ball bearing (1.25 inch I.D., 2.5 inch O.D. and 0.625 inch width dynamic load capacity of 2160 lbf.) For the normal (no fault) bearing case, balls and races are inspected for defects prior to the test runs. In addition, three more bearings were artificially introduced with defect points using a small carbide grinding bit such that each single bearing will have only one fault. The defects are; inner race, outer race and single rolling ball fault. The vibration signals are measured by a PCB accelerometer (model PCB 302A, with sensitivity 10mV/g, and frequency range of 0.7-10000 Hz, and amplitude range of + - 500 g pk). The accelerometer is mounted on the outboard bearing block (Fig. 1). The vibration signals are sampled at 10 KHz and then recorded via a National Instruments USB-6211 data acquisition device. Bearings conditions are evaluated at 10 operating speeds from 500 rpm to 1400 rpm with increments of 100rpm. A small rotating unbalance is added to the rotor disk to create periodic but speed dependent load on the bearings. The raw time vibration signals are analyzed using the Fast Fourier Transform (FFT) and the continuous wavelet transform (CWT) techniques. Figures 2 and 3 show sample time signatures and their corresponding FFT spectra plots for two fault conditions at different running speeds. The time signals are presented by plotting the output of the vibration sensor versus the time sample points. Therefore, at a sampling rate of 10,000 s/sec the segments in Figure 2 represent a duration of 0.5 second. Statistical parameters such as the standard deviation and skewness are also computed. Sample statistical features are shown in Figures 4 and 5 for all four bearing cases.

Wavelet transform is being increasingly used as a signal processing technique [7]. The continuous wavelet transform (CWT) of a signal $x(t)$ is defined as the convolution integral of $x(t)$ with a translated and dilated versions of a mother wavelet function. For this work and after experimenting with several wavelets, it has been found that the Mexican hat wavelet gave acceptable results. Figures 6 and 7show sample wavelet scalograms, over 500 data points, for the two bearing fault cases presented in Figures 2 and 3 respectively. The scalogram is a two-dimensional plot of scales (inverse of frequencies) versus time (or in this case sample points).

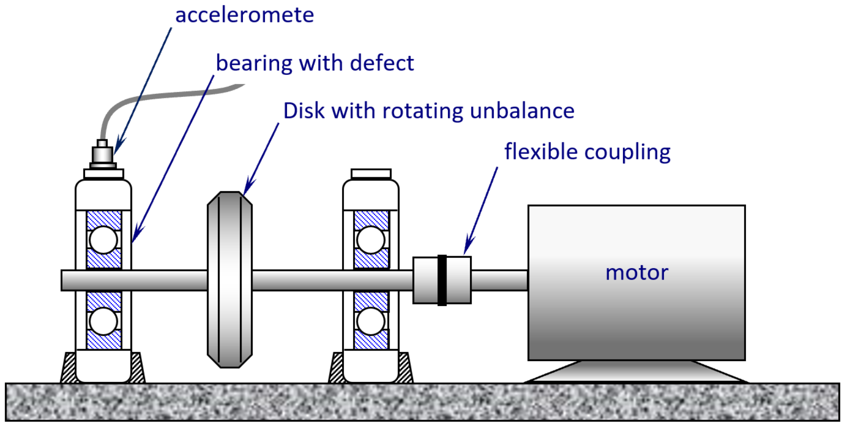

Figure 1. Experiment setup for ball bearing fault.

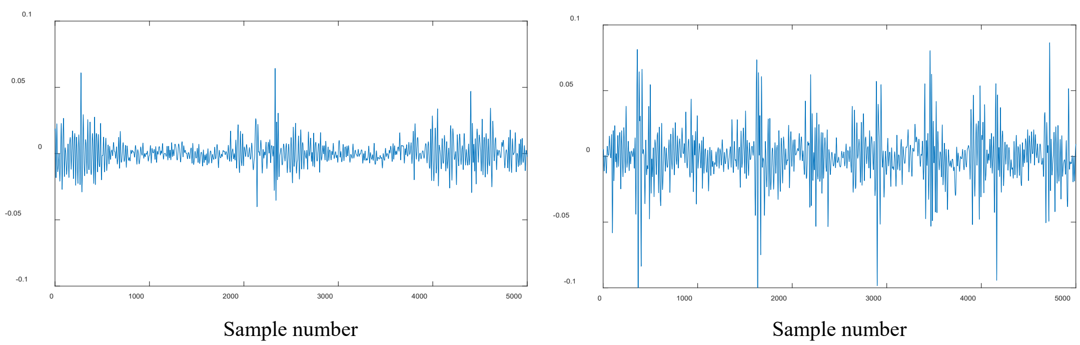

Figure 2. Vibration signals for (a) 500 rpm _ ball fault, and (b) 800 rpm _ inner race fault.

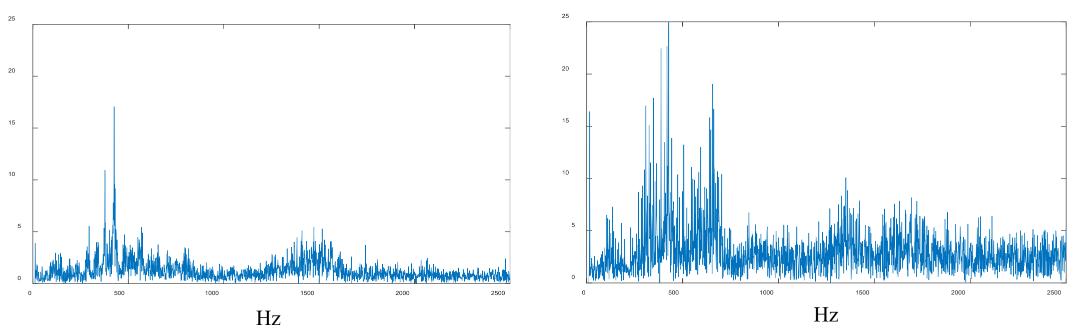

Figure 3. FFT Spectra for (a) 500 rpm _ ball fault, and (b) 800 rpm _ inner race fault.

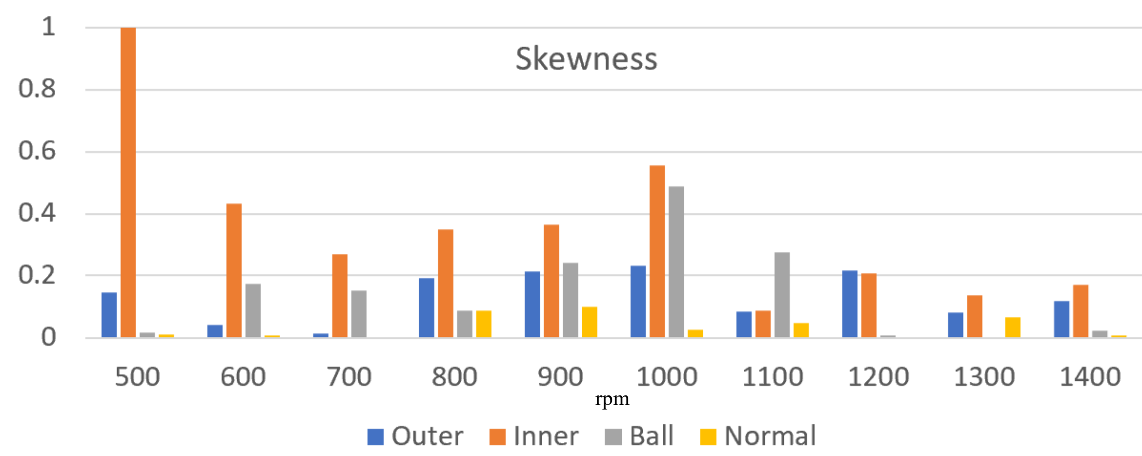

Figure 4. Normalized skewness at different speeds in rpm.

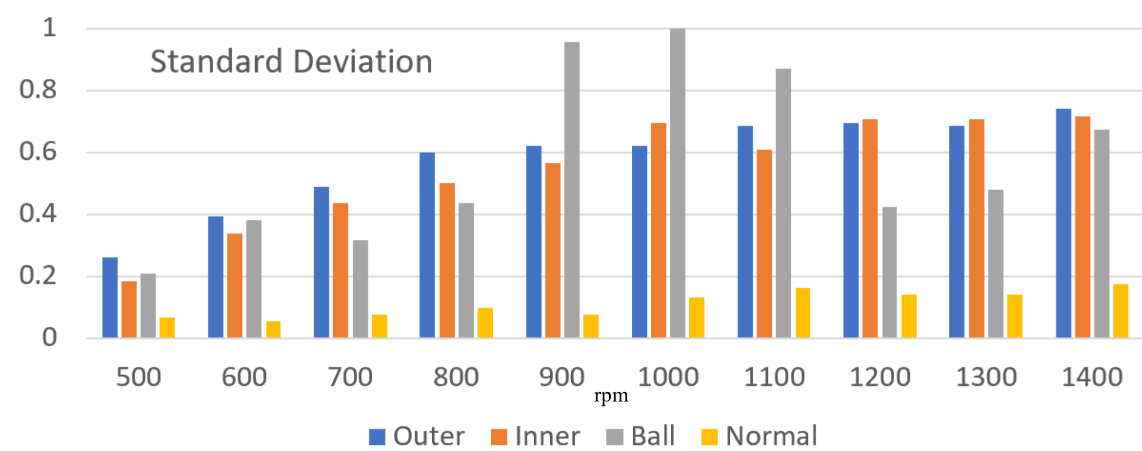

Figure 5. Normalized standard deviation at different speeds in rpm.

Small scaling parameter values result in narrow windows and serve for precisely localized registration of high frequency phenomena. On the other hand, large scaling parameter values result in wide windows and serve for the registration of slow phenomena. This way, information in the time and frequency domains are represented simultaneously. The Mexican hat wavelet, as selected for this investigation, is applied to segments of 5000 data points each. The 64 wavelet scales, for each bearing fault type and at each rotating speed, are averaged over these 5000 points to represent an additional feature vector.

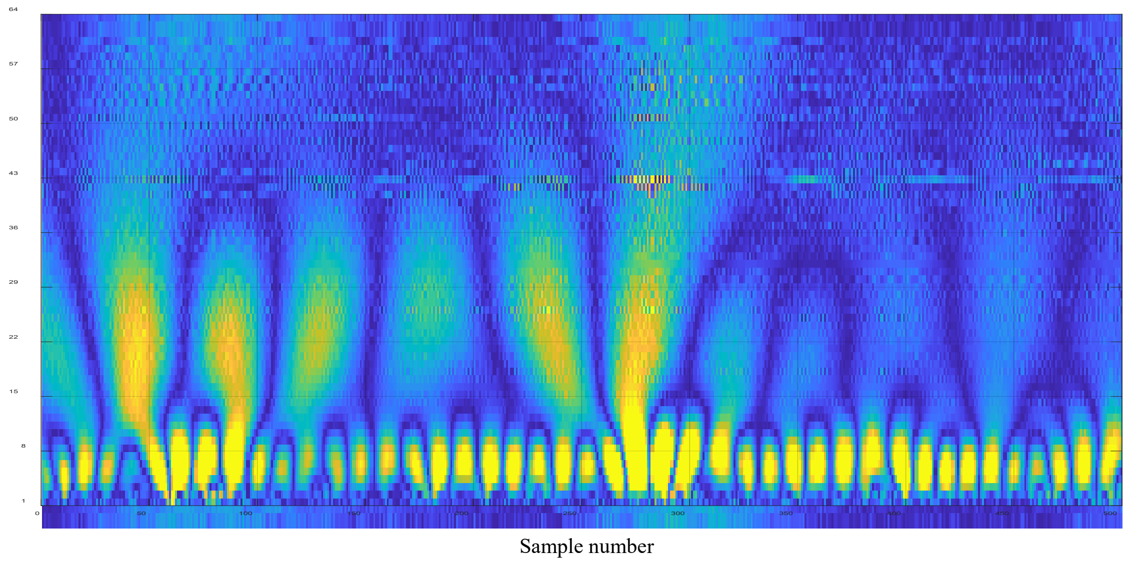

Figure 6. Wavelet scalogram for a bearing with ball fault running at 500 rpm.

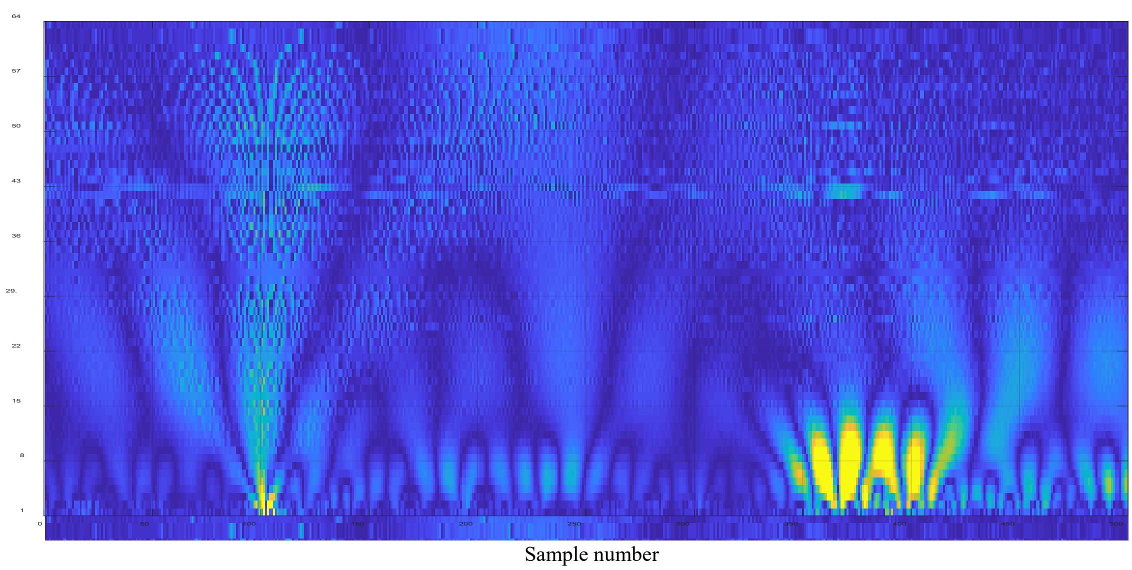

Figure 6. Wavelet scalogram for a bearing with inner race fault running at 800 rpm.

In this paper, we formulate the inverse problem of parameter estimation where the system response is known. The parameters are then estimated by solving a minimization problem in the response space formulated as,

$$f = \min\left[\sum\left\{x(\Theta) - x(\hat{\Theta})\right\}^2\right] \tag{1}$$

where $x(\Theta)$ is the known set of response for an unknown parameter set Θ and $x(\hat{\Theta})$ is the response for the estimated parameter set $\hat{\Theta}$. The known response is in the form of a finite set of either the standard deviation and skewness of the vibration signal, or the normalized averaged FFT bands or the averaged wavelet coefficients. The minimization functional f used by the differential evolution algorithm is the average sum of differences between the measured response and estimated response over the entire response spectrum. The algorithm will guide the search for parameters based on minimizing this functional. The composite differential evolution algorithm is used for optimizing the estimated parameters in this paper and algorithmic performance is compared to basic versions of the differential evolution algorithm.

3. COMPOSITE DIFFERENTIAL EVOLUTION (DE)

A potential candidate solution for the parameter estimation problem is encoded in 6-bits – the first four encoding for speed (rpm) and the other two bits encoding for the four defect types. A population of size N is used and evolved using the differential evolution (DE) algorithm till convergence. In this paper, we define convergence when there is no change (within a threshold) in the population over two consecutive generations. The DE algorithm is an evolutionary algorithm similar to genetic algorithms but instead of using crossover and mutation to evolve new population of candidates, uses a simple differential operator to create new candidate solutions and employs a one-to-one competition scheme to greedily select new candidates [9]. Similar to genetic algorithms, DE is a population-based evolution technique where the starting population is randomly populated by potential candidate solutions. The next generation is evolved by first creating a trial vector for each target vector (from the current population) using a trial vector generation strategy, followed by a crossover operation which creates an offspring vector for every target vector that has aspects of both the target and the trial vector. The offspring vector replaces the target vector in the new generation if it is better (fitter) than the target vector. The performance of the differential evolution algorithm depends on the strategy used to generate trial vectors, and how well the control parameters are tuned to solve the specific problem. To overcome these issues, a variant of the differential evolution algorithm called the composite differential evolution (CoDE) algorithm was proposed [10]. It combines three most popular trial vector generation strategies with three control parameter settings which are used in a random way to generate trial vectors. It has a simple structure and is easy to implement. The three trial vector generation strategies used in this paper are:

1) rand/1
   $$\upsilon_i = x_{r1} + F.[x_{r2} - x_{r3}]$$
2) best/1
   $$\upsilon_i = x_{best} + F.[x_{r1} - x_{r2}] \tag{2}$$
3) current-to-best/2
   $$\upsilon_i = x_i + F.[x_{best} - x_i] + F.[x_{r1} - x_{r2}],$$

where υ_i is the trial vector and x_i is the parent vector. The best candidate solution in a generation is labeled x_{best} and x_{r1} and x_{r2} are two randomly selected vectors from the current generation. Empirical studies have shown that rand/1 maintains a good diversity, while current-to-best/2/bin shows good convergence characteristics. The DE crossover operator implements a discrete (bit-wise) recombination of the trial vector, υ_i, and the parent vector x_i to produce offspring x'_i as follows:

$$x'_{ij}(g) = \begin{cases} \upsilon_{ij} & \text{if } \eta < c_r \text{ and } j \in \Im \\ x_{ij} & \text{otherwise} \end{cases} \quad (3)$$

where x_{ij} refers to the j^{th} bit of the vector x_i, c_r is the crossover probability [0-1] and $\Im$ is the set of bit indices that will undergo crossover The binomial crossover (/bin) is the most widely used method to determine the set $\Im$ where crossover points are randomly selected from a et of possible crossover points, $\{1, 2, .. p\}$ where p is the total number of bits in genotype definition. The larger the crossover probability c_r, the more crossover points will be selected. This also means that more elements of the trial vector will be used to produce the offspring with fewer bits coming from the parent vector. The binomial crossover operator is used with the rand/1 and the best/1 trial vector generation strategies but not with the current-to-best/2.

The scaling factor F controls the amplification of the differential variations and controls what is akin to a mutation operator in genetic algorithms. The smaller the value of F, the smaller the mutation step sizes and the longer it will make the algorithm to converge. On the other hand, larger values help in exploration but may cause the algorithm to miss a good optima. Therefore, the value of F should be small enough to allow differentials to exploit tight spaces in the search space, and large enough to maintain diversity (exploration). In CoDE, three control parameter settings are used because of the drawback in using fixed values of c_r and F throughout the algorithm. In this paper, we use:

1) $F = 1.0$, $c_r = 0.1$
2) $F = 1.0$, $c_r = 0.9$ (4)
3) $F = 0.8$, $c_r = 0.2$

These settings are the same as those used in (Refer to CoDE paper). At each generation, all three trial vector strategies are used to create new trial vectors followed by offspring vectors with a control parameter setting chosen randomly from the three control parameter settings. Therefore, three offspring vectors are generated for each target vector. The best offspring vector then enters the next generation if it is better than its target vector. In this paper we compare the performance of CoDE to that of DE/rand/1/bin and DE/current-to-best/2 with all three control parameter settings used one at a time. The algorithms will be referred to with a number prefix which will correspond to the control parameter setting.

4. Results

All the algorithms are run independently for 5 times for population size of $N = 10$, and number of generations till convergence and the quality of the best solution for the median run are compared for the different algorithms. The CPU run time on MATLAB 2018 running on dual core $i7$ processor is also provided in Tables 1-3 for three different feature sets. The feature sets used are (1) statistical feature set with two features – standard deviation and skewness in Table 1, (2) 16 PSD (FFT) bands in Table 2, and (3) 64 wavelet coefficients in Table 3. As can be seen CoDE outperforms the other algorithms in both convergence characteristics and quality of best solution but takes significantly more run time than the basic versions of the DE algorithm. This is because of multiple trail vectors generated by for every target vector, followed by a comparison phase.

Table 1: Comparative Performance for the DE algorithms for Statistical Feature Set.

	Convergence (# generations)	% Error	CPU Runtime (ms)
CoDE	8	1.924	62
DE/rand/1/bin/1	16	2.996	27
DE/rand/1/bin/2	15	3.412	32
DE/best/1/bin/3	18	2.946	29
DE/current-to-best/2/1	14	3.218	33
DE/current-to-best/2/2	14	3.184	34

Table 2: Comparative Performance for the DE algorithms for FFT Coefficients Feature Set.

	Convergence (# generations)	% Error	CPU Runtime (ms)
CoDE	11	1.247	132
DE/rand/1/bin/1	18	1.895	58
DE/rand/1/bin/2	18	2.887	75
DE/best/1/bin/3	21	1.924	72
DE/current-to-best/2/1	15	2.560	79
DE/current-to-best/2/2	17	2.677	82

Table 3: Comparative Performance for the DE algorithms for Wavelet Coefficients Feature Set.

	Convergence (# generations)	% Error	CPU Runtime (ms)
CoDE	17	4.284	185
DE/rand/1/bin/1	23	5.778	120
DE/rand/1/bin/2	24	6.820	127
DE/best/1/bin/3	20	4.818	121
DE/current-to-best/2/1	19	5.320	125
DE/current-to-best/2/2	21	5.320	125

Among the feature sets tested, the FFT coefficients feature set (16 features) provided the more accurate results than the Wavelet coefficients and the statistical feature set. The statistical feature set (2 features) is the smallest feature set but encapsulates enough information to provide errors < 5% across all algorithms. The statistical feature set also takes the least amount of CPU runtime.

5. CONCLUSION

In this paper, a framework for parameter estimation using differential evolution is created. The framework is based on features extracted from the rotor system vibration response with the purpose of bearing fault identification. It is scalable and can be applied to a larger sample size with very little change to the methodology presented in this paper. The composite differential evolution algorithm, an adaptive variant of the basic differential evolution algorithm is shown to outperform the basic other variants of the algorithm on the parameter identification problem.

References

[1] I. Howard. A Review of Rolling Element Bearing Vibration: Detection, Diagnosis and Prognosis: Melbourne, Australia, DSTO Aeronautical and maritime Research Laboratory, 1994.

[2] T. I. Liu; J. H. Singonahalli; and N. R. Iyer. "Detection of roller bearing defects using expert system and fuzzy logic," Sep., Mech. Syst. Signal Process. Vol. 10, pp. 595–614, 1996.

[3] M. Schwabacher and K. Goebel. "A survey of artificial intelligence for prognostics," NASA Ames Research Center, pp. 107–114, 2006.

[4] S. Tyagi. "Acomparative study of SVM classifiers and artificial neural networks application for rolling element bearing fault diagnosis using wavelet transform preprocessing," World Acad. Sci. Eng. Technol., Vol. 43, pp. 309–317, 2008.

[5] X. Li; A. Zheng; X. Zhang; C. Li and L. Zhang. "Rolling element bearing fault detection using support vector machine with improved ant colony optimization," Oct., Measurement, Vol. 46, pp. 2726–2734, 2013.

[6] I. El-Thalji and E. Jantunen. "A summary of fault modelling and predictive health monitoring of rolling element bearings," Mechanical Systems and Signal Processing, Vol. 60-61, pp. 252–272, 2015.

[7] R. Yana; R. X. Gaob and X. Chenc. "Wavelets for fault diagnosis of rotary machines: A review with applications," Signal Processing, Vol. 96, pp. 1–15, 2014.

[8] R. Storn and K. Price. "Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization, Vol. 11, pp. 341-59, 1997.

[10] Y. Wang, Y.; Z. Cai and Q. Zhang. "Differential evolution with composite trial vector generation strategies and control parameters," IEEE Transactions on Evolutionary Computation, Vol. 15, No. 1, pp. 55-66, 2011.