uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Tony Boutros
AUTEUR DE LA THÈSE / AUTHOR OF THESIS

Ph.D. (Mechanical Engineering)
GRADE / DEGREE

Department of Mechanical Engineering
FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Fault Detection and Diagnosis in Machining Processes and Rotating Machinery Using Fuzzy Approach and Hidden Markov Model

TITRE DE LA THÈSE / TITLE OF THESIS

Ming Liang
DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

Balbir Dhillon

Xi Huang

François Robitaille

Fengfeng Xi

Gary W. Slater
Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

# Fault Detection and Diagnosis in Machining Processes and Rotating Machinery Using Fuzzy Approach and Hidden Markov Model

Tony Boutros

A thesis submitted to the Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements for the degree of

**DOCTOR OF PHILOSOPHY**

In Mechanical Engineering

Ottawa-Carleton Institute for Mechanical and Aerospace Engineering
University of Ottawa
Ottawa, Canada

*Your file Votre référence*
*ISBN: 978-0-494-34121-6*
*Our file Notre référence*
*ISBN: 978-0-494-34121-6*

# Abstract

Over the last three decades, the research for new fault detection and diagnosis techniques in machining processes and rotating machinery has attracted increasing interest worldwide. This development was mainly stimulated by the rapid growth of industrial technologies and the increase in complexity of machining and machinery systems.

In this study, a two-stage monitoring system is proposed. The first stage consists of developing four generic condition indicators to detect transient and continuous anomalies. These indices are then fused to improve the fault detection using two fuzzy methods: a fuzzy-set based fusion and a fuzzy-rule based fusion. In the first approach a condition index is developed based on the min-max technique, whereas in the second method a fuzzy fused index (*FFI*) is defined using the Sugeno-style inference engine and a list of detection rules representing the knowledge gained through learning and experience of domain experts. Additionally, an index of deterioration is developed as part of the first fusion approach.

The second stage forms the fault diagnosis part of the monitoring system. A discrete hidden Markov model is developed to identify the severity of the fault and to isolate its location.

Methods are applied to two types of problems: tool wear/fracture and bearing faults. Results obtained from experimental evaluations show a minimum detection, classification, and isolation rate of 95%. Particularly, the *FFI* is found sensitive to fault severity and not susceptible to speed and load changes. On the diagnosis front, the training required to define the hidden Markov model does not exceed five seconds in both monitoring cases.

## Acknowledgments

Through the following sentences I would like to express my gratitude to every person who contributed directly or indirectly to the success of this research. I'm particularly indebted to my supervisor Professor Ming Liang for his moral and financial support, continuous encouragement, and valuable academic guidance throughout the course of my Ph.D. study. Without his support this work would not have been materialized.

Many thanks go to my Ph.D. thesis committee members, Professor Fengfeng Xi, Professor Xiao Huang, Associate Professor François Robitaille and Professor Balbir Dhillon for their patience in reviewing my thesis and providing me with valuable comments to improve this work.

I wish to thank the University of Ottawa for giving me the chance to reach my academic dream.

I would like to offer my sincere appreciation to GasTOPS, in particular to Dr. Bernie MacIsaac for their support and encouragement.

Finally, I thank my father Hanna for his continuous moral support. I want also to express my great gratitude to my wife Samar and my three sons George, Jonathan and James for their endless patience and for giving me the strength and will to overcome many stopping barriers encountered during this study period. To them I dedicate this work.

# Contents

# List of Figures

## List of Tables

## Nomenclature

| | |
|---|---|
| $A$ | fuzzy subset |
| $\overline{A}$ | state transition probability matrix |
| $\overline{\overline{A}}$ | re-estimated transition probability matrix |
| $a_i$ | filter coefficients for low-pass filter |
| $a_{\bar{i}\bar{j}}$ | transition probability from state $\bar{i}$ to state $\bar{j}$ |
| $\overline{a_{\bar{i}\bar{j}}}$ | re-estimated state transition probability from state $\bar{i}$ to state $\bar{j}$ |
| A/D | analog to digital |
| AI | artificial intelligence |
| ANN | artificial neural network |
| AR | abnormal range (fuzzy set output) |
| $B$ | fuzzy subset |
| $\overline{B}$ | symbol probability matrix |
| $\overline{\overline{B}}$ | re-estimated symbol probability matrix |
| $b_i$ | filter coefficients for low-pass filter |
| $\overline{b_j}$ | re-estimated symbol probability matrix |
| $b_{\bar{j}}(\bar{k})$ | probability of observing symbol $\bar{k}$ while being in state $\bar{j}$ |
| BV | big value (fuzzy set input) |
| $C$ | number of samples |
| $c$ | conditions of machinery or process |

| | |
|---|---|
| $CI$ | condition index |
| $c_k$ | Fourier coefficients |
| $c_t$ | scaling coefficient |
| $corf_P$ | power correlation factor |
| $corf_\sigma$ | standard-deviation correlation factor |
| $CPCF$ | complement of power correlation factor |
| $CSDCF$ | complement of standard deviation correlation factor |
| $D$ | arbitrary number of reference vectors |
| $d$ | city block distance |
| DE | drive end |
| DHMM | discrete hidden Markov model |
| $DM(\lambda,\overline{\lambda})$ | distance measure between models λ and $\overline{\lambda}$ |
| $f_l$ | bandpass filter *l* frequency |
| $F_s$ | sampling frequency |
| FE | fan end |
| $FFI$ | fuzzy fused index |
| $FLIM$ | fault localization index matrix |
| $h$ | height of fuzzy input |
| $h_l$ | impulse response of bandpass filter *l* |
| HMM | hidden Markov model |
| $H_{rl}$ | the minimum height (or membership) associated with rule *rl* |
| $i'$ | index of sub-group |
| $\bar{i}$ | state index |

| | |
|---|---|
| $I'$ | number of sub-groups |
| $IoD$ | index of deterioration |
| $j'$ | index of mini-group |
| $\bar{j}$ | state index |
| $J'$ | number of mini-groups in a sub-group |
| $J_b$ | number of mini-groups |
| $k'$ | index of sample |
| $\bar{k}$ | symbol index |
| $K'$ | number of samples in a mini-group |
| $l$ | filter index |
| $L$ | size of window |
| $LI$ | location index vector |
| $L_{rl}$ | location of the output spike associated with rule $rl$ |
| $\hat{m}$ | index for monitoring indices |
| $M$ | number of distinct observations |
| $\hat{M}$ | number of monitoring indices |
| MA | moving average |
| MISO | multiple input single output model |
| $MIV(l)$ | monitoring index vector after bandpass filter $l$ |
| $MPCI$ | maximum power condition indicator within a sub-group |
| MR | middle range (fuzzy set output) |
| $MSDCI$ | maximum standard deviation condition indicator within a sub-group |

| | |
|---|---|
| MV | medium value (fuzzy set input) |
| $n$ | sample number |
| $\hat{n}$ | index for machine or process conditions |
| $N$ | Number of states |
| $\hat{N}$ | number of conditions |
| $nfl$ | index for number of fault locations |
| $NFL$ | number of fault locations |
| NR | normal range (fuzzy set output) |
| $NSG$ | number of sub-groups |
| $O$ | observations vector |
| $o_t$ | observation at time $t$ |
| $p$ | fuzzy degree of machine or process conditions |
| $P$ | probability |
| $\hat{P}$ | average power for a specific number of mini-groups |
| $\overline{P_{i'}}$ | mean power of sub-group $i'$ |
| $P_{i',\max}$ | maximum sub-group power |
| $P_{i',\min}$ | minimum sub-group power |
| $P_{j'}$ | power of mini-group $j'$ |
| $p_{\hat{n}\hat{u}}$ | fuzzy degree of the $\hat{n}^{\text{th}}$ condition based on the $\hat{u}^{\text{th}}$ sample |
| $P(O/\lambda)$ | probability of observations $O$ given a model $\lambda$ |
| $(Power\text{-}CI)_{j'}$ | power condition indicator for mini-group $j'$ |

| | |
|---|---|
| $q_{\hat{n}\hat{m}\hat{u}}$ | fuzzy relationship for the $\hat{m}^{\text{th}}$ monitoring index and the $\hat{n}^{\text{th}}$ condition as function of the $\hat{u}^{\text{th}}$ sample |
| $\overline{q_t}$ | state at time $t$ |
| $Q$ | fuzzy relationship between conditions and monitoring indices |
| $r$ | fuzzy degree of monitoring indices |
| $rl$ | rule index |
| $\overline{rl}$ | number of activated rules |
| $r_{\hat{m}\hat{u}}$ | fuzzy degree of the $\hat{m}^{\text{th}}$ monitoring index for the $\hat{u}^{\text{th}}$ sample |
| $s$ | Laplace operator |
| $SSR_p$ | sum of the squared errors of power relative to reference mean power $\hat{P}$ |
| $SSR_\sigma$ | sum of the squared errors of standard-deviation relative to reference mean standard deviation $\hat{\sigma}$ |
| $SSV_p$ | sum of the squared errors of power relative to the varying mean power |
| $SSV_\sigma$ | sum of the squared errors of standard deviation relative to the varying mean standard deviation |
| $(Std\text{-}Dev\text{-}CI)_{j'}$ | standard deviation-condition indicator for mini-group $j'$ |
| SV | small value (fuzzy set input) |
| $t$ | time |
| $T$ | sampling period |
| $T_o$ | number of observations |
| $U$ | fuzzy universal set |
| $\hat{u}$ | index for monitoring index samples |

| | |
|---|---|
| $\hat{U}$ | number of samples for monitoring indices |
| $\bar{v}$ | symbol |
| $\bar{V}$ | vector of symbols |
| $V(n)$ | discrete voltage signal |
| $V(t)$ | voltage signal at time $t$ |
| $V_{LP}(n)$ | discrete voltage signal after lowpass filter |
| $V_{LHP}(n)$ | discrete voltage signal after lowpass and highpass filters |
| $w(n)$ | lowpass filter window |
| $w_l$ | center frequency for bandpass filter $l$ (rad/sec) |
| $w_{LP}$ | cut-off frequency of lowpass filter |
| $w_{HP}$ | cut-off frequency of highpass filter |
| WVD | Wigner-Ville distribution |
| $x(n)$ | sample value before filtering |
| $x_l(n)$ | sample values after bandpass filter $l$ |
| $X_n(e^{jw_l})$ | short-time Fourier transform of $x(n)$ |
| $y$ | monitoring indices |
| $y_{\hat{m}\hat{u}}$ | $\hat{u}^{th}$ sample for the $\hat{m}^{th}$ monitoring index |
| $z$ | $z$ domain operator |
| Z | number of filters in the bank |
| ZV | zero value (fuzzy set input) |
| $\alpha_i$ | coefficients of high-pass filter |
| $\alpha_t(\bar{i})$ | forward variable |

| | |
|---|---|
| $\alpha_{T_0}(\bar{i})$ | probability of the sequence of observation of length $T_o$ while being at state $\bar{i}$ and given model $\lambda$ |
| $\beta_i$ | coefficients of high-pass filter |
| $\beta_{t+1}(\bar{j})$ | backward variable |
| $\lambda$ | hidden Markov model parameter |
| $\bar{\lambda}$ | re-estimated hidden Markov model |
| $\delta_t(\bar{i})$ | highest probability along a single path single path, at time $t$, which accounts for the first $t$ observations and ends in state $\bar{i}$. |
| $\mu_A$ | membership function for subset $A$ |
| $\mu_{j'}$ | mean sample value in a mini-group $j'$ |
| $\bar{\pi}$ | initial state probability vector |
| $\bar{\pi}_{\bar{i}}$ | probability of $\bar{i}$ being the initial state |
| $\xi_t(\bar{i},\bar{j})$ | probability of being in state $\bar{i}$ at time $t$, and state $\bar{j}$ at time $t+1$ |
| $\gamma_t(\bar{i})$ | probability of being in state $\bar{i}$ at time $t$, given the model |
| $\sigma_{j'}$ | standard-deviation of mini-group $j'$ |
| $\sigma_{i',\max}$ | maximum standard-deviation of sub-group power $i'$ |
| $\sigma_{i',\min}$ | minimum standard-deviation of sub-group power $i'$ |
| $\overline{\sigma_{i'}}$ | mean standard-deviation of sub-group $i'$ |
| $\hat{\sigma}$ | average standard deviation for a specific number of mini-groups |

# Chapter 1

# Introduction

Traditional maintenance procedures in industry have taken two routes. The first was to perform fixed time interval maintenance, where the maintenance engineers take advantage of slow production cycles to fully inspect all aspects of the machinery. The second route was to simply react to plant failure as and when it happens. The first route was inefficient and costly due to reduced machinery availability and increased maintenance cost and the second could be sometimes catastrophic. However, making use of today's technology, a new scientific approach is becoming a new route to the maintenance management. One of the key elements to this new approach is predictive maintenance through condition monitoring.

Condition monitoring is used for increasing machinery availability and performance, improving manufacturing process productivity and reliability, reducing consequential damage, parts inventory, breakdown maintenance and associated costs. An efficient condition monitoring scheme is capable of providing warnings and predicting faults at early stages. There are many condition monitoring systems installed in different types of engineering plants (Sorensen and Suraland 1988, and Rao 1996). Monitoring systems obtain information from the machine in the form of primary data and through the use of modern signal processing and analyzing techniques, give vital diagnostic information to equipment operator. The acquired information normally covers the condition of the machine (normal or abnormal), fault level (how severe it is) and maintenance recommendations.

As processes and machinery become more and more complex, fault detection and diagnosis emerges as the main task of a monitoring system. This is in fact due to the growing demand for fault tolerance, which can be achieved not only by improving the individual reliabilities of the functional units but also by an efficient fault detection and diagnosis concept (Frank 1990, and Gertler 1988).

According to generally accepted terminology, fault detection and diagnosis consists of the following main tasks (Gertler 1988):

a- *Fault detection*, i.e., the indication that something is going wrong in the system.

b- *Fault isolation*, i.e., determination of the location of the fault.

c- *Fault identification*, i.e., determination of the size or severity of the fault.

In addition to these tasks Pouliezos and Starvrakakis (1994) highlighted the importance of defining a measurement for the degree of deterioration as part of this monitoring system.

In the last three decades fault detection and diagnosis has become a critical issue in the operation of high performance systems (e.g. ships, submarines, air-planes, space vehicles, manufacturing facilities) and structures where safety, mission satisfaction and significant material value are at stake (Frank 1990, Gertler 1988, and Rich and Venkatasubramanian 1987). As such, the research on the development of accurate and robust fault detection and diagnosis models has attracted increasing attention worldwide.

## 1.1 Problem Definition

The study of fault detection and diagnosis is concerned with designing systems that can assist the human operator in detecting and diagnosing equipment faults in order to prevent accidents and maintain the desired system performance and quality requirements. With complex machinery and processes, the need to automate the fault detection and diagnosis becomes imminent, as maintenance decision needs to be taken quickly.

The automation of fault diagnosis was primarily handicapped in the past due to the lack of appropriate techniques to represent the knowledge-based, symbolic reasoning of an expert diagnostician. However, with the advent of the Artificial Intelligence (AI) techniques such as neural network, fuzzy logic, and expert systems (Singh and Al Kazzaz 2003, Pham and Pham 1999, and Filippetti *et al* 2000), this issue can now be addressed effectively.

However, all available fault diagnostic techniques have their drawbacks (Hu *et al* 2000). According to Lee *et al* (2004), all the above techniques are often unreliable and have their own problems to be widely used. Artificial neural network has superior learning and noise suppression abilities and does not require process knowledge (Dornfield and Rangwala 1990, and Grimmelius *et al* 1999). However, it requires some knowledge or a "feel" for the training. In fact, training is time consuming, as a system requires an extensive set of data for all classes of conditions including faults. The system is also not flexible; if dynamic conditions change values outside the trained range, the network must be retrained (Lee *et al* 2004, and Grimmelius *et al* 1999). In addition, there is a difficulty to select the best inputs that allow model optimization in terms of size and processing time (Jack and Nandi 2000).

Fuzzy logic has gained wide acceptance as a useful tool for blending objectivity with flexibility particularly in the area of process control. This method is fast as the decision is based on some simple rules. However its membership function is defined in a relatively subjective manner and the method could suffer from inaccuracy in complex systems (Lee *et al* 2004, and Mechefske 1998).

Expert systems have been used successfully in complicated systems. Nevertheless, complex algorithms have to be updated by an experienced engineer for every target system (Lee *et al* 2004). Furthermore Crupi *et al* (2004) suggested that expert systems could not be applied generically due to the lack of knowledge.

In addition to the drawbacks related to the diagnosis techniques, most of the existing monitoring systems are application-specific, i.e., they are developed either for machinery condition monitoring or for process monitoring, but not for both. However, in many cases, the machinery faults and process malfunctions co-exist. To monitor both types of faults, several such application-specific monitoring systems have to be designed, tested, implemented and maintained separately. This inevitably leads to duplicated efforts and lengthy development processes.

## 1.2 Proposed Solution and Objectives

In view of the limitations of the existing monitoring techniques, the author proposes a two-stage fault detection and diagnosis approach that is applicable to both machinery and machining processes. The main objectives of this model are firstly to optimize the search for a fault and secondly to assess and characterize the fault upon detection based on the requirements described in Section 1.1.

As such, the first stage proposes a generic and fast technique for fault detection using an index fusion and fuzzy logic approach. The second stage describes the development of a fault diagnosis method based on the hidden Markov model (HMM) theory.

## 1.3 Thesis Outline

The main body of this work is composed of five chapters: Chapters 2 to 6. Chapters 3, 4 and 5 are closely organized as a whole system. However they can also be treated as three different studies in the related corresponding areas.

Chapter 2 reviews the fault detection and diagnosis methods described in the literature. The motivations to conduct the present study are also described. Chapter 3 defines a generic technique for the development of condition indicators that can be used for detection of transient or continuous anomalies present in both rotating equipment and machining processes. These indices are evaluated using tool wear/fracture and bearing vibration data. Chapter 4 describes the development of two index fusion methods using fuzzy logic. These methods combine the effects of the condition indicators defined in Chapter 3 in one model to improve the decision-making. In addition, an index of deterioration is proposed to monitor the degradation of the equipment. The approach is validated using the same experimental data utilized in Chapter 3. Chapter 5 contains the details of a discrete hidden Markov model that can employ some of the monitoring indices defined in Chapter 3 for both fault detection and diagnosis (i.e. fault isolation and severity estimation). Similarly, the model is validated using the tool and bearing experimental data. Finally, Chapter 6 summarizes the conclusions and contributions made through this study, and proposes few ideas for future research.

# Chapter 2

# Literature Review

## 2.1 Overview

Nowadays, the study of fault detection and diagnosis has become the most crucial monitoring task. With complex processes and machinery, the need to replace human knowledge with a trained monitoring system becomes imminent. In the last decades few artificial intelligence techniques have been used extensively to automate the monitoring process and provide sufficient information to the operator in a relatively short time. However, a review of previous work indicates that both fault detection methods and diagnosis techniques described in the literature have some limitations in isolating and identifying faults. The main purposes of this literature review are as follows:

i- To show that the existing fault detection methods can be improved

ii- To reveal that the current fault diagnosis techniques have some crucial drawbacks and inadequacies.

## 2.2 Previous Work

### 2.2.1 Fault Detection

A fault is defined as an unacceptable deviation of a characteristic property of the process or machine itself (Poulezos and Stavrakakis 1994). A few methods have been proposed to detect permanent or intermittent changes in the performance of a system in question. These methods can be divided into three main categories: estimation methods, feature extraction and neural networks (Han and Song 2003, Grimmelius *et al* 1999, and Sorsa *et al* 1991). However, as the main advantage of using artificial neural networks is to diagnose the fault (more than to detect it), this technique will be covered under fault diagnosis in Section 2.2.2.

#### 2.2.1.1 Estimation Methods

Estimation methods require mathematical models that represent the real process satisfactorily. The terminology "estimation" is used to reflect the fact that model state variables or parameters are estimated in these methods (Sorsa *et al* 1991).

As part of the cutting process monitoring techniques, Choudhury and Rath (2000) proposed a method that used the relationship between flank wear and the average tangential cutting force coefficient to estimate tool wear. The model was validated through experimental work by comparing the estimated wear size to the real one. The maximum error produced by the model was 8%, but the paper did not provide results for a wide variety of parameter combinations. Elebestawi *et al* (2000) studied the effect of tool wear on the magnitude of various harmonics in the cutting force spectrum. However, no experimental verification of results was performed. Houshmand (1995) presented a dynamic model to monitor processes that can deteriorate from "acceptable" condition to "unacceptable" condition. The mathematical model, which is composed of linear and quadratic discriminant functions, evaluated a $p$-dimensional vector of spectral components of acoustic emission signals at equally spaced time intervals and compared it against a threshold value. This model was applied to the detection of tool wear. The success rate of this method was approximately 70%. Kwon and Ehmann (1994) formulated a general model for the three-dimensional cutting dynamics. In this model, the time-series method was used to identify the imaginary part of the transfer function of the dynamic cutting forces coefficients for tool wear monitoring purposes but the estimation of error or detection efficiency in the experimental work was not reported. Tarng *et al* (1994) defined an analytical model of chatter vibration in metal cutting. The basic cutting mechanics adopted in the model were derived from a predictive machining theory based on shear zone model of chip formation. Although the authors suggested that the results obtained through experiments were satisfactory, the accuracy of the model was not estimated. Liu and Liang (1991) presented an analytical model relating the acoustic emission energy content to the cutting process parameters in peripheral milling. The model was verified using experimental data. Trends of the acoustic energy were defined as functions of the cutting parameters. Despite the reported success, the conclusions were drawn based on relative rather than absolute results as the authors did not discuss the accuracy of the measurements nor define the expected levels. Tarng (1990) established a mechanistic model for prediction of tool breakage using the cutting force. The tool breakage feature in the cutting force signal obtained from the model and cutting tests was examined. Though the results showed that the model was able to detect tool breakage, material properties were not considered in the investigation and the efficiency of the model was not defined.

On the front of machinery condition assessment, Loparo *et al* (2000) developed a model-based approach to the detection of mechanical faults in rotating machinery. The model employed a statistical approach to extract special features associated with fault modes from noise corrupted vibration data. The model worked fine for the inner and outer race faults but did not provide accurate results for the ball bearing faults. Grimmelius (1995) developed a model for a refrigeration plant. The basis for this model was a description of the refrigerant flow in terms of resistors and storage elements. This model was capable of simulating the dynamic behaviour of the plant at various operating conditions with and without the presence of a fault. On the other hand, the model was complex and required tremendous knowledge to perform the implementation. Su *et al* (1993) proposed a mathematical model to illustrate the frequency characteristics of roller bearing vibrations due to surface irregularities, which come from manufacturing errors. The model showed through experimental tests that frequency features of vibration spectra of a normal roller bearing might be indistinguishable from those of damaged bearings. Zheng and McFadden (1993) proposed a vibration parameter identification technique to characterize the transient response of a gear with a fatigue crack in one of the teeth. The technique showed some link between the vibration signal and the quantitative properties of the fault. However, the extraction of knowledge about the crack itself required knowledge of the transfer function.

#### 2.2.1.2 Feature Extraction

The main objective of this task is to extract the necessary information from the signal in order to assess the health of the process or machine in question. The extraction should remove redundant information from measurement data and create simple decision surfaces. There is no *priori* basis for the choice of the feature calculation. It is difficult to know which of the features are essential and which are irrelevant. Inappropriate choices lead to complex decision rules, whereas good choices result in simple rules (Himmelblau 1978 and Pao 1989). The feature extraction can be performed in time, frequency and joint time-frequency domains reviewed as follows.

#### 2.2.1.2.1 Time Domain

Beggan *et al* (1999) used the acoustic emission signal to monitor the surface quality during cutting operation. Correlation between the parameters extracted from the acoustic signal (acoustic power) and surface roughness was defined for various cutting conditions. The experimental results showed the feasibility of this method but the reliability of the system was not high as results showed some inconsistent values. Everson and Cheraghi (1999) investigated the correlation between the quality of a hole drilled in steel with any acoustic emission signal parameter. The acoustic emission energy, number of peak amplitudes above a certain threshold and the root-mean-square were used in this investigation. Experimental work was conducted to validate the method. Their results showed that correlation exists under certain conditions. The use of peak amplitude counts as a condition indicator was inefficient in certain cases where signal is short. The energy was found to be a better measure in this application. Furthermore, Choi *et al* (1999) employed the cutting force and the acoustic signal to monitor the tool fracture in a cutting operation. A burst in the acoustic average energy was used as a triggering signal to inspect the cutting force. A significant drop in cutting force indicated tool breakage. The fracture detection rate based on 15 experimental tests was 100%.

Similarly, a few methods have been proposed for machinery fault detection. Heng and Nor (1998) presented a study on the application of sound pressure and vibration signals to detect the presence of defects in a rolling element bearing using statistical analysis method. The parameters considered in this study included the root-mean-square, crest factor and kurtosis. Results obtained through experiments revealed that the statistical parameters were affected by the shaft speed. In some speed ranges the performance was poor. Morando (1988) reported a successful application of the shock pulse method in the detection of defects in low-speed spherical roller bearings in a paper production line. Miyachi and Seki (1986) extracted the root-mean-square and crest factor from vibration signal to monitor the defects in ball bearings. However, the results were not successful. Along this line, Dyer and Stewart (1978) proposed the use of kurtosis for bearing defect detection based on vibration signal. For an undamaged bearing with Gaussian distribution, the kurtosis value was found equal to three. A value greater than three was judged to be an indication of impeding failure. However, one disadvantage was noted: the kurtosis value comes down to the level of an undamaged bearing when the damage is well advanced. Moreover, Collacott (1977) used the probability density and kurtosis of vibration signal for bearing defects detection. They found that the probability density of acceleration of a bearing in good condition has a Gaussian distribution, whereas a damaged bearing resulted in a non-Gaussian distribution with dominant tails because of a relative increase in the number of high levels of acceleration.

#### 2.2.1.2.2 Frequency Domain

In condition monitoring of metal cutting processes, Dimla (2002) proposed a tool wear detection method based on the fast Fourier transform and contour plots of vibration signal. These two features were validated using various experimental tests. Roth and Pandit (1999) utilized the modal energies contained in the vibration signal to detect tool wear. Six different tests describing various cutting conditions were conducted to validate the detection method. The detection scheme was able to provide a warning of impending failure several centimetres before the failure occurrence. Weck *et al* (1975) employed the sound pressure for chatter detection in milling. The peak-to-peak values of the signal in the frequency domain were compared to a maximum given threshold before triggering the fault detection flag.

As part of an on-line machinery monitoring system, Hou *et al* (1999) employed the acoustic power spectrum density to monitor the pipeline flows. The mass flow rate, volume flow rate and solid concentration were estimated. The estimated errors for all measurements were insignificant. Martin (1996) demonstrated the limitation of using Fourier transform approach for bearing damage monitoring. The averaging technique used to treat vibration fluctuation tends to hide some features of short duration. Tandon (1994) has used successfully the vibration signal cepstrum to detect the outer race defects effectively but failed to detect the inner race defects. Liu and Mengel (1992) used the peak amplitude in the frequency domain, peak root-mean-square and the power spectrum as indirect indices in the monitoring of ball bearing vibration. The success rate claimed in the study was 100%.

#### 2.2.1.2.3 Time-Frequency Domain

Joint time-frequency methods were employed in the process monitoring systems. For example, Xiaoli (1999) proposed the breakage detection of small diameter drills by monitoring the AC servo-motor current. The continuous wavelet transform was used in this method. Experimental results were relatively reliable.

Wavelet transform has been also employed for machinery fault detection. Recently, Liu (2005) proposed the use of wavelet packet basis function to detect the transient anomalies generated by localized defects in rotary machinery. The proposed method did not need training samples. It was applied successfully to gearbox and bearing problems. Similarly, Singh *et al* (2004) processed signal obtained from monitoring system using wavelet transform to extract detailed information for induction machine fault diagnosis. Although the results obtained from the experimental tests suggested that wavelet transform could be used as an effective tool to extract important information about the machine condition it could not effectively deal with multiple faults simultaneously. Baydar and Ball (2003) examined whether acoustic signal can be used effectively along with vibration signals to detect various local faults in a gearbox using the wavelet transform. Two commonly encountered local faults were simulated: tooth breakage and tooth crack. The results suggested that acoustic signals are very effective for the early detection of faults. However, the investigation did not cover the influence of load variation on the fault detection capability of the acoustic approach. Yesilyurt (1997) applied wavelet analysis and the instantaneous power spectrum to detect gear faults.

Along this line, Stander and Heyns (2002) utilized successfully the pseudo-Wigner-Ville distribution (WVD) to identify the influence of the fluctuating load conditions on gearboxes. Similar work was also reported by Baydar and Ball (2001) who investigated whether acoustic signal in conjunction with the WVD can be used effectively to detect the various local faults in gearboxes. The method was successfully employed to detect broken teeth, gear crack and localized wear in gearboxes. McFadden

(1993) extended the application by using the normal WVD and the weighted version of the WVD to gear failure to improve detection capability of the method proposed below by Forrester. Forrester (1989) applied the WVD to averaged gear vibration signals and showed that different faults such as tooth cracks and pitting can be detected in the WVD plot.

#### 2.2.1.3 Summary

Several monitoring methods were developed to detect faults in processes or machinery. Two categories were described above. The first category covers the estimation methods whereas the second contains the feature extraction techniques.

The estimation methods employ a mathematical model to monitor the desired parameters or state variables. The model could be complex and requires a thorough knowledge.

On the other hand, the feature extraction techniques can be used in time, frequency and joint time-frequency domains. Necessary information is extracted from signal and monitored through time to assess the health of the process or the equipment in question. The extraction removes redundant information from measurement data and creates decision surfaces. These methods are easy to implement, as they do not require exact knowledge about the system in question.

However, all the pre-described methods were application-specific; i.e. developed for a specific process or type of machinery but not both.

### 2.2.2 Fault Diagnosis

The next step following fault detection is to post-process the information in order to identify the severity of the fault and isolate its location. As processes and machinery became more complex, efforts in the last three decades were concentrated on the development of automated and knowledge-based monitoring systems. This was mostly done through the use of artificial intelligence techniques such as expert systems, neural networks and fuzzy logic. In the last few years, the hidden Markov model was also explored as an alternative technique by a few researchers.

### 2.2.2.1 Artificial Intelligence

#### 2.2.2.1.1 Expert Systems

Expert systems are computer programs incorporating knowledge about a narrow domain for solving problems related to that domain. It usually consists of two elements: a knowledge base and an inference mechanism. The knowledge base contains domain knowledge, which may be represented by a combination of "*if-then*" rules and procedures. The inference mechanism has the role of manipulating the stored knowledge to provide solutions to problems.

Peng and Goodwin (2001) developed a computerized smart system to interpret comprehensive data obtained from particle analysis, in order to assess wear modes and wear rates of gearboxes. The system was designed to emulate the human ability to perform expert decision-making in the wear particle analysis field.

Yadava and Chakrabarti (1996) and Chakrabarti *et al* (1995) presented the application of expert system for machine condition monitoring. They described the typical faults of a steam turbo-generator and its diagnostic parameters, which are used to establish machine-monitoring expert system. Along this line, Biber *et al* (1990) demonstrated the specifications of first Soviet expert system for vibration control of large turbo-generator.

Siegel and Roedersheimer (1990) developed an expert system to diagnose rotating machines based on vibration signals. However, these systems presented some disadvantages in terms of generality. The model could only be applied to simple machines.

Tzafestas (1989) listed a few expert systems used in different critical applications, including FIS, NDS, and Diamon. FIS is a Fault Isolation System developed at the United States Naval Research Laboratory. The purpose of this expert system is to assist a technician in diagnosing faults in a piece of electronic equipment. NDS is an expert system that detects and locates multiple faults in a nationwide communication network. Diamon is a knowledge-based diagnosis and maintenance planning system aimed at increasing the efficiency of systems for condition monitoring of rotating machinery.

Rich and Venkatasubramanian (1987) described an expert system called MODEX (Model Oriented Diagnostic EXpert), which used model-based reasoning for diagnosing chemical plants. The flow of information (knowledge) was based on physical links between various equipment units and causal relationships among process variables. The model was used successfully to diagnose different configuration of the processes. However, the system did not allow the user to intercede at any time during the course of the diagnostic session in order to provide the system with additional process information.

Shapiro *et al* (1986) designed an expert system called *VMES* (versatile maintenance expert system) for troubleshooting circuits that use structural and functional descriptions. While troubleshooting, it indicated dynamically the state of the reasoning process.

#### 2.2.2.1.2 Artificial Neural Networks

Artificial neural network (ANN) is a computational model organised in layers each consisting of processing elements called neurons. The neurons are interconnected and operate in parallel. Neural networks can capture domain knowledge from training. The training consists of selecting the input patterns and allocating the corresponding output patterns. The error between the actual and expected outputs is used to adjust the weights of the connections between the neurons.

ANN was used to monitor processes conditions. As for example, Tsai *et al* (1999) employed the spindle speed, feed rate, depth cut and vibration average per revolution as inputs to a neural network model to determine the surface roughness during a cutting process. The model was evaluated through experimental work by comparing the estimated roughness value against the actual one. The model was able to classify the roughness with a minimum accuracy rate of 96%. Similarly, Choudhury *et al* (1999) utilized an optoelectronic sensor in conjunction with a multilayered neural network for predicting the flank wear size on the cutting tool. The values predicted using the model and those calculated using the geometrical relation were compared with the actual values measured using a tool maker's microscope. The identification of the wear size was accurately estimated. Ghasempoor *et al* (1999) employed a neural network model that used the cutting force signals as inputs in order to detect the tool wear. A range of cutting conditions was used for the verification runs. In each test the estimated wear size was compared to the actual value to determine the error. The maximum error encountered during experiments was 15%. In addition, Khajavi and Komanduri (1995) used the thrust, torque and strains values as inputs to a neural network model for detection of drill wear. The authors found that when sensor signals are noisy the system could actually result in the deterioration of the correct estimation of drill wear. Along this line, Tansel (1994) trained a neural network model with seven characteristic features extracted from thrust force and micro-drill velocity in order to detect the pre-failure phase of micro-drilling. The model estimated the pre-failure phase 0.2-1 seconds ahead of the failure. Although the results in general were acceptable, the model wrongly classified the problem in some cases. Yao and Fang (1993) utilized a neural network model to quantify the complicated interrelationship between the change of chip breakability and that of comprehensive wear states. The results showed that the method was valid and effective, however, the authors did not explain how "appropriate" selection of neural network structure was conducted. Furthermore, Tansel *et al* (1991) used two synthetically trained neural networks to recognize the harmonics acceleration signals and their frequency collected during cutting process. Based on these observations the future vibration characteristics of the system were estimated leading as such to the detection of chatter. The use of the two neural networks was time consuming in training and model preparation. Dornfeld and Rangwala (1990) used the acoustic emission and force signals as inputs to an ANN model to detect tool wear. The authors highlighted the learning and noise suppression abilities of the model. The neural network model provided a tool classification with a success rate around 85%.

ANN was in parallel employed to monitor machinery conditions. Crupi *et al* (2004) used an ANN to diagnose rotating machinery of the Refinery of Milazzo (Italy). The authors concluded that ANN is a better dynamic tool than expert systems as it can be adaptable to continuously variable data. One of the general disadvantages highlighted by the authors is that neural network models need to be trained by real examples of different fault typologies. Paya *et al* (1997) proposed a neural network model using the wavelet transforms of the vibration signals as inputs to detect and localize single and multi faults in a drive-line model. The faults were induced into the bearing and gearbox. Different combinations of gear/bearing conditions were investigated in this study. The model was able to detect and isolate the faults, however, the bearing fault detection was in some cases inefficient. In a similar way, Baillie and Mathew (1996) utilized vibration signals as input to a neural network model for fault diagnosis of rolling element bearing. The model was able to detect the faults induced into the bearing and to isolate the corresponding locations (inner and outer race). Chow *et al* (1991) demonstrated the effectiveness of a neural network model to detect incipient faults in induction motors. The model was able to detect stator winding and rotor-bearing incipient faults.

Sorsa *et al* (1991) conducted a fault diagnosis investigation using various neural network architectures. Three models were used in this study: single layer perceptron, multilayer perceptron, and counterpropagation networks. The models were tested using a heat exchanger. The multilayer perceptron gave the best results. The authors found that "teaching of the neural network seems fairly slow". Also, if a new fault is introduced, teaching should start all over again. Hoskins and Himmelblau (1988) proposed an ANN approach to automate knowledge representation in process engineering. They demonstrated the ability of ANN models to learn and adapt themselves to different statistical distributions of inputs.

#### 2.2.2.1.3 Fuzzy Logic

With fuzzy logic, the precise value of a variable is replaced by a linguistic description, the meaning of which is represented with a fuzzy set. The inference is carried out based on this representation.

Recently, Sokolowski (2004) proposed an intelligent monitoring system based on fuzzy rule base to monitor the tool wear during a cutting process. One of the results obtained by this study showed that the increase of rule number can lead to a deterioration in the generalization ability and consequently to the development of a corrupted model. Similarly, Susanto and Chen (2003) developed a fuzzy logic based in-process tool wear. The fuzzy membership function and rule bank were based on observations obtained during cutting experiments using artificial tool-wear inserts in face milling operations. The maximum resultant force, feed rate and depth of cut were used as independent variables. In this study the triangular membership function was adopted. Tool wear was identified based on six classes of wear. The accuracy of estimating the wear size during cutting tests was greater than 90%. Insaki (1999) proposed an intelligent database to monitor and control grinding processes. The database used genetic algorithms and fuzzy reasoning so that the system could learn the relationship between the input and the output from the grinding results and transformed them into explicit rules that described the quality requirements for the workpiece. The inputs were obtained through acoustic and power sensors. The use of genetic algorithms optimized the performance of the model, however, it highlighted the weakness of the fuzzy logic to establish the optimum relationship between the inputs and the outputs of the model. Similarly, Ming *et al* (1999) used training examples to construct fuzzy *if-then* rules and the membership functions of the fuzzy sets involved in these rules. Genetic algorithm was used to find the optimal combination of rules to compose a reasoning mechanism and options related to the membership functions. The model was employed to estimate flank wear length. The mean error calculated based on the cutting tests and in relative to the actual wear length was less than 13%. In addition, Du *et al* (1992) presented a study on tool condition monitoring in turning using the fuzzy set theory. The tool conditions considered in this investigation included tool breakage and several states of wear. Eleven monitoring indices were utilized to describe the signature characteristics. A linear fuzzy equation was proposed to describe the relationship between the tool conditions and the monitoring indices. The results obtained from the experimental work indicated a 90% reliability. The authors proposed a membership function based on the probability concept.

Fuzzy logic was also utilized in the monitoring of machinery conditions. For example, Mechefske (1998) developed and applied a fuzzy logic technique to frequency spectra representing various rolling element bearing faults. The application of basic fuzzy logic led to the generation of fuzzy numbers that represented the similarity between frequency spectra. Different fuzzy membership functions were employed in this study.

#### 2.2.2.1.4 Fuzzy-Neural Network

A combination of fuzzy/neural network was proposed by some researchers to improve the model performance, robustness and accuracy. These methods were employed to monitor processes and machinery conditions.

As for example, Mesina and Langari (2001) used a fuzzy neural network model to predict the condition of the tool in a milling process. The relationship between the sensor readings and tool wear state was first captured via a neural network and was subsequently reflected in linguistic form in terms of fuzzy logic based diagnostic algorithm. The model incorporated an adaptation module to adjust and optimize the performance of the model. The authors suggested that the detection rate of tool wear was 97% using this adaptation feature, but did not explain how the adjustments to the membership functions changed the fuzzy rules. Along this line, Li *et al* (2000) developed a fuzzy neural network to describe the relationship between the characteristics of vibration and the tool wear condition. The authors suggested that the combined model has a higher learning speed and better flexibility comparing with the conventional neural network. However, the results obtained through experiments do not suggest a high accuracy for the system as minimum recognition rate attained 68% for severe state.

Zhang and Morris (1996) proposed a fuzzy neural network to model a process. Process knowledge was used to initially divide the process operation into several fuzzy operating regions and to set-up the initial fuzzification layer weights. The fuzzification layer converted the increments in on-line measurements and controller outputs into three fuzzy sets: "increase", "steady" and "decrease". The authors stated that the combined model is "not purely a black box". This may suggest that the system has been substantially improved.

Goode and Chow (1995) used the fuzzy logic as a mean of adding some heuristic reasoning to the neural network model which gives normally accurate results but does not provide heuristic interpretation of the solution. The purpose of this combined model was to detect and diagnose faults in induction motors. The training was performed using a set of sensor data such as current and speed. The trained model was then used to detect and classify bearing wear. The detection and classification rate was 98%. Chow *et al* (1993) presented the general design considerations of artificial neural networks to perform motor fault detection. The design considerations covered in their study were network performance, network implementation, size of training data set, and assignment of training parameter values. A fuzzy logic approach was used to optimize the selection of the inputs values and consequently reduce the time required to obtain a satisfactory network configuration.

#### 2.2.2.2 Hidden Markov Model

Hidden Markov Model (HMM) has been a dominant method in speech recognition technique since 1960s and become popular in various areas in the last few years. The increasing popularity of HMM is based on its rich mathematical structure and proven accuracy on critical applications (Rabiner and Juang 1993). Although it has become popular in various areas like signal analysis and pattern recognition, its use in mechanical engineering field has been very limited (Lee *et al* 2004).

Recently, Lee *et al* (2004) proposed a continuous HMM to monitor and diagnose mechanical faults. Six types of faults were simulated in the experimental work and isolated successfully. The authors claimed high accuracy for their model but did not show the detection rate or efficiency. In addition they did not specify the required training time. It was noted also from the probability matrix prepared for the six types of failures that all these failures are uncoupled in terms of frequency contents, which makes the problem easier to solve. Similarly, Ertunc and Oysu (2004), and Ertunc and Loparo (2001) developed a monitoring technique namely the bargraph method based on continuous HMM. This method was applied for the detection and identification of drill wear. Although the authors were satisfied with the results, there was no description about the HMM, nor the training time or detection efficiency.

Couvreur *et al* (1998) used a discrete HMM to build an environmental noise recognition system. The authors used the cepstral coefficients to classify five types of noise events: car, truck, moped, aircraft and train. The results of the tests performed with the model showed an average recognition rate of 93%.

#### 2.2.2.3 Summary

As seen above, artificial intelligence methods were used in the last three decades to assist with human reasoning process and create monitoring systems for processes and machinery fault diagnosis. As such, expert systems, neural networks, fuzzy logic and fuzzy-neural networks were employed to detect and assess faults prior to decision-making. Despite the successful results reported in different applications, researchers identified the weaknesses of these methods. For this reason, researchers started lately looking for alternative methods to improve monitoring systems. The hidden Markov model, utilized extensively in the domain of speech recognition, was selected as a possible replacement method capable of filling the "gaps" created by the pre-listed techniques.

## 2.3 Motivations and Objectives

As technology is evolving rapidly and tending towards complex processes and machinery, feature extraction becomes a more attractive tool especially with the availability of sophisticated computing and processing techniques.

As described earlier, the estimation methods require thorough process knowledge to develop the model. Depending on the complexity of the system, the implementation of the model may require significant efforts and computation time. In addition, the model is normally not flexible; a new model has to be designed for every type of machinery and any changes to the model parameters may lead to model reconfiguration (Sorsa *et al* 1991 and Grimmelius *et al* 1999). Moreover, the model is always an estimation of the real process or machinery. This is in fact reflected in the poor accuracy results of some of the cases listed above.

On the other hand, the feature extraction techniques require little process knowledge to define the appropriate features that lead to simple rules (Grimmelius *et al* 1999, Himmelblau 1978, and Pao 1989). In addition, these methods are flexible and easy to implement.

Nevertheless, all the fault detection methods described earlier are mostly application-specific, i.e., they are developed either for machinery condition monitoring or for process monitoring, but not for both. However, in many cases, the machinery faults and process malfunctions co-exist. To monitor both types of faults, several such application-specific monitoring systems have to be designed, tested, implemented and maintained separately. This inevitably leads to duplicated efforts, lengthy development processes and consequently high costs. For these reasons, the author proposes few generic condition indicators that can be utilized for both process and machinery fault detection. This is highly desirable in practice as stated by Ghamri *et al* (2002).

Similarly, on the fault diagnostic front, all available artificial intelligence techniques have their drawbacks (Hu *et al* 2000). According to Lee *et al* (2004) these methods are often unreliable.

Expert systems have been used successfully in complicated systems, however complex algorithms have to be updated by an experienced engineer for every target system (Lee *et al* 2004). Furthermore, Crupi *et al* (2004) suggested that expert systems could not be applied generically due to lack of knowledge. Add to that, its disability to adapt to dynamic environment or to handle new situations that are not covered in its knowledge bases (Pham and Pham 1999).

Fuzzy logic has gained wide acceptance as a useful tool for blending objectivity with flexibility particularly in the area of process control. The method is fast as decision is based on some simple rules, however its membership function is defined in a relatively subjective manner and the method could suffer inaccuracy with complex systems (Lee *et al* 2004, Mechefske 1998, and Goode and Chow 1995).

Artificial neural network has superior learning and noise suppression abilities and does not require process knowledge (Dornfield and Rangwala 1990, and Grimmelius *et al* 1999). However, it requires some knowledge or a "feel" for the training. In fact, training is time consuming as the system requires an extensive set of data for all classes of conditions including faults. The system is also inflexible: if dynamic conditions change values outside the trained range, the network must be retrained (Lee *et al* 2004, and Grimmelius *et al* 1999). In addition, it is difficult to select the best inputs to the model that allow model optimization in terms of size and processing time (Jack and Nandi 2000).

Combining fuzzy logic with neural network improves the input selection problem but does not alleviate the training or the flexibility concerns.

On the other hand, the application of HMM in process monitoring is of great interest because of their strong capability in representing non-stationary machinery signals. Furthermore, in fault diagnosis, there always exists uncertainty, randomness and incompleteness originating from various sources. Stochastic models are known to deal with these problems efficiently by using probabilistic models, thus widely used in signal modeling. Among these stochastic approaches, HMM has proven very effective and accurate in modeling both dynamic and static signals (Cho *et al.* 1995, Bunks *et al.* 2000, and Couvreur *et al* 1998). Though the HMM requires also patterns for every symptom of the system, new symptoms can be simply trained and added to existing library (Lee *et al* 2004).

As the main purpose of this study is to propose a robust generic fault detection and diagnosis technique for both machinery and processes and as the author did not find many (to not say any) papers that cover all the monitoring aspects described earlier, the author chooses a two-stage monitoring strategy:

i- Develop generic condition indicators and fuse them into a fuzzy model to improve fault detection effectiveness

ii- Diagnose the fault (estimation of fault severity and isolation of fault location) using HMM.

# Chapter 3

# Condition Indices

As technology is evolving rapidly, manufacturing enterprises are nowadays attracted towards the development of automatic machine and process condition monitoring systems to improve machine reliability, increase process productivity and enhance product quality. The ability to recognize process anomalies and recommend actions is the key to success of such systems. This in turn relies on the availability of reliable input data and anomaly indicators. This chapter describes the sensor and signal processing method to be used for data acquisition and presents several indicators for fault detection. Experimental studies are also reported to illustrate the applications of the proposed fault indicators in both tool and bearing condition monitoring.

## 3.1 Monitoring System

The monitoring system to be used in this chapter is composed of the following elements (Figure 3.1): a sensor, a signal conditioning unit, a number of condition indicators, and a decision-making module. The details of these elements are explained in the following sections.

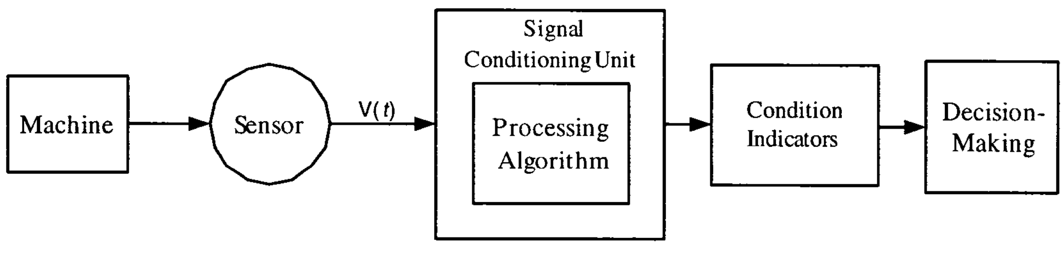



Figure 3.1 Monitoring system description.

### 3.1.1 Sensor

The sensor is used to capture the machine signal and transmit it to the conditioning unit for processing. The sensor signal is usually analog shown in Figure 3.1 as a voltage in function of time ($V(t)$).

### 3.1.2 Signal Processing Method

The signal obtained from the sensor is conditioned and processed to extract useful information. The following tasks are performed as part of this activity.

#### 3.1.2.1 Discretization

The discretization consists of sending the analog signal through an analog to digital (A/D) converter prior to processing as shown in Figure 3.2. The A/D converter samples the signal at a frequency called the sampling frequency. This sampling frequency must be at least two times of the maximum frequency detected by the sensor in order to prevent aliasing (Nyquisit Criteria). The main advantages of working in the digital domain (instead of the analog domain) are the flexibility in reconfiguring the digital signal processing operations and the control of the accuracy requirements.

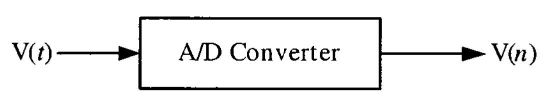



Figure 3.2 Signal through analog to digital converter.

#### 3.1.2.2 Filtering

##### 3.1.2.2.1 Scanning Spectrum

The main objective of this task is to define the regions of interest in the signal spectrum. This can be achieved by scanning the spectrum using the power density function. Upon completion of this activity, the filters cut-off frequencies can be determined. Therefore, some experimental work (learning phase) is required to determine the necessary spectrum bandwidth.

Let's take $c_k$ as the Fourier coefficient for series $V(n)$, where $k$ can vary between 0 and $C$-1. $C$ is a specific number of samples. $c_k$ and $V(n)$ are related as follows (Proakis and Manolakis 1996):

$$c_k = \frac{1}{C}\sum_{n=0}^{C-1} V(n) e^{-j2\pi kn/C} \qquad k=0,1,\ldots, C-1 \tag{3.1}$$

The sequence $|c_k|^2$ for $k$=0,1,...., $C$-1, is the distribution of power as a function of frequency and is called the *power spectrum density* of the signal.

#### 3.1.2.2.2 Bandpass Filter

The filters are used to remove the undesired frequencies in order to improve the signal to noise ratio in the regions of interest defined in the above paragraph. In this study it is proposed to use a 3rd order lowpass Butterworth filter and a 3rd order highpass Butterworth filter to build a bandpass filter. The third order is chosen in this case to minimize the processing time without significantly affecting the accuracy of the results. The cut-off frequencies of these filters are defined by the investigation performed in Section 3.2.1.2.

The digital signal is filtered first by the lowpass filter as shown in Figure 3.3. The output from the filter, $V_{LP}(n)$, is expressed as follows:

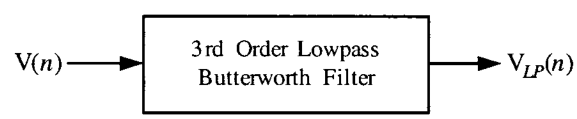



Figure 3.3 Signal through lowpass filter.

$$\begin{aligned} V_{LP}(n) = {} & a_1 V_{LP}(n-1) + a_2 V_{LP}(n-2) + a_3 V_{LP}(n-3) - b_0 V(n) - b_1 V(n-1) - \\ & b_2 V(n-2) - b_3 V(n-3) \end{aligned} \tag{3.2}$$

where $V_{LP}(n)$ is the filtered signal, and $a_i$ and $b_i$ are the filter coefficients. Those coefficients depend on the cut-off frequency and sampling frequency. Their values are determined in the following way.

For a third order Butterworth filter where $T$ is the sampling period (equal to $1/F_s$) and $w_{LP}$ is the cut-off frequency, the transfer function is as follows.

$$G(s)=\frac{w_{LP}{}^{3}}{s^3+2w_{LP}s^2+2w_{LP}^2s+w_{LP}^3} \tag{3.3}$$

Using the bilinear transform $s=\frac{2}{T}\frac{(z-1)}{z+1}$ to switch to the $z$ domain and assuming that the warping constant is equal to $2/T$, equation (3.4) leads to the following:

$$G(z)=\frac{w_{LP}^3}{(\frac{2}{T}\frac{(z-1)}{(z+1)})^3+2w_{LP}(\frac{2}{T}\frac{(z-1)}{(z+1)})^2+2w_{LP}^2(\frac{2}{T}\frac{(z-1)}{(z+1)})+w_{LP}^3} \tag{3.4}$$

Developing the numerator and denominator and then dividing them by $z^3(\frac{2}{T})^3+2w_{LP}(\frac{2}{T})^2+2w_{LP}^2(\frac{2}{T})+w_{LP}^3$ gives the above equation the following form:

$$G(z)=\frac{b_0+b_1z^{-1}+b_2z^{-2}+b_3z^{-3}}{1+a_1z^{-1}+a_2z^{-2}+a_3z^{-3}} \tag{3.5}$$

This manipulation leads to the following:

$$b_0=\frac{w_{LP}^3}{(\frac{2}{T})^3+2w_{LP}(\frac{2}{T})^2+2w_{LP}^2(\frac{2}{T})+w_{LP}^3} \tag{3.6}$$

$$b_1=\frac{3w_{LP}^3}{(\frac{2}{T})^3+2w_{LP}(\frac{2}{T})^2+2w_{LP}^2(\frac{2}{T})+w_{LP}^3} \tag{3.7}$$

$$b_2=\frac{3w_{LP}^3}{(\frac{2}{T})^3+2w_{LP}(\frac{2}{T})^2+2w_{LP}^2(\frac{2}{T})+w_{LP}^3} \tag{3.8}$$

$$b_3=\frac{w_{LP}^3}{(\frac{2}{T})^3+2w_{LP}(\frac{2}{T})^2+2w_{LP}^2(\frac{2}{T})+w_{LP}^3} \tag{3.9}$$

$$a_1=\frac{-3(\frac{2}{T})^3-2w_{LP}(\frac{2}{T})^2+2w_{LP}^2(\frac{2}{T})+3w_{LP}^3}{(\frac{2}{T})^3+2w_{LP}(\frac{2}{T})^2+2w_{LP}^2(\frac{2}{T})+w_{LP}^3} \tag{3.10}$$

$$a_2 = \frac{3(\frac{2}{T})^3 - 2w_{LP}(\frac{2}{T})^2 - 2w_{LP}^2(\frac{2}{T}) + 3w_{LP}^3}{(\frac{2}{T})^3 + 2w_{LP}(\frac{2}{T})^2 + 2w_{LP}^2(\frac{2}{T}) + w_{LP}^3} \tag{3.11}$$

$$a_3 = \frac{-(\frac{2}{T})^3 + 2w_{LP}(\frac{2}{T})^2 - 2w_{LP}^2(\frac{2}{T}) + w_{LP}^3}{(\frac{2}{T})^3 + 2w_{LP}(\frac{2}{T})^2 + 2w_{LP}^2(\frac{2}{T}) + w_{LP}^3} \tag{3.12}$$

After the lowpass filtering, the signal is further filtered by the highpass filter as shown in Figure 3.4 and the output, $V_{LHP}(n)$, is:

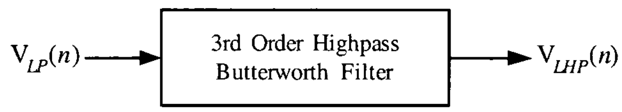



Figure 3.4 Signal through highpass filter.

$$\begin{aligned} V_{LHP}(n) = {} & \alpha_1 V_{LHP}(n-1) + \alpha_2 V_{LHP}(n-2) + \alpha_3 V_{LHP}(n-3) - \beta_0 V_{LP}(n) - \\ & \beta_1 V_{LP}(n-1) - \beta_2 V_{LP}(n-2) - \beta_3 V_{LP}(n-3) \end{aligned} \tag{3.13}$$

where $\alpha_i$ and $\beta i$ are the coefficients of the high-pass filter. Those coefficients depend strongly on the cut-off frequency and sampling frequency. The calculation of these coefficients is shown below.

To obtain a highpass filter, one should replace $s$ in equation (3.4) by $\frac{w_{HP}^{\;2}}{s}$ as follows:

$$\begin{aligned} H(s) &= \frac{w_{HP}^{\;3}}{\left(\frac{w_{HP}^2}{s}\right)^3 + 2w_{HP}\left(\frac{w_{HP}^2}{s}\right)^2 + 2w_{HP}^2\left(\frac{w_{HP}^2}{s}\right) + w_{HP}^3} \\ &= \frac{s^3}{w_{HP}^3 + 2w_{HP}^2 s + 2w_{HP} s^2 + s^3} \end{aligned} \tag{3.14}$$

Using the binomial transformation (replace $s$ by $\frac{2}{T}\frac{(z-1)}{(z+1)}$), equation (3.15) becomes:

$$H(z)=\frac{(\frac{2}{T}\frac{(z-1)}{(z+1)})^3}{(\frac{2}{T}\frac{(z-1)}{(z+1)})^3+2w_{HP}(\frac{2}{T}\frac{(z-1)}{(z+1)})^2+2w_{HP}^2(\frac{2}{T}\frac{(z-1)}{(z+1)})+w_{HP}^3} \tag{3.15}$$

Developing the numerator and denominator and then dividing them by $z^3\left((\frac{2}{T})^3+2w_{HP}(\frac{2}{T})^2+2w_{HP}^2(\frac{2}{T})+w_{HP}^3\right)$ leads to the following filter coefficients:

$$\beta_0=\frac{(\frac{2}{T})^3}{(\frac{2}{T})^3+2w_{HP}(\frac{2}{T})^2+2w_{HP}^2(\frac{2}{T})+w_{HP}^3} \tag{3.16}$$

$$\beta_1=\frac{-3(\frac{2}{T})^3}{(\frac{2}{T})^3+2w_{HP}(\frac{2}{T})^2+2w_{HP}^2(\frac{2}{T})+w_{HP}^3} \tag{3.17}$$

$$\beta_2=\frac{3(\frac{2}{T})^3}{(\frac{2}{T})^3+2w_{HP}(\frac{2}{T})^2+2w_{HP}^2(\frac{2}{T})+w_{HP}^3} \tag{3.18}$$

$$\beta_3=\frac{-(\frac{2}{T})^3}{(\frac{2}{T})^3+2w_{HP}(\frac{2}{T})^2+2w_{HP}^2(\frac{2}{T})+w_{HP}^3} \tag{3.19}$$

$$\alpha_1=\frac{-3(\frac{2}{T})^3-2w_{HP}(\frac{2}{T})^2+2w_{HP}^2(\frac{2}{T})+3w_{HP}^3}{(\frac{2}{T})^3+2w_{HP}(\frac{2}{T})^2+2w_{HP}^2(\frac{2}{T})+w_{HP}^3} \tag{3.20}$$

$$\alpha_2=\frac{3(\frac{2}{T})^3-2w_{HP}(\frac{2}{T})^2-2w_{HP}^2(\frac{2}{T})+3w_{HP}^3}{(\frac{2}{T})^3+2w_{HP}(\frac{2}{T})^2+2w_{HP}^2(\frac{2}{T})+w_{HP}^3} \tag{3.21}$$

$$\alpha_3=\frac{-(\frac{2}{T})^3+2w_{HP}(\frac{2}{T})^2-2w_{HP}^2(\frac{2}{T})+w_{HP}^3}{(\frac{2}{T})^3+2w_{HP}(\frac{2}{T})^2+2w_{HP}^2(\frac{2}{T})+w_{HP}^3} \tag{3.22}$$

### 3.1.3 Monitoring Technique

As the need to use a high sampling rate increases significantly in the processing methods of recent monitoring systems, the degree of complexity grows simultaneously due to the size of data to be handled in a very short time. Therefore, looking for anomalies in a huge data package is a very challenging task. In order to efficiently accomplish this task, the data acquired after the filtering stage must be processed and manipulated to produce the desired results. This data manipulation is described below.

#### 3.1.3.1 "Classes" Definition

To improve the detection efficiency and the robustness of the monitoring technique, it is necessary first to understand the nature and the characteristics of the anomalies that could be embedded into the sensor signal. In general, two types of abnormalities exist: transient and continuous.

The transient anomaly exists for a relatively short period of time (e.g. fracture). The duration of this event can be less than 0.01 second (Beranek 1971). However, the continuous anomaly can be repeatable and extends normally for a relatively long duration (e.g. bearing failure). For this reason, the data is assembled into mini-groups and sub-groups as shown in Figure 3.5. Each mini-group will contain $K'$ data points and each sub-group will consist of $J'$ mini-groups. The size of the mini-group represents the search resolution. Increasing the number of points in each mini-group reduces the number of processing steps and consequently the analysis time, but also reduces at the same time the detection efficiency of transient abnormal signals. The author recommends to use a size equivalent to a duration varying between 0.001 and 0.020 seconds. It is based on what has been recommended as window size in the speech recognition techniques (Lee *et al.* 1996). On the other hand, the size of the sub-group defines the size or the duration of the monitoring interval. This interval can be composed of 6 to 30 mini-groups depending on the type of transient noise.

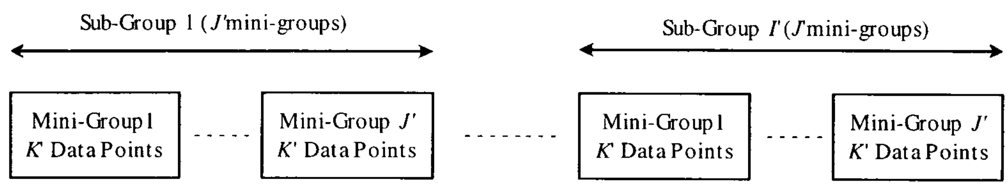



Figure 3.5 Description of mini-groups and sub-groups of samples.

All the processing calculations leading to the definition of the condition indicators are performed based on these two levels of groups or classes as described below.

### 3.1.3.2 Condition Indicators

The condition indicators must be used to detect both types of anomalies previously defined and to support the "decision-making" phase. In this study, four (4) condition indicators are defined, i.e., power, standard-deviation, power correlation factor and standard-deviation correlation factor.

The condition indicators are computed with respect to the "classes" defined earlier and are obtained as follows.

**Step 1.** Calculate the power and standard-deviation with respect to each mini-group.

*Mini-group power*

$$P_{j'} = \frac{1}{K'} \sum_{k'=(j'-1)K'+1}^{j'K'} V_{LHP}^{2}(k') \qquad \forall j' \tag{3.23}$$

where $K'$ is the number of samples in a mini-group and $V_{LHP}(k')$ represents the stream of data obtained after filtering.

*Standard-deviation of mini-group power*

$$\sigma_{j'} = \sqrt{\frac{\sum_{k'=(j'-1)K'+1}^{j'K'} (V_{LHP}(k') - \mu_{j'})^2}{K' - 1}} \qquad \forall j' \tag{3.24}$$

where $\mu_{j'}$ is the mean sample value of in a mini-group defined as:

$$\mu_{j'} = \frac{\sum_{k'=(j'-1)K'}^{j'K'} V_{LHP}(k')}{K'} \qquad \forall j' \tag{3.25}$$

**Step 2.** Calculate the average power and standard-deviation with respect to each sub-group.

Given the above mini-group parameters, the "sub-group" parameters can be expressed as follows:

*Mean sub-group power*

$$\overline{P_{i'}} = \frac{1}{J'} \sum_{j'=(i'-1)J'+1}^{i'J'} P_{j'} \qquad \forall i' \tag{3.26}$$

*Mean standard-deviation of sub-group power*

$$\overline{\sigma_{i'}} = \frac{1}{J'} \sum_{j'=(i'-1)J'+1}^{i'J'} \sigma_{j'} \qquad \forall i' \tag{3.27}$$

**Step 3.** Find the maximum power and standard-deviation in each sub-group.

*Maximum sub-group power*

$$P_{i',\max} = \max_{(i'-1)J'+1 \le j' \le i'J'} (P_{j'}) \qquad \forall i' \tag{3.28}$$

*Maximum standard-deviation of sub-group power*

$$\sigma_{i',\max} = \max_{(i'-1)J'+1 \le j' \le i'J'} (\sigma_{j'}) \qquad \forall i' \tag{3.29}$$

**Step 4.** Find the minimum power and standard-deviation in each sub-group.

*Minimum sub-group power*

$$P_{i',\min} = \min_{(i'-1)J'+1 \le j' \le i'J'} (P_{j'}) \qquad \forall i' \tag{3.30}$$

*Minimum standard-deviation of sub-group power*

$$\sigma_{i',\min} = \min_{(i'-1)J'+1 \le j' \le i'J'} (\sigma_{j'}) \qquad \forall i' \tag{3.31}$$

**Step 5.** Calculate the power and standard-deviation correlation factors**.**

The correlation factor can be applied to both the mean power and mean standard-deviation matrices. This is done by taking the first $J_b$ mini-groups, calculating their corresponding power and standard-deviation values, and then using these values as reference in the correlation factor calculation as shown below.

Let's define for example the following parameters:

$$\hat{P} = \frac{\sum_{j'=1}^{J_b} P_{j'}}{J_b} \tag{3.32}$$

where $\hat{P}$ is the average power of the $J_b$ mini-groups. Using the following coefficients:

$$SSV_p = \sum_{j'=J_b+1}^{J_b+J'} P_{j'}^2 - \frac{(\sum_{j'=J_b+1}^{J_b+J'} P_{j'})^2}{J_b} \tag{3.33}$$

$$SSR_p = \sum_{j'=J_b+1}^{J_b+J'} (P_{j'} - \hat{P})^2 \tag{3.34}$$

where $SSV_p$ is the sum of the squared deviations of power relative to the varying mean power and $SSR_p$ is the sum of the squared errors of power relative to the reference mean power $\hat{P}$ (Sheaffer and McClave 1990). Hence, the power correlation factor is defined as:

$$corf_P = \sqrt{\frac{SSV_p}{SSR_p}} \tag{3.35}$$

Similarly, the correlation factor for the standard-deviation is defined as follows:

$$SSV_\sigma = \sum_{j'=J_b+1}^{J_b+J'} \sigma_{j'}^2 - \frac{(\sum_{j'=J_b+1}^{J_b+J'} \sigma_{j'})^2}{J_b} \tag{3.36}$$

$$SSR_\sigma = \sum_{j'=J_b+1}^{J_b+J'} (\sigma_{j'} - \hat{\sigma})^2 \tag{3.37}$$

$$\hat{\sigma} = \frac{\sum_{j'=1}^{J_b} \sigma_{j'}}{J_b} \tag{3.38}$$

$$corf_\sigma = \sqrt{\frac{SSV_\sigma}{SSR_\sigma}} \tag{3.39}$$

**Step 6.** Calculate the power and standard-deviation condition indicators.

A successful condition indicator must be sensitive to machinery conditions but insensitive to the variation of the operation aspects. Therefore, the power and the standard-deviation condition indicators must be normalized to satisfy the pre-defined criteria. In this study the normalization is performed in terms of the average, maximum and minimum values.

The power and standard-deviation condition indicators are proposed as follows.

*Power condition indicator*

$$(Power-CI)_{j'} = \frac{\left|P_{j'} - \overline{P_{i'}}\right|}{P_{i',\max} - P_{i',\min}} \qquad (i'-1)J' \leq j' \leq i'J',\ \forall i' \tag{3.40}$$

*Standard-Deviation condition indicator*

$$(Std-Dev-CI)_{j'} = \frac{\left|\sigma_{j'} - \overline{\sigma_{i'}}\right|}{\sigma_{i',\max} - \sigma_{i',\min}} \qquad (i'-1)J' \leq j' \leq i'J',\forall i' \tag{3.41}$$

The power correlation factor, standard-deviation correlation factor, power condition indicator and standard-deviation condition indicator will be examined and monitored simultaneously in the decision-making module.

### 3.1.4 Decision-Making

The decision about the equipment condition is made based on a time duration represented by the *I'* sub-groups and using all the pre-defined condition indicators. The continuous anomalies can be detected using the correlation factors. As the continuous anomaly becomes "stronger" or much severe, the correlation factors decrease towards zero. However, the power and standard-deviation condition indicators can be used especially to detect the transient anomaly. During a transient anomaly detection, the power and standard-deviation condition indicators increase towards one as the maximum values of the condition indicators in the corresponding sub-group become much greater than the mean values. This is expressed as:

$$\max(\lim_{anomaly\uparrow}(Power-CI)_{j'}) = \frac{P_{i',\max} - \overline{P_{i'}}}{P_{i',\max} - P_{i',\min}} \approx \frac{P_{i',\max}}{P_{i',\max}} = 1 \qquad (i'-1)J' \leq j' \leq i'J',\forall i' \tag{3.42}$$

$$\max(\lim_{anomaly\uparrow}(Std-Dev-CI)_{j'}) = \frac{\sigma_{i',\max} - \overline{\sigma_{i'}}}{\sigma_{i',\max} - \sigma_{i',\min}} \approx \frac{\sigma_{i',\max}}{\sigma_{i',\max}} = 1 \qquad (i'-1)J' \leq j' \leq i'J',\forall i' \tag{3.43}$$

The notation $lim_{anomaly\uparrow}$ is used to emphasize the severity of the anomaly.

## 3.2 Experimental Work

In order to test the efficiency of the pre-described method, the condition indicators defined above are used for tool and bearing monitoring.

### 3.2.1 Tool Monitoring

#### 3.2.1.1 Set-Up

A manual milling machine was used to gather the experimental data (Figure 3.6). The machine incorporates some semi-automatic features (e.g. power feeds) to control the feed-rate and cutting depth. Two types of cutters were utilized to conduct the tests (Figure 3.7). The acoustic signal was detected using a microphone attached to the milling machine at the following distances from the bottom center of the cutter (Figure 3.8):

horizontal distance = 1" and vertical distance = 0.25"

A laptop computer was used to collect the acoustic data and perform the necessary processing tasks. Table 3.1 summarizes the equipment specifications. Full immersion milling process was performed without coolant. The collected acoustic data also contain background noise. The experiments were conducted with three different tool conditions: normal cutting, tool wear and tool breakage.

Table3.1 Test equipment specifications.

| Item | Brand | Model | Other Specifications |
|---|---|---|---|
| Milling Machine | DoAll | GPM-200S | 2 hp motor, 570 V, 60 Hz |
| Cutter1 | B | - | 9/16" HSS, 4 flutes, 60 degrees helix angle |
| Cutter 2 | YG | - | 9/16" HSS, 4 flutes, 60 degrees helix angle |
| Microphone | Shure | VR116-L | Dynamic, unidirectional, 0-15 kHz |
| Workpiece | - | - | Flat, cold steel, grade 1040 |
| Laptop | Compaq | Presario | Pentium 4 processor |

For normal cutting, sharp tools or tools whose flank or crater wear are smaller than 0.1 mm and have no chipping, were used under stable cutting conditions. On the other hand, the tool wear considered in these tests was quite severe (between 0.4mm and

0.7mm flank wear). Finally, in order to induce or initiate the tool fracture, a *v notch* was created in one of the cutters. Each of the new and worn cutters was utilized at two different speeds using the settings shown in Table 3.2.

Table 3.2 Milling machine set-up specifications.

| Item | Setting |
| --- | --- |
| Feed rate | 0.07 in/s |
| Cutting depth | 0.04" |
| Spindle speed | 135 and 210 rpm |

**3.2.1.2 Frequency Regions of Interest**

The first step is to define regions of interest that can lead to the cut-off frequencies as described in Section 3.1.2.2.1. Using a sampling frequency of 22 kHz and the power spectrum density function, a significant increase of the acoustic level was found for the worn cutters in the frequency range of 2.5- 4.8 kHz (refer to Figures 3.9, 3.10, 3.11 and 3.12). This energy increase occurred due to the rubbing action of the tool and workpiece. This result seems to be in conformity with the conclusions drawn by Sadat and Raman (1987) who found that the noise level increases in the frequency range 2.75-3.5 kHz using a worn tool. In addition, the tool fracture increases energy in the overall spectrum including the pre-defined range but most of the increase is noted in the frequency range 6.3-7.3 kHz (Figure 3.13). Based on these observations, frequency range chosen for this study is [2000 Hz – 5000 Hz].

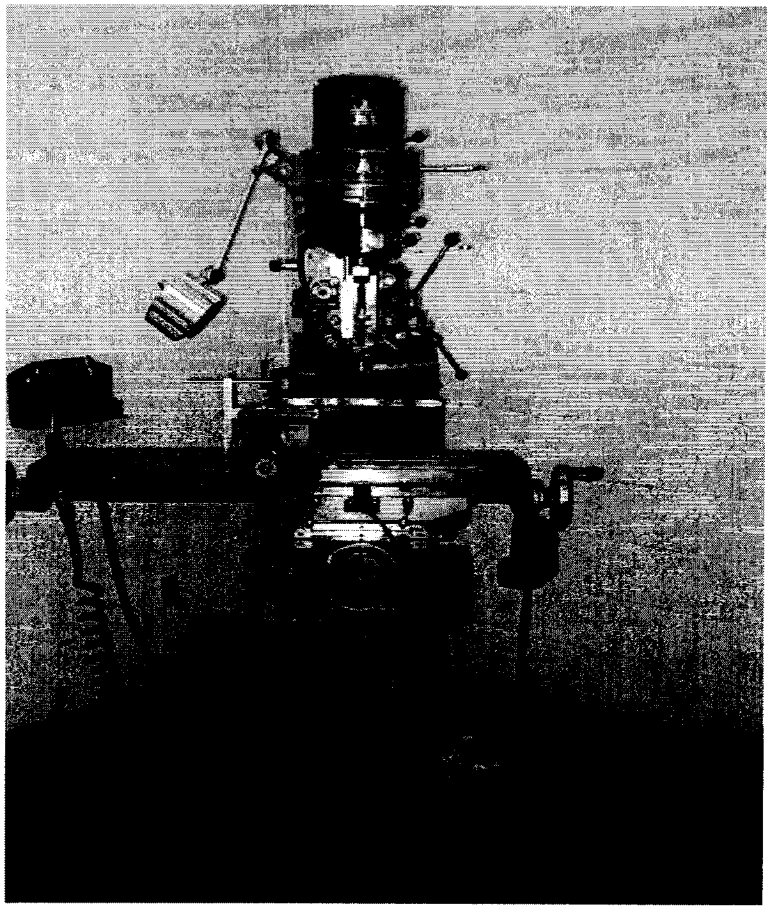


Figure 3.6 Milling machine used in experimental work.

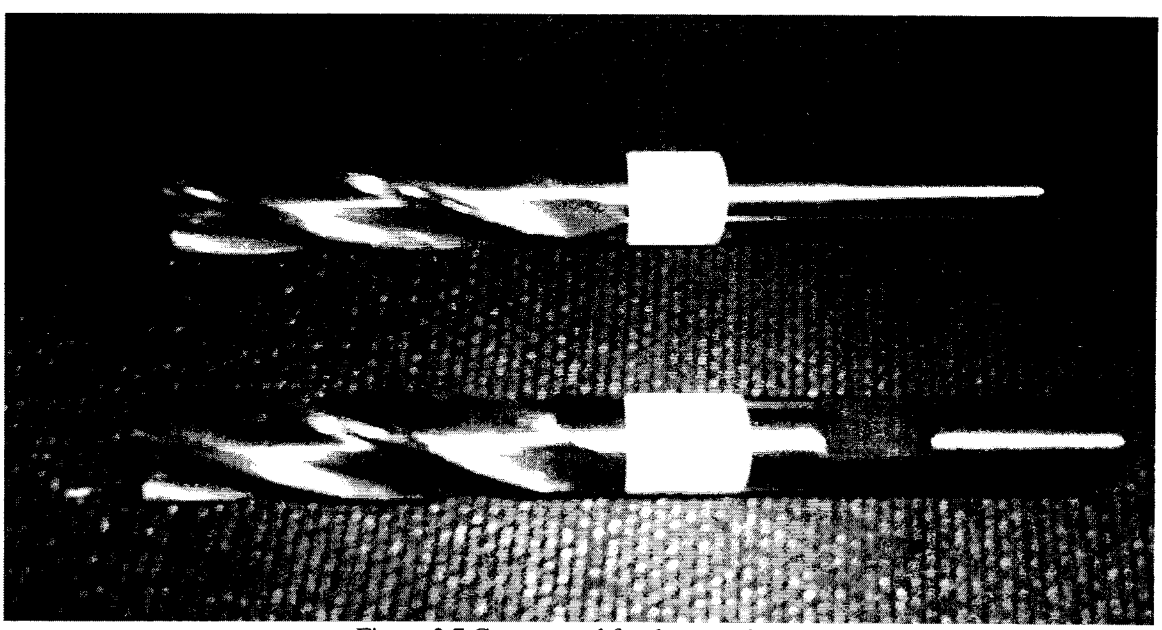


Figure 3.7 Cutters used for the experiments.

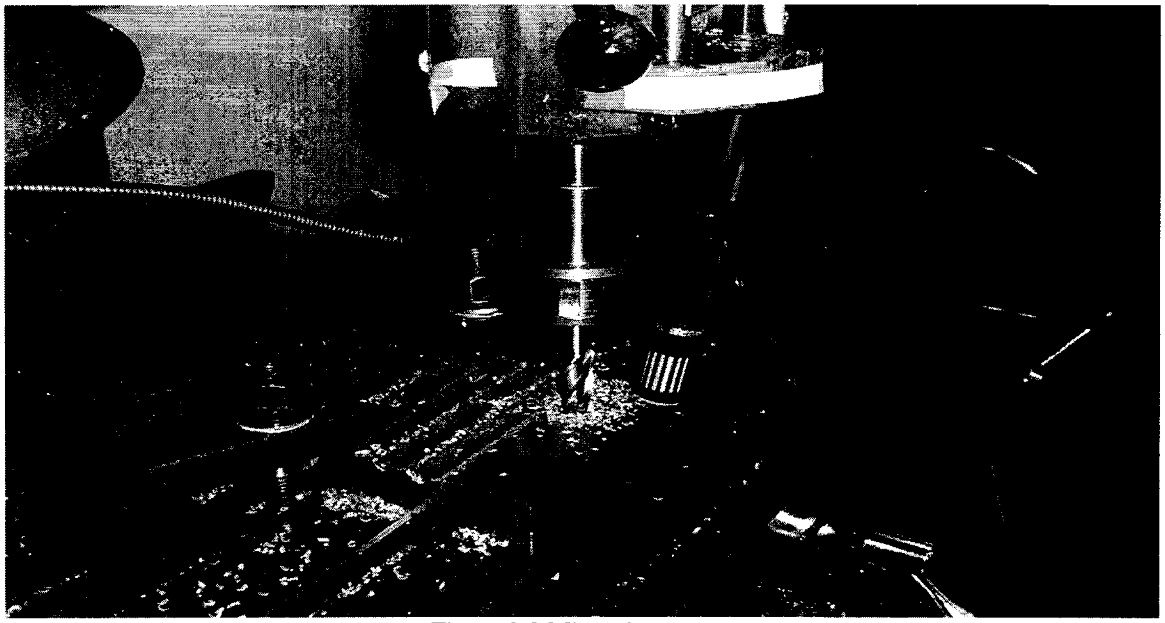


Figure 3.8 Microphone set-up.

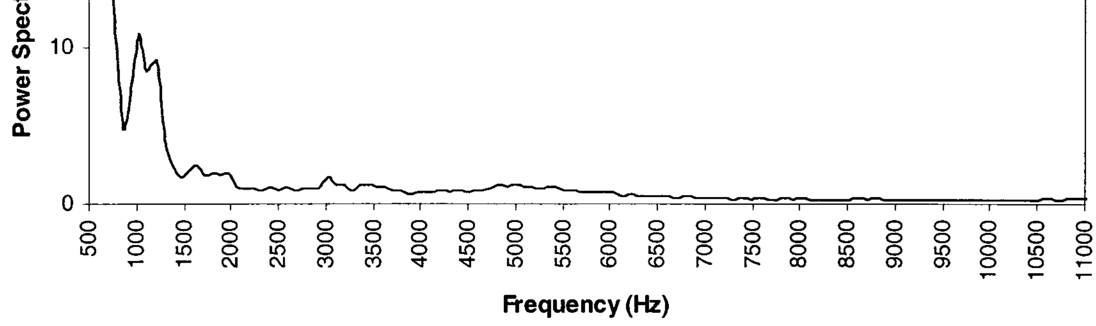



Figure 3.9 Power spectrum density using new cutter at 135 rpm

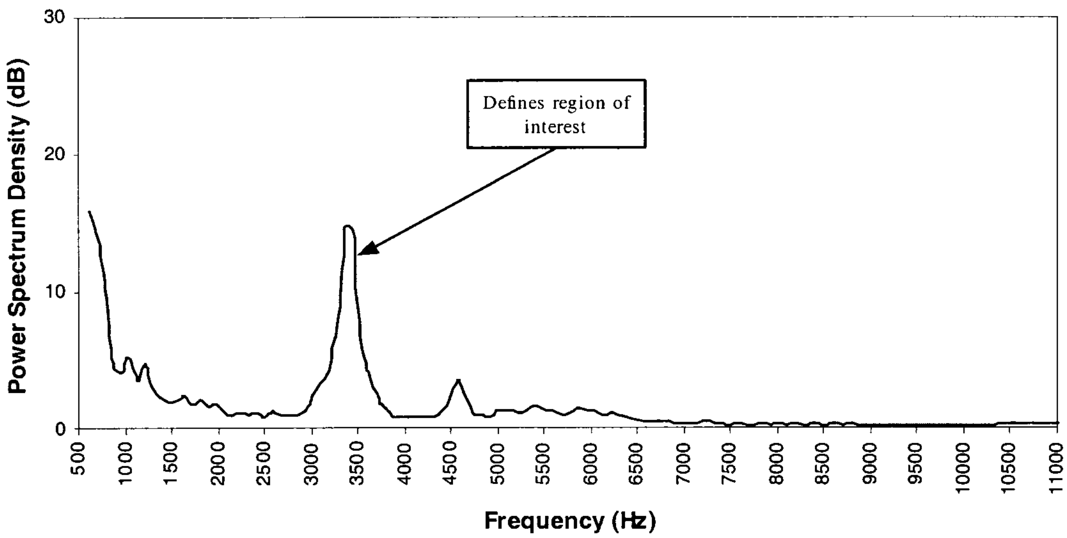



Figure 3.10 Power spectrum density using worn cutter at 135 rpm.

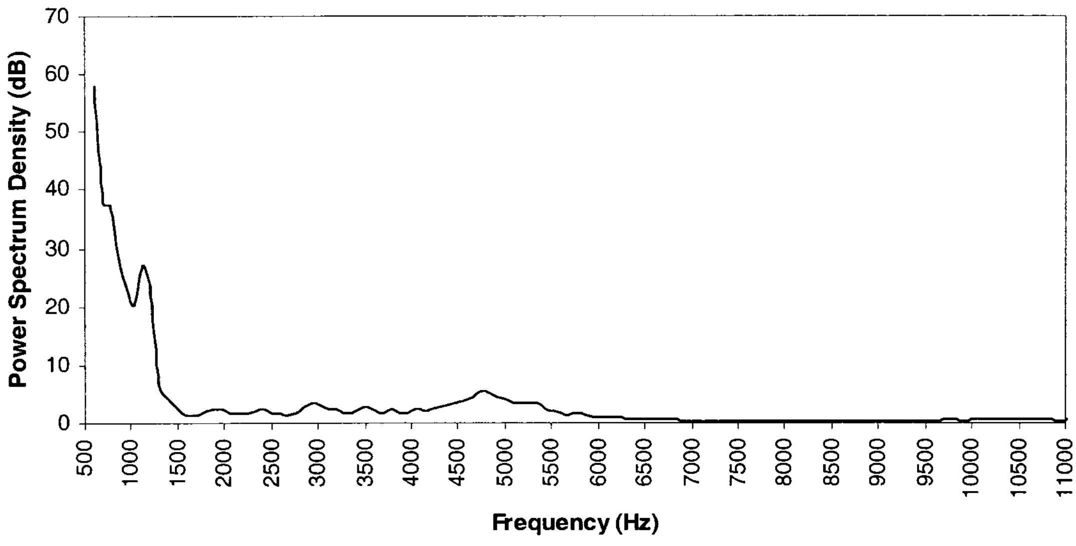



Figure 3.11 Power spectrum density using new cutter at 210 rpm.

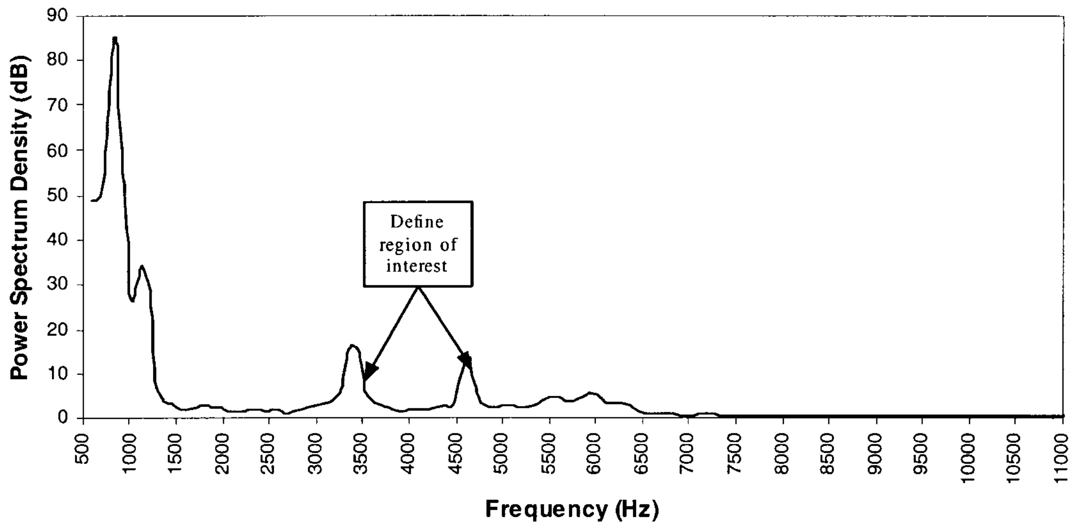



Figure 3.12 Power spectrum density using worn cutter at 210 rpm.

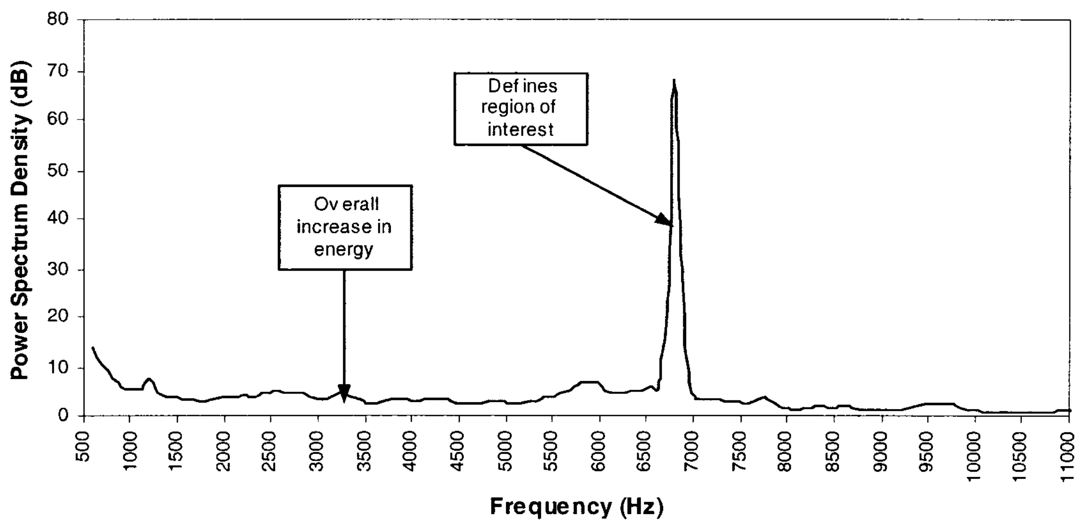



Figure 3.13 Power spectrum density during tool breakage.

### 3.2.1.3 Implementation and Results

The analysis described in Section 3.1.3 and the findings of Section 3.2.1.2 were implemented using a *C* code MATLAB program named *Monitoring*. The program code is attached to Appendix A. In order to simplify the case, one passband filter was used to filter the input signal as per Section 3.2.1.2 (i.e. bandwidth [2000 Hz-5000 Hz]).

The number of samples used in each mini-group was 200 (equivalent to 0.0091 seconds based on a sampling frequency of 22 kHz). The size of the mini-group is much smaller than the time required for one spindle rotation (0.444 seconds for 135 rpm and 0.285 seconds for 210 rpm). This facilitates the detection of possible transient anomalies during a full spindle rotation. In addition, each sub-group was composed of 25 mini-groups (equivalent to 0.2275 seconds). Samples of the results showing the efficiency of the condition indicators are represented below.

Figures 3.14 and 3.16 depict a typical representation of the power and standard deviation condition indicators for a new cutter. Figures 3.15, 3.17 and 3.18 show how the condition indicators increase towards one (1) at the instant of wear detection or tool breakage.

Similarly, Figures 3.19 and 3.21 illustrate the variation of the correlation factors for new cutters. These factors stay close to one (1) indicating as such the normal condition of the tool. Figures 3.20, 3.22 and 3.23 describe clearly the change of the sub-group power and correlation factors with respect to wear formation or fracture. It is clearly noted that in Figure 3.23, the correlation factors decrease significantly reaching a value of 0.1 for power, and hence indicating the occurrence of a severe event. This observation can be made also for Figure 3.18 where several successive peaks occur in a very short time leading to the conclusion that an unusual and severe incident has happened.

To examine the detection efficiency of the proposed technique and condition indicators, forty sub-groups (equivalent to approximately 9 seconds) of each of the collected data files were used for decision-making. Making the decision based on four successive monitoring intervals or sub-groups (equivalent to 0.9 seconds) (the number four (4) was arbitrarily chosen for decision-making), a failure classification was assigned upon finding at least two anomalies in two different intervals as per the following criteria:

- ➢ Power condition indicator above a threshold equal to 0.8, or
- ➢ Standard-deviation condition indicator above a threshold equal to 0.75, or
- ➢ One of the correlation factors below a threshold equal to 0.7

The above thresholds were assigned based on a learning period of 5 seconds where experiments were conducted with the new cutters. On the other hand, the minimum number of anomalies per interval, required for failure classification, was used mainly to eliminate false alarms. The success rate of the tool state detection was then computed by dividing the number of successful detections by the total number of decisions. Results are tabulated below.

Table 3.3 Success rates of tool state detection

| Case | Success Rate |
| --- | --- |
| Worn tool at 135 rpm | 100 % |
| Worn tool at 210 rpm | 100 % |
| Tool breakage | 100 % |

Table 3.3 shows that the proposed monitoring method is highly effective in detecting tool wear and tool breakage despite the dynamic nature of the process and other factors that may affect the signal characteristics.

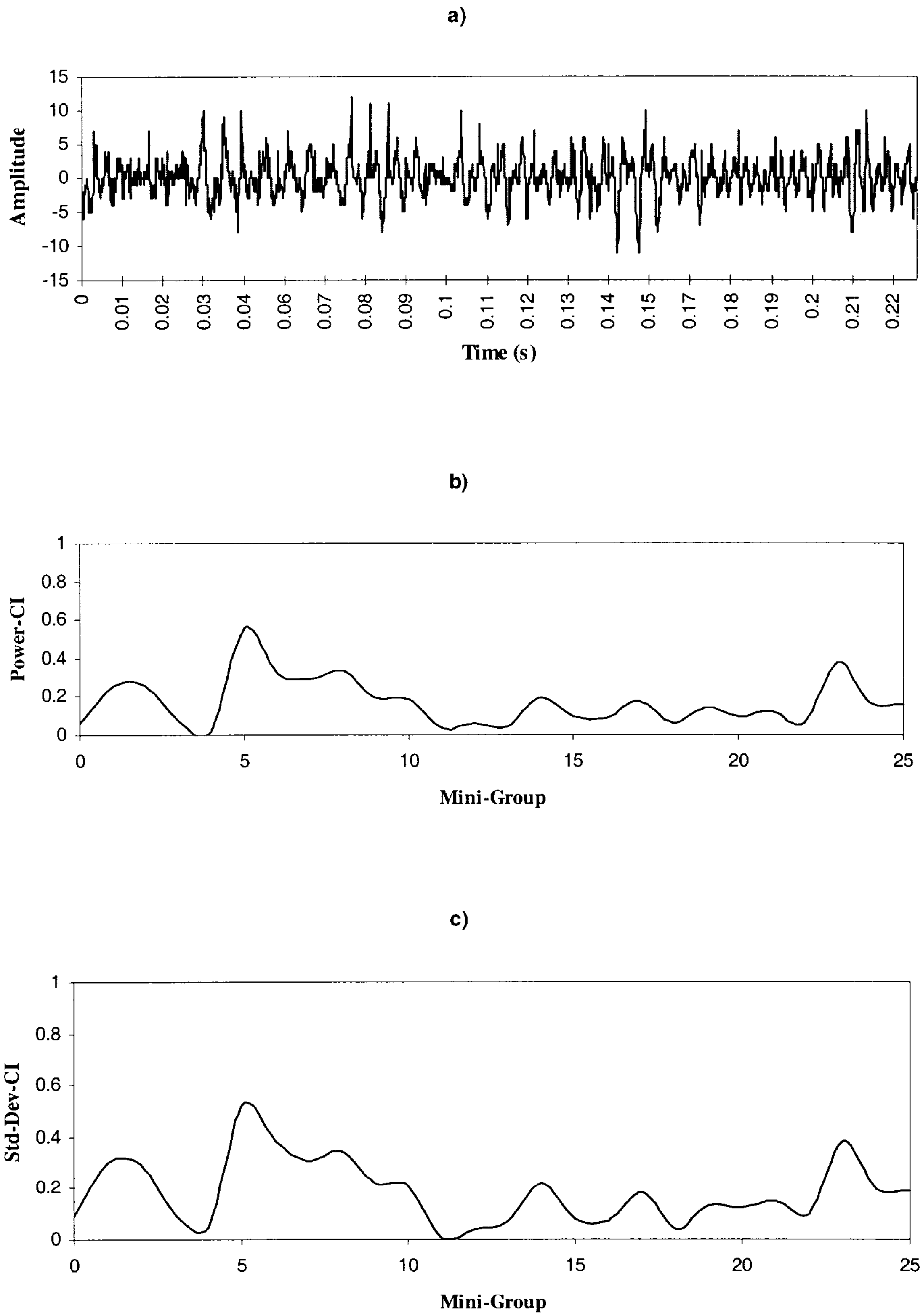



Figure 3.14 New cutter at 135 rpm: a) time signal, b) signal represented by power condition indicator, c) signal represented by standard-deviation condition indicator.

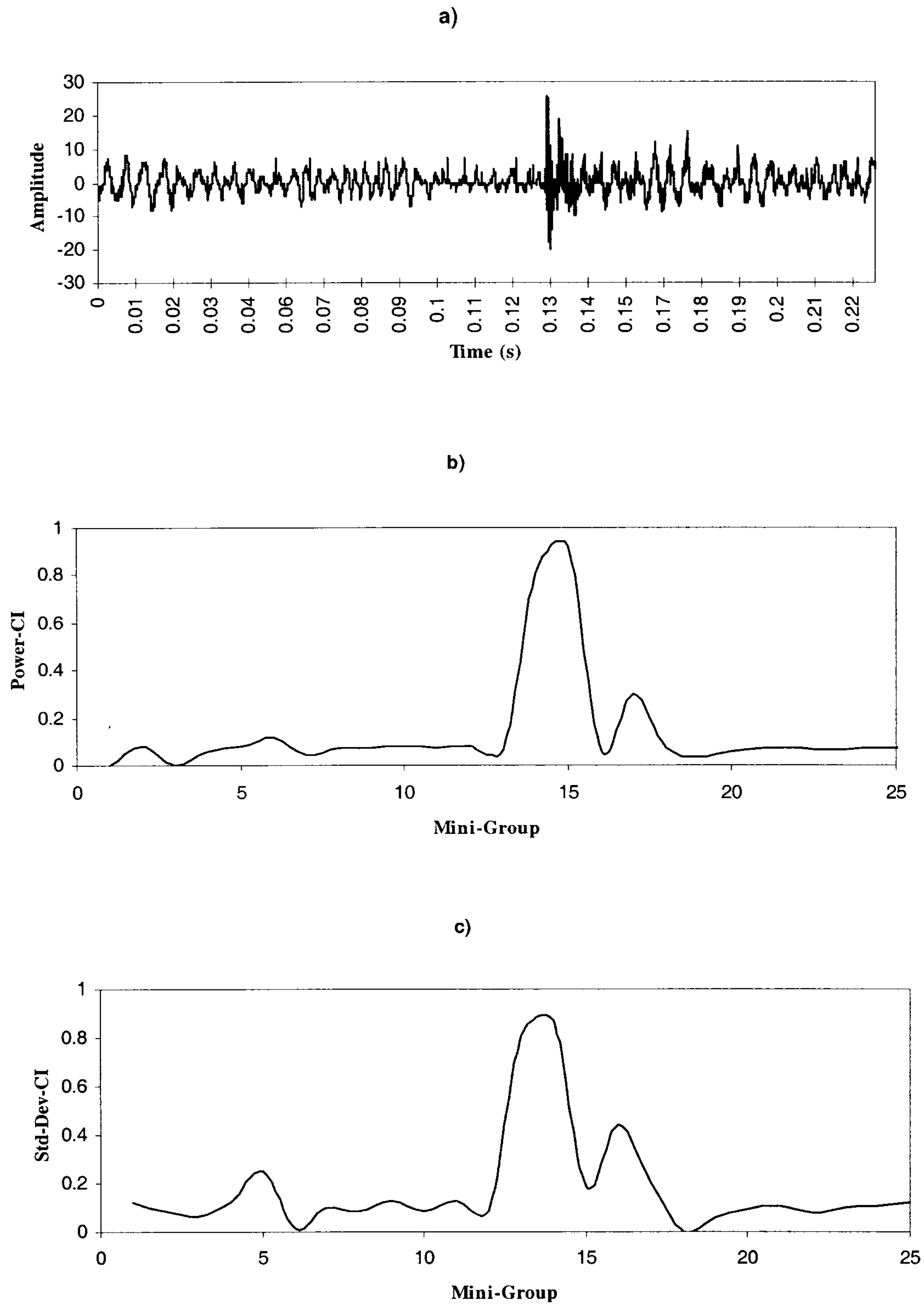



Figure 3.15 Worn cutter at 135 rpm: a) time signal, b) signal represented by power condition ndicator, c) signal represented by standard-deviation condition indicator.

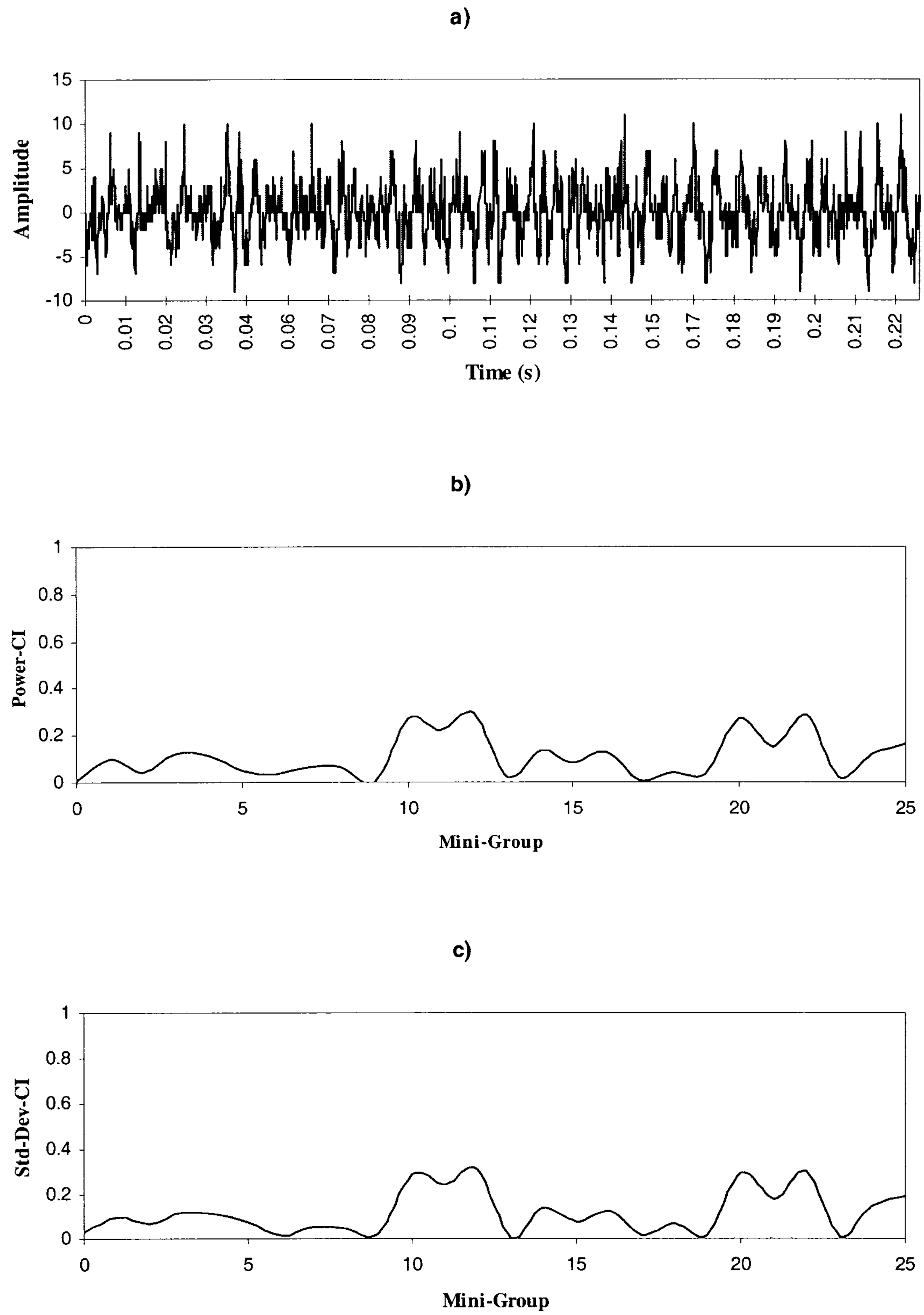



Figure 3.16 New cutter at 210 rpm: a) time signal, b) signal represented by power condition indicator, c) signal represented by standard-deviation condition indicator.

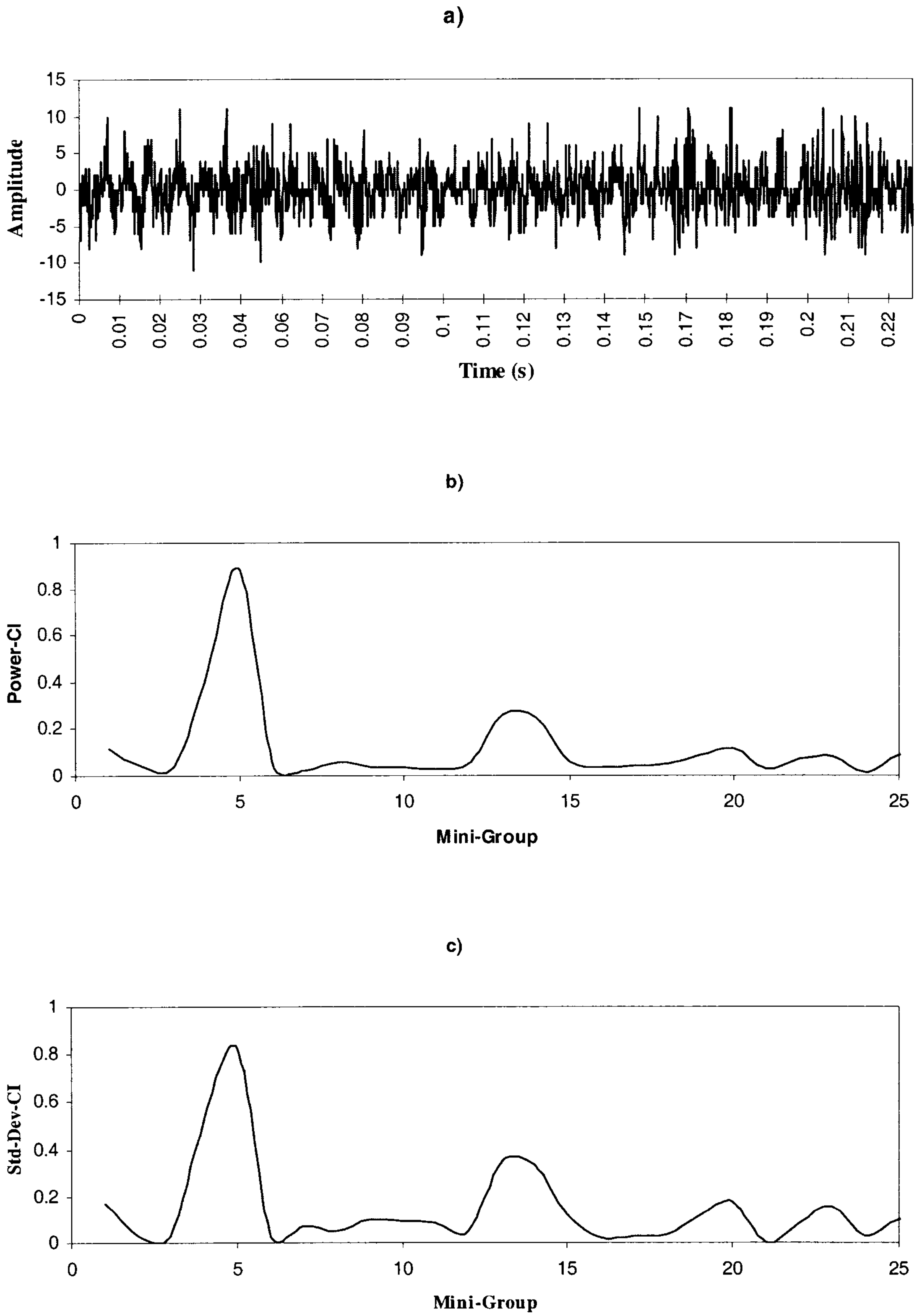



Figure 3.17 Worn cutter at 210 rpm: a) time signal, b) signal represented by power condition indicator, c) signal represented by standard-deviation condition indicator.

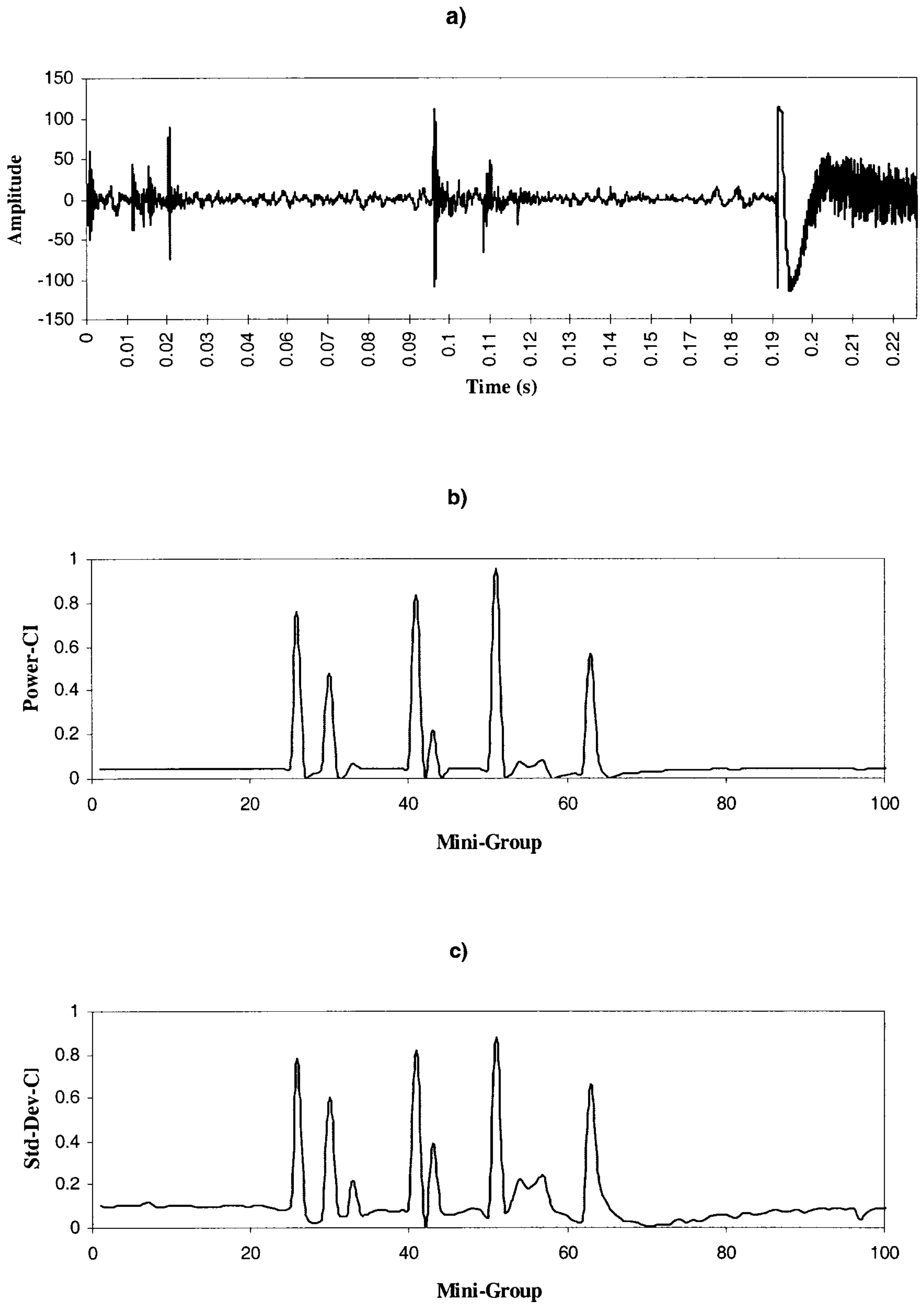



Figure 3.18 Tool breakage: a) time signal at fracture, b) signal represented by power condition indicator, c) signal represented by standard-deviation condition indicator.

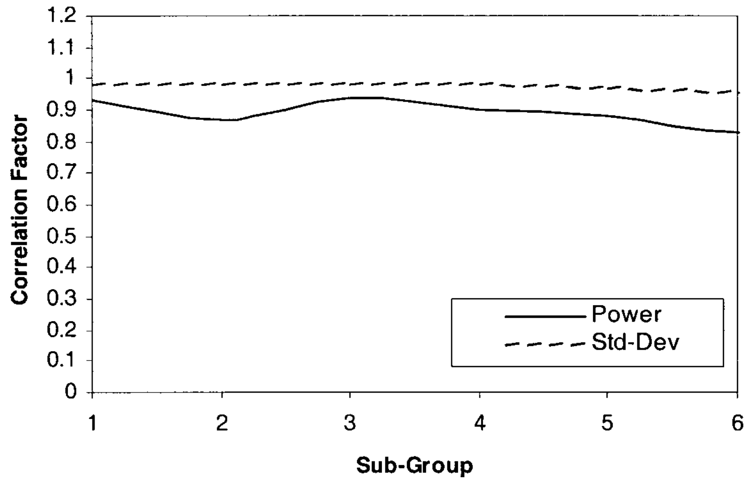



Figure 3.19 Correlation factors obtained using new cutter at 135 rpm.

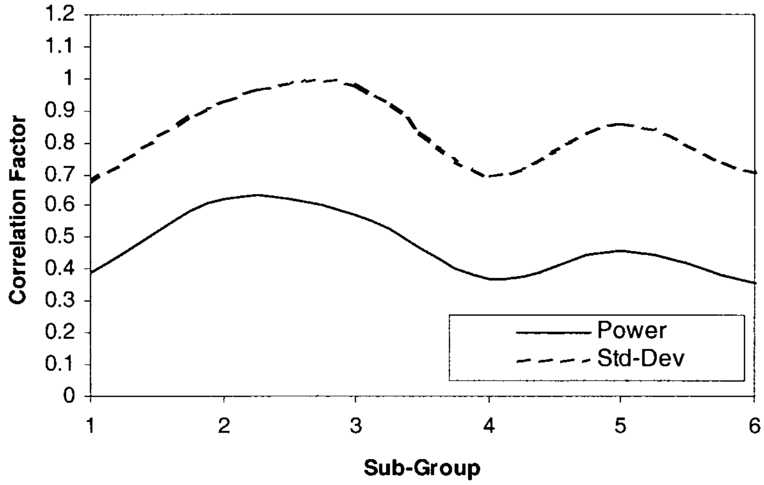



Figure 3.20 Correlation factors obtained using worn cutter at 135 rpm.

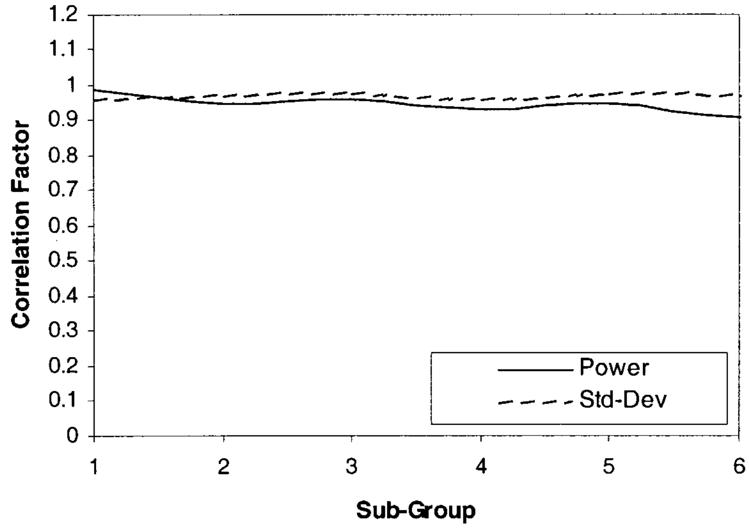



Figure 3.21 Correlation factors obtained using new cutter at 210 rpm.

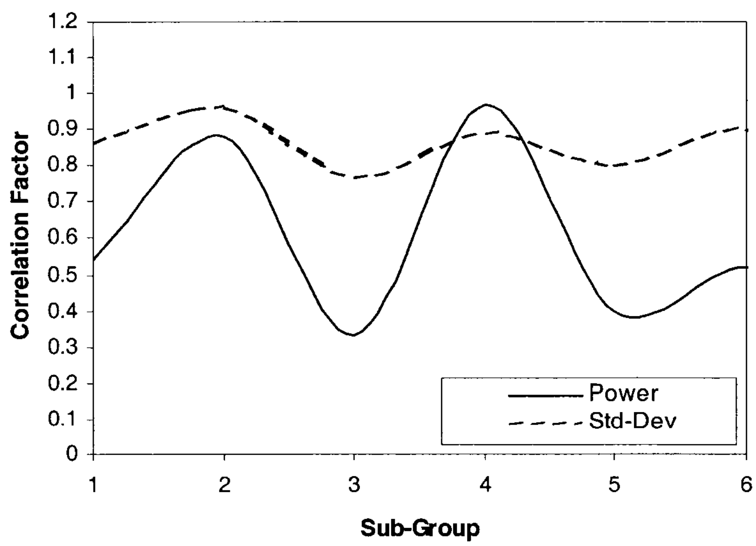



Figure 3.22 Correlation factors obtained using worn cutter at 210 rpm.

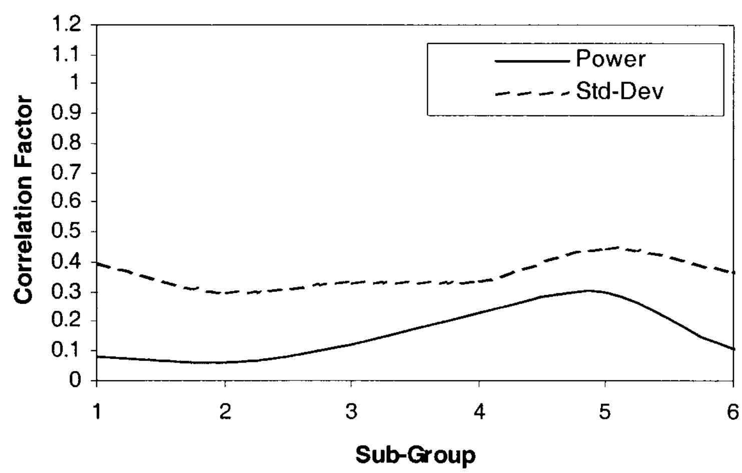



Figure 3.23 Correlation Factors obtained during tool breakage at 210 rpm.

### 3.2.2 Bearing Monitoring

As stated earlier, one of the objectives of this study is to develop a versatile approach for detecting both machinery and machining process anomalies. The above algorithm and condition indicators are therefore further tested using bearing data.

#### 3.2.2.1 Data Collection

The data used in this Section was obtained from the Case Western Reserve University Bearing Data Center. This data center provides normal and abnormal vibration data at different motor speeds. Data can be found in the following website (http://www.eecs.cwru.edu/laboratory/bearing/welcome overview.htm).

The motor bearings under consideration were seeded with faults using Electro-Discharge Machining (EDM). Faults ranging from 0.007" in diameter to 0.028" were introduced separately at the inner raceway, rolling element (i.e. ball) and outer raceway. The fault seeded at the outer race was placed at position equivalent to 6:00 o'clock time configuration. Faulted bearings were reinstalled into the test motor and vibration data were recorded for motor loads of 0 to 3 hp (at motor speeds of 1720 to 1797 rpm). The data used in this study, was collected at 12000 samples/sec for normal bearings, single point drive end (DE) and fan end (FE) defects. Tables 3.4, 3.5 and 3.6 describe the available data used in this study. In these tables, “X” indicates data available and “NA” means data not available.

Table 3.4 Normal baseline data.

| Case No. | Motor Load (hp) | Motor Speed (rpm) | Drive End Bearing | Fan End Bearing |
|---|---|---|---|---|
| 1 | 0 | 1797 | X | X |
| 2 | 1 | 1772 | X | X |
| 3 | 2 | 1750 | X | X |
| 4 | 3 | 1730 | X | X |

Table 3.5 Drive end bearing fault data.

| Case No. | Fault Diameter | Motor Load | Motor Speed | Inner Race | Ball | Outer Race |
|---|---|---|---|---|---|---|
| 1 | 0.007” | 0 | 1797 | X | X | X |
| 2 | 0.007” | 1 | 1772 | X | X | X |
| 3 | 0.007” | 2 | 1750 | X | X | X |
| 4 | 0.007” | 3 | 1730 | X | X | X |
| 5 | 0.014” | 0 | 1797 | X | X | X |
| 6 | 0.014” | 1 | 1772 | X | X | X |
| 7 | 0.014” | 2 | 1750 | X | X | X |
| 8 | 0.014” | 3 | 1730 | X | NA | X |
| 9 | 0.021” | 0 | 1797 | X | X | X |
| 10 | 0.021” | 1 | 1772 | X | X | X |
| 11 | 0.021” | 2 | 1750 | X | X | X |
| 12 | 0.021” | 3 | 1730 | X | X | X |
| 13 | 0.028” | 0 | 1797 | X | X | NA |
| 14 | 0.028” | 1 | 1772 | X | X | NA |
| 15 | 0.028” | 2 | 1750 | X | X | NA |
| 16 | 0.028” | 3 | 1730 | X | X | NA |

Table 3.6 Fan end bearing fault data.

| Case No. | Fault Diameter | Motor Load | Motor Speed | Inner Race | Ball | Outer Race |
|---|---|---|---|---|---|---|
| 1 | 0.007" | 0 | 1797 | X | X | X |
| 2 | 0.007" | 1 | 1772 | X | X | X |
| 3 | 0.007" | 2 | 1750 | X | X | X |
| 4 | 0.007" | 3 | 1730 | X | X | X |
| 5 | 0.014" | 0 | 1797 | X | X | X |
| 6 | 0.014" | 1 | 1772 | X | NA | NA |
| 7 | 0.014" | 2 | 1750 | NA | X | NA |
| 8 | 0.014" | 3 | 1730 | X | NA | NA |
| 9 | 0.021" | 0 | 1797 | X | X | X |
| 10 | 0.021" | 1 | 1772 | X | X | NA |
| 11 | 0.021" | 2 | 1750 | X | X | NA |
| 12 | 0.021" | 3 | 1730 | X | X | NA |
| 13 | 0.028" | 0 | 1797 | NA | NA | NA |
| 14 | 0.028" | 1 | 1772 | NA | NA | NA |
| 15 | 0.028" | 2 | 1750 | NA | NA | NA |
| 16 | 0.028" | 3 | 1730 | NA | NA | NA |

### 3.2.2.2 Frequency Regions of Interest

Using the sampling frequency pre-defined above and the power spectrum density function, the frequency spectrum was scanned to determine the regions of interest. By comparing the results obtained for normal and faulty runs, it has been found that the region of interest covers all the frequencies between 100 Hz and 5000 Hz. To illustrate this observation, a few typical examples of drive end bearing are shown below. The examples cover the 2 extremities of the load and speed ranges. One can note easily how the energy in Figures 3.26-29 is increasing throughout the pre-defined bandwidth relatively to Figures 3.24 and 3.25.

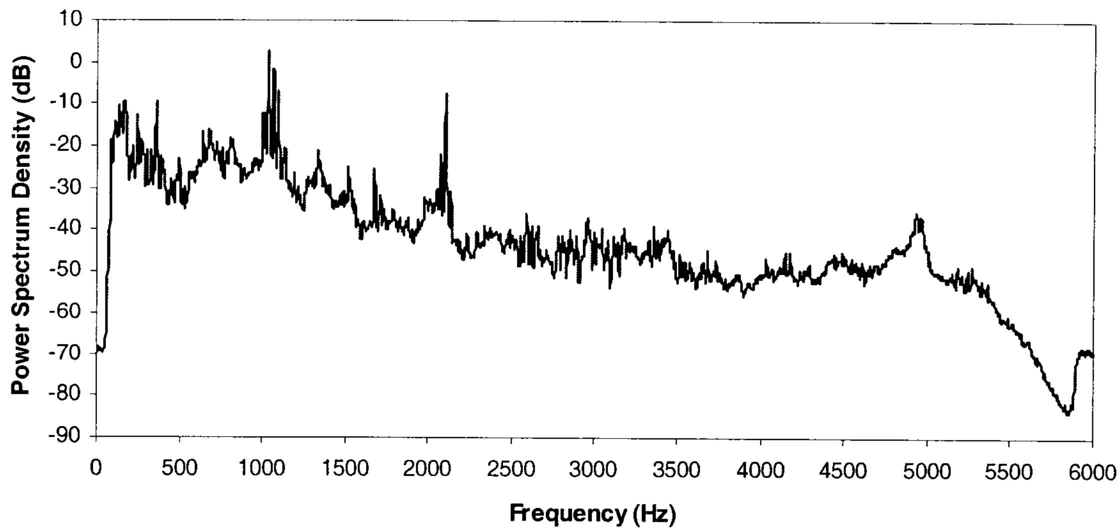



Figure 3.24 Power spectrum density for drive end bearing, normal condition, at 1797 rpm and 0 hp load.

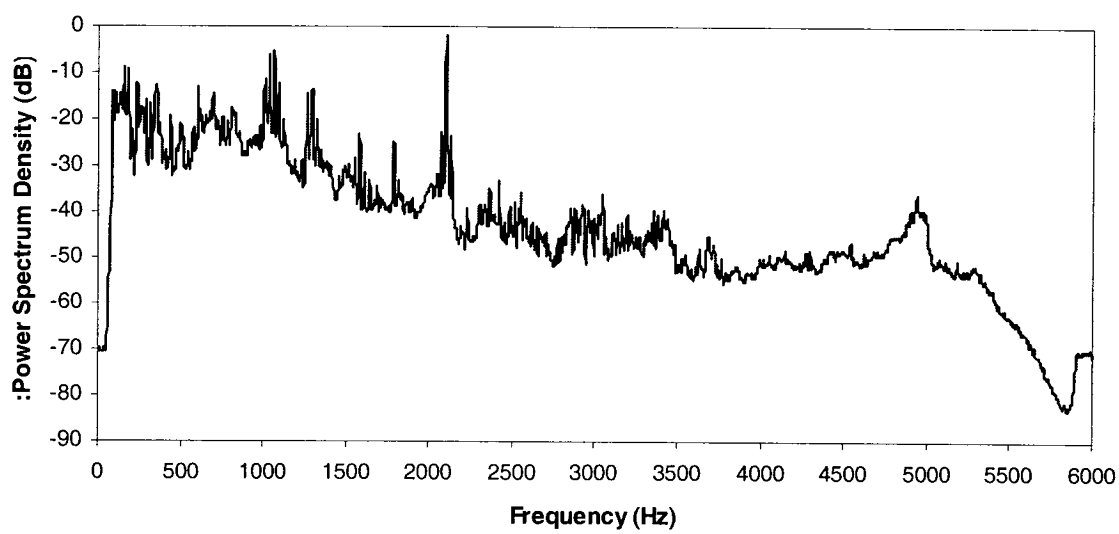



Figure 3.25 Power spectrum density for drive end bearing, normal condition, at 1730 rpm and 3 hp load.

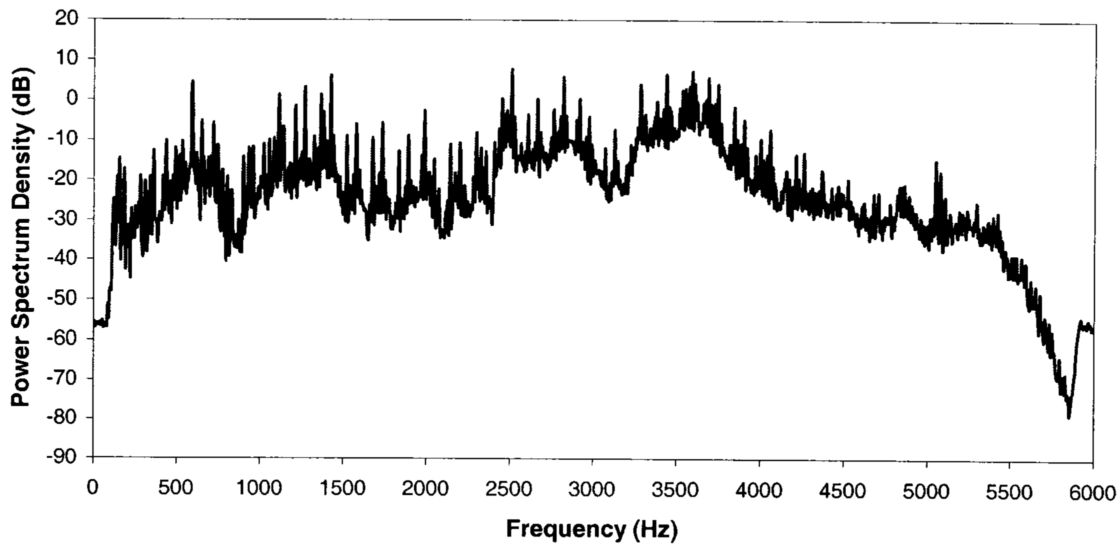



Figure 3.26 Power spectrum density for drive end bearing, 0.007" fault, at 1797 rpm and 0 hp load.

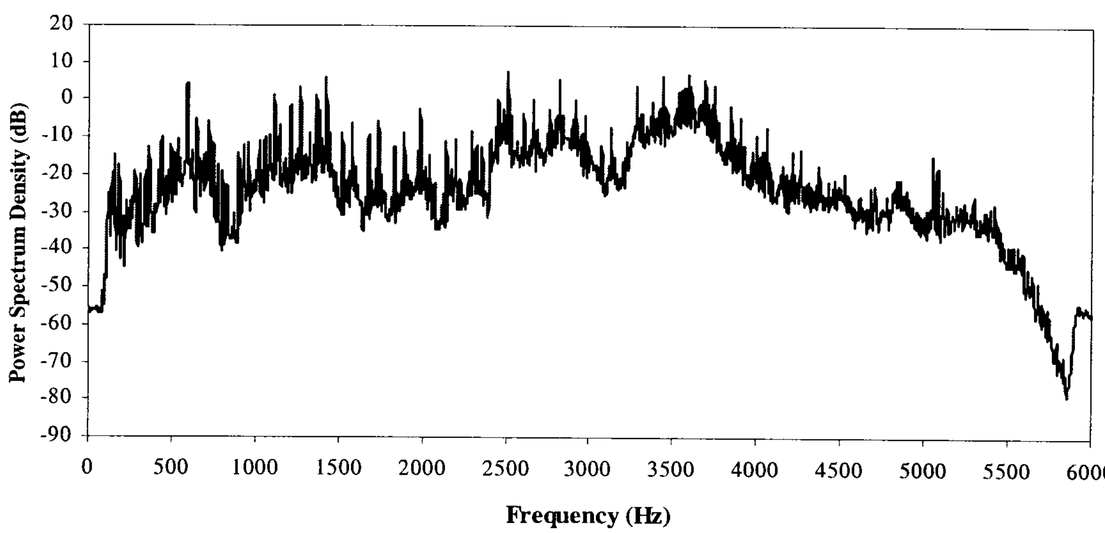



Figure 3.27 Power spectrum density for drive end bearing, 0.007" fault, at 1730 rpm and 3 hp load.

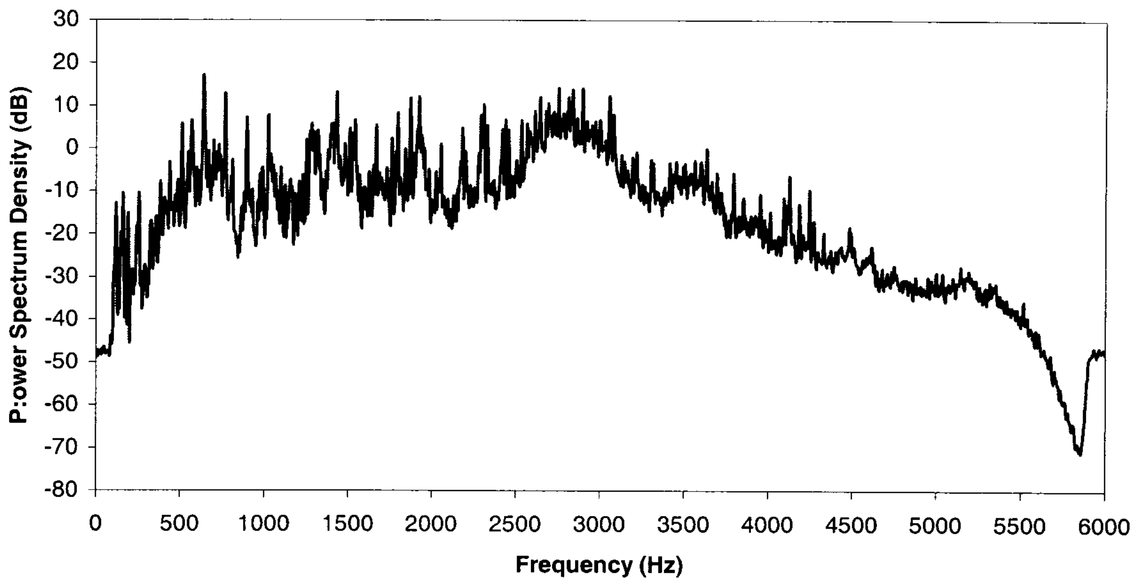



Figure 3.28 Power spectrum density for drive end bearing, 0.028" fault, at 1797 rpm and 0 hp load.

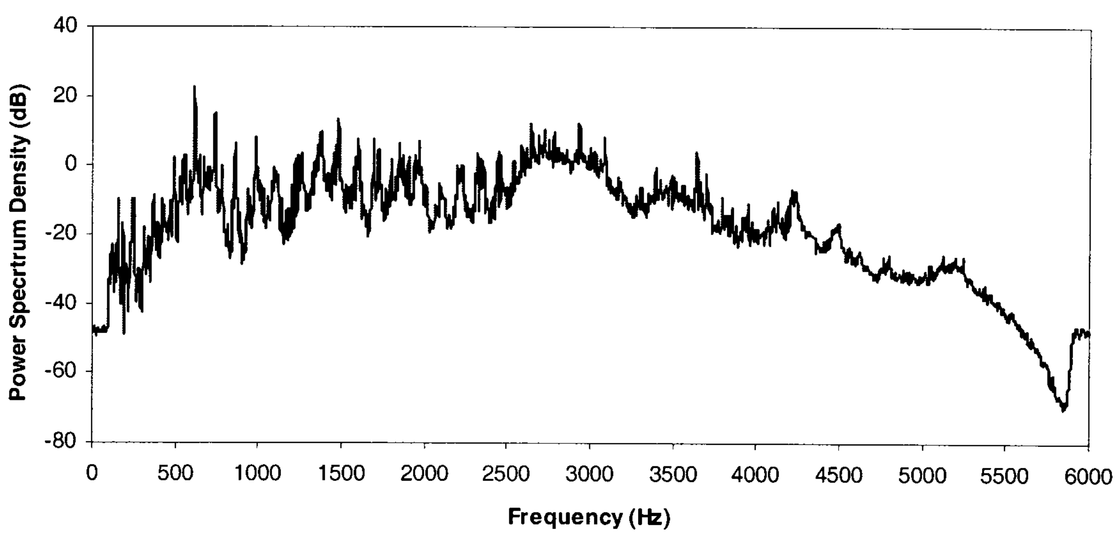



Figure 3.29 Power spectrum density for drive end bearing, 0.028" fault, at 1730 rpm and 3 hp load.

### 3.2.2.3 Implementation and Results

The analysis described in Section 3.1.3 and the findings of Section 3.2.2.2 were implemented using a *C* code MATLAB program named *Monitoring*, the same program used for tool condition monitoring (Appendix A). Based upon the sampling frequency (12000 Hz) and the nature of this monitoring problem (vibration), each mini-group was formed of 60 sample points (equivalent to 5 ms) to improve the detection sensitivity of transient anomalies (note that the minimum required time for one rotation is 33.3 ms corresponding to 1797 rpm). Each sub-group was composed of 10 mini-groups equivalent to a time window of 50 ms. As the condition indicators are used again in the fusion methods described in Chapter 4, a few typical sample results are represented below to illustrate the effectiveness of the proposed approach. Other examples are illustrated in Appendix B.

Figures 3.30 to 3.33 show typical representations of the condition indicators for normal condition. One can observe in Figures 3.31 and 3.33 some fluctuations in the correlation factors indicating the dynamic behaviour of the signal.

Figures 3.34, 3.36, 3.38 and 3.40 depict the detection of the transient anomaly produced upon contacting the fault at the inner race. The power condition indicator seems more sensitive to transient anomalies than the standard deviation condition indicator. However, the standard deviation correlation factor demonstrates a better capability than the power correlation factor, in terms of describing the continuous change in the signal. This observation can be illustrated in Figures 3.35, 3.37, 3.39 and 3.41.

As the fault size increases, the vibration amplitude increases accordingly. This can be reflected in the behaviour of the correlation factors. Comparing Figures 3.35 and 3.37 to Figures 3.39 and 3.41 respectively, one can note a significant decrease of the correlation factors' magnitudes. This decrease means that a dramatic change has occurred in the signal characteristics.

Similarly to tool condition monitoring as described in Section 3.2.1.3, one hundred and twenty monitoring intervals or sub-groups (equivalent to 6 seconds) of each of the collected data files were used for decision-making and for examining the effectiveness of the approach in the bearing context. Based on four (similar to the tool monitoring case this is a small arbitrary number selected for decision-making) successive sub-groups (equivalent to 200 ms), a failure classification was assigned upon finding at least two anomalies in two different sub-groups according to the following criteria:

- Power condition indicator above a threshold equal to 0.70, or
- Standard deviation condition indicator above a threshold equal to 0.65, or
- Power correlation factor below a threshold equal to 0.70, or
- Standard-deviation correlation factor below a threshold equal to 0.65

The thresholds were obtained based on assessing the data provided in normal conditions for a period of 5 seconds. As explained earlier, the minimum number of anomalies per condition indicator, required for failure classification, was used mainly to eliminate false alarms. As such, the detection efficiency was calculated for both DE and FE bearings by dividing in each case the number of successful detections by the total number of decisions. The detection was 100% in all cases. This result demonstrated the robustness of the proposed approach. It is important to note that the power and standard deviation condition indicators were very efficient in detecting the transient anomalies. A typical example of that is illustrated in Figures B.18, B.19, B.20, and B60-B63. In fact this is due to the nature of the problem: the faults seeded into the bearing produce a transient event. If this event is repeatable or continuous to affect the average value, the correlation factors will reflect this change as shown in Figures B.28-B.31, otherwise, the correlation factors will not perform the detection (e.g. Figure B.18).

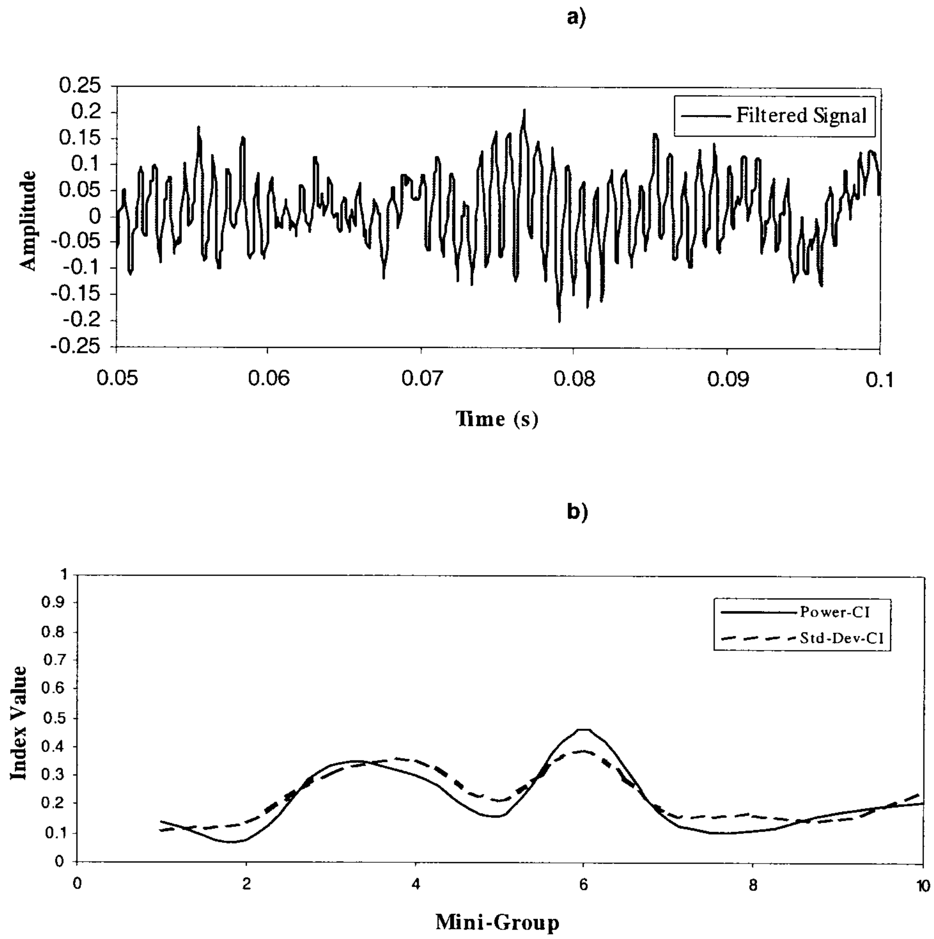



Figure 3.30 Drive end bearing, normal condition, at 1797 rpm and 0 hp load: a) time signal, b) signal represented by power condition indicator, and standard-deviation condition indicator.

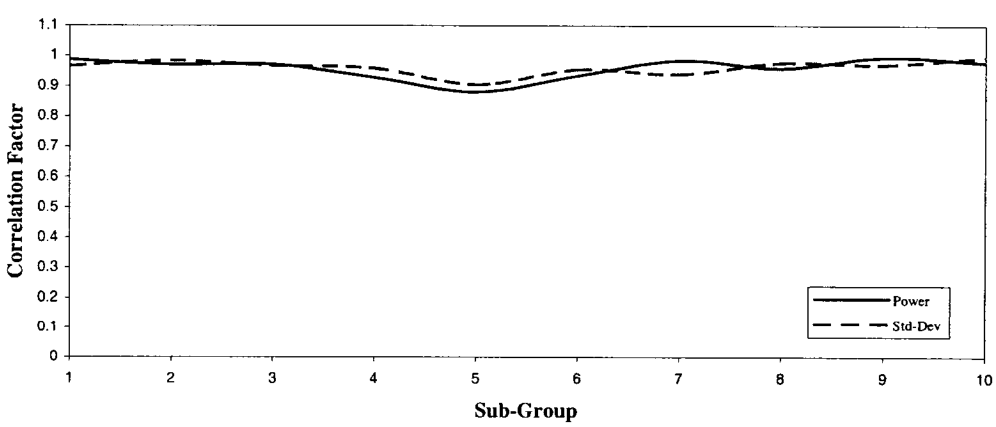



Figure 3.31 Drive end bearing, normal condition, at 1797 rpm and 0 hp load: power and standar-deviation correlation factors.

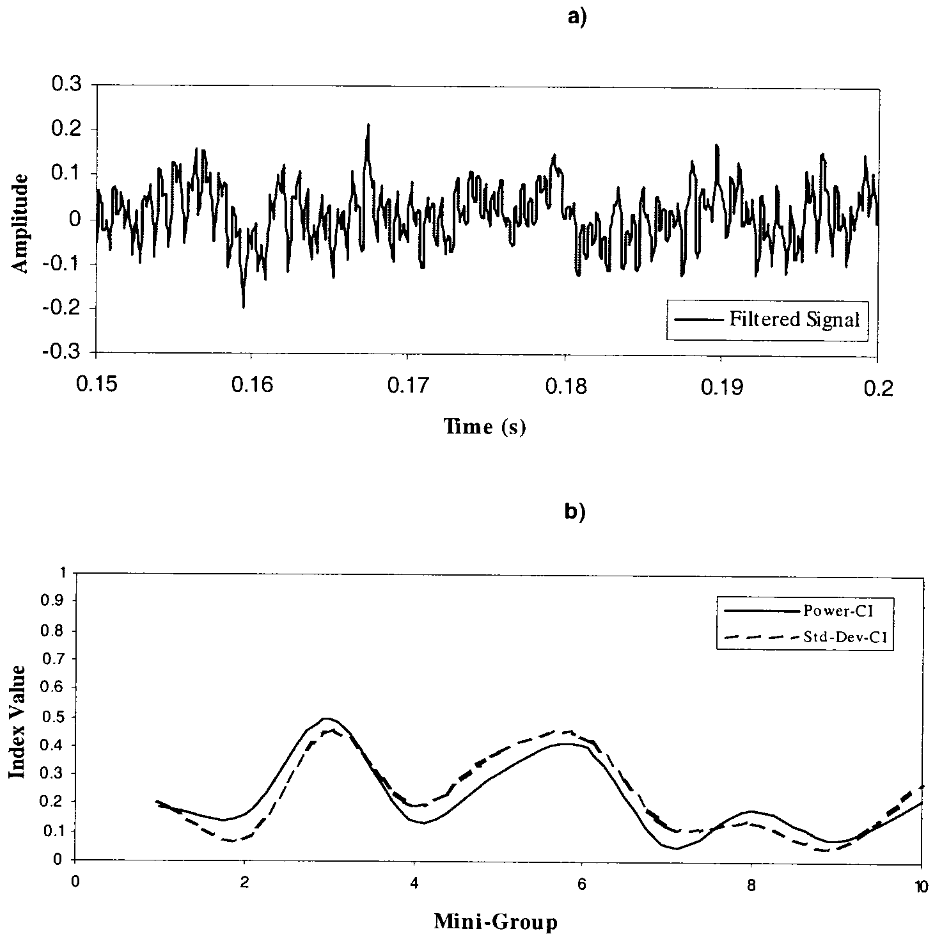



Figure 3.32 Drive end bearing, normal condition, at 1730 rpm and 3 hp load: a) time signal, b) signal represented by power condition indicator and standard-deviation condition indicator.

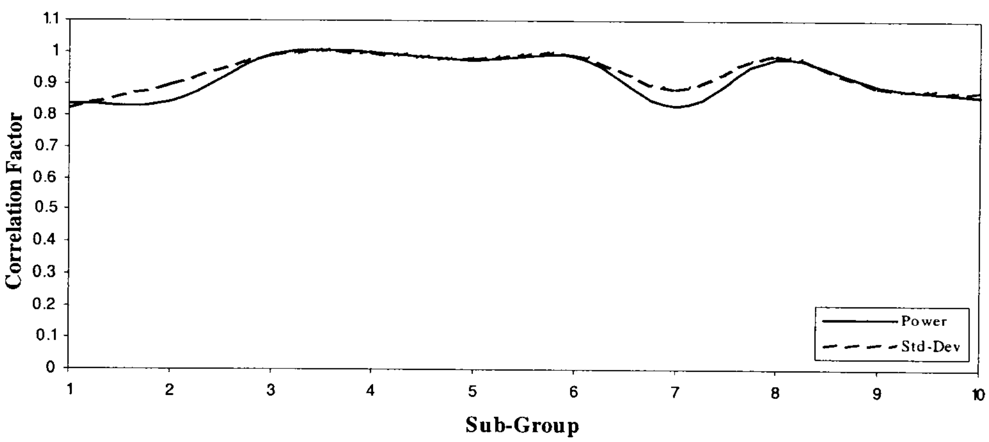
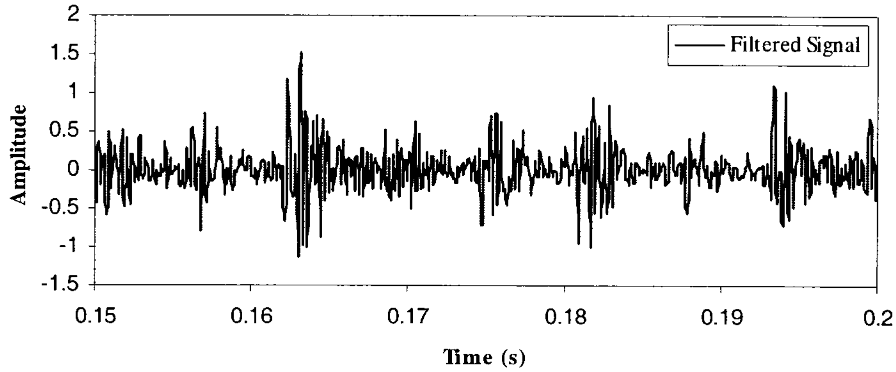



Figure 3.33 Drive end bearing, normal condition, at 1730 rpm and 3 hp load: power and standard deviation correlation factors.

a)




b)

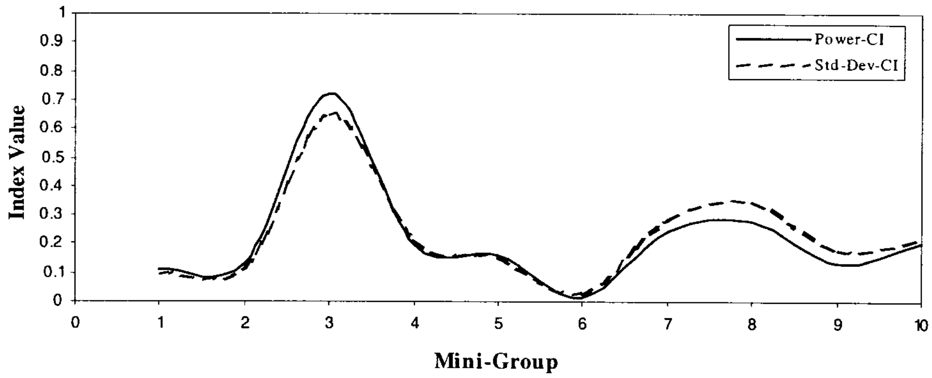



Figure 3.34 Drive end bearing, 0.007" fault at inner race, 1797 rpm and 0 hp load: a) time signal, b) signal represented by power condition indicator and standard-deviation condition indicator.

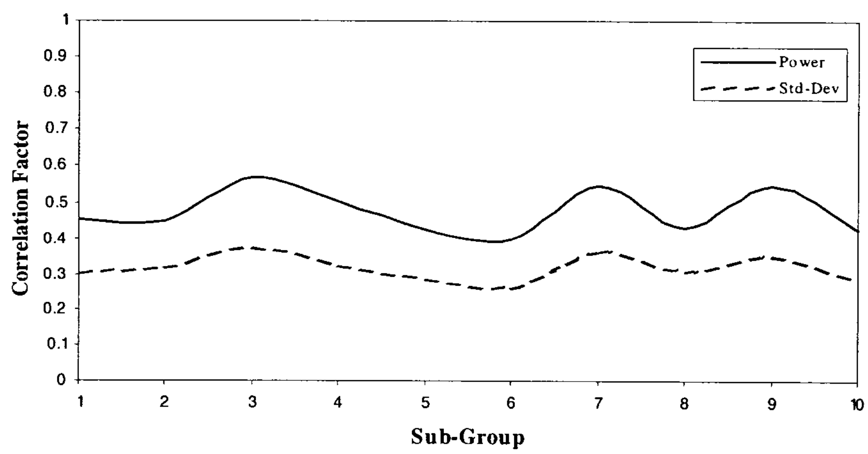
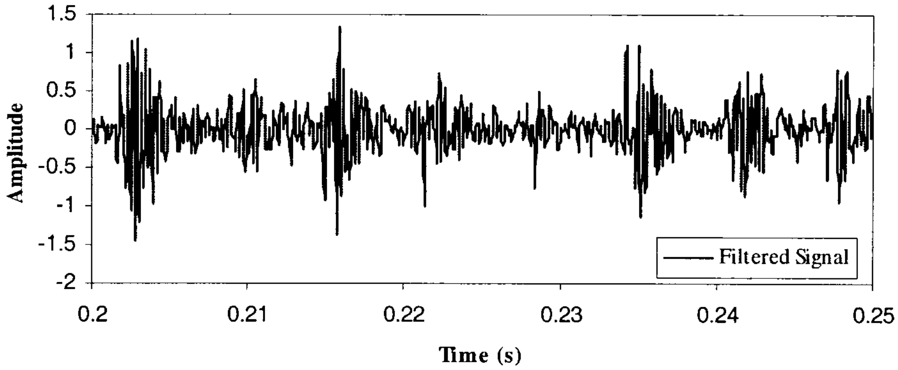



Figure 3.35 Drive end bearing, 0.007" fault at inner race, 1797 rpm and 0 hp load: power and standard deviation correlation factors.

a)




b)

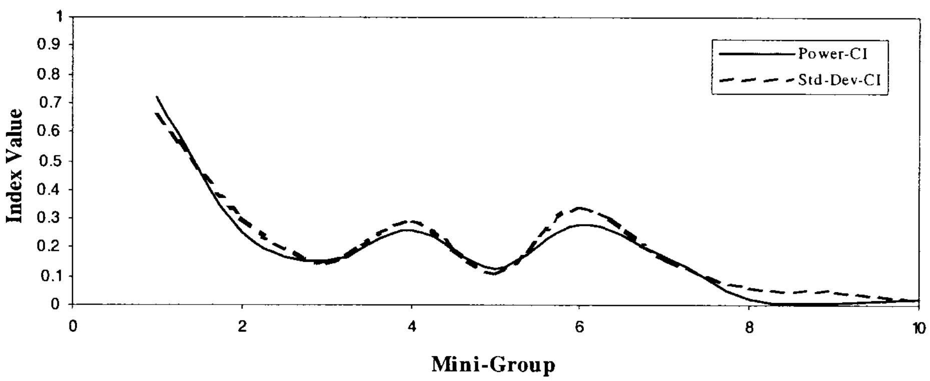



Figure 3.36 Drive end bearing, 0.007" fault at inner race, 1730 rpm and 3 hp load: a) time signal, b) signal represented by power condition indicator and standard-deviation condition indicator.

a)

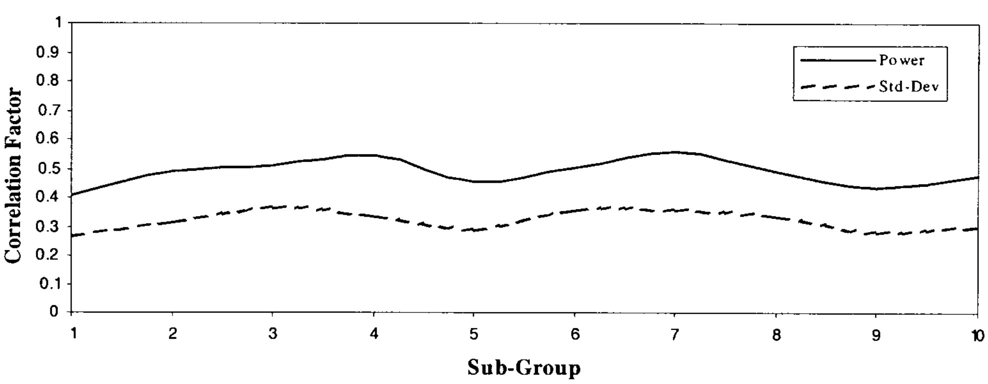
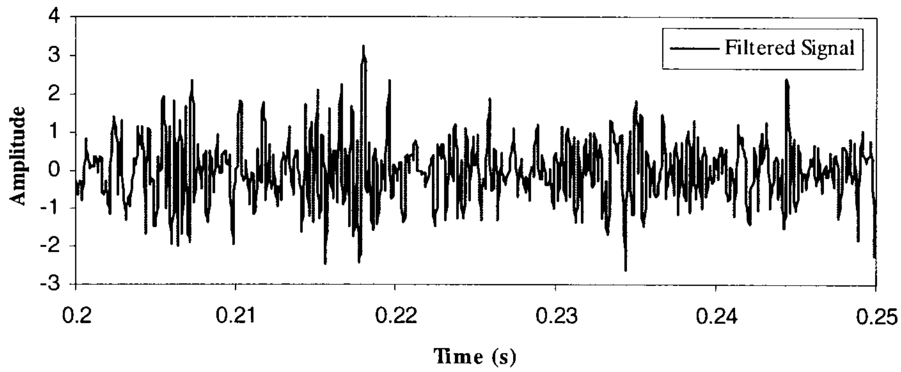



Figure 3.37 Drive end bearing, 0.007" fault at inner race, 1730 rpm and 3 hp load: power and standard deviation correlation factors.

a)




b)

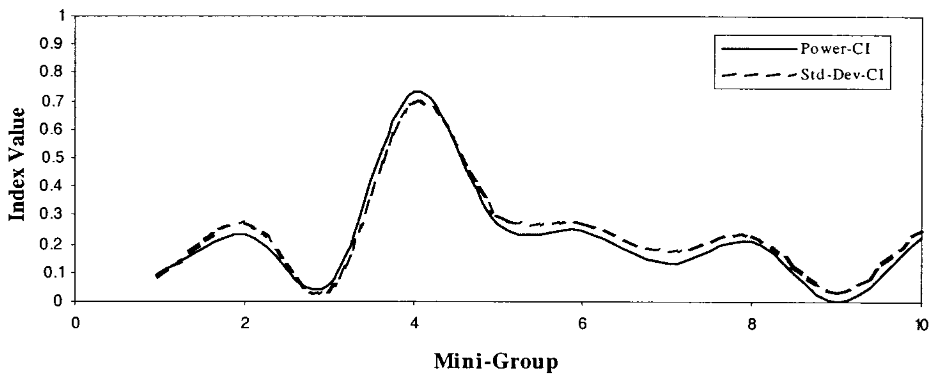



Figure 3.38 Drive end bearing, 0.028" fault at inner race, 1797 rpm and 0 hp load: a) time signal, b) signal represented by power condition indicator and standard-deviation condition indicator.

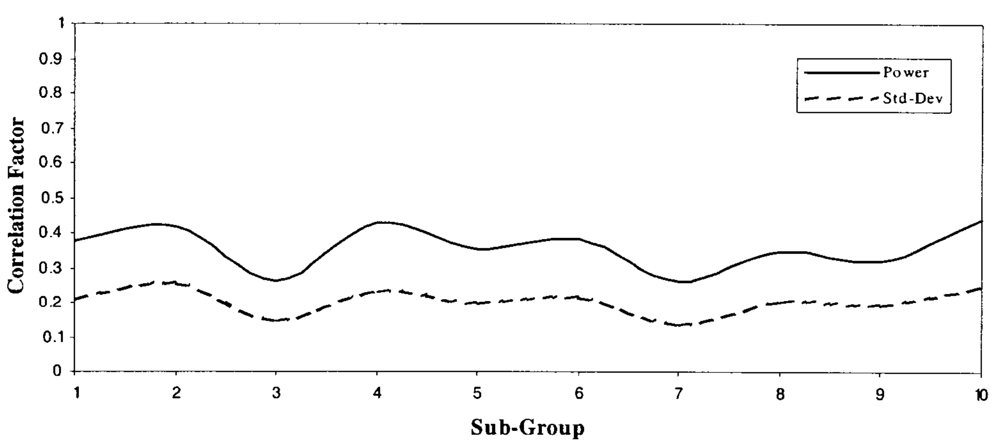
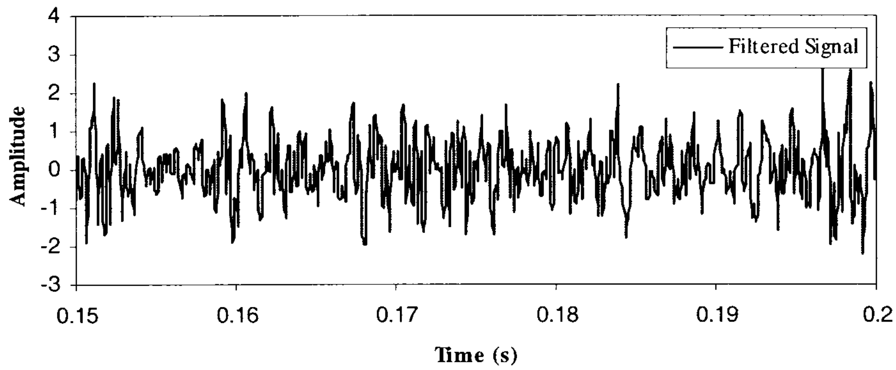



Figure 3.39 Drive end bearing, 0.028" fault at inner race, 1797 rpm and 0 hp load: power and standard deviation correlation factors.

a)




b)

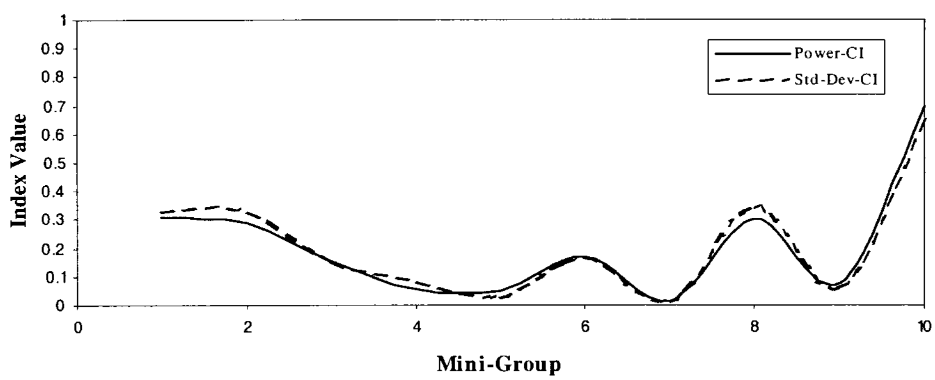



Figure 3.40 Drive end bearing, 0.028" fault at inner race, 1730rpm and 3 hp load: a) time signal, b) signal represented by power condition indicator and standard-deviation condition indicator.

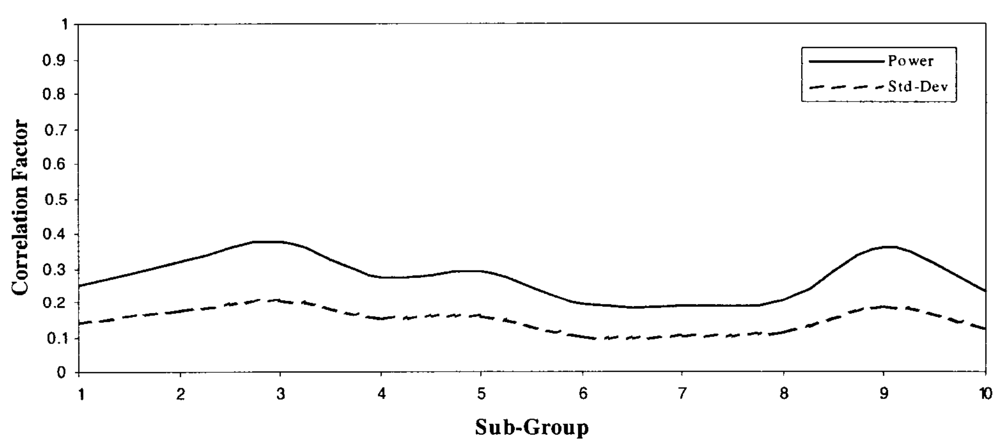



Figure 3.41 Drive end bearing, 0.028" fault at inner race, 1730 rpm and 3 hp load: power and standard deviation correlation factors.

## 3.3 Conclusions

In this chapter a new generic monitoring technique based on four condition indicators was presented. The proposed method was applied successfully to tool and bearing condition monitoring. The detection efficiency calculated in the experimental work suggests that these indices are accurate and robust enough to deal with the dynamic aspects of the signal and to be used for different faults (tool wear, tool breakage and bearing fault) in different applications (machining process and motor bearing). The power and standard deviation condition indicators have shown a high sensitivity to transient anomalies. However, the correlation factors have demonstrated a good capability to detect continuous anomalies.

# Chapter 4

# Fault Detection Using Fuzzy Index Fusion

The assessment of machine or process conditions becomes increasingly difficult due to the complexity of machine structure and operation dynamics. A machine can contain more than a few thousands parts (e.g. gas turbine in a power plant) and a cutter may be used to machine a variety of workpieces of different materials using different machining parameters. This poses a significant challenge to the designer of a monitoring system. This is why the sensor or condition indicators fusion theory becomes more attractive to researchers (Inasaki 1999, Choi *et al.* 1999) as different fault indicators derived from the same or different sensors can be combined to make the final maintenance decision. In such cases, the most important roles of the desired monitoring system are to detect problems, which occur during the process, to provide information to improve the process and to contribute to establishing the database where reference information are saved for future operations and maintenance.

Processes are mostly dynamic, non-linear and complex. As such, equipment conditions can vary significantly in a very short time and produce several profiles or signatures. These signatures carry in general a degree of uncertainty or "fuzzy" events (e.g. type of failure, severity, etc.). In Chapter 3 a condition monitoring technique was developed for machinery and process diagnosis. The detection method was based upon the proposed condition indicators. There was no real linkage between the condition indicators, therefore, the decision-making could be executed based on a single monitoring index. This process can be inefficient especially when dealing with complex systems. In this chapter, two methods of fuzzy fusion are proposed to incorporate the effects of all monitoring indices or condition indicators in the decision-making system. The first technique is a fuzzy-set based fusion and the second is a fuzzy-rule based method.

Depending on the complexity of the application, one of the methods can be selected. The first method can be used to integrate condition indicators, which have been assigned an equal weight and have well defined thresholds. The second technique is more robust and flexible as it provides a weighted solution based on all the condition indicators in the inference process and a set of predefined detection rules. It can be especially employed when the user has no means of determining the boundaries between normal and abnormal conditions. The advantage and disadvantages of each method are listed in the following sections. The resulting decision at the end of these two processes should confirm if the rotating equipment or machining process in question is healthy or not. Other information related to the classification and isolation of the fault will be determined in Chapter 5 through the use of the hidden Markov model approach. In addition, an index of deterioration is defined in the first method to monitor machinery or process performance degradation.

## 4.1 Fuzzy-set Based Fusion

The fuzzy-set based fusion method allows the integration of the condition indicators based on the fuzzy relation concept (Klir and Folger 1988). The technique is suitable for applications where thresholds for monitoring indices are clearly defined.

Fuzzy methods are used to deal with imprecision or uncertainty of facts by means of fuzzy degree (also called possibility function). More specifically, if $U$ is a universal set, and a fuzzy concept $A$ is a subset of $U$, then $A$ can be described as:

$$A = \{x, \mu_A(x)\} \tag{4.1}$$

where $x \in U$ is the value of $A$ and $\mu_A(x)$ is the membership function (Klir and Folger 1988). The membership function, $\mu_A(x)$ is a monotonous function:

$$0 \le \mu_A(x) \le 1 \tag{4.2}$$

It increases with respect to the increase of certainty of $A$. In addition, if $B$ is also a subset of $U$ and is more certain than $A$, then

$$\mu_A(x) > \mu_B(x) \tag{4.3}$$

The fuzzy-set based model that illustrates machinery or process conditions is shown in Figure 4.1. In this model, the monitoring indices such as power, standard deviation, power correlation and standard deviation indicators derived in Chapter 3, $y$, are the inputs and the machinery or process conditions (i.e. normal or abnormal), $c$, are the outputs of the model.

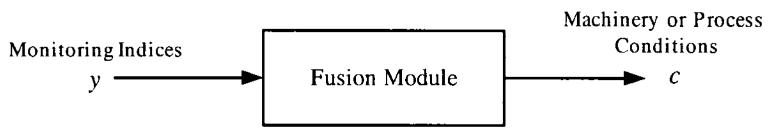



Figure 4.1 Fusion concept for equipment condition monitoring.

The relationship between the inputs and the outputs are described by a linear fuzzy equation:

$$p = Q \circ r \tag{4.4}$$

where $p$ represents the fuzzy degree of the machine or process conditions ($c$), $r$ represents the fuzzy degree of the monitoring indices ($y$), and $Q$ is the joint fuzzy relationship between the conditions and the monitoring indices, and the symbol "$\circ$"is the fuzzy operator.

Given a system that has $\hat{N}$ different conditions and $\hat{M}$ monitoring indices, and performing the calculation based on $\hat{U}$ samples of each index collected during one monitoring interval then, $p$ is an $\hat{N} \times \hat{U}$ matrix, $r$ is an $\hat{M} \times \hat{U}$ matrix and $Q$ is an $\hat{M} \times \hat{N} \times \hat{U}$ matrix.

The main steps of this method are illustrated in Figure 4.2. Similar to other decision-making methods (e.g. pattern recognition, neural network, etc.), this approach consists of 2 phases, i.e., learning and classification. Learning is aimed at building the fuzzy relationship matrix from available samples (i.e. determining $Q$ based on $p$ and $r$). In fact all the training or experimental work is used to support the learning phase and gives the theory a practical and realistic shape. On the other hand, the classification is to solve the linear fuzzy equation (i.e. determining $p$ when $Q$ and $r$ are known).

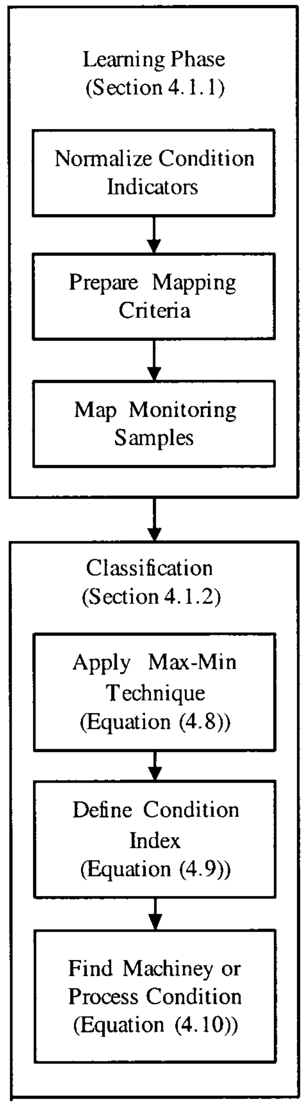



Figure 4.2 Main steps of fuzzy-set based fusion.

### 4.1.1 Learning Phase

Using $\hat{m}$ to label the monitoring indices ($\hat{m} = 1,2,...,\hat{M}$), $\hat{n}$ to label the machinery or process conditions ($\hat{n} = 1,2,...,\hat{N}$), and $\hat{u}$ to label the samples in a monitoring interval ($\hat{u} = 1,2,...,\hat{U}$), the relationship can be described as follows:

$$
\begin{bmatrix} p_{\hat{1}\hat{u}} \\ p_{\hat{2}\hat{u}} \\ \cdot \\ p_{\hat{n}\hat{u}} \\ \cdot \\ \cdot \\ p_{\hat{N}\hat{u}} \end{bmatrix} = \begin{bmatrix} q_{11\hat{u}} & q_{12\hat{u}} & \cdot & q_{1\hat{m}\hat{u}} & \cdot & \cdot & q_{1\hat{M}\hat{u}} \\ q_{21\hat{u}} & q_{22\hat{u}} & \cdot & q_{2\hat{m}\hat{u}} & \cdot & \cdot & q_{2\hat{M}\hat{u}} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ q_{\hat{n}1\hat{u}} & q_{\hat{n}2\hat{u}} & \cdot & q_{\hat{n}\hat{m}\hat{u}} & \cdot & \cdot & q_{\hat{n}\hat{M}\hat{u}} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ q_{\hat{N}1\hat{u}} & q_{\hat{N}2\hat{u}} & \cdot & q_{\hat{N}\hat{m}\hat{u}} & \cdot & \cdot & q_{\hat{N}\hat{M}\hat{u}} \end{bmatrix} \circ \begin{bmatrix} r_{1\hat{u}} \\ r_{2\hat{u}} \\ \cdot \\ r_{\hat{m}\hat{u}} \\ \cdot \\ \cdot \\ r_{\hat{M}\hat{u}} \end{bmatrix} \quad \forall \hat{u} \tag{4.5}
$$

where $p_{\hat{n}\hat{u}}$ represents the fuzzy degree of the $\hat{n}^{\text{th}}$ condition based on the $\hat{u}^{\text{th}}$ sample, $r_{\hat{m}\hat{u}}$ is the fuzzy degree of the $\hat{m}^{\text{th}}$ monitoring index for the $\hat{u}^{\text{th}}$ sample and $q_{\hat{n}\hat{m}\hat{u}}$ defines the relationship between the $\hat{m}^{\text{th}}$ monitoring index and the $\hat{n}^{\text{th}}$ condition as function of the $\hat{u}^{\text{th}}$ sample. Equation (4.5) can be also presented as follows:

$$
p_{\hat{n}\hat{u}} = q_{\hat{n}1\hat{u}} \otimes r_{1\hat{u}} \oplus q_{\hat{n}2\hat{u}} \otimes r_{2\hat{u}} \ldots \oplus q_{\hat{n}\hat{m}\hat{u}} \otimes r_{\hat{m}\hat{u}} \oplus . \quad . \; .q_{\hat{n}\hat{M}\hat{u}} \otimes r_{\hat{M}\hat{u}} \tag{4.6}
$$

where $\hat{n} = 1,....,\hat{N}$ , $\hat{u} = 1,....,\hat{U}$ , $\otimes$ is the fuzzy multiplication operator, $\oplus$ is the fuzzy addition operator. The fuzzy multiplication defines an intersection fuzzy relationship that results in the choice of the minimum value of the two multiplied parameters. On the other hand, the addition operator represents a union fuzzy relationship that yields at the maximum value of the two added parameters.

All the monitoring indices or the condition indicators are tested and verified through appropriate experiments. Index samples collected at different times and conditions lead to the composition of the membership matrices.

The first step is to ensure that indices are normalized (i.e. values ranging between [0,1]). The normalization is performed in this case using the same steps described in Chapter 3 as part of the condition indicators preparation.

The second step is to divide, based on this training activity, each monitoring index interval [0,1] into several segments each representing an equipment condition or state. For example, [0,0.8) represents a normal condition and [0.8,1] reflects an abnormal condition. These segments define the criteria that allow the mapping activity defined later. For example, these criteria can be the same ones defined in Chapter 3 for fault detection. It is also recommended to use the same progressive condition scheme, i.e., (normal $\rightarrow$ bad) is corresponding to the interval ($0 \rightarrow 1$) in the same order, to prevent conflicting conclusions in the classification phase.

The last step in this phase is to define the fuzzy relation matrix ($Q$). As the normalized monitoring indices or condition indicators can reflect the change in the equipment conditions as shown above, and assuming that samples are reliable, especially at the proximity of the segment boundaries, the samples are mapped into the matrix $Q$ based on the criteria defined earlier and using the fuzzy relation theory (Klir and Folger 1988) i.e.,

$$y_{\hat{m}\hat{u}} \Rightarrow q_{\hat{n}\hat{m}\hat{u}} \quad \forall \hat{n}, \hat{m}, and\, \hat{u} \tag{4.7}$$

where $y_{\hat{m}\hat{u}}$ represents the $\hat{u}^{\text{th}}$ sample for the $\hat{m}^{\text{th}}$ monitoring index.

This mapping activity will result in filling the columns of the matrix $Q$ with the value of the samples or with zeros if samples do not belong to the condition in question. Illustrative examples are shown in Section 4.1.4.

### 4.1.2 Classification

This phase is intended to determine the equipment condition based on the fuzzy relationship, defined in equation (4.6). The decision-making can be executed by solving this equation using the max-min technique described by Klir and Folger (1988).

$$p_{\hat{n}\hat{u}} = \max(\min(q_{\hat{n}\hat{m}\hat{u}}, r_{\hat{n}\hat{u}}) \quad \hat{m} = 1,2,\ldots,\hat{M}\ ) \quad \forall \hat{n}, \hat{u} \tag{4.8}$$

Averaging (moving average can be also used) $p_{\hat{n}\hat{m}}$ values with respect to the number of samples used in the monitoring interval leads to the definition of a *condition index* (*CI*) as follows:

$$CI(\hat{n}) = \frac{\sum_{\hat{u}=1}^{\hat{U}} p_{\hat{n}\hat{u}}}{\hat{U}} \quad \forall \hat{n} \tag{4.9}$$

The value of $CI(\hat{n})$ represents the degree of belongingness to condition $\hat{n}$. It is logical to judge the condition based on the $\hat{n}$ that is associated with the highest $CI$ value. Therefore, the estimated machinery or process condition is determined by finding such an $\hat{n}$ given by:

$$\hat{n}^* = \arg\max_{\hat{n}}(CI(\hat{n})) \tag{4.10}$$

### 4.1.3 Index of Deterioration

In the previous sections an index fusion method based on fuzzy-set concept was developed to determine the machinery or the process condition (normal or abnormal). These matrices were composed of the condition indicators defined in Chapter 3 such as power, standard deviation, power correlation factor and standard deviation correlation factor. As explained earlier, the first two condition indicators ($\hat{m} = 1,2$) are developed to detect transient anomalies, whereas the last two indicators ($\hat{m} = 3,4$) are effective in detecting continuous type of faults. In many processes or applications, the fault can be tolerated if the system is able to withstand malfunctions whilst still maintaining tolerable performance. This is in fact described by the notion mainly used in industries; *fault tolerance*. The main purposes of using *fault tolerance* are to maximize equipment availability, increase productivity and minimize costs. Although numerous technologies have been developed for monitoring and diagnosis, one still needs techniques for detecting the progress of various deteriorations at their early stages (Pouliezos and Stavrakakis 1994). In this section, an index of deterioration is developed as shown below.

As the monitoring interval in this study is based on one sub-group, and as one sample of each of the correlation factors is collected during this time-window, the index of deterioration (*IoD*) can be defined as follows:

$$IoD = \frac{\sum_{\hat{u}=1}^{NSG}\sum_{\hat{m}=3}^{4} y_{\hat{m}\hat{u}}}{2NSG} \tag{4.11}$$

where $NSG$ is the total number of sub-groups used in the calculation of the index of deterioration. This index is in fact the average value of all correlation factors within the specified time window. Hence, this index becomes more stable with a relatively big number of data samples.

### 4.1.4 Experimental Work

Using the data collected in Chapter 3, the condition index (as a moving average) is computed for tool and bearing condition assessment. As the same data is also used in the rule-based method (described below), the author selects several typical examples herein to just demonstrate the technique. A complete set of results are prepared as part of the second method.

#### 4.1.4.1 Tool Condition Monitoring

The development of the condition indicators described previously forms the most important part of the learning phase. The four normalized condition indicators are used to fill the fuzzy relation matrix defined above. As described in Section 4.1.1, to keep the same progressive condition scheme with respect to the interval [0,1] (i.e., the perfect normal condition corresponds to 0 and the worst abnormal condition corresponds to 1), the complement of the power and standard–deviation correlation factors (*CPCF* and *CSDCF* respectively) are used. As described earlier, the power and standard deviation condition indicators are usually utilized to detect transient anomalies within the sub-group interval. The correlation factors are monitored throughout a number of sub-groups to determine if the anomaly is repeatable or continuous. Hence, to keep the same time-scale and eliminate unnecessary computation steps, it is desirable to employ the complement of the correlation factors with the maximum values of power and standard deviation indicators (*MPCI* and *MSDCI* respectively) within the each sub-group interval.

As the main objective of this method is to only determine the tool condition, two different states are defined ($\hat{N}=2$): normal ($\hat{n}=1$) and abnormal ($\hat{n}=2$). Therefore, the samples of the monitoring indices are mapped into the membership matrices based on the thresholds defined in Section 3.2.1.3 and as per the following criteria:

(1) Abnormal if the *MPCI* is above a threshold equal to 0.8, or

(2) Abnormal if the *MSDCI* is above a threshold equal to 0.75, or

(3) Abnormal if any of the complement correlation factors is above a threshold equal to 0.3.

After the composition of the fuzzy relation matrix, equations (4.8)-(4.10) are used to determine the tool condition. In this calculation, the fuzzy degrees of the monitoring indices are assumed equal to one. This means that the accuracy of all measurements is assumed very high.

To illustrate the technique, consider the sharp tool used at the spindle speed of 135 rpm (refer to Figure 3.14 (1 sample shown) and Figure 3.19). The *MPCI*, *MSDCI*, *CPCF* and *CSDCF* for 4 successive sub-groups (i.e. $\hat{U} = 4$, the same number used in Chapter 3 for monitoring decision) are listed below.

$$[(MPCI)_{\hat{u}}] = [y_{1\hat{u}}] = [0.55, 0.61, 0.59, 0.59] \qquad \hat{u} = 1,\ldots,4$$

$$[(MSDCI)_{\hat{u}}] = [y_{2\hat{u}}] = [0.52, 0.59, 0.56, 0.57] \qquad \hat{u} = 1,\ldots,4$$

$$[(CPCF)_{\hat{u}}] = [y_{3\hat{u}}] = [0.07, 0.13, 0.06, 0.10] \qquad \hat{u} = 1,\ldots,4$$

$$[(CSDCF)_{\hat{u}}] = [y_{4\hat{u}}] = [0.02, 0.02, 0.02, 0.02] \qquad \hat{u} = 1,\ldots,4$$

Mapping these samples into fuzzy relation matrix using the criteria above and assuming that the fuzzy degrees for all monitoring indices are equal to one ($r_{\hat{m}\hat{u}} = 1$ i.e. measurements are precise and accurate), equation (4.5) leads to the following:

$$\begin{bmatrix} p_{11} \\ p_{21} \end{bmatrix} = \begin{bmatrix} q_{111} & q_{121} & q_{131} & q_{141} \\ q_{211} & q_{221} & q_{231} & q_{241} \end{bmatrix} \circ \begin{bmatrix} r_{11} \\ r_{21} \\ r_{31} \\ r_{41} \end{bmatrix} = \begin{bmatrix} 0.55 & 0.52 & 0.07 & 0.02 \\ 0.00 & 0.00 & 0.00 & 0.00 \end{bmatrix} \circ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} p_{12} \\ p_{22} \end{bmatrix} = \begin{bmatrix} q_{112} & q_{122} & q_{132} & q_{142} \\ q_{212} & q_{222} & q_{232} & q_{242} \end{bmatrix} \circ \begin{bmatrix} r_{12} \\ r_{22} \\ r_{32} \\ r_{42} \end{bmatrix} = \begin{bmatrix} 0.61 & 0.59 & 0.13 & 0.02 \\ 0.00 & 0.00 & 0.00 & 0.00 \end{bmatrix} \circ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} p_{13} \\ p_{23} \end{bmatrix} = \begin{bmatrix} q_{113} & q_{123} & q_{133} & q_{143} \\ q_{213} & q_{223} & q_{233} & q_{243} \end{bmatrix} \circ \begin{bmatrix} r_{13} \\ r_{23} \\ r_{33} \\ r_{43} \end{bmatrix} = \begin{bmatrix} 0.59 & 0.56 & 0.06 & 0.10 \\ 0.00 & 0.00 & 0.00 & 0.00 \end{bmatrix} \circ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} p_{14} \\ p_{24} \end{bmatrix} = \begin{bmatrix} q_{114} & q_{124} & q_{134} & q_{144} \\ q_{214} & q_{224} & q_{324} & q_{424} \end{bmatrix} \circ \begin{bmatrix} r_{14} \\ r_{24} \\ r_{34} \\ r_{44} \end{bmatrix} = \begin{bmatrix} 0.59 & 0.57 & 0.10 & 0.02 \\ 0.00 & 0.00 & 0.00 & 0.00 \end{bmatrix} \circ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Using the max-min technique described in equation (4.8), the above equations become:

$$\begin{bmatrix} p_{11} \\ p_{21} \end{bmatrix} = \begin{bmatrix} 0.55 \\ 0.00 \end{bmatrix} \qquad \begin{bmatrix} p_{12} \\ p_{22} \end{bmatrix} = \begin{bmatrix} 0.61 \\ 0.00 \end{bmatrix} \qquad \begin{bmatrix} p_{13} \\ p_{23} \end{bmatrix} = \begin{bmatrix} 0.59 \\ 0.00 \end{bmatrix} \qquad \begin{bmatrix} p_{14} \\ p_{24} \end{bmatrix} = \begin{bmatrix} 0.59 \\ 0.00 \end{bmatrix}$$

Consequently, the *condition index* can be calculated using equation (4.9), i.e.:

$$CI(1) = \frac{\sum_{\hat{u}=1}^{4} p_{1\hat{u}}}{4} = \frac{(0.55 + 0.61 + 0.59 + 0.59)}{4} = 0.585$$

$$CI(2) = \frac{\sum_{\hat{u}=1}^{4} p_{2\hat{u}}}{4} = \frac{(0.00 + 0.00 + 0.00 + 0.00)}{4} = 0.00$$

Therefore, based on the highest value of $CI$ (equation (4.10)), $\hat{n}^* = 1$ meaning that the condition is normal. This can be interpreted as that the degree of belongingness to condition 1 (normal) is 0.585, much higher than that to condition 2 (abnormal) and hence the condition should be classified as normal.

In the following, the $CI$ values for both normal and abnormal are computed as moving averages and then plotted in function of time. The condition should be classified as the one corresponding to the higher $CI$ value. Figures 4.3 and 4.4 illustrate the variation of the condition index for normal case at two different speeds. One can notice a few sudden increases in the values of $CI$ for the abnormal case. This is due to transient false detections obtained during some sub-groups. However, the effect of these false alarms is neutralized using the averaging technique. In fact, the $CI$ for the normal case maintains the highest level in both plots. Therefore the condition can be classified as normal in both cases (Figures 4.3 and 4.4), which is good agreement with the actual tool condition. By examining the margin between the normal and abnormal cases, one can feel the importance of selecting the appropriate condition indicator thresholds for such applications. This margin can be reduced significantly and in a relative short time as shown in both figures.

For worn tool the *CI* for abnormal condition seems stable as depicted in Figures 4.5 and 4.6. For these cases the margin between normal and abnormal conditions is wide indicating the effectiveness of the monitoring indices in detecting anomalies.

For tool breakage, it is interesting to observe in Figure 4.7 how the *CI* for both conditions fluctuates rapidly indicating the occurrence of a severe incident. It is also important to note how after breakage (at 6.5 sec in Figure 4.7), the *CI* for abnormal condition maintains the highest level. This is mainly driven by the two complement of correlation factors.

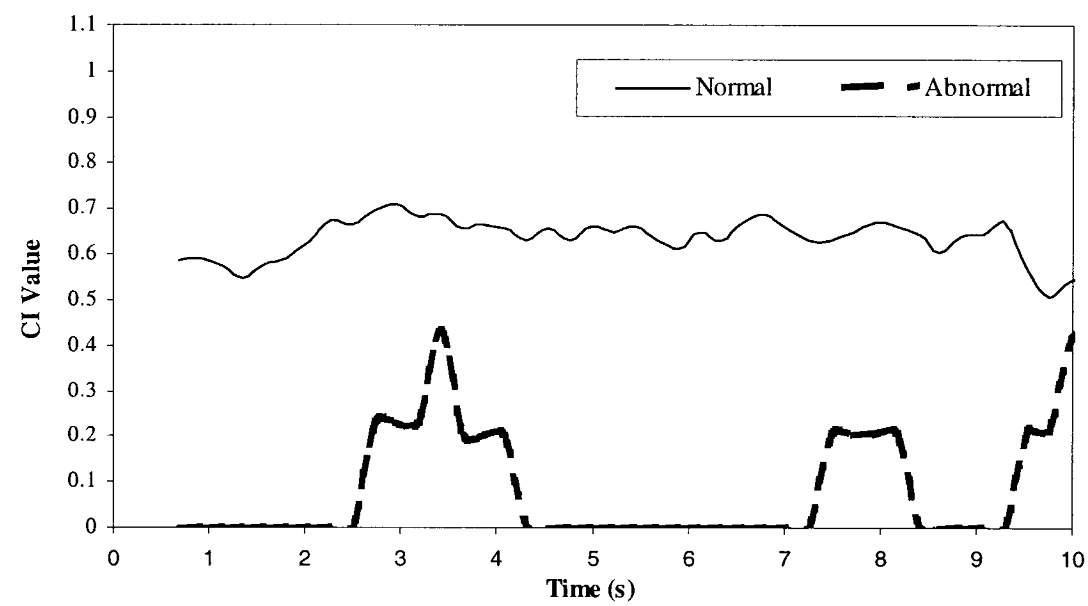



Figure 4.3 Condition index as a moving average for sharp tool at 135 rpm.

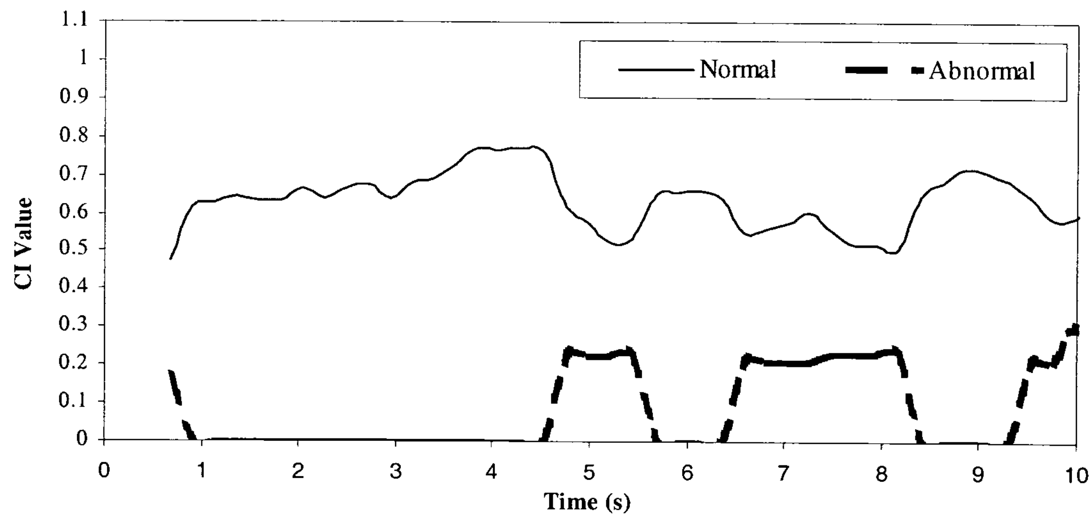



Figure 4.4 Condition index as a moving average for sharp tool at 210 rpm.

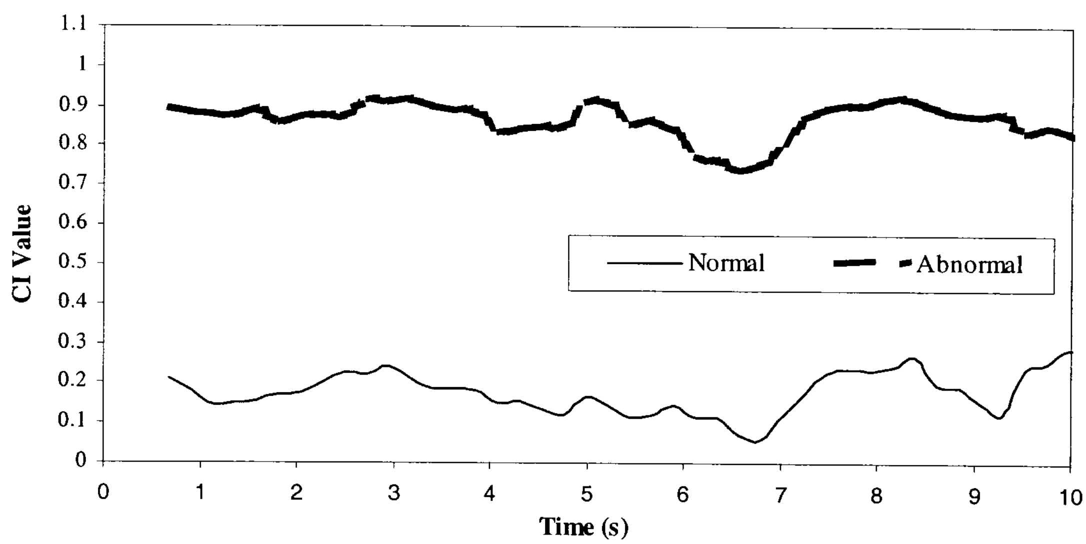



Figure 4.5 Condition index as a moving average for worn tool at 135 rpm.

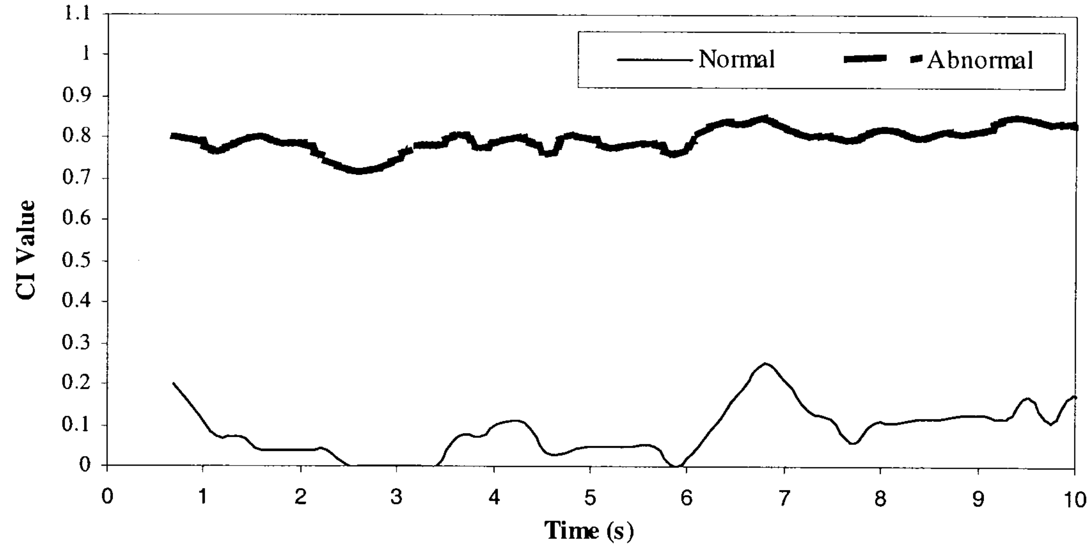



Figure 4.6 Condition index as a moving average for worn tool at 210 rpm.

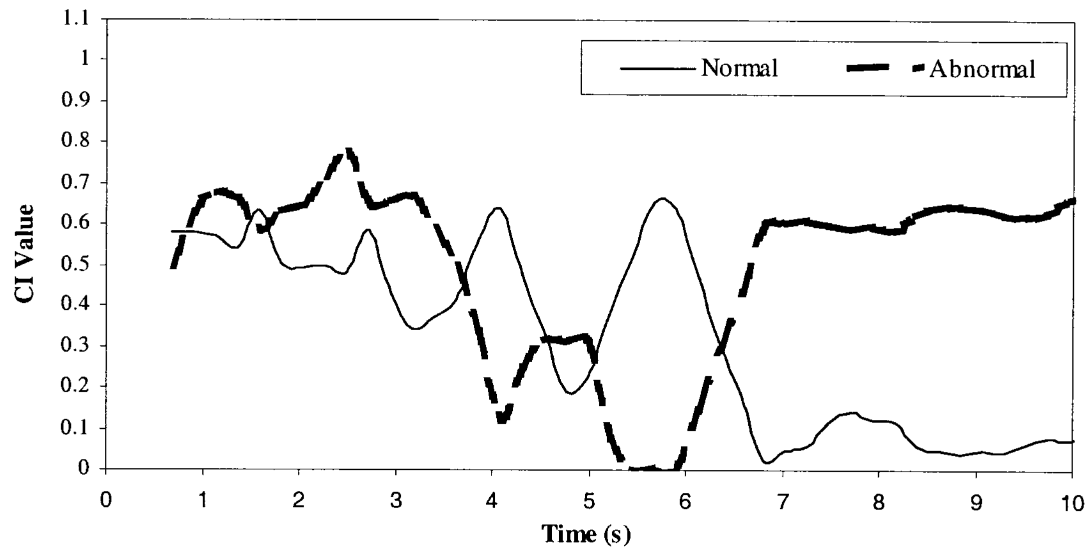



Figure 4.7 Condition index as a moving average for tool breakage.

### 4.1.4.2 Bearing Condition Monitoring

The same methodology described in the tool condition monitoring is utilized herein. The normalized condition indicators are used to determine the fuzzy relation matrix based on the data collected in Chapter 3. Again, two states ($\hat{N}=2$) are defined for this bearing study, i.e., normal ($\hat{n}=1$) and abnormal ($\hat{n}=2$).

Based on the technique described above, the data samples are mapped into the fuzzy relation matrix using the thresholds defined in Chapter 3 and the following criteria:

(1) Abnormal if *MPCI* is above a threshold equal to 0.70, or

(2) Abnormal if *MSDCI* is above a threshold equal to 0.65, or

(3) Abnormal if *CPCF* is above a threshold equal to 0.30, or

(4) Abnormal if *CSDCF* is above a threshold equal to 0.35.

The *CI* as a moving average value is computed for several typical cases for the DE and FE bearings and plotted below for validation purpose. These examples cover the two extremities of the speed range, load range and fault size.

Figures 4.8 and 4.9 describe the *CI* for DE in normal condition at two different speeds and loads. One can notice numerous increases in the value of the abnormal *CI*. These pulses represent the false detections obtained at sub-group level. However, their effect is reduced using the averaging technique. The same observation can be made for Figures 4.10 and 4.11 which display the FE normal bearing conditions. The number of pulses in Figure 4.11 seems bigger than the ones in the other cases but the normal *CI* maintains the highest level as shown below.

In Figures 4.12 and 4.13 the *CI* for abnormal case is higher than that of "normal" indicating as such the presence of a fault. The margin between the normal and abnormal conditions is not wide as illustrated in these figures. This is due to the fact that some of the condition indicators are not sensitive enough to the presence of the light fault with a small diameter (0.007"). This scenario changes with the 0.014" fault diameter. The margin between the two types of *CI* increases significantly as depicted in Figures 4.14 and 4.15. This can be explained in this case by the high sensitivity of all condition indicators with respect to this fault size. It is interesting also to note that the *CI* in Figure 4.14 is showing some repeatable patterns at a period of 0.5 seconds. This observation will be discussed further in the second fusion method. Using a 0.021" fault size, the margin between the two *CI* is narrowed again as per Figures 4.16 and 4.17. The abnormal *CI* is driven in this case by the complement of the correlation factors only. This means that the fault produces a continuous anomaly signal. As the fault size increases to 0.028", the *CI* for abnormal case increases from 0.7 to approximately 0.8 as illustrated in Figures 4.18-4.19. This increase is caused mainly by the complement of the correlation factors. At the same time, the margin between the two *CI* becomes wider comparing with the previous case. Looking at the level of abnormal *CI* in the above examples, one can note that this index does not show in a consistent way the condition degradation of the bearing. The index varies up and down with the increase of the fault size.

By examining the FE bearing examples, a few important observations can be highlighted. The margin between the two *CI* for 0.007" fault is much wider in Figures 4.20 and 4.21 than in the ones obtained for the DE bearing (Figures 4.12 and 4.13). In fact, both *CI* in these figures show a repeatable pattern with a period of 0.5 seconds. It is interesting to note how the normal and abnormal *CI* values are varying in an opposite way, so when the first increases the second decreases and vice versa. The same type of signatures is shown in Figures 4.22 and 4.23 when the fault size increases to 0.014". However, the abnormal *CI* level does not reflect this increase in the fault size. Along this line, another type of periodic pattern is presented in Figures 4.24 and 4.25 when the fault size grows to 0.021". The periodic waves are oscillating in this case at 1 Hz. Similar observations are made and explained in the second fusion approach.

In addition, the index of deterioration enhances the monitoring and diagnosis system and provides a useful tool for the maintenance department to evaluate the performance of the equipment in function of the fault severity.

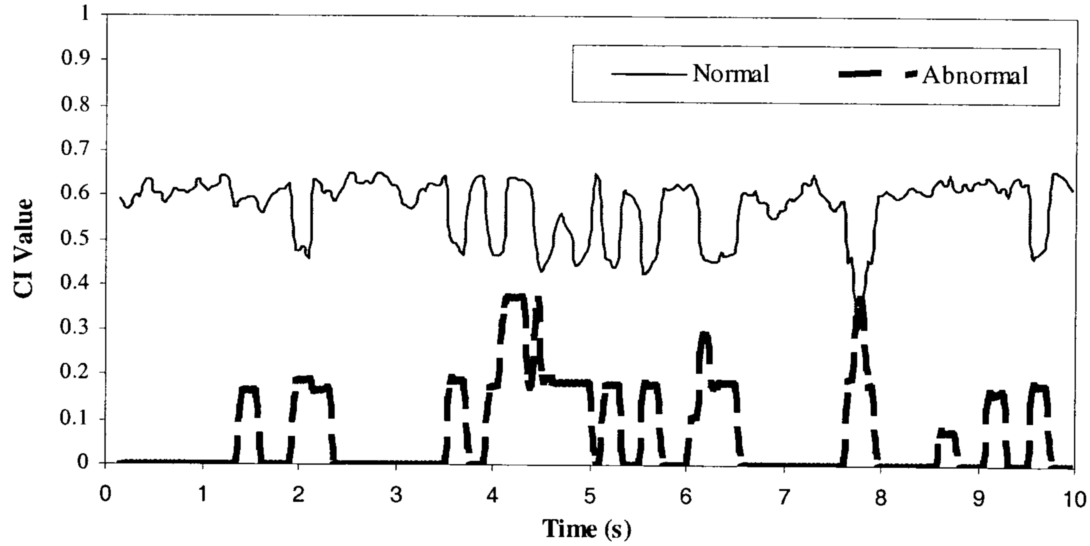



Figure 4.8 Condition index as a moving average for drive end bearing, normal condition at 1797 rpm and 0 hp motor load.

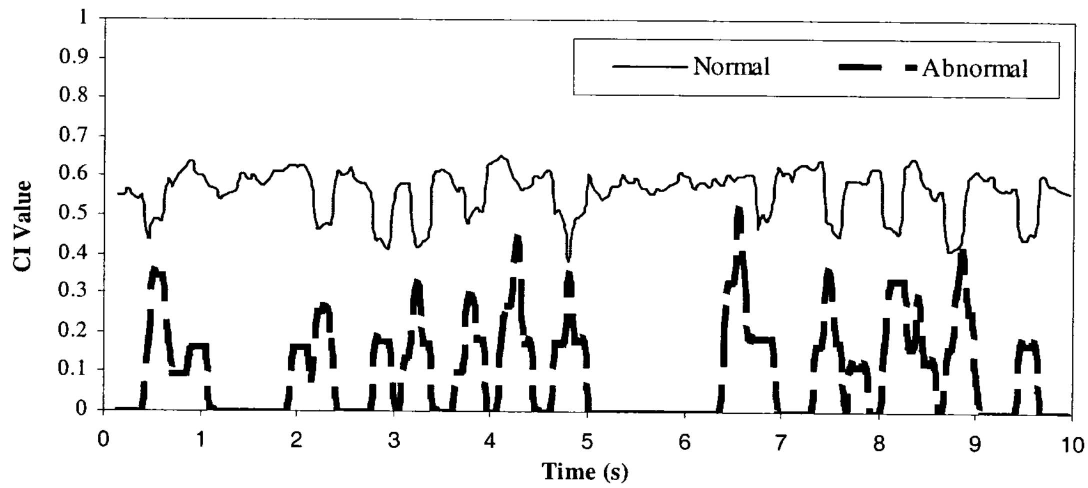



Figure 4.9 Condition index as a moving average for drive end bearing, normal condition at 1730 rpm and 3 hp motor load.

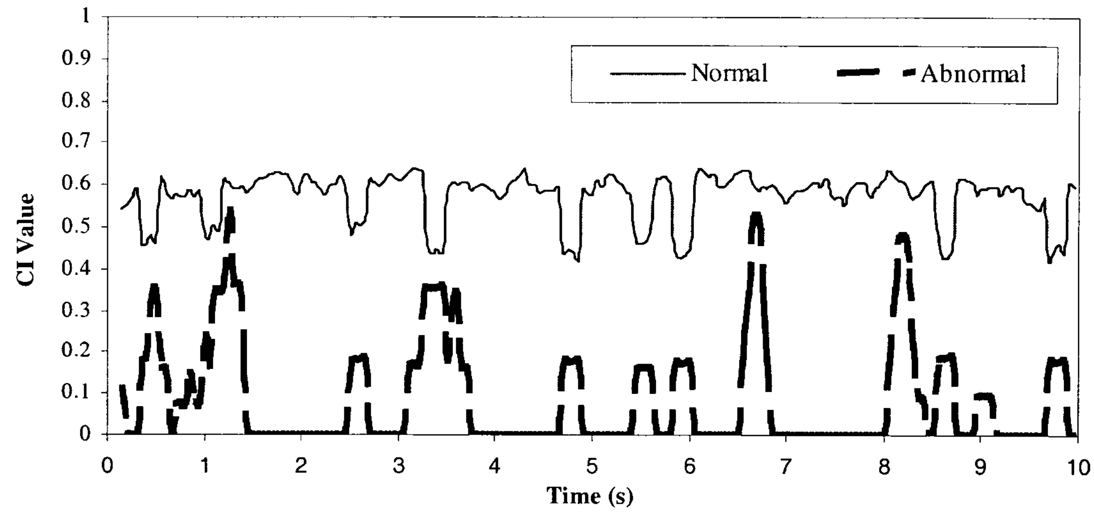



Figure 4.10 Condition index as a moving average for fan end bearing, normal condition at 1797 rpm and 0 hp motor load.

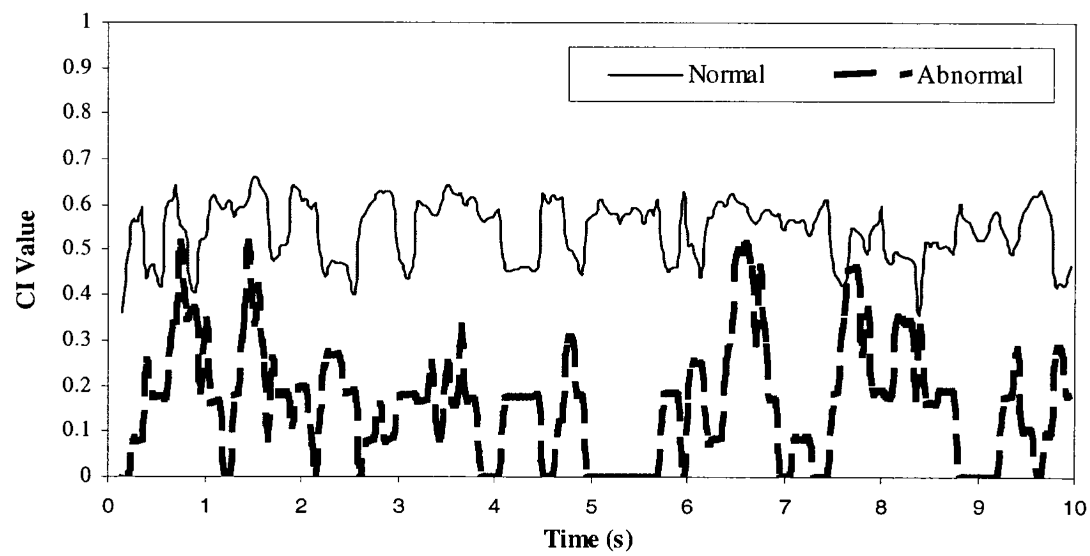



Figure 4.11 Condition index as a moving average for fan end bearing, normal condition at 1730 rpm and 3 hp motor load.

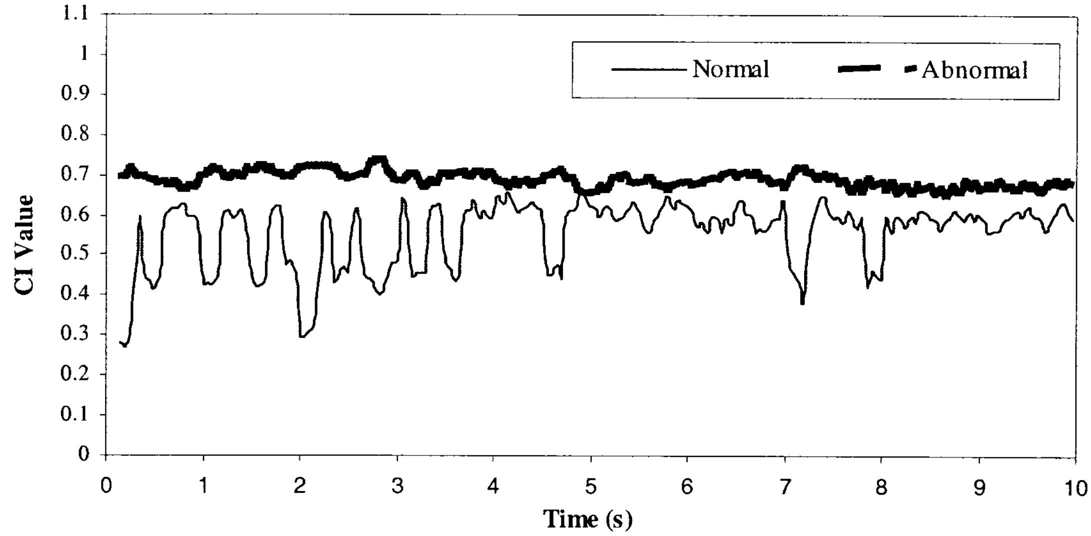



Figure 4.12 Condition index as a moving average for drive end bearing, 0.007" fault diameter at inner race, 1797 rpm and 0 hp motor load.

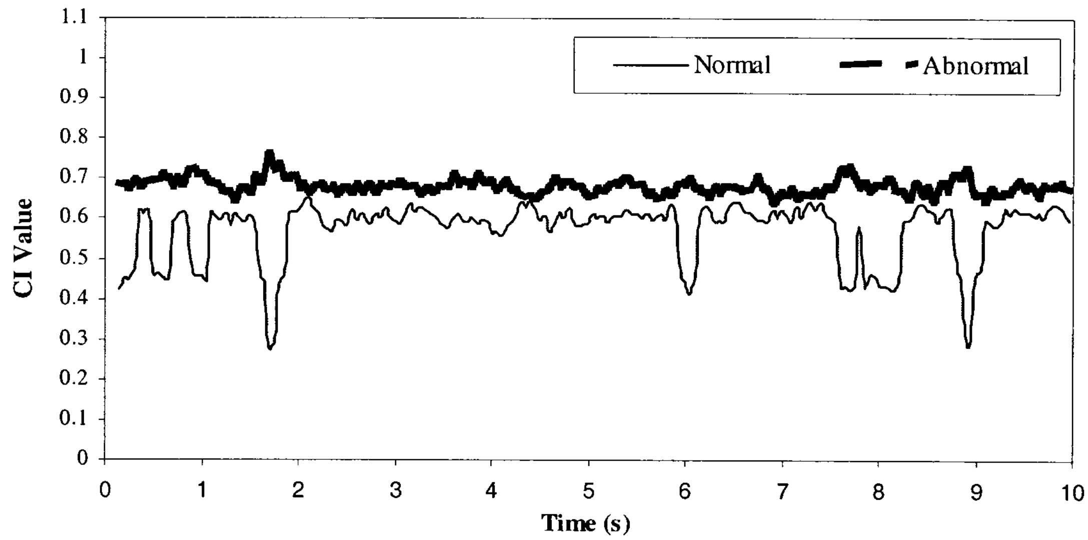



Figure 4.13 Condition index as a moving average for drive end bearing, 0.007" fault diameter at inner race, 1730 rpm and 3 hp motor load.

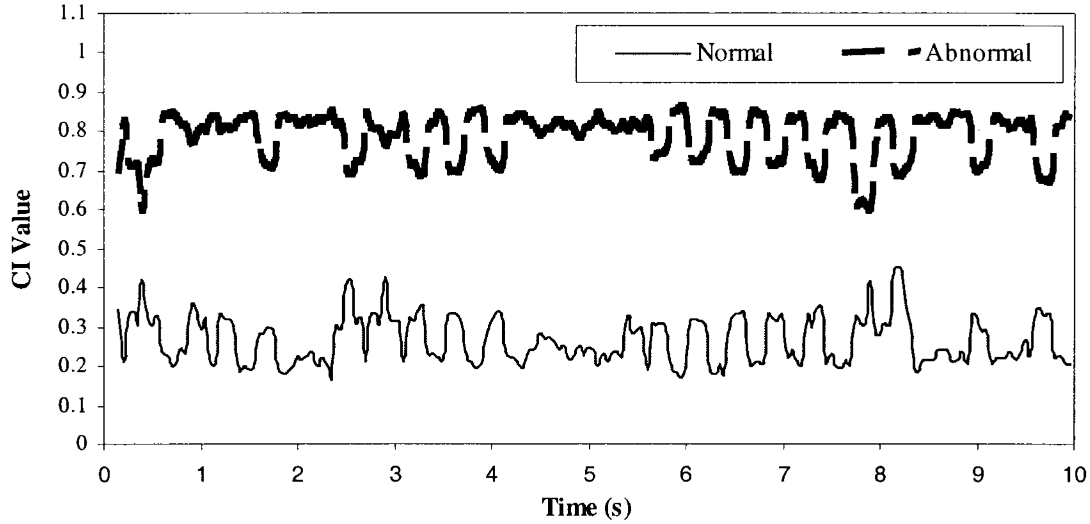



Figure 4.14 Condition index as a moving average for drive end bearing, 0.014" fault diameter at inner race, 1797 rpm and 0 hp motor load.

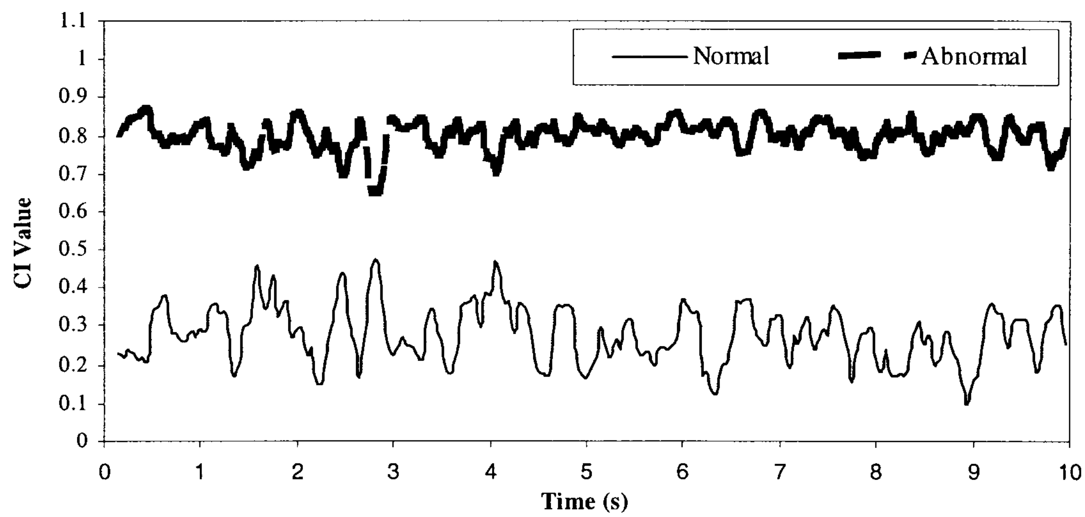



Figure 4.15 Condition index as a moving average for drive end bearing, 0.014" fault diameter at inner race, 1730 rpm and 3 hp motor load.

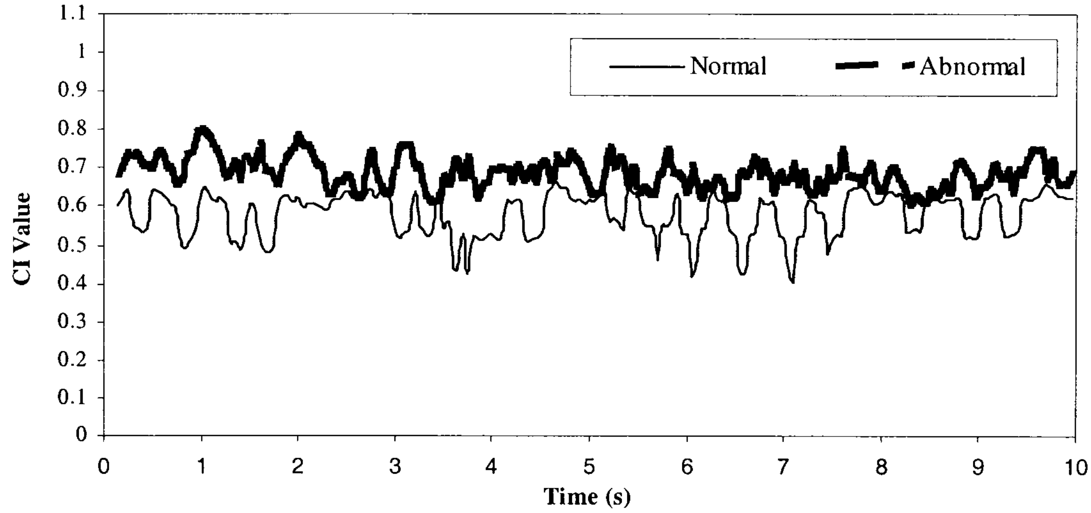



Figure 4.16 Condition index as a moving average for drive end bearing, 0.021" fault diameter at inner race, 1797 rpm and 0 hp motor load.

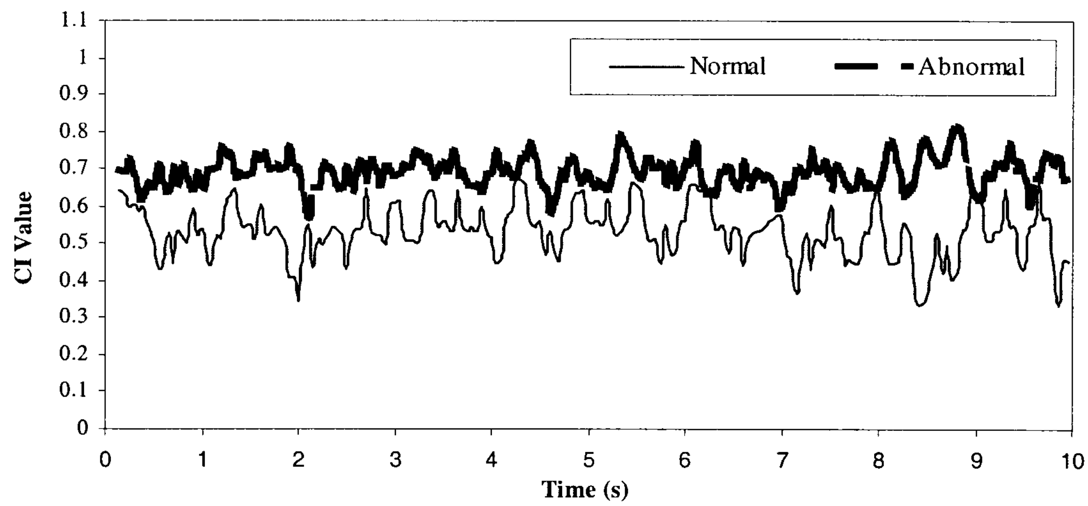



Figure 4.17 Condition index as a moving average for drive end bearing, 0.021" fault diameter at inner race, 1730 rpm and 3 hp motor load.

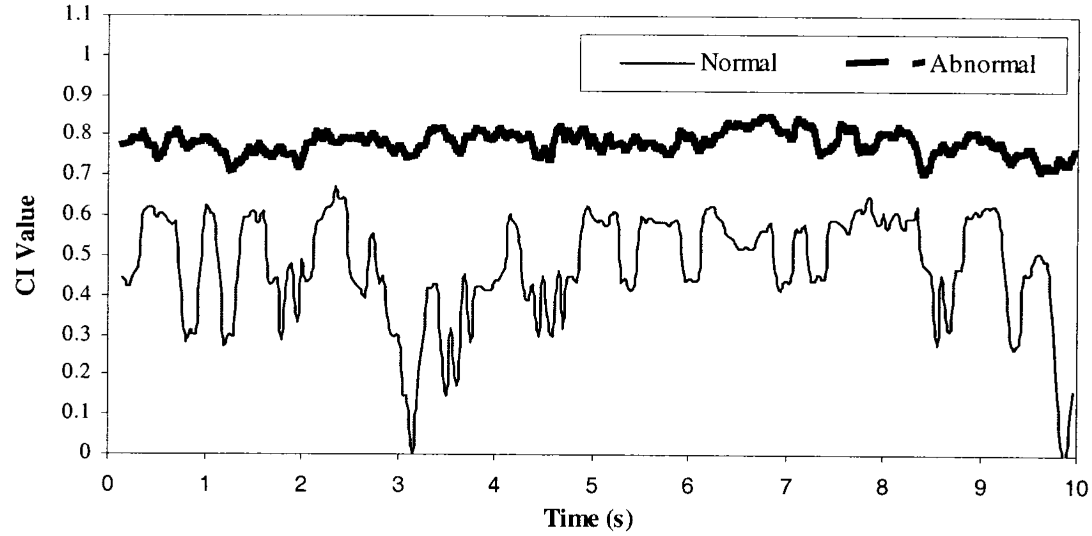



Figure 4.18 Condition index as a moving average for drive end bearing, 0.028" fault diameter at inner race, 1797 rpm and 0 hp motor load.

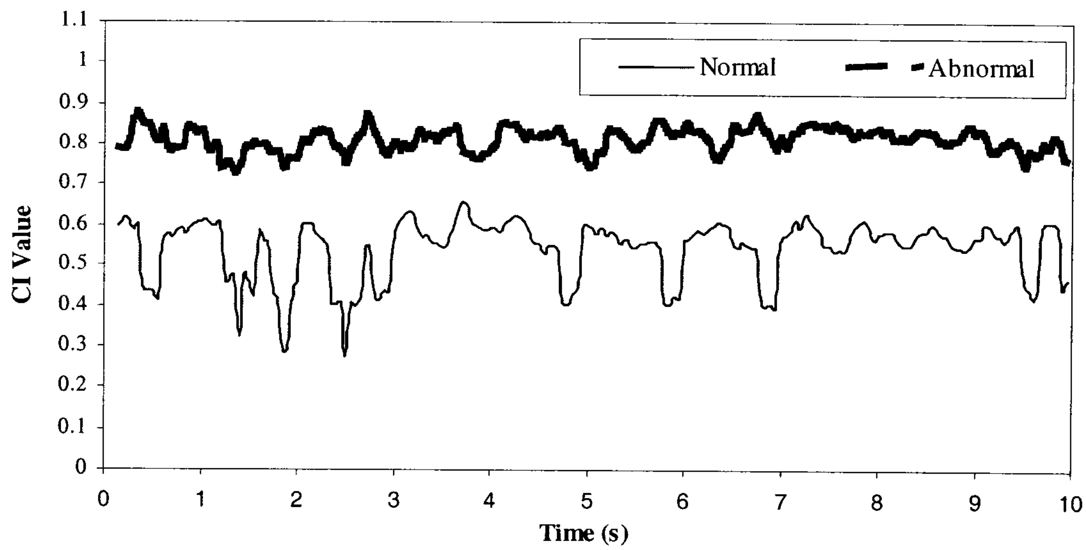



Figure 4.19 Condition index as a moving average for drive end bearing, 0.028" fault diameter at inner race, 1730 rpm and 3 hp motor load.

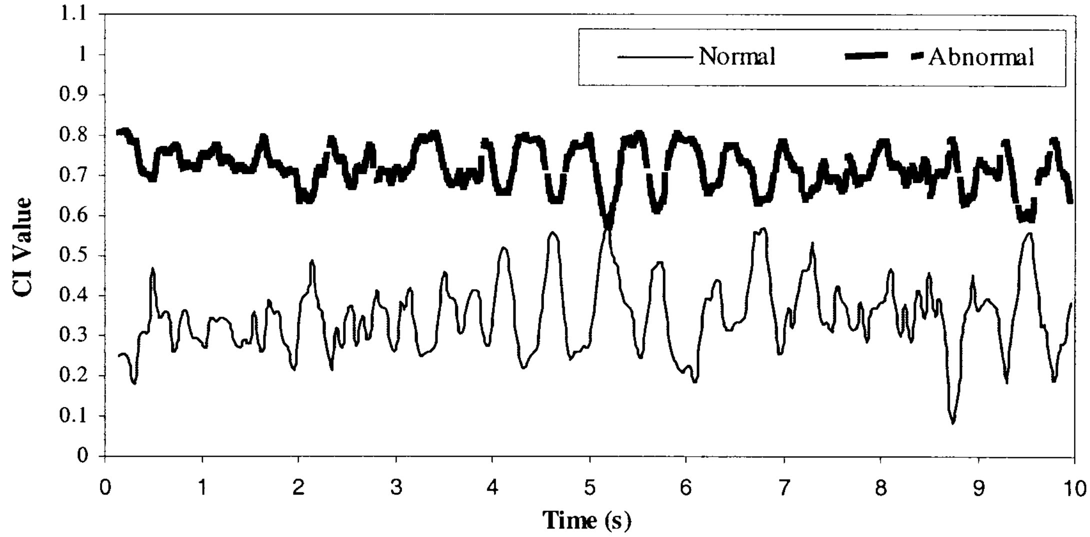



Figure 4.20 Condition index as a moving average for fan end bearing, 0.007” fault diameter at inner race, 1797 rpm and 0 hp motor load.

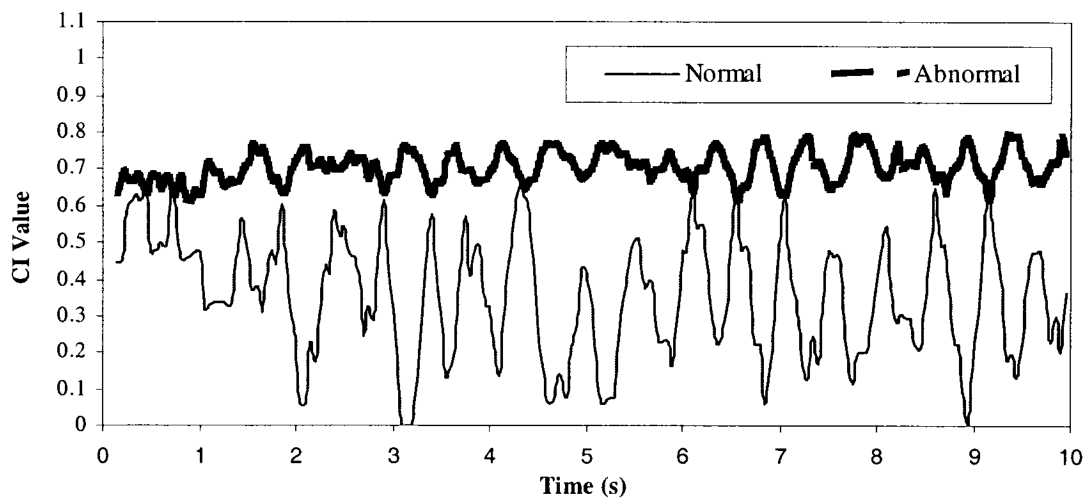



Figure 4.21 Condition index as a moving average for fan end bearing, 0.007” fault diameter at inner race, 1730 rpm and 3 hp motor load.

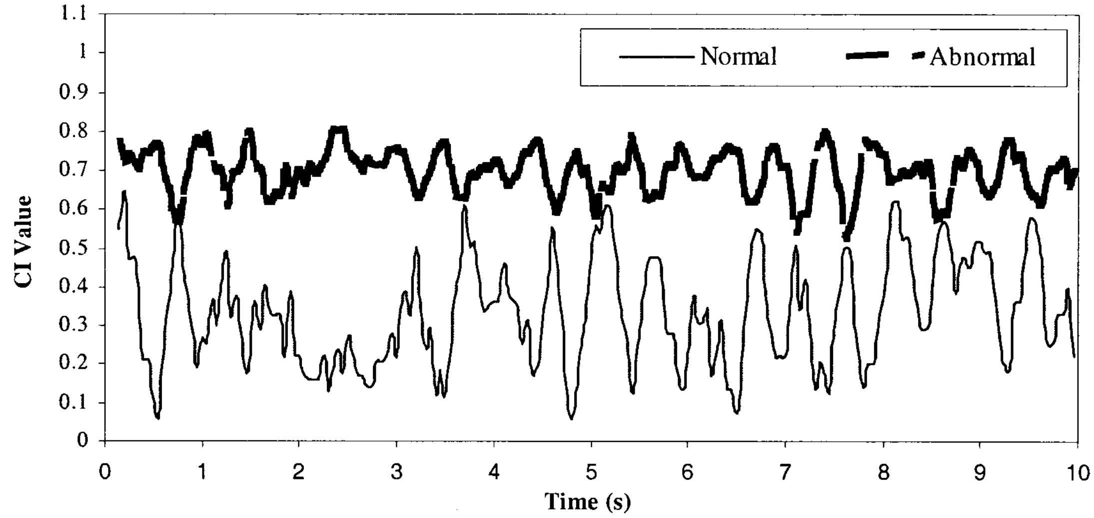



Figure 4.22 Condition index as a moving average for fan end bearing, 0.014" fault diameter at inner race, 1797 rpm and 0 hp motor load.

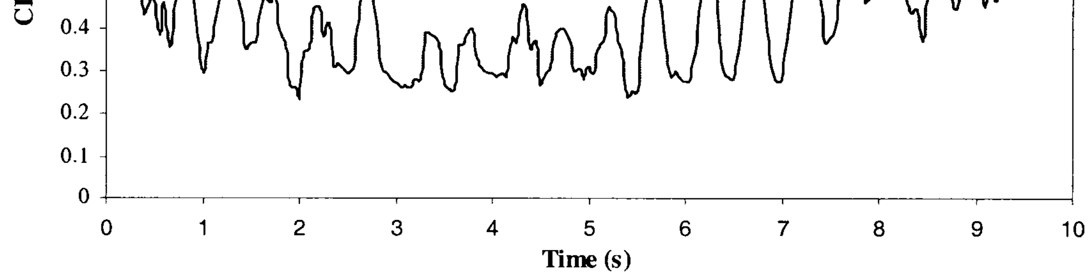



Figure 4.23 Condition index as a moving average for fan end bearing, 0.014" fault diameter at inner race, 1730 rpm and 3 hp motor load.

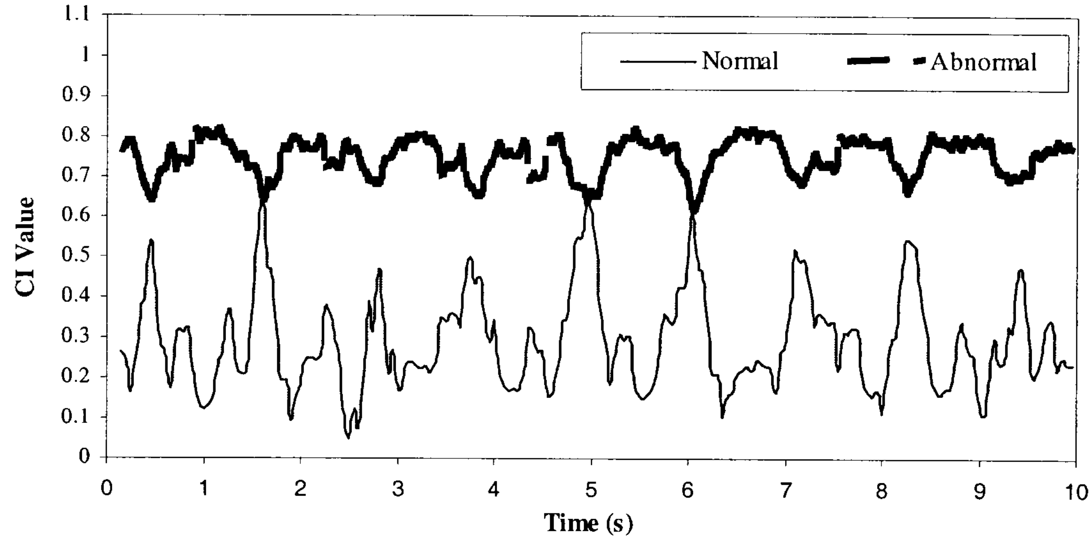



Figure 4.24 Condition index as a moving average for fan end bearing, 0.021" fault diameter at inner race, 1797 rpm and 0 hp motor load.

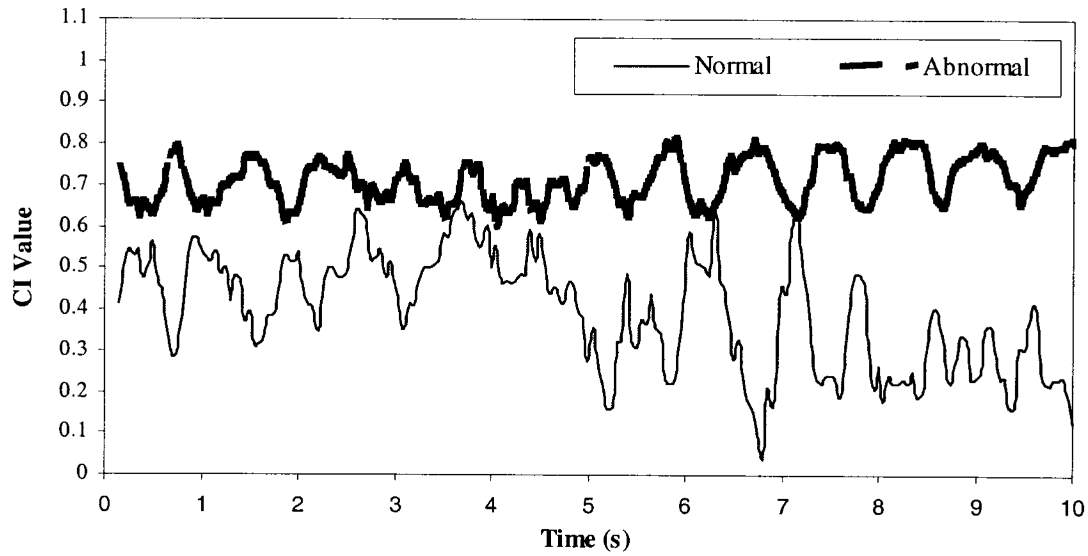



Figure 4.25 Condition index as a moving average for fan end bearing, 0.021" fault diameter at inner race, 1730 rpm and 3 hp motor load.

#### 4.1.4.3 Index of Deterioration

To illustrate the use of the index of deterioration, two examples are depicted below. The index of deterioration was calculated based on sixteen values of each of the complement of power and standard-deviation correlation factors (i.e. total number of samples is 32). This calculation was repeated two more times to show the data trend.

Placing the measurements in series to simulate the time effect on the bearing condition (this is presented in the below figures as the *time index*), Figures 4.26 and 4.27 were prepared to show the bearing health degradation in terms of the fault diameter. Figure 4.26 depicts the bearing deterioration when faults with various sizes are placed at the inner race of the fan end bearing, using a load of 0 hp, and a speed of 1797 rpm. On the other hand, Figure 4.27 illustrates the same type of tests but for the drive end bearing. One can notice in Figure 4.26 that the index of deterioration increased suddenly in parallel with the transition from normal to 0.007" faulty state. Along the same line, the deterioration was increasing marginally as the fault diameter was changing from 0.007" to 0.014" and 0.021" successively. This could be due to the type of anomaly (transient) produced by the fault size. The same observation can be applied to Figure 4.27 as the index of deterioration increased when the fault grew from 0.014" to 0.021" and 0.028". However, the 0.007" fault in Figure 4.27 produced an index of deterioration higher than the ones for 0.014" and 0.021". There is no real explanation yet for this unreasonable event.

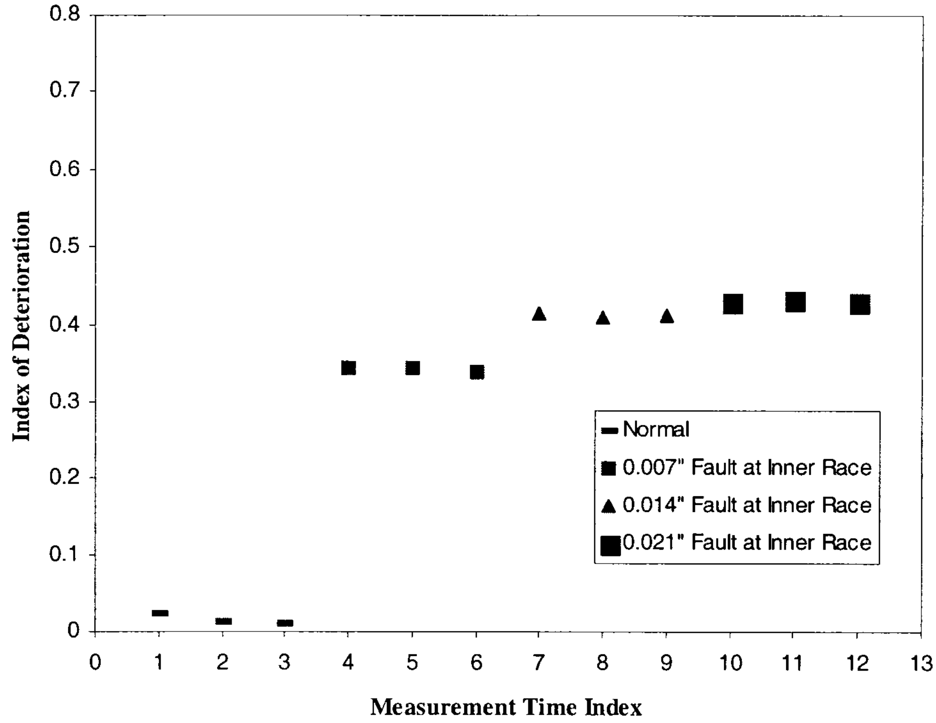



Figure 4.26 Index of deterioration for fan end bearing at 0 hp and 1797 rpm.

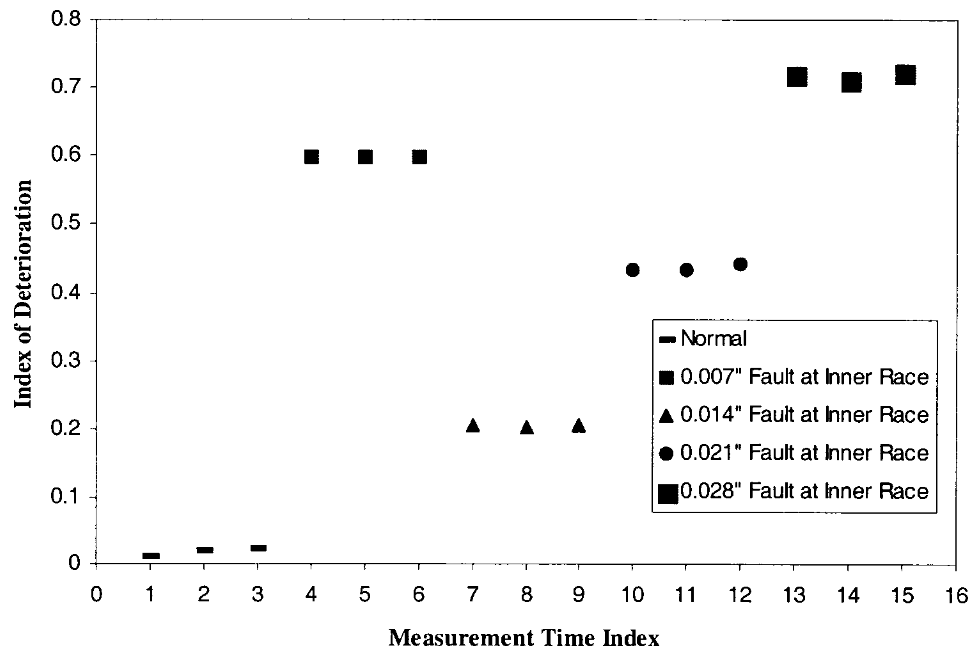



Figure 4.27 Index of deterioration for drive end bearing at 0 hp and 1797 rpm.

### 4.1.5 Conclusions

A fuzzy-set based fusion method was proposed to combine the effects of the four condition indicators, defined in Chapter 3, in one model in order to facilitate the decision-making. In addition, a deterioration index has been developed for *fault tolerance* investigations.

This method was applied successfully to both tool and bearing fault detection. Results obtained from experimental work suggest that this method is quick and suitable for on-line monitoring but is not flexible and robust enough to deal with sensitivity variation of the condition indicators in certain cases.

## 4.2 Fuzzy-rule Based Fusion

The fusion method described above is based on fuzzy set theory. This method can be used to integrate the condition indicators and provide a quick check on tool or machinery conditions. The selection of this technique means that the boundaries between normal and abnormal conditions are well defined. This assumption in general is not completely valid especially when dealing with complex systems. There is always an uncertain zone between normal and abnormal ranges. Furthermore, as the number of monitoring indices increases, the selection of the associated limits becomes a very difficult task. To eliminate the need for various thresholds and make the solution more realistic, the monitoring indices can be fused using a fuzzy-rule based method.

In addition to the prelisted advantages, the fuzzy-rule based technique provides a weighted solution based on all condition indicators during the inference process. This is very useful especially when some monitoring indices are not too sensitive as seen in the first method to certain types of anomalies. The model in this case takes that into account through the predefined detection rules and maintains as such the same detection efficiency. These flexibility and robustness characteristics make this technique suitable for continuous on-line monitoring for cases or applications involving rapid condition changes from normal to abnormal. On the other hand, the definition of the rule base can be a lengthy process depending on the number of input indices, and may require training and domain knowledge about the monitored process or application.

In this thesis, it is proposed to use a multiple inputs, single output (MISO) fuzzy fusion module. The condition indicators developed in Chapter 3 are fed into the module and then fused as a single output (*FFI*, fuzzy fused index) which describes the condition of the monitored machine or process in question: i.e. normal or abnormal.

### 4.2.1 The Rule-based Fuzzy Fusion Module

The rule-based fuzzy fusion module is shown in Figure 4.28 and described below.

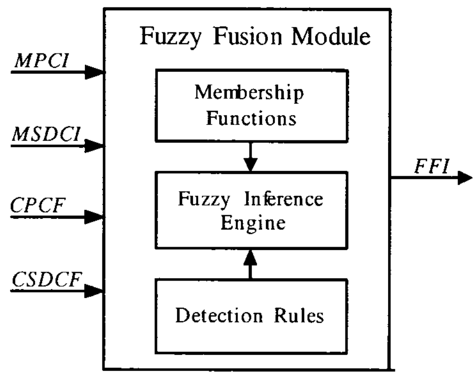



Figure 4.28 Fuzzy index fusion module.

#### 4.2.1.1 Module Inputs

The condition indicators developed in Chapter 3 are used to define the model inputs. The maximum power condition indicator (*MPCI*) and maximum standard-deviation condition indicator (*MSDCI*) within each sub-group, and the complement of the power correlation factor (*CPCF*) and the complement of the standard-deviation correlation factor (*CSDCF*) are utilized as inputs. All these inputs are already normalized (i.e. varying within [0,1]).

#### 4.2.1.2 Membership Functions

To associate crisp inputs with fuzzy sets, membership functions have to be defined. For computational efficiency, the triangular membership functions are used for inputs and spike membership functions are employed for outputs.

In this study, four fuzzy sets are proposed for the inputs and three others for the outputs (Figure 4.29). The input fuzzy sets are selected as follows: ZV, zero value; SV, small value; MV, medium value; BV, big value. On the other hand, the output fuzzy sets are defined in this order: NR, normal range; MR, middle range; and AR, abnormal range. The universe of discourse of all inputs and outputs is in the range of [0,1].

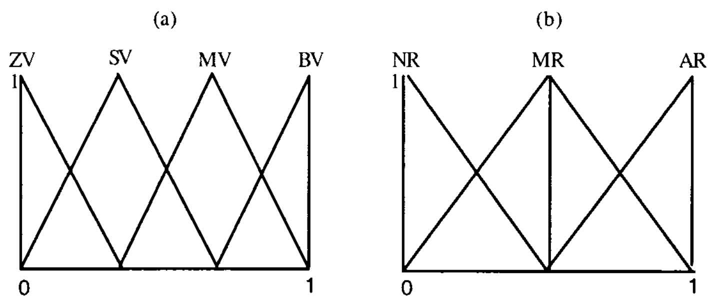



Figure 4.29 Membership functions for input (a) and output (b).

### 4.2.1.3 Detection Rules

One of the most important components of this fuzzy module is the detection rule base. The fuzzy rules, which determine the output norm based on the inputs, represents the knowledge gained through learning and experience of domain experts. Each of these rules is written as an *IF* (condition indicator's range) – THEN (consequences, i.e. machine or process condition). As shown in Figures 4.28 and 4.29, there are four input fuzzy sets associated with four condition indicators. This leads to a total of 256 rules ($4^4$) as illustrated in Table 4.1. This list of rules will be used for both tool and bearing condition monitoring as explained in the experimental section.

Each set of condition indicators generally activates several fuzzy rules. For example, consider Figure 4.30. If *MPCI* and *MSDCI* fall respectively between MV and BV, and SV and MV, *CPCF* triggers ZV and SV and *CSDCF* activates SV and MV. Eight points in total are triggered at different heights (*h*). This leads to sixteen combinations of fuzzy sets. Referring to Table 4.1, the following detection rules will be activated:

*Rule* (146): *IF MPCI is MV and MSDCI is SV and CPCF is ZV and CSDCF is SV THEN Condition is NR*

*Rule* (147): *IF MPCI is MV and MSDCI is SV and CPCF is ZV and CSDCF is MV THEN Condition is AR*

*Rule* (150): *IF MPCI is MV and MSDCI is SV and CPCF is SV and CSDCF is SV THEN Condition is AR*

*Rule* (151): *IF MPCI is MV and MSDCI is SV and CPCF is SV and CSDCF is MV THEN Condition is AR*

*Rule* (162): *IF MPCI is MV and MSDCI is MV and CPCF is ZV and CSDCF is SV THEN Condition is NR*

*Rule* (163): *IF MPCI is MV and MSDCI is MV and CPCF is ZV and CSDCF is MV THEN Condition is AR*

*Rule* (166): *IF MPCI is MV and MSDCI is MV and CPCF is SV and CSDCF is SV THEN Condition is AR*

*Rule* (167): *IF MPCI is MV and MSDCI is MV and CPCF is SV and CSDCF is MV THEN CONDITION is AR*

*Rule* (210): *IF MPCI is BV and MSDCI is SV and CPCF is ZV and CSDCF is SV THEN Condition is AR*

*Rule* (211): *IF MPCI is BV and MSDCI is SV and CPCF is ZV and CSDCF is MV THEN CONDITION is AR*

*Rule* (214): *IF MPCI is BV and MSDCI is SV and CPCF is SV and CSDCF is SV THEN Condition is AR*

*Rule* (215): *IF MPCI is BV and MSDCI is SV and CPCF is SV and CSDCF is MV THEN Condition is AR*

*Rule* (226): *IF MPCI is BV and MSDCI is MV and CPCF is ZV and CSDCF is SV THEN Condition is AR*

*Rule* (227): *IF MPCI is BV and MSDCI is MV and CPCF is ZV and CSDCF is MV THEN Condition is AR*

*Rule* (230): *IF MPCI is BV and MSDCI is MV and CPCF is SV and CSDCF is SV THEN Condition is AR*

*Rule* (231): *IF MPCI is BV and MSDCI is MV and CPCF is SV and CSDCF is MV THEN Condition is AR*

To explain briefly the meaning of these rules, let's take for example rule (166). This can be translated as, if the maximum power condition indicator (*MPCI*), the maximum standard-deviation condition indicator (*MSDCI*) fall within a medium range (MV) and the complement of the power and standard-deviation correlation factors have small values (SV), the machine or process condition for that rule will be assigned to the normal range (NR). All these rules will jointly contribute to the strength of the output (fuzzy fused index), which will be defined by the fuzzy inference engine as illustrated in the below section.

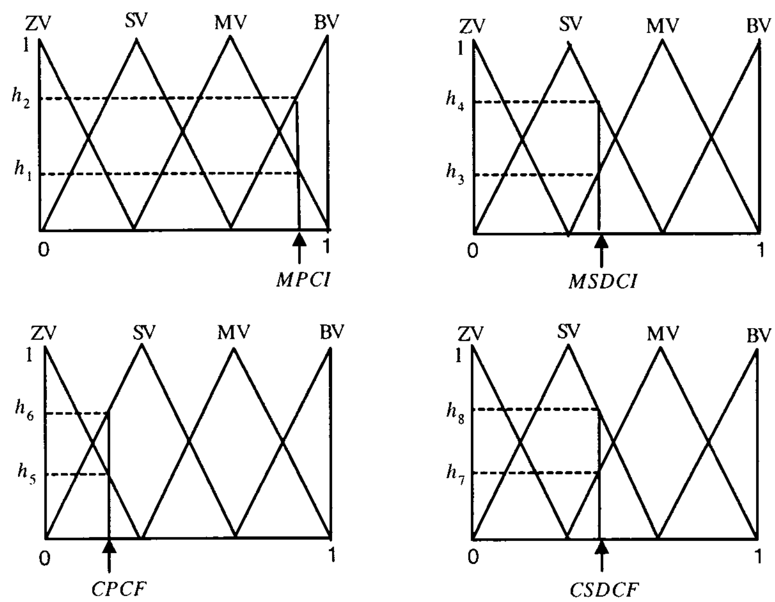



Figure 4.30 Illustration of inputs and associated fuzzy sets.

Table 4.1 Detection rules for the fuzzy fusion module.

| Rule No. | *MPCI* | *MSDCI* | *CPCF* | *CSDCF* | *Cond-ition* |
|---|---|---|---|---|---|
| 1 | ZV | ZV | ZV | ZV | NR |
| 2 | ZV | ZV | ZV | SV | NR |
| 3 | ZV | ZV | ZV | MV | AR |
| 4 | ZV | ZV | ZV | BV | AR |
| 5 | ZV | ZV | SV | ZV | NR |
| 6 | ZV | ZV | SV | SV | AR |
| 7 | ZV | ZV | SV | MV | AR |
| 8 | ZV | ZV | SV | BV | AR |
| 9 | ZV | ZV | MV | ZV | MR |
| 10 | ZV | ZV | MV | SV | AR |
| 11 | ZV | ZV | MV | MV | AR |
| 12 | ZV | ZV | MV | BV | AR |
| 13 | ZV | ZV | BV | ZV | MR |
| 14 | ZV | ZV | BV | SV | AR |
| 15 | ZV | ZV | BV | MV | AR |
| 16 | ZV | ZV | BV | BV | AR |
| 17 | ZV | SV | ZV | ZV | NR |
| 18 | ZV | SV | ZV | SV | NR |
| 19 | ZV | SV | ZV | MV | AR |
| 20 | ZV | SV | ZV | BV | AR |
| 21 | ZV | SV | SV | ZV | NR |
| 22 | ZV | SV | SV | SV | AR |
| 23 | ZV | SV | SV | MV | AR |
| 24 | ZV | SV | SV | BV | AR |
| 25 | ZV | SV | MV | ZV | MR |
| 26 | ZV | SV | MV | SV | AR |
| 27 | ZV | SV | MV | MV | AR |
| 28 | ZV | SV | MV | BV | AR |
| 29 | ZV | SV | BV | ZV | MR |
| 30 | ZV | SV | BV | SV | AR |
| 31 | ZV | SV | BV | MV | AR |
| 32 | ZV | SV | BV | BV | AR |
| 33 | ZV | MV | ZV | ZV | NR |
| 34 | ZV | MV | ZV | SV | NR |
| 35 | ZV | MV | ZV | MV | MR |
| 36 | ZV | MV | ZV | BV | AR |
| 37 | ZV | MV | SV | ZV | NR |
| 38 | ZV | MV | SV | SV | AR |
| 39 | ZV | MV | SV | MV | AR |
| 40 | ZV | MV | SV | BV | AR |
| 41 | ZV | MV | MV | ZV | MR |
| 42 | ZV | MV | MV | SV | AR |
| 43 | ZV | MV | MV | MV | AR |
| 44 | ZV | MV | MV | BV | AR |
| 45 | ZV | MV | BV | ZV | MR |
| 46 | ZV | MV | BV | SV | AR |
| 47 | ZV | MV | BV | MV | AR |
| 48 | ZV | MV | BV | BV | AR |
| 49 | ZV | BV | ZV | ZV | MR |
| 50 | ZV | BV | ZV | SV | NR |
| 51 | ZV | BV | ZV | MV | AR |
| 52 | ZV | BV | ZV | BV | AR |
| 53 | ZV | BV | SV | ZV | NR |
| 54 | ZV | BV | SV | SV | AR |
| 55 | ZV | BV | SV | MV | AR |
| 56 | ZV | BV | SV | BV | AR |
| 57 | ZV | BV | MV | ZV | MR |
| 58 | ZV | BV | MV | SV | AR |
| 59 | ZV | BV | MV | MV | AR |
| 60 | ZV | BV | MV | BV | AR |
| 61 | ZV | BV | BV | ZV | MR |
| 62 | ZV | BV | BV | SV | AR |
| 63 | ZV | BV | BV | MV | AR |
| 64 | ZV | BV | BV | BV | AR |
| 65 | SV | ZV | ZV | ZV | NR |
| 66 | SV | ZV | ZV | SV | NR |
| 67 | SV | ZV | ZV | MV | AR |
| 68 | SV | ZV | ZV | BV | AR |
| 69 | SV | ZV | SV | ZV | NR |
| 70 | SV | ZV | SV | SV | AR |
| 71 | SV | ZV | SV | MV | AR |
| 72 | SV | ZV | SV | BV | AR |
| 73 | SV | ZV | MV | ZV | MR |
| 74 | SV | ZV | MV | SV | AR |
| 75 | SV | ZV | MV | MV | AR |
| 76 | SV | ZV | MV | BV | AR |
| 77 | SV | ZV | BV | ZV | MR |
| 78 | SV | ZV | BV | SV | AR |
| 79 | SV | ZV | BV | MV | AR |
| 80 | SV | ZV | BV | BV | AR |
| 81 | SV | SV | ZV | ZV | NR |
| 82 | SV | SV | ZV | SV | NR |
| 83 | SV | SV | ZV | MV | AR |
| 84 | SV | SV | ZV | BV | AR |
| 85 | SV | SV | SV | ZV | NR |
| 86 | SV | SV | SV | SV | AR |
| 87 | SV | SV | SV | MV | AR |
| 88 | SV | SV | SV | BV | AR |
| 89 | SV | SV | MV | ZV | MR |
| 90 | SV | SV | MV | SV | AR |
| 91 | SV | SV | MV | MV | AR |
| 92 | SV | SV | MV | BV | AR |
| 93 | SV | SV | BV | ZV | MR |
| 94 | SV | SV | BV | SV | AR |
| 95 | SV | SV | BV | MV | AR |
| 96 | SV | SV | BV | BV | AR |
| 97 | SV | MV | ZV | ZV | NR |
| 98 | SV | MV | ZV | SV | NR |

Table 4.1 (continued)

| Rule No. | *MPCI* | *MSDCI* | *CPCF* | *CSDCF* | *Cond-ition* |
|---|---|---|---|---|---|
| 99 | SV | MV | ZV | MV | AR |
| 100 | SV | MV | ZV | BV | AR |
| 101 | SV | MV | SV | ZV | NR |
| 102 | SV | MV | SV | SV | AR |
| 103 | SV | MV | SV | MV | AR |
| 104 | SV | MV | SV | BV | AR |
| 105 | SV | MV | MV | ZV | MR |
| 106 | SV | MV | MV | SV | AR |
| 107 | SV | MV | MV | MV | AR |
| 108 | SV | MV | MV | BV | AR |
| 109 | SV | MV | BV | ZV | MR |
| 110 | SV | MV | BV | SV | AR |
| 111 | SV | MV | BV | MV | AR |
| 112 | SV | MV | BV | BV | AR |
| 113 | SV | BV | ZV | ZV | NR |
| 114 | SV | BV | ZV | SV | NR |
| 115 | SV | BV | ZV | MV | AR |
| 116 | SV | BV | ZV | BV | AR |
| 117 | SV | BV | SV | ZV | NR |
| 118 | SV | BV | SV | SV | AR |
| 119 | SV | BV | SV | MV | AR |
| 120 | SV | BV | SV | BV | AR |
| 121 | SV | BV | MV | ZV | MR |
| 122 | SV | BV | MV | SV | AR |
| 123 | SV | BV | MV | MV | AR |
| 124 | SV | BV | MV | BV | AR |
| 125 | SV | BV | BV | ZV | MR |
| 126 | SV | BV | BV | SV | AR |
| 127 | SV | BV | BV | MV | AR |
| 128 | SV | BV | BV | BV | AR |
| 129 | MV | ZV | ZV | ZV | NR |
| 130 | MV | ZV | ZV | SV | NR |
| 131 | MV | ZV | ZV | MV | AR |
| 132 | MV | ZV | ZV | BV | AR |
| 133 | MV | ZV | SV | ZV | NR |
| 134 | MV | ZV | SV | SV | AR |
| 135 | MV | ZV | SV | MV | AR |
| 136 | MV | ZV | SV | BV | AR |
| 137 | MV | ZV | MV | ZV | MR |
| 138 | MV | ZV | MV | SV | AR |
| 139 | MV | ZV | MV | MV | AR |
| 140 | MV | ZV | MV | BV | AR |
| 141 | MV | ZV | BV | ZV | MR |
| 142 | MV | ZV | BV | SV | AR |
| 143 | MV | ZV | BV | MV | AR |
| 144 | MV | ZV | BV | BV | AR |
| 145 | MV | SV | ZV | ZV | NR |
| 146 | MV | SV | ZV | SV | NR |
| 147 | MV | SV | ZV | MV | AR |
| 148 | MV | SV | ZV | BV | AR |
| 149 | MV | SV | SV | ZV | NR |
| 150 | MV | SV | SV | SV | AR |
| 151 | MV | SV | SV | MV | AR |
| 152 | MV | SV | SV | BV | AR |
| 153 | MV | SV | MV | ZV | MR |
| 154 | MV | SV | MV | SV | AR |
| 155 | MV | SV | MV | MV | AR |
| 156 | MV | SV | MV | BV | AR |
| 157 | MV | SV | BV | ZV | MR |
| 158 | MV | SV | BV | SV | AR |
| 159 | MV | SV | BV | MV | AR |
| 160 | MV | SV | BV | BV | AR |
| 161 | MV | MV | ZV | ZV | NR |
| 162 | MV | MV | ZV | SV | NR |
| 163 | MV | MV | ZV | MV | AR |
| 164 | MV | MV | ZV | BV | AR |
| 165 | MV | MV | SV | ZV | NR |
| 166 | MV | MV | SV | SV | AR |
| 167 | MV | MV | SV | MV | AR |
| 168 | MV | MV | SV | BV | AR |
| 169 | MV | MV | MV | ZV | MR |
| 170 | MV | MV | MV | SV | AR |
| 171 | MV | MV | MV | MV | AR |
| 172 | MV | MV | MV | BV | AR |
| 173 | MV | MV | BV | ZV | MR |
| 174 | MV | MV | BV | SV | AR |
| 175 | MV | MV | BV | MV | AR |
| 176 | MV | MV | BV | BV | AR |
| 177 | MV | BV | ZV | ZV | AR |
| 178 | MV | BV | ZV | SV | AR |
| 179 | MV | BV | ZV | MV | AR |
| 180 | MV | BV | ZV | BV | AR |
| 181 | MV | BV | SV | ZV | AR |
| 182 | MV | BV | SV | SV | AR |
| 183 | MV | BV | SV | MV | AR |
| 184 | MV | BV | SV | BV | AR |
| 185 | MV | BV | MV | ZV | AR |
| 186 | MV | BV | MV | SV | AR |
| 187 | MV | BV | MV | MV | AR |
| 188 | MV | BV | MV | BV | AR |
| 189 | MV | BV | BV | ZV | AR |
| 190 | MV | BV | BV | SV | AR |
| 191 | MV | BV | BV | MV | AR |
| 192 | MV | BV | BV | BV | AR |
| 193 | BV | ZV | ZV | ZV | MR |
| 194 | BV | ZV | ZV | SV | AR |
| 195 | BV | ZV | ZV | MV | AR |
| 196 | BV | ZV | ZV | BV | AR |

Table 4.1 (continued)

| Rule No. | *MPCI* | *MSDCI* | *CPCF* | *CSDCF* | *Condition* |
|---|---|---|---|---|---|
| 197 | BV | ZV | SV | ZV | AR |
| 198 | BV | ZV | SV | SV | AR |
| 199 | BV | ZV | SV | MV | AR |
| 200 | BV | ZV | SV | BV | AR |
| 201 | BV | ZV | MV | ZV | AR |
| 202 | BV | ZV | MV | SV | AR |
| 203 | BV | ZV | MV | MV | AR |
| 204 | BV | ZV | MV | BV | AR |
| 205 | BV | ZV | BV | ZV | AR |
| 206 | BV | ZV | BV | SV | AR |
| 207 | BV | ZV | BV | MV | AR |
| 208 | BV | ZV | BV | BV | AR |
| 209 | BV | SV | ZV | ZV | AR |
| 210 | BV | SV | ZV | SV | AR |
| 211 | BV | SV | ZV | MV | AR |
| 212 | BV | SV | ZV | BV | AR |
| 213 | BV | SV | SV | ZV | AR |
| 214 | BV | SV | SV | SV | AR |
| 215 | BV | SV | SV | MV | AR |
| 216 | BV | SV | SV | BV | AR |
| 217 | BV | SV | MV | ZV | AR |
| 218 | BV | SV | MV | SV | AR |
| 219 | BV | SV | MV | MV | AR |
| 220 | BV | SV | MV | BV | AR |
| 221 | BV | SV | BV | ZV | AR |
| 222 | BV | SV | BV | SV | AR |
| 223 | BV | SV | BV | MV | AR |
| 224 | BV | SV | BV | BV | AR |
| 225 | BV | MV | ZV | ZV | AR |
| 226 | BV | MV | ZV | SV | AR |
| 227 | BV | MV | ZV | MV | AR |
| 228 | BV | MV | ZV | BV | AR |
| 229 | BV | MV | SV | ZV | AR |
| 230 | BV | MV | SV | SV | AR |
| 231 | BV | MV | SV | MV | AR |
| 232 | BV | MV | SV | BV | AR |
| 233 | BV | MV | MV | ZV | AR |
| 234 | BV | MV | MV | SV | AR |
| 235 | BV | MV | MV | MV | AR |
| 236 | BV | MV | MV | BV | AR |
| 237 | BV | MV | BV | ZV | AR |
| 238 | BV | MV | BV | SV | AR |
| 239 | BV | MV | BV | MV | AR |
| 240 | BV | MV | BV | BV | AR |
| 241 | BV | BV | ZV | ZV | AR |
| 242 | BV | BV | ZV | SV | AR |
| 243 | BV | BV | ZV | MV | AR |
| 244 | BV | BV | ZV | BV | AR |

| Rule No. | *MPCI* | *MSDCI* | *CPCF* | *CSDCF* | *Condition.* |
|---|---|---|---|---|---|
| 245 | BV | BV | SV | ZV | AR |
| 246 | BV | BV | SV | SV | AR |
| 247 | BV | BV | SV | MV | AR |
| 248 | BV | BV | SV | BV | AR |
| 249 | BV | BV | MV | ZV | AR |
| 250 | BV | BV | MV | SV | AR |
| 251 | BV | BV | MV | MV | AR |
| 252 | BV | BV | MV | BV | AR |
| 253 | BV | BV | BV | ZV | AR |
| 254 | BV | BV | BV | SV | AR |
| 255 | BV | BV | BV | MV | AR |
| 256 | BV | BV | BV | BV | AR |

#### 4.2.1.4 Fuzzy Inference Engine

The role of the fuzzy inference engine is to perform the fuzzy operations necessary for the determination of the *FFI*. During this phase, the fused index is computed based on the fuzzy inputs and the activated rules. As the need to reduce the computation time in a monitoring system becomes a generic requirement for most of today's industrial applications, the Sugeno-style fuzzy inference engine is selected in this analysis. This type of engines provides a very quick response comparing with the well known Mandani-style inference engine. The difference between the two techniques is mainly due to the selection of the output membership function. In the Sugeno-style the output membership is presented by a spike (Figure 4.29) instead of a complete triangle as suggested by the Mandani-style. In addition to its computation efficiency the Sugeno-style has other attractive characteristics such as suitability for mathematical analyses and effectiveness to work with optimization and adaptive techniques (Liang *et al* 2003).

The output, i.e. fuzzy fused index, is simply the weighted (with respect to the spike heights) average of the spike locations (Liang *et al* 2003) i.e.:

$$FFI = \frac{\left(\sum_{rl=1}^{\overline{rl}} H_{rl} L_{rl}\right)}{\sum_{rl=1}^{\overline{rl}} H_{rl}} \tag{4.12}$$

where $L_{rl}$ is the location of the output spike associated with rule *rl*, $H_{rl}$ is the minimum height (or membership) associated with rule *rl* and $\overline{rl}$ is the total number of activated rules.

To illustrate the method, let's consider a numerical example. Assume that the model inputs have the following values in Figure 4.30:

$MPCI = 0.87$

$MSDCI = 0.43$

$CPCF = 0.20$

$CSDCF = 0.38$

These four inputs trigger the fuzzy sets at a total of eight points as illustrated in Figure 4.30 and Table 4.2.

Table 4.2 Input membership degrees associated with fuzzy inputs.

| Input | Triggered Sets | Height of Triggered Points |
|---|---|---|
| $MPCI = 0.87$ | MV and BV | $h_1 = 0.39$ and $h_2 = 0.61$ |
| $MSDCI = 0.43$ | MV and SV | $h_3 = 0.29$ and $h_4 = 0.71$ |
| $CPCF = 0.22$ | ZV and SV | $h_5 = 0.40$ and $h_6 = 0.60$ |
| $CSDCF = 0.10$ | MV and SV | $h_7 = 0.19$ and $h_8 = 0.86$ |

As such, the pre-defined points result in a total of sixteen fuzzy set permutations, each one associated with a rule as described in the previous section. The possible permutations, associated rule number and output spike location are depicted in Table 4.3. The associated height (column 4 of Table 4.3) is determined by taking the minimum height within each permutation. On the other hand, the output spike location (column five of Table 4.3) is determined based on the selected rule and the output membership as illustrated in Figure 4.29. For example, the output membership fuzzy set for rule (146), in the sixth row of Table 4.3, is NR. Looking at Figure 4.29, the NR set spike is located at fuzzy zero (0) set.

Table 4.3 Sugeno-style elements of the illustrative example.

| $rl$ | Fuzzy Set Permutation | Activated Rule | Associated Height ($H_{rl}$) | Output Spike Location ($L_{rl}$) |
|---|---|---|---|---|
| 1 | (MV, MV, ZV, MV) | Rule (163) | Min(0.39, 0.29, 0.40, 0.14)=0.14 | 1 |
| 2 | (MV, MV, ZV, SV) | Rule (162) | Min(0.39, 0.29, 0.40, 0.86)=0.29 | 0 |
| 3 | (MV, MV, SV, MV) | Rule (167) | Min(0.39, 0.29, 0.60, 0.14)=0.14 | 1 |
| 4 | (MV, MV, SV, SV) | Rule (166) | Min(0.39, 0.29, 0.60, 0.86)=0.29 | 1 |
| 5 | (MV, SV, ZV, MV) | Rule (147) | Min(0.39, 0.71, 0.40, 0.14)=0.14 | 1 |
| 6 | (MV, SV, ZV, SV) | Rule (146) | Min(0.39, 0.71, 0.40, 0.86)=0.39 | 0 |
| 7 | (MV, SV, SV, MV) | Rule (151) | Min(0.39, 0.71, 0.60, 0.14)=0.14 | 1 |
| 8 | (MV, MV, ZV, MV) | Rule (150) | Min(0.39, 0.71, 0.60, 0.86)=0.39 | 1 |
| 9 | (BV, MV, ZV, MV) | Rule (227) | Min(0.61, 0.29, 0.40, 0.14)=0.14 | 1 |
| 10 | (BV, MV, ZV, SV) | Rule (226) | Min(0.61, 0.29, 0.40, 0.86)=0.29 | 1 |
| 11 | (BV, MV, SV, MV) | Rule (231) | Min(0.61, 0.29, 0.60, 0.14)=0.14 | 1 |
| 12 | (BV, MV, SV, SV) | Rule (230) | Min(0.61, 0.29, 0.60, 0.86)=0.29 | 1 |
| 13 | (BV, SV, ZV, MV) | Rule (211) | Min(0.61, 0.71, 0.40, 0.14)=0.14 | 1 |
| 14 | (BV, SV, ZV, SV) | Rule (210) | Min(0.61, 0.71, 0.40, 0.86)=0.40 | 1 |
| 15 | (BV, SV, SV, MV) | Rule (215) | Min(0.61, 0.71, 0.60, 0.14)=0.14 | 1 |
| 16 | (BV, MV, ZV, MV) | Rule (240) | Min(0.61, 0.71, 0.60, 0.86)=0.60 | 1 |

The output spike locations and the associated heights are presented in Figure 4.31. Using equation (4.12) and Table 4.6, the fuzzy fused index can be calculated as follows:

$FFI = [(0.14\times1+0.29\times0+0.14\times1+0.29\times1+0.14\times1+0.39\times0+0.14\times1+0.39\times1+0.14\times10.29\times1+0.14\times1+029\times1+0.14\times1+0.40\times1+0.14\times1+0.60\times1)/(0.14+0.29+0.14+0.29+0.14+0.39+0.14+0.39+0.14+0.29+0.14+0.29+0.14+0.40+0.14+0.60)] = 0.83$

The fuzzy fused index is shown in Figure 4.31.

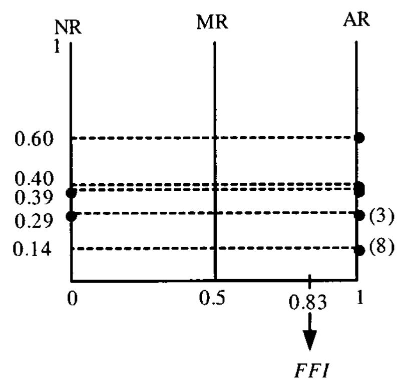



Figure 4.31 Sugeno-style output illustration (Note: (3) means there are three occurrences of 0.29. Similarly, (8) indicates eight occurrences of 0.14).

### 4.2.2 Experimental Work

The rule-based fuzzy fusion module is applied to tool and bearing condition monitoring. The results are described in the sections below.

#### 4.2.2.1 Tool Condition Monitoring

The data collected previously are used herein to validate the pre-described detection method. A MATLAB *C* code program called *Fuzzy* (refer to Appendix C) is developed to compute the fuzzy fused index (*FFI*) using the detection rules and the process described above. The index is calculated for each tool case and then plotted with respect to time. The moving average (MA) for *FFI* is also computed based on four sub-groups, as chosen in Chapter 3, and added thereafter to the same *FFI* plot to show the trend of the mean values.

Figures 4.32 and 4.33 describe the results for a sharp tool. The MA values fluctuate in both cases between 0.1 and 0.4. At the same time, the *FFI* does not exceed

0.5. However, looking at Figures 4.34 and 4.35 for worn cutters one can notice a clear difference. For both speeds 135 rpm and 210 rpm, the *FFI* varies between 0.8 and 1. This result suggests that all four condition indicators are contributing to the wear detection. In parallel, the MA of *FFI* is above 0.8 for both spindle speeds, a clear indication of tool wear. Another important observation can be made with respect to the sensitivity of *FFI* to the change of speed. Looking at Figures 4.32-4.35 and comparing the results obtained at the two available speeds, one can note that the *FFI* is not sensitive to speed change.

Regarding the tool breakage, one can observe in Figure 4.36 that the *FFI* varies suddenly and severely before reaching a steady level at one. This fluctuation within a short time (2 seconds) indicates clearly that a severe incident has occurred. This can be confirmed by looking at the *FFI* and MA values following this significant variation. In fact, both indices reach and stay at one, the highest fuzzy level.

Based on these results, one can conclude that the proposed technique is efficient in detecting anomalies. It is important also to highlight the fact that fusing the condition indicators in one model reduces the number of individual thresholds (one instead of four) and improves consequently the detection capability as all inputs contribute at the same time to the decision-making.

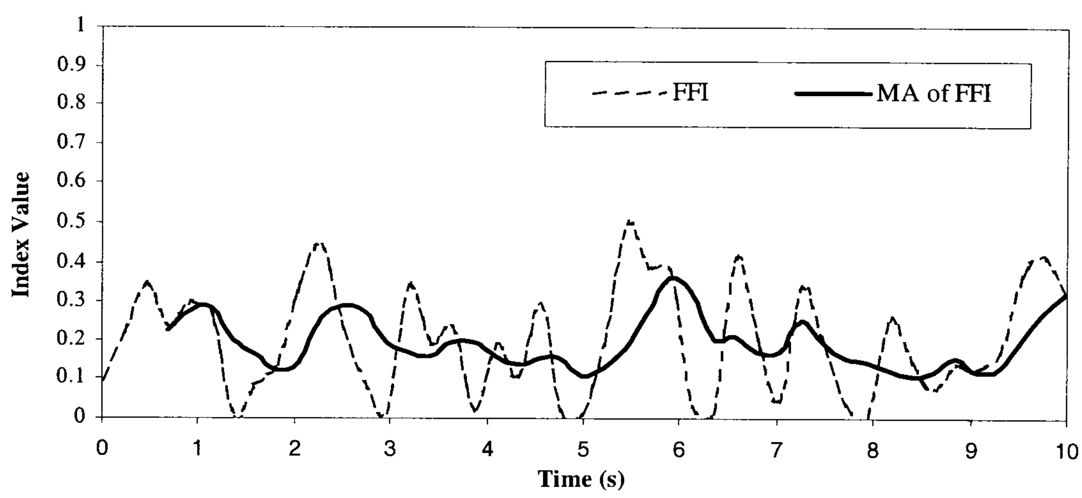



Figure 4.32 Fuzzy fused index and its moving average for sharp cutter at 135 rpm.

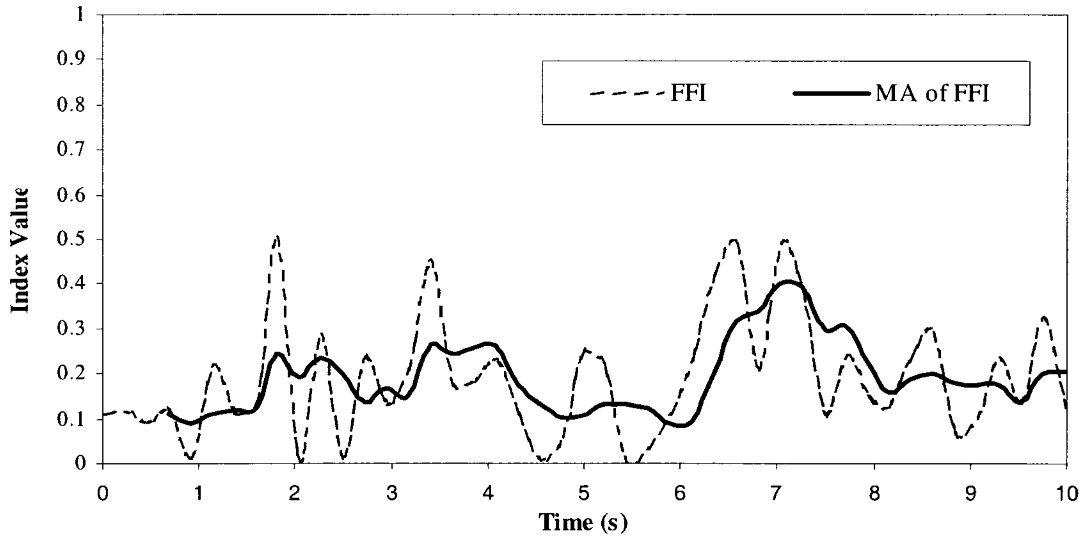



Figure 4.33 Fuzzy fused index and its moving average for sharp cutter at 210 rpm.

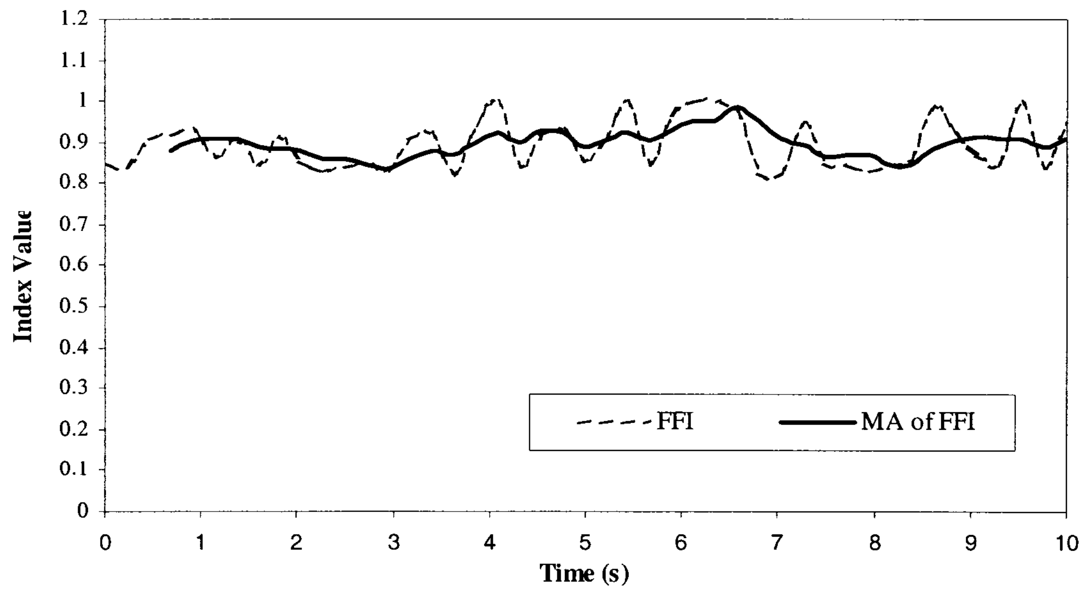



Figure 4.34 Fuzzy fused index and its moving average for worn cutter at 135 rpm.

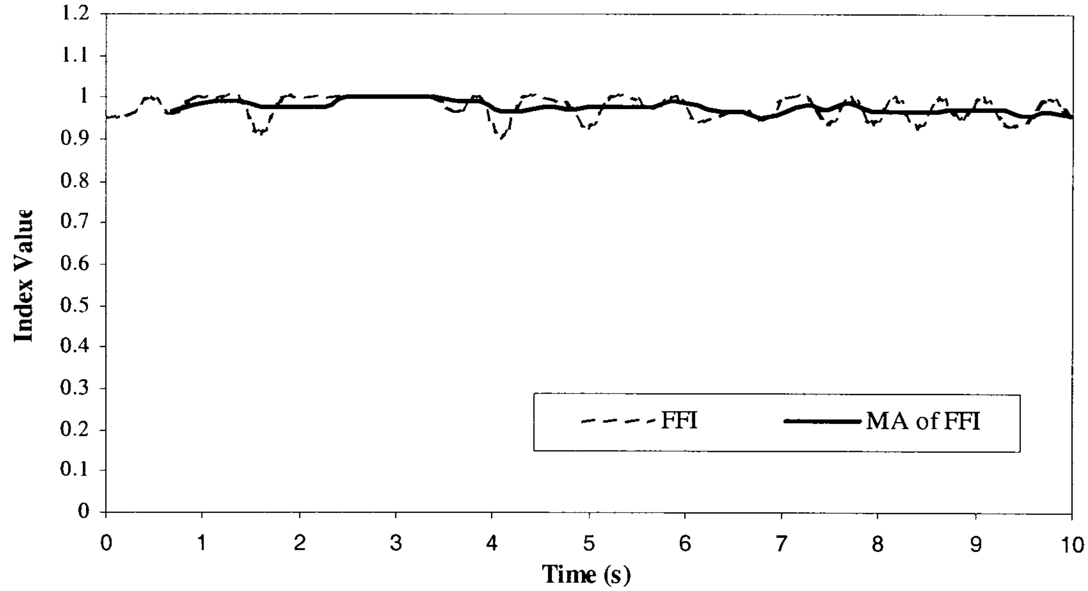



Figure 4.35 Fuzzy fused index and its moving average for worn cutter at 210 rpm.

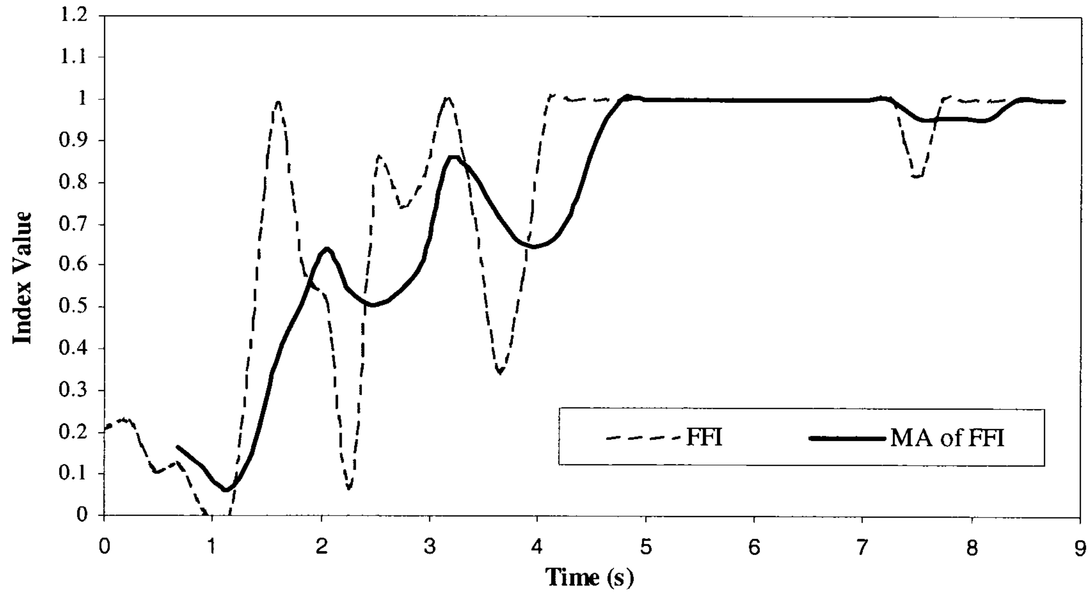



Figure 4.36 Fuzzy fused index and its moving average for tool breakage.

### 4.2.2.2 Bearing Condition Monitoring

The same methodology described in the tool wear case is applied herein. The MATLAB program *Fuzzy* is employed to compute the *FFI* and its MA values using the condition indicators described above and the vibration data collected in Chapter 3. The MA of *FFI* is calculated using four sub-groups as specified in Chapter 3. Both *FFI* and its MA are then plotted for each case with respect to time. In addition, it is important to mention that the same detection rules utilized in the tool monitoring case are employed in this analysis. This clearly shows the generic nature of the condition indicators.

To demonstrate the fault detection efficiency of the fusion method, several typical examples for drive end (DE) and fan end (FE) bearings are considered below. The examples cover both extremities of the load range (0-3 hp), speed range (1797-1730 rpm) and fault size (0.007"-0.028"). The results of the remaining cases are shown in Appendix D. Based on these results, the following discussions can be made with respect to the effects of fault size and location, and the robustness of the method when involving load and speed changes.

#### 4.2.2.2.1 Effect of Fault Size

Figures 4.37-4.38 and 4.39-4.40 show the variations of the *FFI* and its MA for the DE and FE bearings respectively at normal conditions. One can note the *FFI* fluctuates between 0 and 0.5. At the same time, the MA index varies in a narrower range below 0.4. These results are similar to the ones obtained for the sharp tool.

When the bearings are seeded with a 0.007" fault in the inner races, the *FFI* and its MA for the DE bearing, increase and stabilize at one (1) as illustrated in Figures 4.41 and 4.42. This result of maximum *FFI* (equal to 1) is due to the contribution of all condition indicators in the detection of the anomaly. The same scenario occurs for the FE bearing but the *FFI* and its MA in this case reach an average level of 0.90 approximately as shown in Figures 4.43 and 4.44.

With a fault of size 0.014" is placed at the inner race, the MA index is around 0.76 for the DE bearing (Figures 4.45-4.46) and 0.94 for the FE bearing (Figures 4.47-4.48). This can be explained by the fact that the contribution from all the condition indicators is a little bit less than what is shown in the previous case.

In the case of 0.021" fault diameter, the *FFI* and MA indices for both bearings increase again towards one as illustrated in Figures 4.49-4.52. This result is reasonable as discussed below.

When the fault diameter further increases to 0.028", the *FFI* for the DE bearing reaches again the maximum value of one (1) as shown in Figures 4.53 and 4.54. For this specific fault size, the data for the FE is not available as explained in Chapter 3.

In all above examples, the MA levels of *FFI* for abnormal conditions are above 0.6. On the other hand, the MA for normal cases is below 0.4. Therefore, the normal and abnormal conditions can be clearly differentiated. It is also interesting to note that the MA values for the DE bearing increase steadily from around 0.76, to 0.96, and 1.0 when the fault diameter grows from 0.014", to 0.021, and 0.028" respectively. This indicates that the fused index is sensitive to fault size and hence higher FFI or its MA values could be used to estimate severity of the anomaly. However, the MA also reached 1.0 when the fault diameter is only 0.007". In fact, the same observation was made during the calculation of the index of deterioration in the fuzzy-set based technique. The cause of this exception is yet to be investigated. Along this line, the fan end bearing data look more stable. The MA values increase from approximately 0.90 to 0.94 and 0.95 when the fault size varies from 0.007" to 0.014" and 0.021" respectively. Similar observations can be made based on other cases shown in Appendix D.

#### 4.2.2.2.2 Effect of Fault Location

Based on the results shown below and in Appendix D, this method shows a high degree of robustness and effectiveness in identifying all anomalies in the various locations of the two bearings.

It is interesting to note that the *FFI* and its MA values are always closer to one (1) when the fault is seeded at the ball comparing with the other two locations, inner and outer races. This observation could be explained by the fact that when the fault is posed at the ball, it is in continuous contact with the bearing races, increasing as such the complement of the correlation factors and consequently the fused index. As for example the MA of *FFI* for FE bearing when a 0.007" fault is seeded at the ball is 0.98 (Figure D.64) comparing with 0.96 (Figure D.53) and 0.8 (Figure D.75) if the fault is placed at the inner race and outer race respectively. In general, when the fault is placed at the ball, the MA for the FE bearing is above 0.95 and the one for the DE is above 0.9 (Figures D.63-D.73 for FE and Figures D.21-D.35 for DE).

Another significant note can be made with respect to some repeatable patterns observed only in the inner race and outer race studies (e.g. Figures 4.43, 4.44, 4.47, 4.48, 4.51, 4.52, D.75-D.79 etc.). Looking for example at Figure 4.43, one can notice that the MA values are varying periodically in form of waves and at an approximate frequency of 2 Hz. This reflects the same variation shown by the signal envelope in Figure 4.55. The index is indicating the existence of some repeatable oscillations. The author thinks the fault is inducing a low frequency engine imbalance/oscillation picked by the accelerometers. Similar observation can be made for Figures 4.47 and 4.48. The MA fluctuations occur at a frequency of 2Hz. Along this line, Figures 4.51,4.52 and 4.56 illustrate a similar type of variation but at lower frequency (1 Hz). The shape of this variation is identical to the time signal envelope shown in Figure 4.57 and to the *CI* signatures illustrated in Figures 4.24 and 4.25.

#### 4.2.2.2.3 Robustness with Respect to Load and Speed

In most of the cases illustrated below and in Appendix D, the *FFI* and its MA do not seem to be affected by the speed or load. No significant MA variation is observed when different speed and/or load are applied. A typical example is shown in Figures D.13-D.16. Even when the fused index shows a certain degree of sensitivity to these two parameters (e.g. Figure 4.45 and 4.46) the MA maintains at approximately identical level: the MA is 0.75 in Figure 4.45 and 0.8 in Figure 4.46, the variation as such is less than 5%. This characteristic makes the method robust and very suitable for dynamic applications.

### 4.2.3 Conclusions

A fuzzy fusion method based on the Sugeno-style process has been applied successfully to tool and bearing monitoring studies. The proposed *FFI* provides a clear distinction between normal and abnormal conditions and does not show a direct sensitivity to load or speed changes. In addition the *FFI* or its MA can be used to estimate the health degradation of the monitored process or machine. As all monitoring indices simultaneously contribute to the decision making process, fewer misclassifications can be expected. The maintenance decision can be made with much more confidence. The reduced number of thresholds also helps to reduce the effort in specifying their values and the subjectivity involved in the process.

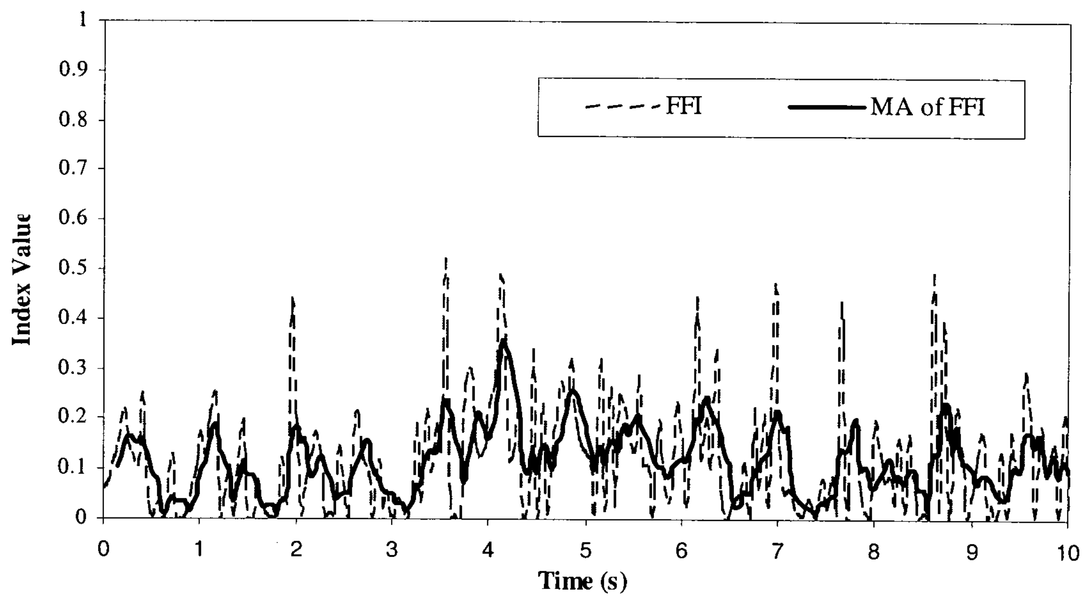



Figure 4.37 Fuzzy fused index and its moving average for drive end bearing, normal condition at 1797 rpm and 0 hp motor load.

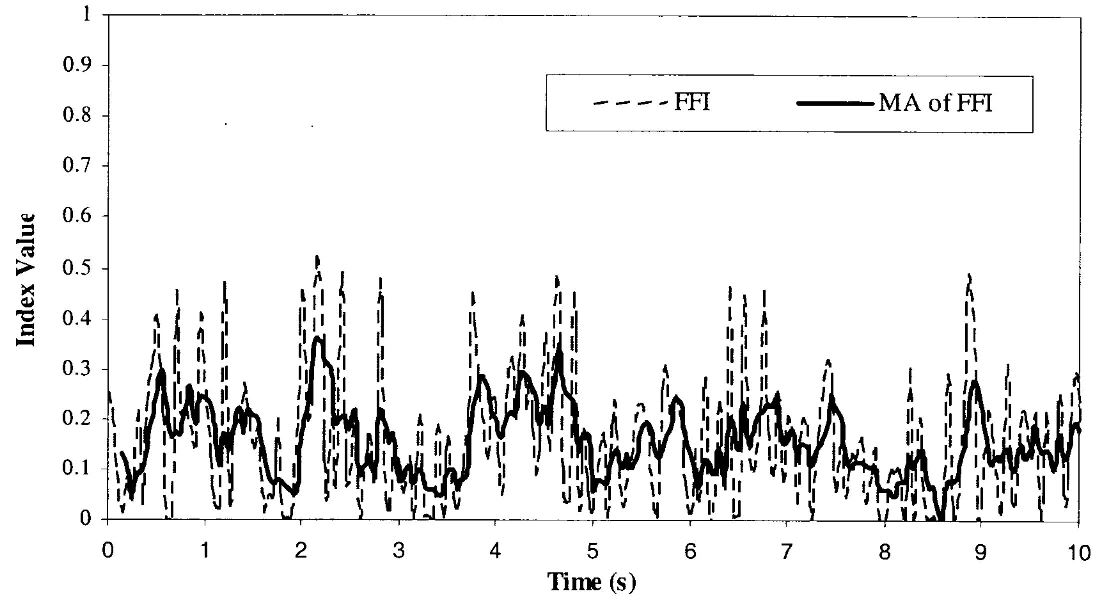



Figure 4.38 Fuzzy fused index and its moving average for drive end bearing, normal condition at 1730 rpm and 3 hp motor load.

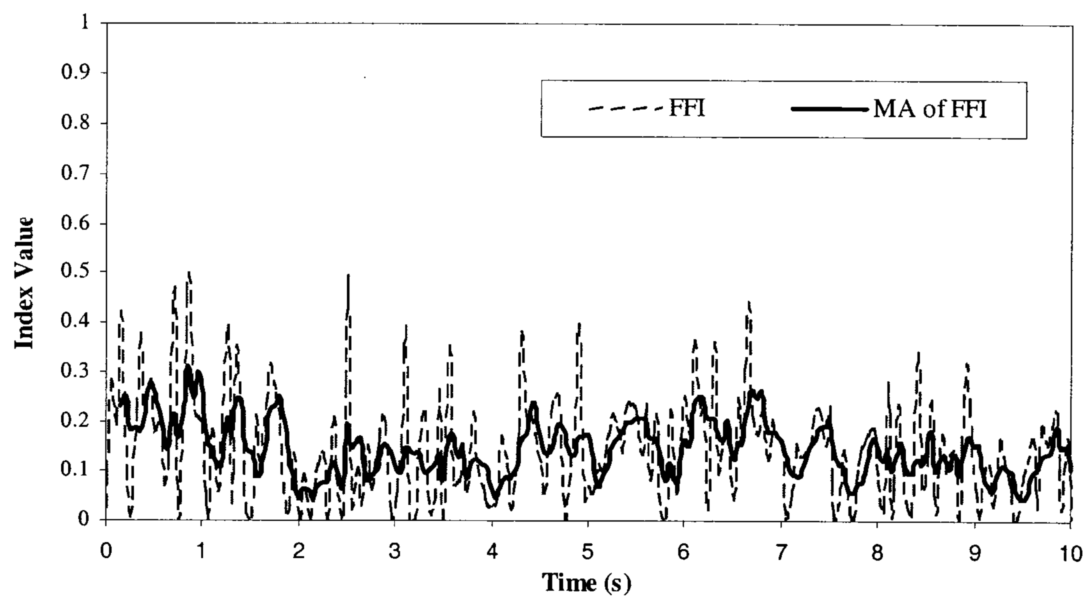



Figure 4.39 Fuzzy fused index and its moving average for fan end bearing, normal condition at 1797 rpm and 0 hp motor load.

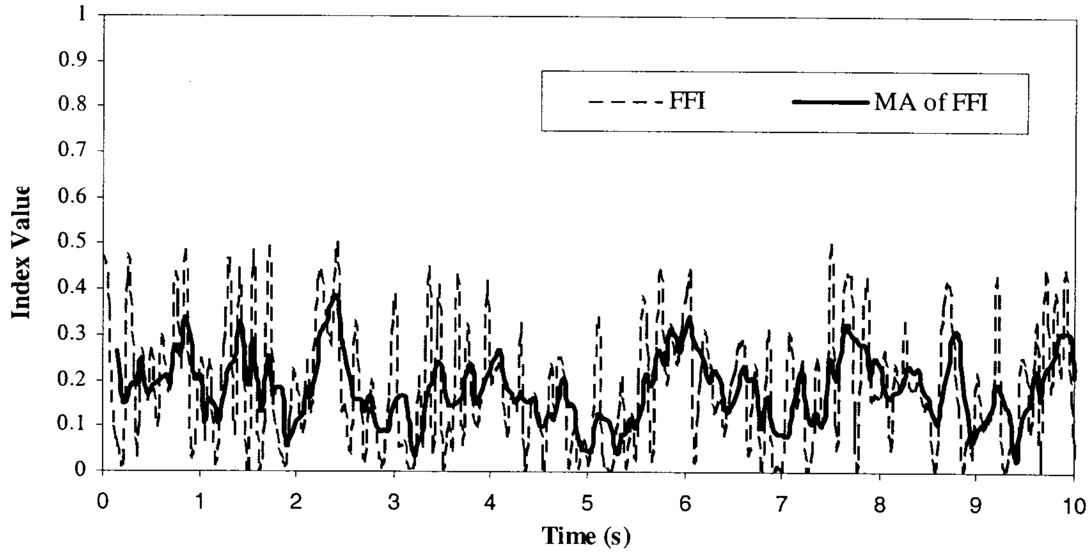



Figure 4.40 Fuzzy fused index and its moving average for fan end bearing, normal condition at 1730 rpm and 3 hp motor load.

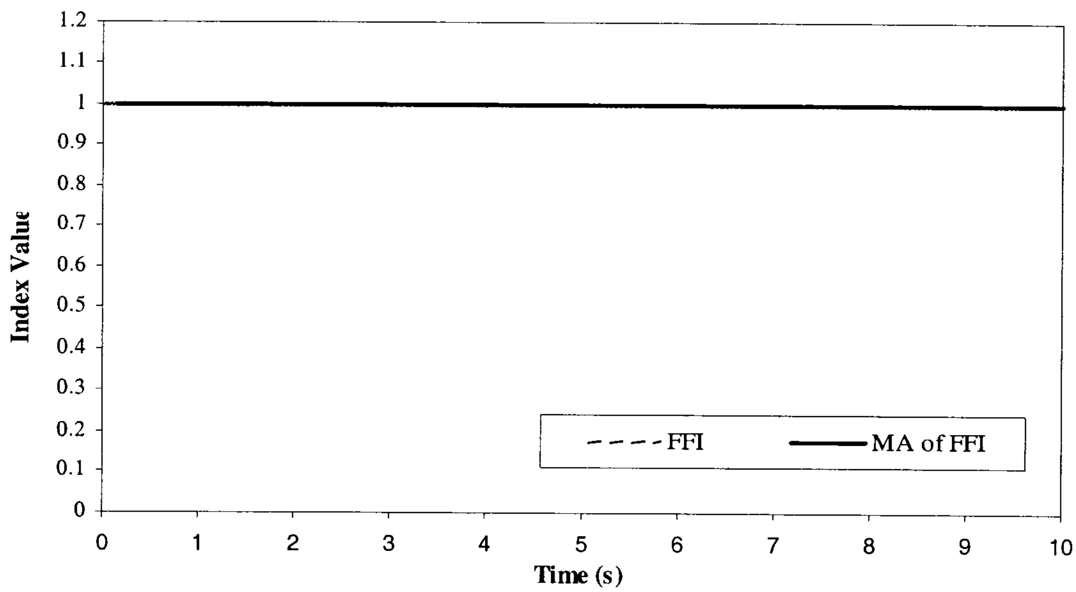



Figure 4.41 Fuzzy fused index and its moving average for drive end bearing, 0.007" fault diameter at inner race, 1797 rpm and 0 hp motor load.

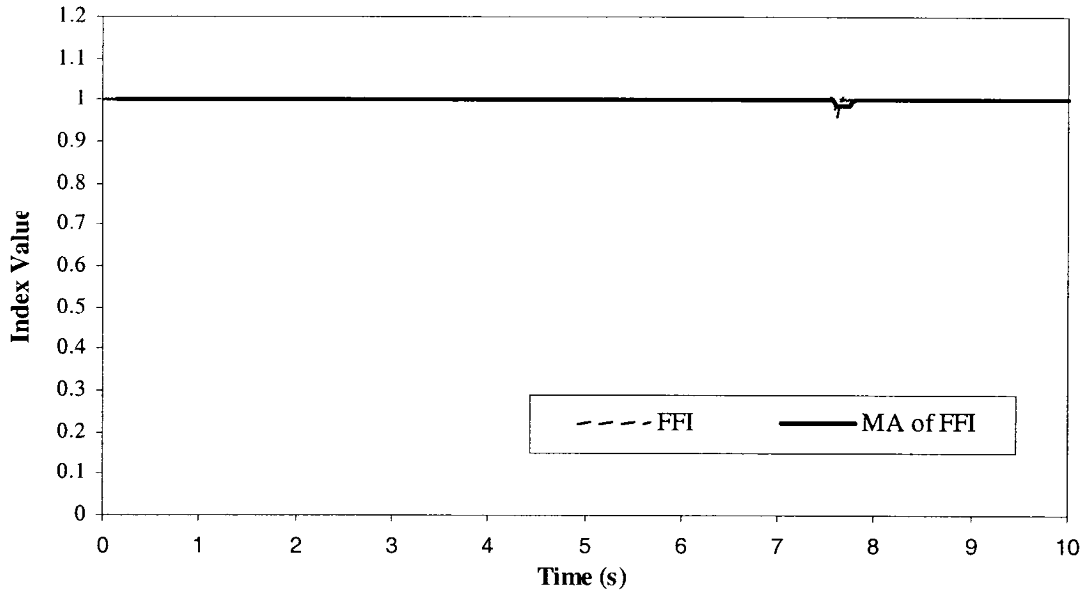



Figure 4.42 Fuzzy fused index and its moving average for drive end bearing, 0.007" fault diameter at inner race, 1730 rpm and 3 hp motor load.

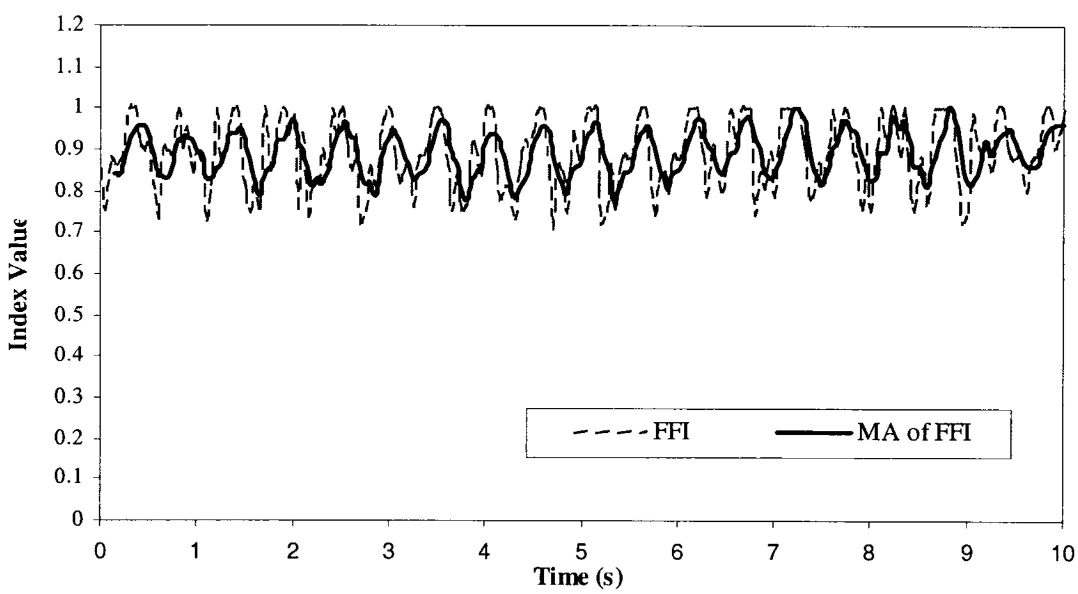



Figure 4.43 Fuzzy fused index and its moving average for fan end bearing, 0.007" fault diameter at inner race, 1797 rpm and 0 hp motor load.

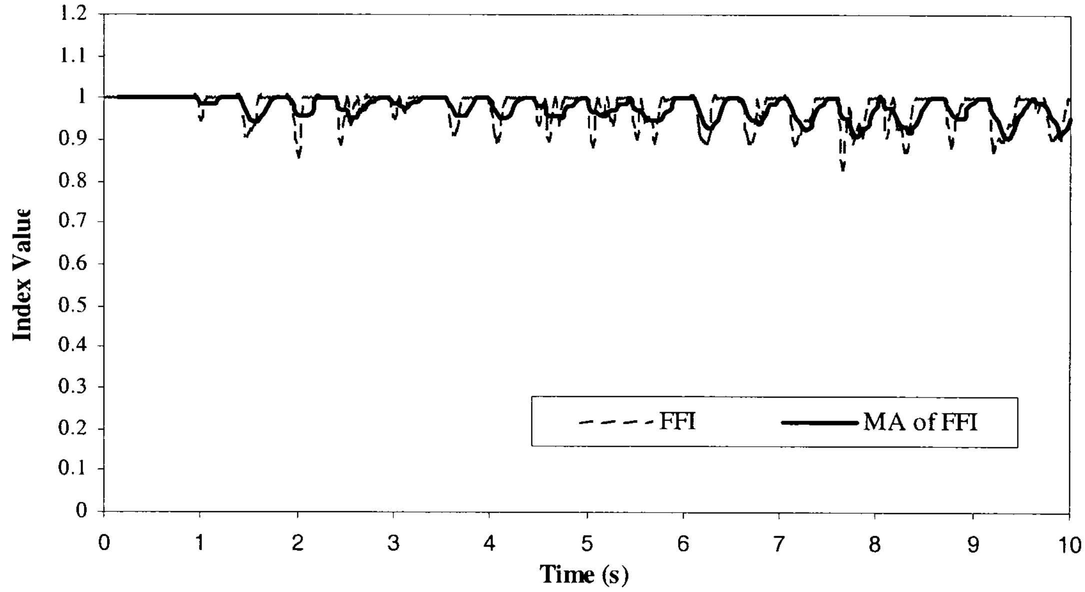



Figure 4.44 Fuzzy fused index and its moving average for fan end bearing, 0.007" fault diameter at inner race, 1730 rpm and 3 hp motor load.

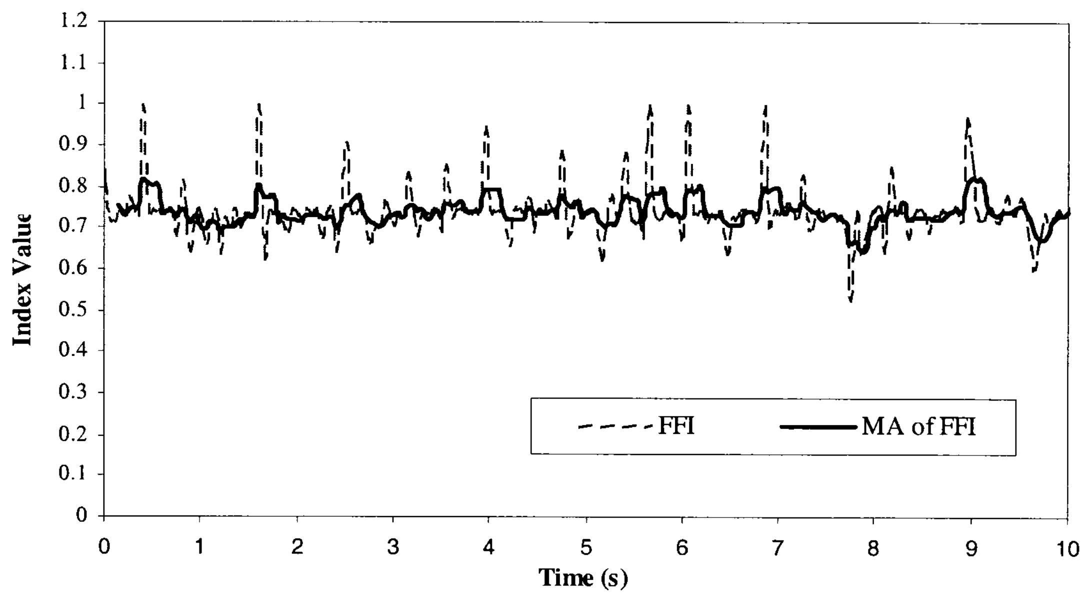



Figure 4.45 Fuzzy fused index and its moving average for drive end bearing, 0.014" fault diameter at inner race, 1797 rpm and 0 hp motor load.

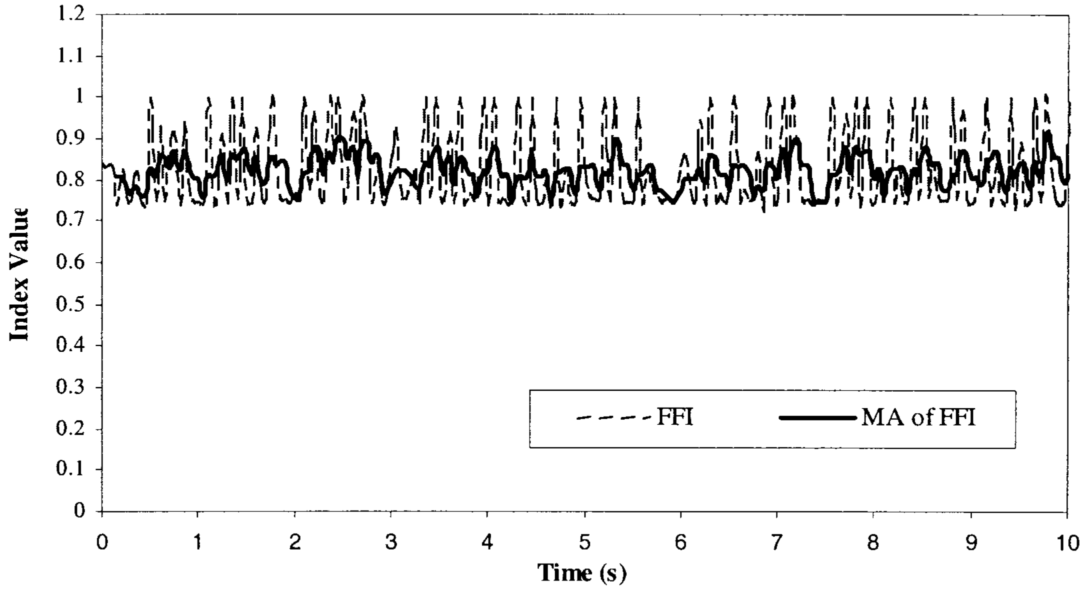



Figure 4.46 Fuzzy fused index and its moving average for drive end bearing, 0.014" fault diameter at inner race, 1730 rpm and 3 hp motor load.

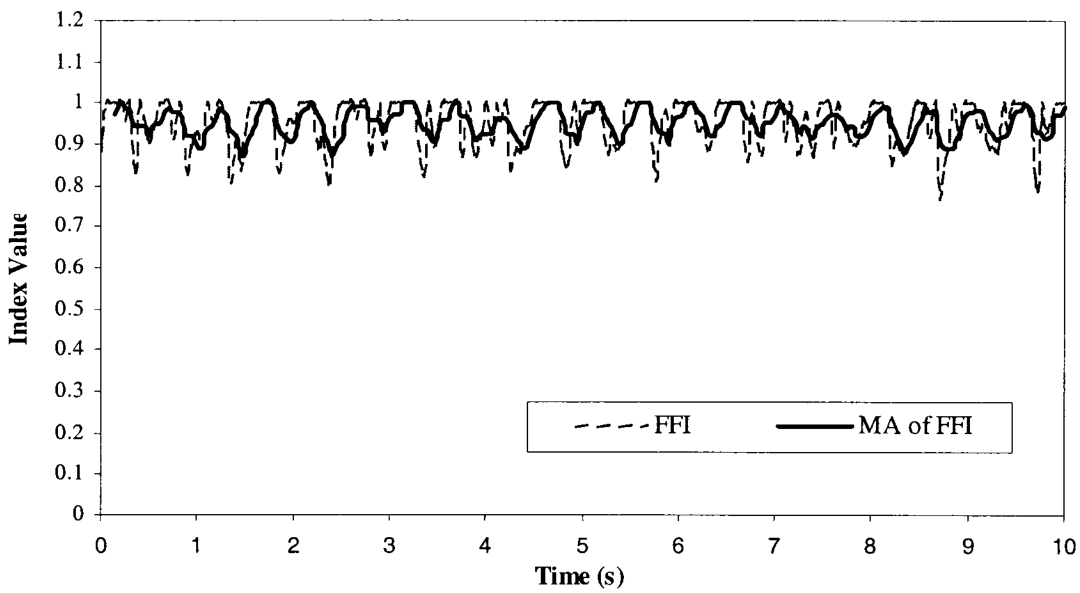



Figure 4.47 Fuzzy fused index and its moving average for fan end bearing, 0.014" fault diameter at inner race, 1797 rpm and 0 hp motor load.

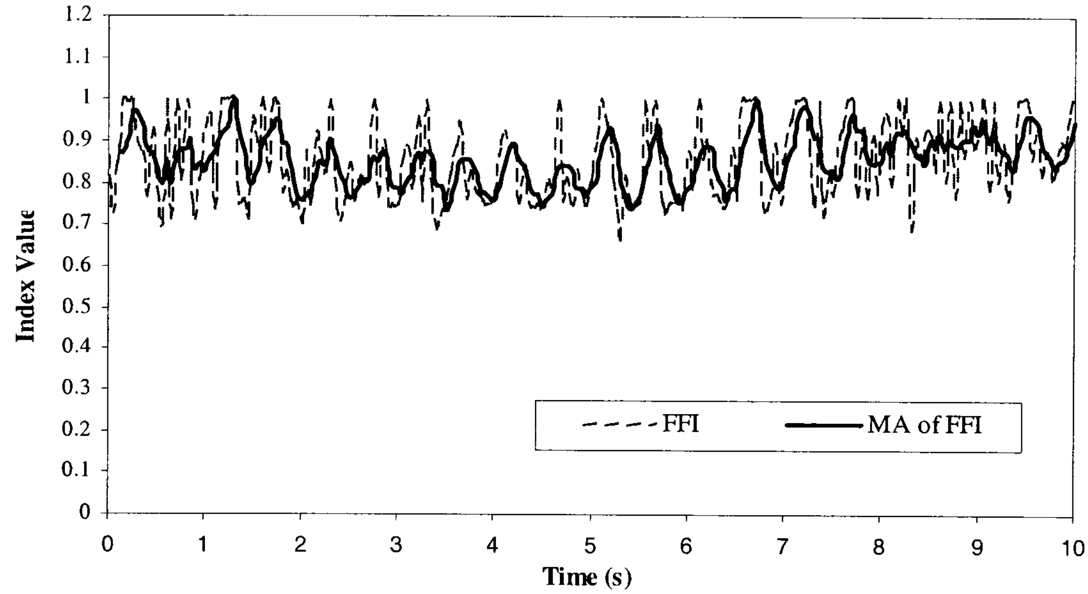



Figure 4.48 Fuzzy fused index and its moving average for fan end bearing, 0.014" fault diameter at inner race, 1730 rpm and 3 hp motor load.

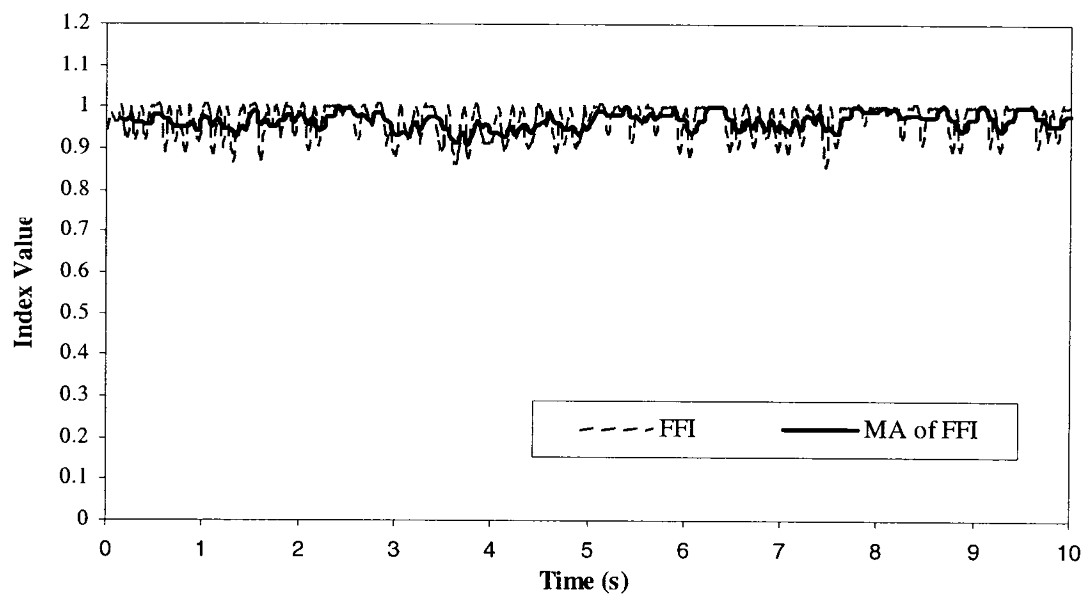



Figure 4.49 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at inner race, 1797 rpm and 0 hp motor load.

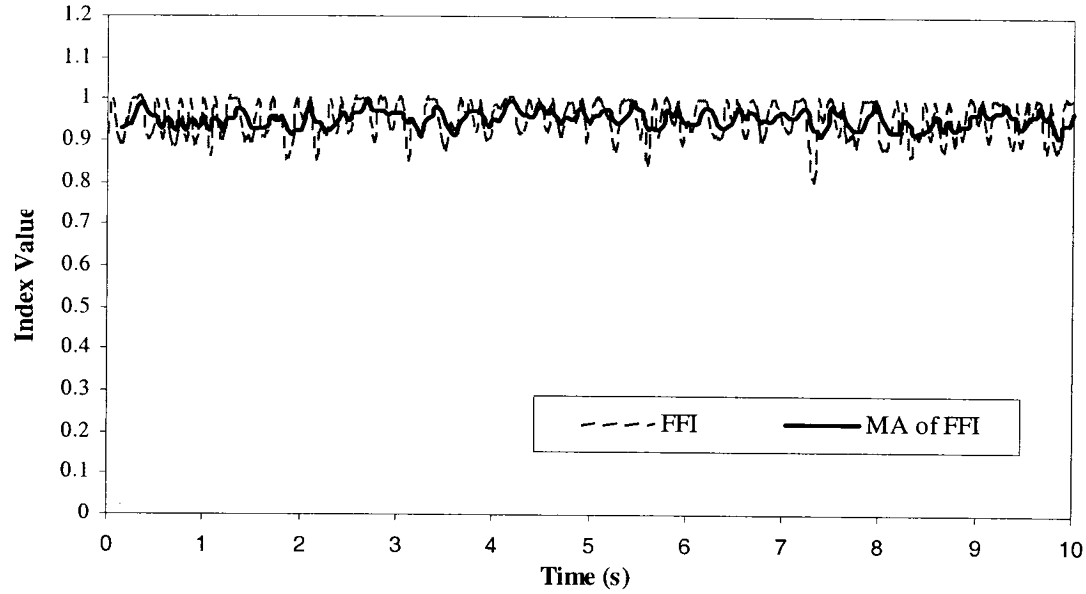



Figure 4.50 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at inner race, 1730 rpm and 3 hp motor load.

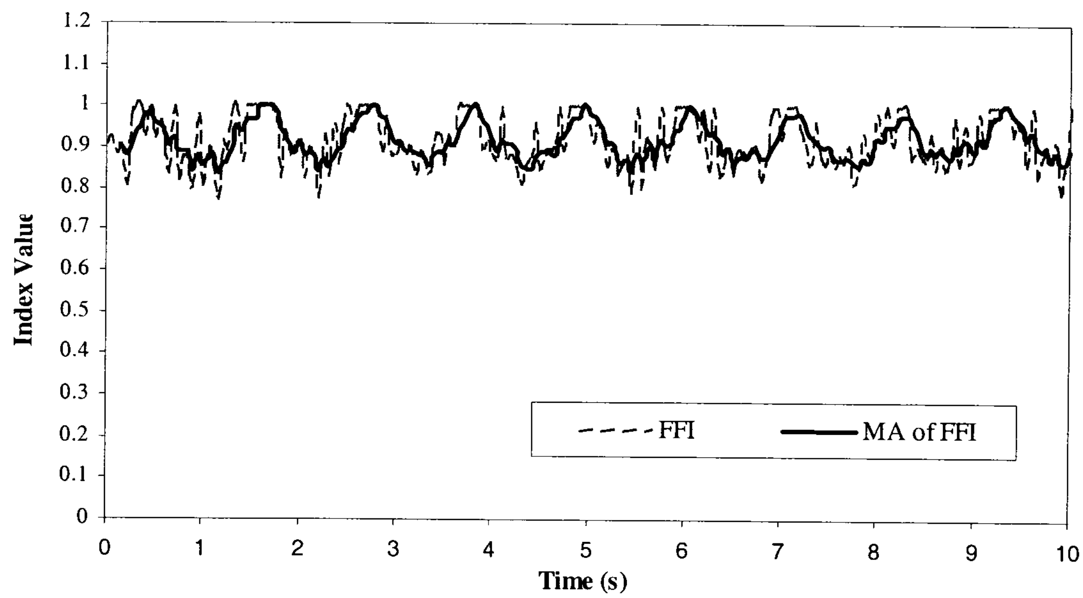



Figure 4.51 Fuzzy fused index and its moving average for fan end bearing, 0.021" fault diameter at inner race, 1797 rpm and 0 hp motor load.

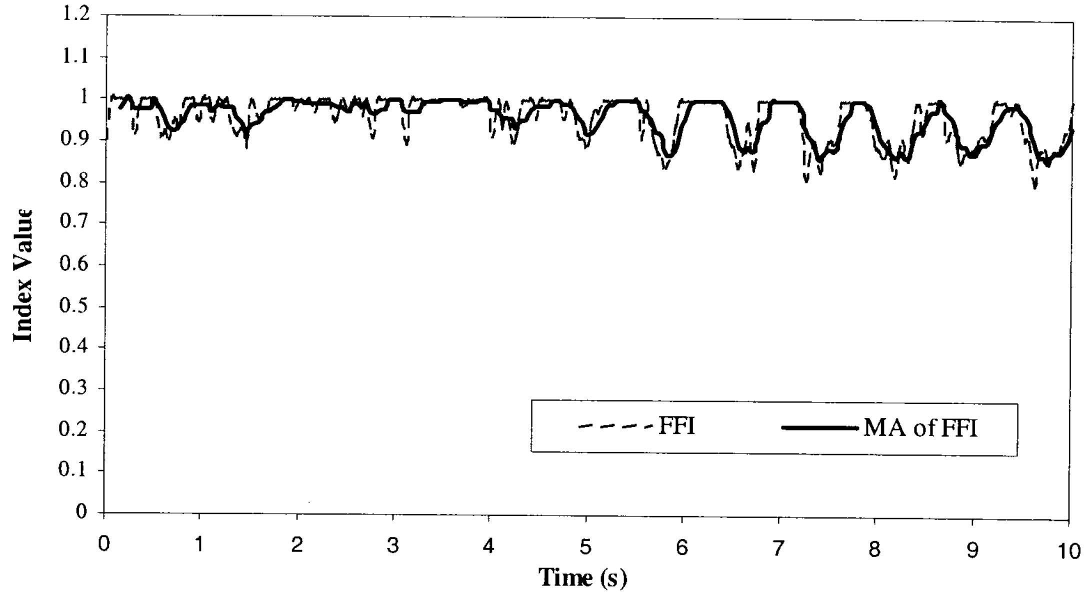



Figure 4.52 Fuzzy fused index and its moving average for fan end bearing, 0.021" fault diameter at inner race, 1730 rpm and 3 hp motor load.

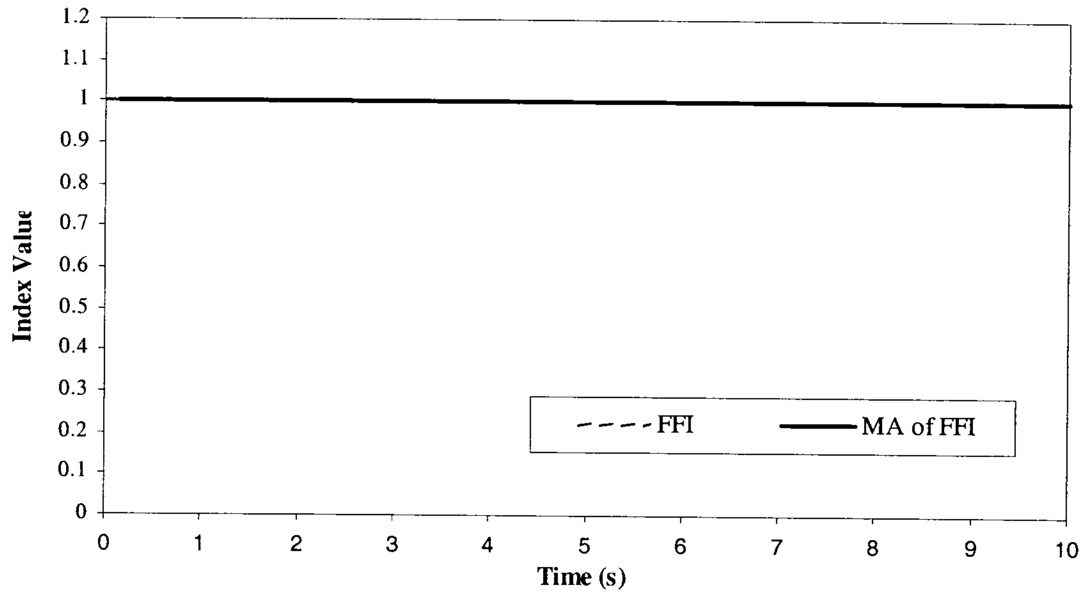



Figure 4.53 Fuzzy fused index and its moving average for drive end bearing, 0.028" fault diameter at inner race, 1797 rpm and 0 hp motor load.

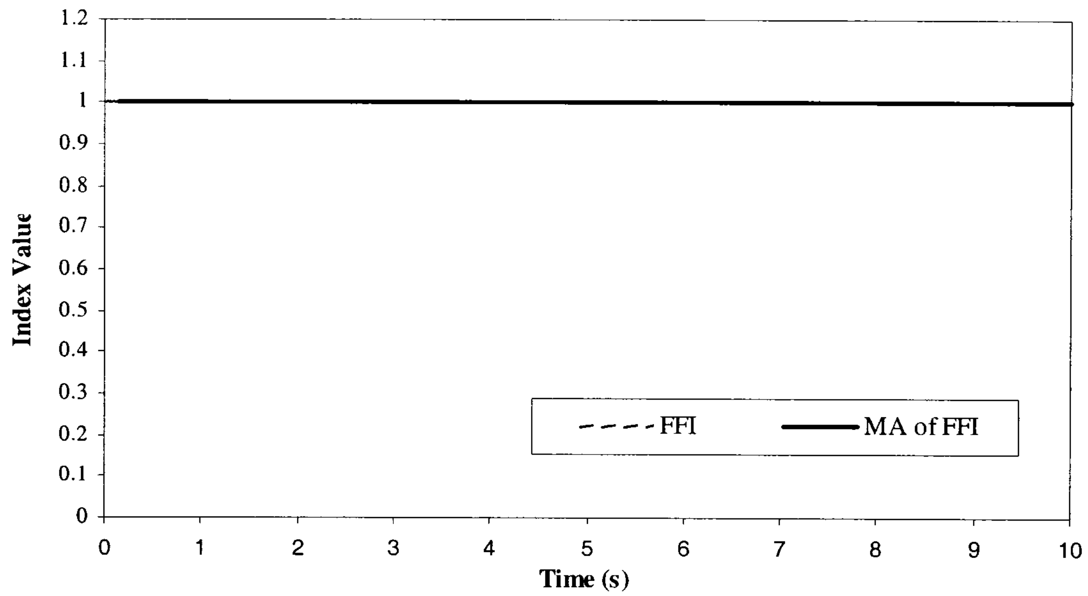



Figure 4.54 Fuzzy fused index and its moving average for drive end bearing, 0.028" fault diameter at inner race, 1730 rpm and 3 hp motor load.

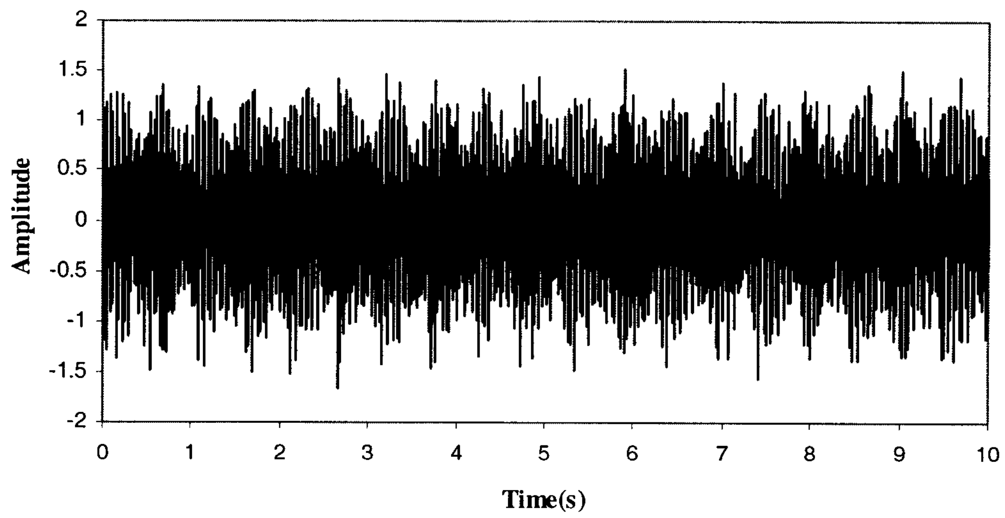



Figure 4.55 Filtered signal for fan end bearing, 0.007" fault diameter at inner race, 1797 rpm and 0 hp motor load.

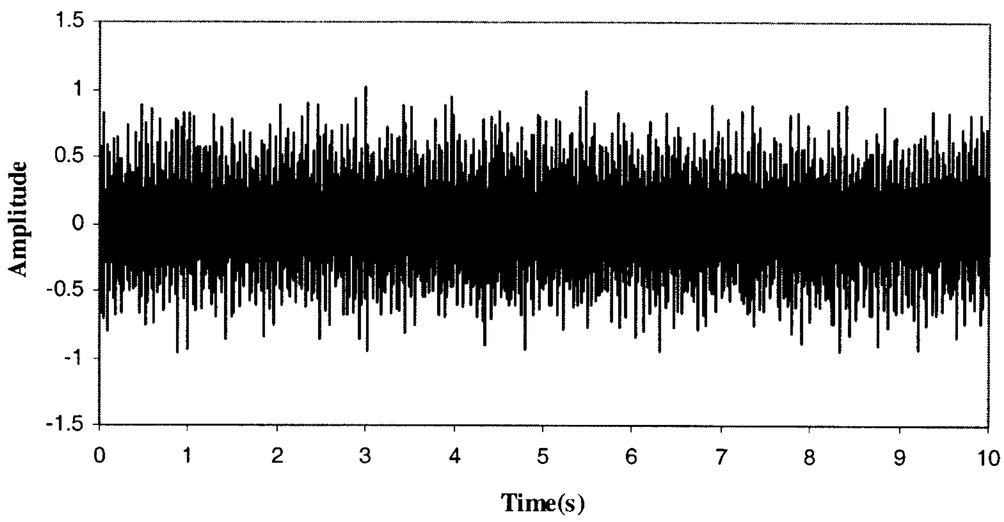



Figure 4.56 Filtered signal for fan end bearing, 0.014" fault diameter at inner race, 1797 rpm and 0 hp motor load.

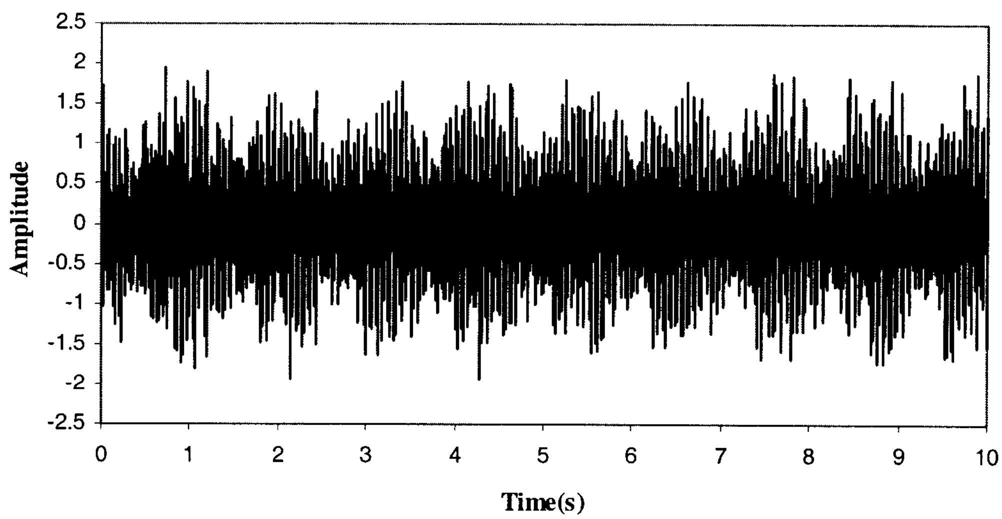



Figure 4.57 Filtered signal for fan end bearing, 0.021" fault diameter at inner race, 1797 rpm and 0 hp motor load.

# Chapter 5

## Fault Diagnosis Using Hidden Markov Models

Human being is probably the best source of intelligent sensors (e.g. ears, eyes). It is always desirable to emulate the knowledge embedded in a human brain. For example, an experienced machinist is capable to classify anomalies based on "trained data" stored in his brain. Through his years of experience, the machinist is able to gather acoustic data, set thresholds for normal operations and store information in his brain. During a machining operation, the acoustic signal produced by the machine is processed and compared to a set of "reference" thresholds before taking a decision about the status of the whole system as shown in Figure 5.1: normal or abnormal.

This technique is similar to the one used in speech pattern recognition (refer to Figure 5.2), particularly the Hidden Markov Model (HMM) that has been a dominant method since 1960s. HMM is an extension to Markov chains. It is a doubly embedded stochastic process; that is not directly observable (hidden) but can be observed only through another set of stochastic processes that produce a sequence of observations.

In Chapter 3, a monitoring technique and some condition indicators were developed to detect transient and continuous anomalies. These condition indicators were fused in Chapter 4 to improve the decision-making and confirm the state of the machine or process under monitoring, i.e. normal or abnormal. In addition, an index of deterioration was developed to monitor the performance degradation of the system in question. However, the methods described in the previous chapters are limited to fault detection only. In many applications (e.g. gas turbine in a power plant), the fault detection requirement becomes insufficient as the maintenance department is aiming at optimizing the equipment availability while keeping the cost low. Therefore, fault diagnosis becomes crucial for decision-making. A system capable of identifying fault severity and location would reduce the uncertainty and hence improve maintenance decision quality substantially. For this reason, a discrete HMM module is introduced to boost system diagnosis ability.

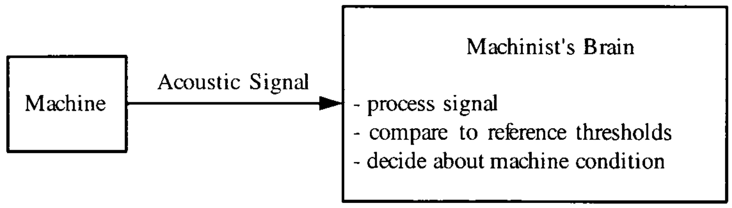



Figure 5.1 Monitoring technique used by an experienced machinist.

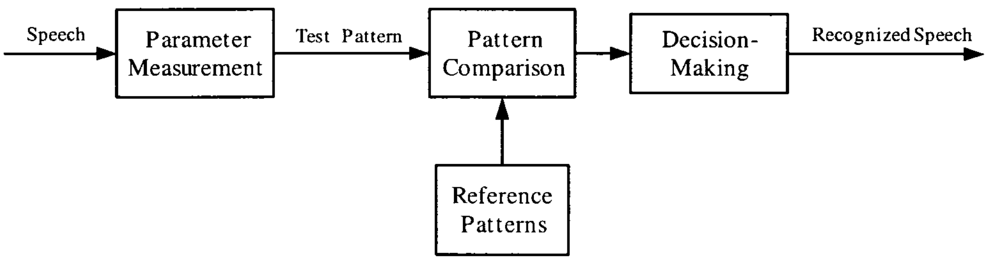



Figure 5.2 Speech pattern recognition process.

## 5.1 The Hidden Markov Model Approach

Before detailing the HMM procedures, let's illustrate the concept of this approach using a simple example. If one lives beside a football field surrounded by concrete walls so he cannot "observe" from outside ("hidden") what the "state" of the field is: "empty" or "occupied". The "transition" between these two states can occur at any "time *t*" of the day. For example, if the field is "empty" at "time *t*" it may become "occupied" at "time *t*+1". To determine the field "state", he observed the "the number of cars" in the parking lot. This "observation" collected at "different time" of the "day" gave him an idea about the nature of the traffic around the field so he could "learn" how to estimate the "probable" "state" in the "next day", based on "similar observations".

In order to "train himself", he stood one day at the entrance of the field to check the field state and at the same time he conducted a "sequence of observations" at "every hour" and recorded the results in the following table.

| Time (hr) | 8:00 | 9:00 | 10:00 | 11:00 | 12:00 | 13:00 | 14:00 | 15:00 | 16:00 | 17:00 |
|---|---|---|---|---|---|---|---|---|---|---|
| Number of cars | <10 | <10 | <10 | >10 | <10 | >10 | >10 | >10 | >10 | <10 |
| Field State | empty | empty | empty | occupied | empty | occupied | occupied | occupied | occupied | empty |

He realized that the field was "initially" "empty" in the morning. He also took note of the "state transition" during the day. He presented his "observations" in the table using "symbols" (i.e. "number of cars" <10 or > 10). After this "training day", he established an idea about this problem. He could estimate for example the probability of having a "number of cars" less than ten when the field is "occupied" (assume probability equal to 0.1). Knowing that the field was "occupied" at time "12:00" of the day, he could also "estimate" the "probability" of having the field "empty" at "13:00" (assume probability equal to 0.5).

The HMM approach is very similar to what has been described in the above example. It is a doubly stochastic technique that uses the observation probabilities (i.e. probability of being in a state $\bar{j}$ for example and observing observation $\bar{k}$) in addition to the state transition probabilities (i.e. Markov chain: probability to switch from state $\bar{i}$ to state $\bar{j}$) to define the condition of the system in question. Using the same logic described in the above example, a machine or process has a number of failure modes that can be used to define the states of the system. Suppose there are three states which describe the tool conditions (sharp, worn, broken) associated with $\bar{i}$=1,2,3 respectively and three observations ($o$) of acoustic power vectors represented by the following symbols (low, medium, high) corresponding to $\bar{k}$=1,2,3.

Assume also that through a training exercise, the state probability matrix ($\overline{A} = \{a_{\bar{i}\bar{j}}\}$, i.e. probability of switching from state $\bar{i}$ at time $t$-1 to state $\bar{j}$ at time $t$) and the observation probability matrix ($\overline{B} = \{b_{\bar{j}}(\bar{k})\}$, i.e. probability of observing symbol $\bar{k}$ while being in state $\bar{j}$) are defined as follows:

| $a_{\bar{i}\bar{j}}$ | | State | | |
|---|---|---|---|---|
| | | Normal | Worn | Broken |
| State | Normal | 0.80 | 0.15 | 0.01 |
| | Worn | 0.00 | 0.90 | 0.10 |
| | Broken | 0.00 | 0.00 | 1.00 |

| $b_{\bar{j}}(\bar{k})$ | | Observed Power Symbol | | |
|---|---|---|---|---|
| | | Low | Medium | High |
| State | Normal | 0.90 | 0.08 | 0.02 |
| | Worn | 0.08 | 0.85 | 0.07 |
| | Broken | 0.04 | 0.06 | 0.90 |

Denoting by $\overline{q_t}$ the state at time $t$, and knowing that the tool was sharp at initial time ($t$=0), one can estimate for example the probability of the tool staying sharp during a sequence of three observations (low, medium, high) collected at time $t$=1,2,3. This can be translated in HMM terminology as:

$P(\overline{q_1}$ =sharp, $\overline{q_2}$ =sharp, $\overline{q_3}$ =sharp/$o_1$=low, $o_2$=medium, $o_3$=high) =

$P(o_1$=low/$\overline{q_1}$ =sharp)$\cdot P(o_2$=medium/$\overline{q_2}$ =sharp) $\cdot P(o_3$=high/$\overline{q_3}$ =sharp) $\cdot P(\overline{q_1}$ =sharp)

$\cdot P(\overline{q_2}$ =sharp/$\overline{q_1}$ =sharp) $\cdot P(\overline{q_3}$ =sharp/$\overline{q_2}$ =sharp) = $b_1(1)\cdot b_1(2)\cdot b_1(3)\cdot a_{11}\cdot a_{11}\cdot a_{11}$ = 0.90 $\times$0.08$\times$0.02$\times$0.80$\times$0.80$\times$0.80=0.000737.

Similarly, one can compute the probability for the following state transition (sharp → worn → broken) when observing the same sequence of observations (low, medium, high):

$P(\overline{q_1}$ =sharp, $\overline{q_2}$ =worn, $\overline{q_3}$ =broken/$o_1$=low, $o_2$=medium, $o_3$=high) =

$P(o_1$=low/$\overline{q_1}$ =sharp) $\cdot P(o_2$=medium/$\overline{q_2}$ =worn) $\cdot P(o_3$=high/$\overline{q_3}$ =broken)$\cdot P(\overline{q_1}$ =sharp) $\cdot$

$P(\overline{q_2}$ =worn/$\overline{q_1}$ =sharp) $\cdot P(\overline{q_3}$ =broken/$\overline{q_2}$ =worn) = $b_1(1)\cdot b_2(2)\cdot b_3(3)\cdot a_{11}\cdot a_{12}\cdot a_{23}$ =

0.95$\times$0.85$\times$0.90$\times$0.80$\times$0.15$\times$0.10= 0.00872.

Comparing the above two probabilities (0.000737 and 0.00872), one can notice that the first transition sequence is probably not correct as it has the lowest probability. The tool is more probably broken and not sharp at the end of the observation sequence.

The above calculations mean in summary, that the state of the tool is "hidden" during machining, i.e., the observation (acoustic power level) does not directly tell the state with 100% certainty. Therefore, the HMM technique employs the sequence of observations and the associated probabilities to determine the most probable states sequence.

Based on above illustrations, the method consists of two stages as shown in Figure 5.3: training and diagnosis. Before describing these steps, a short definition of the type and elements of the model is made below.

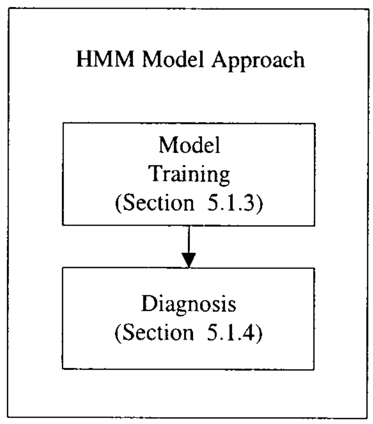



Figure 5.3 HMM main steps description.

### 5.1.1 Type of HMM To Be Used

Two types of HMM exist: continuous and discrete. In the continuous model, the observations are assumed to follow a Gaussian probability density function. However, in the discrete model the observations are obtained directly from the real data. Although, in the discrete model there is a minor loss of information due to sampling, this model can still provide accurate results (Rabiner and Juang 1993). For the pre-listed reasons a discrete model is proposed in this study.

The discrete model can be also ergodic (each state is reachable from each and every other state) or left-right (named also Bakis) (i.e. transition flow occurring in one direction) (Jelinek 1976 and Rabiner 1989).

### 5.1.2 Elements of HMM

Elements of the HMM include the following:

(1) $N$, the number of states in the model. These states have some real meaning in this study (e.g. wear, breakage, etc.) and are defined based on the user's interest to study some specific failure modes. The individual states are denoted at time $t$ as $\overline{q_t}$ and labelled as {1, 2, ..., $N$} (e.g. $N$=3 in the above illustration: "sharp", "worn" and "breakage").

(2) $M$, the number of distinct observation symbols per state: the observation symbols correspond to the physical output of the system being modeled. The individual symbols are denoted as $\overline{V} = \left\{ \overline{v_1}, \overline{v_2}, ..., \overline{v_M} \right\}$ (e.g. $M$=3 in the above illustration: "low", "medium" and "high"). The user needs to keep the list of symbols as short as possible to reduce the computation time.

(3) $\overline{A} = \left\{ a_{\bar{i}\bar{j}} \right\}$, the state transition probability distribution (e.g. matrix shown in the machining tool example above) where

$$a_{\bar{i}\bar{j}} = P\left[ \overline{q_{t+1}} = \bar{j} / \overline{q_t} = \bar{i} \right] \quad 1 \le \bar{i}, \bar{j} \le N \tag{5.1}$$

with the following properties:

$$a_{\bar{i}\bar{j}} \ge 0 \quad \forall \bar{i}, \bar{j} \tag{5.2}$$

$$\sum_{\bar{j}=1}^{N} a_{\bar{i}\bar{j}} = 1 \quad \forall \bar{i} \tag{5.3}$$

where $\bar{i}$ and $\bar{j}$ denote the states indexes.

(4) $\overline{B} = \left\{ b_{\bar{j}}(\bar{k}) \right\}$, the observation symbol probability distribution (e.g. matrix illustrated in the machining tool example above), in which,

$$b_{\bar{j}}(\bar{k}) = P\left[ o_t = v_{\bar{k}} / \overline{q_t} = \bar{j} \right] \quad 1 \le \bar{k} \le M \tag{5.4}$$

defines the symbol distribution in state $\bar{j}$, $\bar{j} = 1,2,...,N$. In addition, $o_t$ in the above equation is the observation at time $t$.

Similar to $a_{\bar{i}\bar{j}}$, $b_{\bar{j}}(\bar{k})$ has the following property

$$\sum_{\bar{k}}^{M} b_{\bar{j}}(\bar{k}) = 1 \quad \forall \bar{j} \tag{5.5}$$

(5) $\bar{\pi} = \{\bar{\pi}_i\}$, the initial state distribution (e.g., initial state is "sharp" in the machining tool example), in which

$$\bar{\pi}_i = P\left[\ \bar{q}_1 = \bar{i}\ \right] \quad 1 \le \bar{i} \le N \tag{5.6}$$

In summary, a complete specification of an HMM requires specification of two model parameters, $N$ and $M$, specification of observation symbols, and the specification of the three sets of probability measures: $\bar{A}$, $\bar{B}$, $\bar{\pi}$. The model is noted as:

$$\lambda = \left(\bar{A}, \bar{B}, \bar{\pi}\right) \tag{5.7}$$

This parameter set defines a probability measure for a vector of observation $O$, i.e., $P(O/\lambda)$, where $O$ is an observation sequence

$$O = \left(o_1, o_2, \ldots, o_{T_o}\right) \tag{5.8}$$

in which each observation $o_t$ is one of the symbols from $\bar{V}$ and $T_o$ is the number of observations in the sequence.

Three assumptions are associated with the use of the HMM theory (Ertunc *et al* 2001):

(1) The probability of being in a state at a given instant of time $t$ depends only on the state at the previous instant of time $t$-1. This is known as Markov assumption and is described by equation (5.1).

(2) The state transition probabilities are independent of the actual time at which the transition occurs. This is called the stationary assumption.

(3) The current observation depends only on the current state and it is statistically independent of the previous observations.

The first assumption can be justified, as the main monitoring purpose is to define the current state based on the one determined at previous time, which reflects the accumulative effects of all previous states. Therefore, it may not be necessary to track the states found at earlier stages. For example, if the tool is normal at time $t$ and worn at time $t$+1, the information recorded at time earlier than $t$ is not necessary for the monitoring decision.

The second assumption means that the state transition probabilities follow an exponential distribution (memory-less). In fact, most of the rotating equipment and machining tools follow a Weibull probability distribution. However, if the slope or shape factor (beta) in such distributions is equal to one (1), they become exponential. Harris (1991) has shown that for rolling bearings and based on real experimental data, the slope varies between 0.9 and 1.1. Similarly, Shigley (1977) has employed a slope value of 1.17. Based on these facts, the assumption of using an exponential distribution is quite valid. Along the same line, Levi and Rossetto (1975) estimate that the tool life follows an exponential distribution if the sudden failure such as fracture, and chipping is the main reason and normal distribution if wear is the main failure.

Another important issue that can be added to support the use of exponential assumption is that the HMM is an adaptive method capable of alleviating or tolerating the partial non-conformance to such assumption through training. In fact, the states transition probabilities are defined and adjusted through training.

Regarding the third assumption, the observation such as the strength of the acoustic power vector taken at time $t$ depends mainly on the state at that exact time. For example, our extensive experiments have shown that the "high" power vector observation is indeed mainly related to the current state: "broken" regardless whether the previous state is "sharp", "worn" or "broken". Similarly, the "medium" power vector observation is largely a result of a current "worn" state. Hence, justification of the third assumption is practically acceptable in this study.

### 5.1.3 Model Training

Similar to other pattern recognition techniques (e.g. template matching, neural network, etc.), the HMM needs to be trained to generate accurate and reliable results. The training occurs at different stages as illustrated in Figure 5.4 and described below. It consists of preparing the codebook (library where unique symbols or reference vectors are stored) and estimating the model parameters ($\overline{\pi,A},\overline{B}$) prior to diagnosis.

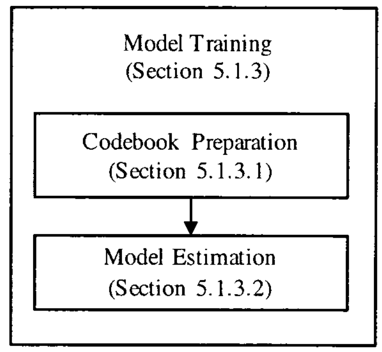



Figure 5.4 Model training process description.

### 5.1.3.1 "Codebook" Preparation

The codebook contains all the unique symbols or reference vectors that characterize the states of the process or machine in question (e.g. wear, breakage, etc.). It is prepared for matching purpose, i.e. each observation is compared to these symbols to find the closest one. The codebook is prepared through different steps as illustrated in Figure 5.5 and described below.

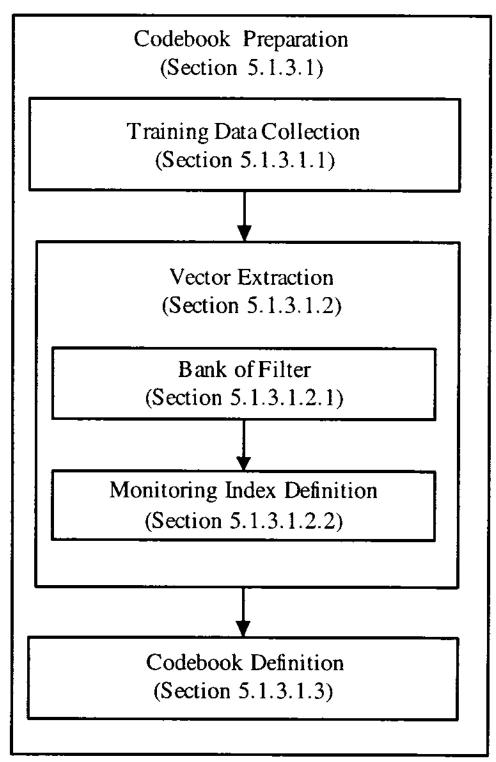



Figure 5.5 Codebook generation method.

#### 5.1.3.1.1 Training Data Collection

Data for each of the state symptoms (e.g. sharp, wear, fracture, bearing fault etc.) is collected to train the model. This is the raw data gathered directly from the sensors in use (e.g. acoustic or vibration data).

#### 5.1.3.1.2 Vector Extraction

This step is required to extract the necessary information from the raw data to define the unique pattern or symbol for each state. For this reason, the training data collected in the previous step needs to be processed. The signal processing consists of several steps as shown in Figure 5.5 and described below.

##### 5.1.3.1.2.1 Bank of Filters

The bank of filters is used to study the frequency content of the signal at different spots of the spectrum. The structure of the filters bank is shown in Figure 5.6. The sampled signal $x(n)$ is subjected to a set of bandpass filters structured in parallel. There is no specific rule defining how many filters are required. Increasing the number of filters improves the frequency resolution but on the other hand requires significantly more computation time. The choice of the cut-off frequencies is based on the regions of interests defined in Chapter 3. These regions of interest indicate the spot where the most useful information is lying. For a bank of $Z$ filters, the signal becomes (Rabiner and Juang 1993):

$$x_l(n) = x(n) * h_l(n) \quad 1 \le l \le Z \tag{5.9}$$

$$x_l(n) = \sum_{m}^{L-1} h_l(m)x(n-m) \tag{5.10}$$

where $h_l(m)$ is assumed the impulse response of the $l^{th}$ bandpass filter with a duration of $L$ samples (chosen as a multiplier of the number of samples in a sub-group). The size of $L$ defines the size of the window used below.

The bandwidths of the filters are chosen in such a way to emphasize the desired frequency range. In order to reduce the computational requirement of equation (5.10), each bandpass filter impulse response can be represented as a fixed lowpass window, $w(n)$ modulated by the complex exponential $e^{jw_l n}$, i.e.:

$$h_l(n) = w(n)e^{jw_l n} \tag{5.11}$$

where $w_l = 2\pi f_l$ and $f_l$ is the center frequency of bandpass filter $l$. In this case, equation (5.10) becomes:

$$x_l(n) = \sum_{m=0}^{L-1} w(m)e^{jw_l m} x(n-m) \qquad 0 \le n \le L-1 \tag{5.12}$$

Looking at equation (5.12) and as explained previously $L$ defines the size of the window or observation. There is no simple theoretical way to determine the optimum window size. The choice is dependent upon type of problem and signal characteristics (e.g. sampling frequency). With short window (equivalent to a maximum of 30 ms as proposed by Rabiner and Schafer (1978)), the resolution is poor but a good estimate of the gross spectral shape is obtained. On the other hand, with relatively large window, the frequency resolution is much better and the computation time is much shorter.

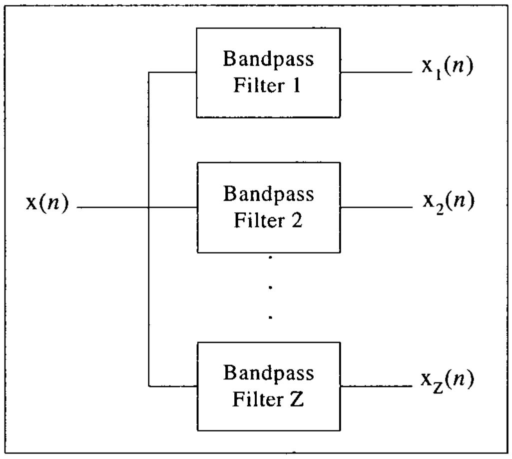



Figure 5.6 Bank filters description.

An alternative form of equation (5.12) is obtained by change of summation index, which yields the expression:

$$x_l(n) = \sum_{m=0}^{L-1} x(n)w(n-m)e^{jw_l(n-m)} = e^{jw_l n}\sum_{m=0}^{L-1} x(n)w(n-m)e^{-jw_l m}$$

$$= e^{jw_l n} X_n(e^{jw_l}) \tag{5.13}$$

where $X_n(e^{jw_l}) = \sum_{m=0}^{L-1} x(n)w(n-m)e^{-jw_l m}$ (5.14)

$X_n(e^{jw_l})$ is the short-time Fourier transform of $x(n)$ at frequency $w_l$. It is noted that for a fixed $n = n_0$, $X_n(e^{jw_l})$ denotes the conventional Fourier transform of the windowed signal $x(m)w(n_0 - m)$ evaluated at frequency $w_l$. It is also important to note that if $L$ is large, relative to signal periodicity, $X_n(e^{jw_l})$ gives good frequency resolution but produces rough overall spectral envelope of signal within the window. However, if $L$ is small, $X_n(e^{jw_l})$ gives poor frequency resolution, but a good estimate of the gross spectral shape is obtained (Rabiner and Juang 1993, Rabiner and Schafer 1978).

#### 5.1.3.1.2.2 Monitoring Index Vector

The next step prior to codebook generation is the calculation of the monitoring index vector. This monitoring index is selected based on its sensitivity to reflect any possible changes in the performance of the monitored system. In Chapter 4, a fused fuzzy index (*FFI*) is developed based on the four condition indicators defined in Chapter 3. However, as some of the condition indicators (e.g., the correlation factors) are not sensitive to the frequency content of the signal obtained from the bank of filters, the *FFI* cannot lead to the best solution. On the other hand, the power index is very effective in reflecting the signal change in all of the frequency ranges. Hence, the power is employed in this study to form the monitoring index vector. Based on equation (5.13), the size of the monitoring index vector is equal to the number of filters used in the bank of filters (*Z*).

To normalize this vector, each component is divided first by the sum of all components and then by the maximum component obtained as a result of the first step. The first step gives an idea about the distribution of the monitoring index vector over the whole desired frequency spectrum and the second step highlights the maximum value and provides a vector with components less or equal to one. As a result of that, the monitoring index vector (that can be used as observation) becomes independent of the signal magnitude. The vector obtained through this computation is a "unique" vector that corresponds to some characteristics of the machinery or process states. This vector noted by *MIV*(*l*) where $l = 1,2,...,Z$, will be used below to define the "codebook". To illustrate the process, let's take a numerical example. Assume that the regions of interest are: (1000-2000 Hz), (2000-3000 Hz) and (3000-4000 Hz). Therefore, three bandwidth filters are required ($Z$=3). Assume also that for a given input data, the power is calculated in these regions of interests and computation leads to a vector with the following components (22, 33, 15). By summing the components of this vector (22+33+15=70), one can have an idea about the total power carried by the input data. Dividing each component of the power vector by the sum (70) leads to the following results: (22/70=0.31, 33/70=0.47, 15/70=0.22). This means that 31% of the power is embedded in the first region of interest (1000-2000 Hz), 47% in the second range (2000-3000 Hz) and finally 22% in the last one (3000-4000 Hz). Therefore, the second frequency spot contains the highest energy. The last step is to normalize the pre-calculated components by the maximum value (0.47). This calculation results in the following monitoring index vector *MIV* (0.31/0.47=0.66, 0.47/0.47=1,0, 22/0.47=0.46). It should be noted that instead of simply using 0.31, 0.47, and 0.22, this will highlight easily the maximum component in the power vector and make the maximum value identical in all vectors, i.e., equal to one instead of different values.

#### 5.1.3.1.3 Codebook Definition

The objective of this step is to find the best and shortest list of symbols that are unique and can explain accurately the change in the state of the milling process or rotating equipment. This list is used as a reference library for matching purpose.

Assuming that the number of training *MIVs* defined in the above section is "*R*", these vectors can be clustered into a set of "*D*" (an arbitrary number used only as reference) reference vectors (or symbols), where "*D*" is much smaller than "*R*". This procedure is based on the generalized Llyod algorithm or the *K-means clustering algorithm* (Rabiner and Juang 1993).

The procedures applied in this study can be summarized as follows:

(1) Arbitrarily choose "*D*" reference vectors

(2) For each training vector choose the "reference" vector that is closest (in terms of city block distance) and assign that vector to the corresponding cell (or state represented by that reference vector). The city block distance is defined as follows:

$$d(MIV(l) - MIV_{ref}(l)) = \sum_{l=1}^{Z} \left| MIV(l) - MIV_{ref}(l) \right| \qquad (5.15)$$

where *MIV(l)* is the symbol vector representing the training vector, $MIV_{ref}(l)$ is the reference symbol vector, and $Z$ is the number of bank filters. This distance is calculated for each observation.

(3) Update the reference vector by calculating the centroid (i.e. adding the effect of the trained vector to the reference vector by considering the average values).

(4) Repeat steps 2 and 3 until the distance falls below a preset threshold.

#### 5.1.3.2 Model Estimation

The next step in the training process is to estimate the model parameters such as $\overline{\pi, A}, \overline{B}$. The procedure proposed in this study is shown in Figure 5.7 and is explained as follows.

**Step 1.** Take training data of each symptom describing the state of the equipment or process in question (e.g., data for sharp (state) tool at 135 rpm), extract the monitoring index vectors (i.e., power index vector as described in Section 5.1.3.1.2.2) and then compare them to the reference vectors or symbols (defined in Section 5.1.3.1.3) using the distance function defined in equation (5.15). This step assigns symbols to the given observations.

**Step 2.** The second step consists of defining the optimum state sequence using the *Viterbi algorithm* based on initial model randomly defined (Rabiner and Juang 1993). It is important to note that the selection of random values for the model parameters can have sometimes an effect on the training duration. This step finds the sequence of states that best explains the observations.

**Step 3.** The model is re-estimated using the *Baum Welch* method (Rabiner and Juang 1993). In this step, the matrices $\overline{A}, \overline{B}, \overline{\pi}$ are re-estimated based on the maximum likelihood concept to best adapt to training data representing real phenomena.

**Step 4.** The previous two steps are repeated until convergence is reached.

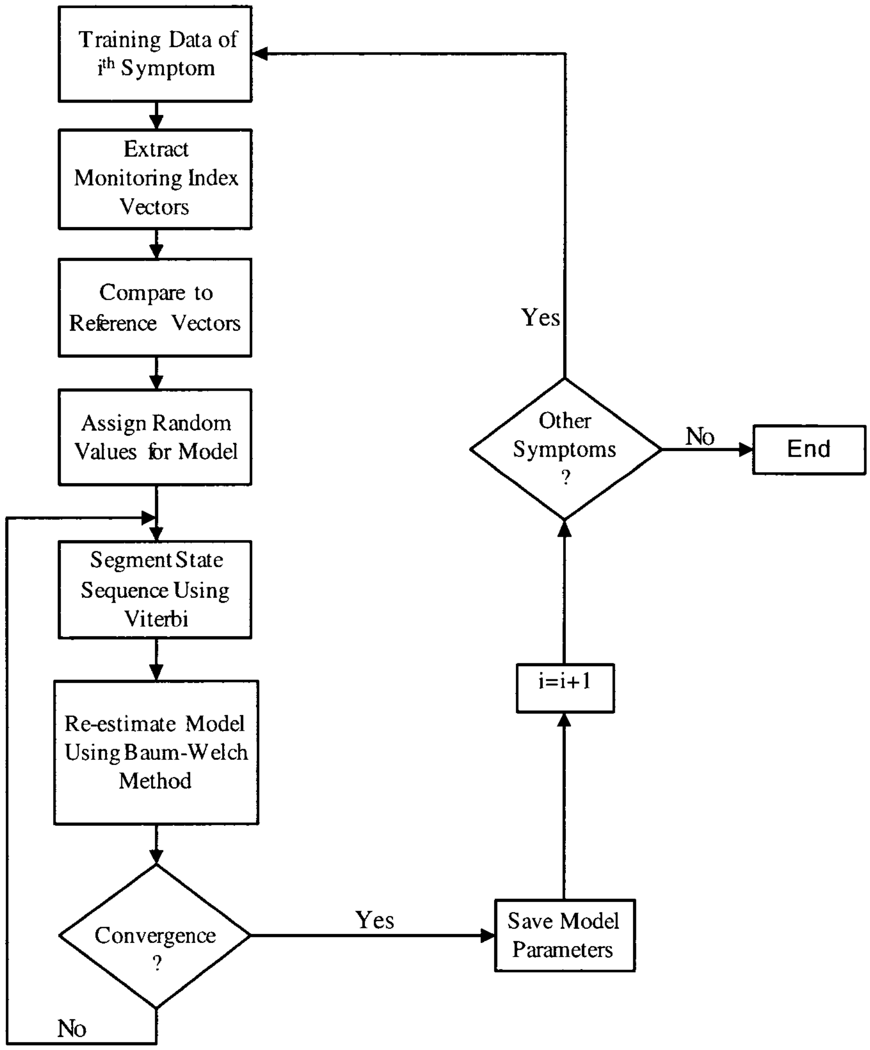



Figure 5.7 Model estimation procedures.

#### 5.1.3.2.1 State Sequence

State sequence is defined using the *Viterbi algorithm*. This algorithm is used to find the best state sequence, $\bar{q} = (\bar{q}_1, \bar{q}_2, ..., \bar{q}_N)$ for the given observation $O = (o_1, o_2, ..., o_{T_o})$.

Defining the quantity

$$\delta_t(\bar{i}) = \max_{\bar{q}_1, \bar{q}_2, ..., \bar{q}_{t-1}} P\left[ \bar{q}_1, \bar{q}_2, ..., \bar{q}_{t-1}, \bar{q}_t = \bar{i}, o_1 o_2 ... o_t / \lambda \right] \tag{5.16}$$

as the highest probability along a single path, at time *t*, which accounts for the first *t* observations and ends in state $\bar{i}$.

By induction:

$$\delta_{t+1}(\bar{i}) = \left[ \max_{i} \delta_t(\bar{i}) a_{\bar{i}\bar{j}} \right] b_{\bar{j}}(o_{t+1}) \tag{5.17}$$

The procedure is summarized as follows:

(1) Initialization

$$\delta_1(\bar{i}) = \bar{\pi}_{\bar{i}} b_{\bar{i}}(o_1) \quad \bar{i} = 1,2,\ldots,N \tag{5.18}$$

(2) Recursion

$$\delta_t(\bar{j}) = \max_{1 \le \bar{i} \le N} \left[ \delta_{t-1}(\bar{i}) a_{\bar{i}\bar{j}} \right] b_{\bar{j}}(o_t) \qquad 2 \le t \le T_o \text{ and } \bar{j} = 1,2,\ldots,N \tag{5.19}$$

(3) Termination

$$q^*_{T_o} = \arg\max_{1 \le \bar{i} \le N} \left[ \delta_{T_o}(\bar{i}) \right] \tag{5.20}$$

#### 5.1.3.2.2 Re-Estimation Method

Re-estimation method is applied using the *Baum-Welch* iterative procedures (Rabiner and Juang 1993). Let's define $\xi_t(\bar{i},\bar{j})$ as the probability of being in state $\bar{i}$ at time *t,* and state $\bar{j}$ at time *t+1*:

$$\xi_t(\bar{i},\bar{j}) = P(\bar{q}_t = \bar{i}, \bar{q}_{t+1} = \bar{j} / O, \lambda) \tag{5.21}$$

Using the forward and backward variables, $\xi_t(\bar{i},\bar{j})$ can be written in the form

$$\xi_t(\bar{i},\bar{j}) = \frac{\alpha_t(\bar{i}) a_{\bar{i}\bar{j}} b_{\bar{j}}(o_{t+1}) \beta_{t+1}(\bar{j})}{\sum_{\bar{i}=1}^{N} \sum_{\bar{j}=1}^{N} \alpha_t(\bar{i}) a_{\bar{i}\bar{j}} b_{\bar{j}}(o_{t+1}) \beta_{t+1}(\bar{j})} \tag{5.22}$$

where $\alpha_t(\bar{i})$ and $\beta_{t+1}(\bar{j})$ represent the forward and backward variables as described below.

Given the entire observation sequence and the model, the probability of being in state $\bar{i}$ at time *t* is denoted $\gamma_t(\bar{i})$ and given by

$$\gamma_t(\bar{i}) = \sum_{\bar{j}=1}^{N} \xi_t(\bar{i},\bar{j}) \tag{5.23}$$

Baum and his colleagues have proven (Rabiner and Juang 1993) that reasonable re-estimation formulas for $\overline{A}$, $\overline{B}$ and $\overline{\pi}$ are $\overline{\overline{A}}$, $\overline{\overline{B}}$ and $\overline{\overline{\pi}}$ respectively and are defined as follows (Note: After re-estimating a number of times, $\overline{\overline{A}}$, $\overline{\overline{B}}$ and $\overline{\overline{\pi}}$ will converge to $\overline{A}$, $\overline{B}$ and $\overline{\pi}$ respectively. Then the converged $\overline{\overline{A}}$, $\overline{\overline{B}}$ and $\overline{\overline{\pi}}$ will be denoted as $\overline{A}$, $\overline{B}$ and $\overline{\pi}$ accordingly):

$$\overline{\overline{\pi}}_{\bar{i}} = \text{expected frequency (number of times) in state } \bar{i} \text{ at time } (t=1) = \gamma_1(\bar{i}) \tag{5.24}$$

$$\overline{\overline{a}}_{\bar{i}\bar{j}} = \frac{\textit{Expected number of transitions from state } \bar{i} \textit{ to state } \bar{j}}{\textit{Expected number of transitions from state } \bar{i}} = \frac{\sum_{t=1}^{T_o-1} \xi_t(\bar{i},\bar{j})}{\sum_{t=1}^{T_o-1} \gamma_t(\bar{i})} \tag{5.25}$$

$$\overline{\overline{b}}_{\bar{j}}(\bar{k}) = \frac{\textit{Expected number of times in state } \bar{j} \textit{ and observing symbol } v_{\bar{k}}}{\textit{Expected number of times in state } \bar{j}} = \frac{\sum_{\substack{t=1 \\ s.t.\, o_t = v_k}}^{T_o} \gamma_t(\bar{j})}{\sum_{t=1}^{T_o} \gamma_t(\bar{j})} \tag{5.26}$$

Using $\overline{\lambda} = (\overline{\overline{A}}, \overline{\overline{B}}, \overline{\pi})$, where $\overline{\overline{A}}$ and $\overline{\overline{B}}$ are composed of $\overline{\overline{a}}_{\bar{i}\bar{j}}$ and $\overline{\overline{b}}_{\bar{j}}$ coefficients respectively, and iteratively repeat the re-estimation calculation to improve the probability of $O$ being observed from the model (i.e. $P(O/\overline{\lambda}) > P(O/\lambda)$) until convergence is reached. Convergence is checked using distance measure $DM(\lambda, \overline{\lambda})$ where

$$DM\left(\lambda, \overline{\lambda}\right) = \frac{1}{T_o}\left[\log_{10}(P(O/\lambda)) - \log_{10}(P(O/\overline{\lambda}))\right] \tag{5.27}$$

### 5.1.4 Diagnosis Using Discrete HMM

The diagnosis procedures using the discrete HMM (DHMM) are shown in Figure 5.8. After defining the sequence of observations $O$ (i.e. sequence of symbols too), the probabilities of $O$ given the states model are calculated. The maximum probability at the end determines the actual state of the machine or process in question.

The probability of $O$ given a model λ is defined as:

$$P(O/\lambda) = \sum_{\bar{q}} P(O/\bar{q},\lambda)P(\bar{q}/\lambda) \tag{5.28}$$

where for $T_o$ observations $O = (o_1, o_2, ..., o_{T_o})$, the state sequence is

$$\bar{q} = (\bar{q}_1 \bar{q}_2 ... \bar{q}_{T_o}) \tag{5.29}$$

$$P(O/\bar{q},\lambda) = b_{\bar{q}_1}(o_1) b_{\bar{q}_2}(o_2) ... b_{\bar{q}_{T_o}}(o_{T_o}) \tag{5.30}$$

$$P(\bar{q}/\lambda) = \pi_{\bar{q}_1} a_{\bar{q}_1 \bar{q}_2} a_{\bar{q}_2 \bar{q}_3} ... a_{\bar{q}_{T_o-1} \bar{q}_{T_o}} \tag{5.31}$$

However, equation (5.28) involves a tremendous number of calculations. To reduce the computation time, two methods can be used: forward and backward procedures. In this study, the forward procedure is utilized.

#### 5.1.4.1 Forward Procedure

The forward variable $\alpha_t(\bar{i})$ is defined as follows:

$$\alpha_t(\bar{i}) = P(o_1 o_2 ... o_t, \bar{q}_t = \bar{i}/\lambda) \tag{5.32}$$

i.e., the probability of the partial observations sequence (until time $t$) and state $\bar{i}$ and time $t$, given the model $\lambda$. This variable can be solved through few steps described below:

1- Initialization

$$\alpha_1(\bar{i}) = \pi_{\bar{i}} b_{\bar{i}}(o_1) \qquad \bar{i} = 1,2,...,N \tag{5.33}$$

2- Induction

$$\alpha_{t+1}(\bar{j}) = \left[\sum_{\bar{i}=1}^{N} \alpha_t(\bar{i}) a_{\bar{i}\bar{j}}\right] b_{\bar{j}}(o_{t+1}) \qquad \bar{j} = 1,2,...,N \text{ and } 1 \le t \le T_o - 1 \tag{5.34}$$

where $T_0$ is the number of observations.

3- Termination

$$P(O/\lambda)=\sum_{i=1}^{N}\alpha_{T_0}(\bar{i}) \tag{5.35}$$

As the main purpose of this study is to find the optimum state, $\alpha_{T_0}(\bar{i})$ (i.e. the probability of the sequence of observation of length $T_o$ while being at state $\bar{i}$ and given model $\lambda$) is used as the indicator to define the maximum state probability.

#### 5.1.4.2 Backward Procedure

Similar to the forward variable, the backward variable employed in Section 5.1.3.2.2 is defined as

$$\beta_t(\bar{i})=P(o_{t+1}o_{t+2}...o_{T_o},\overline{q}_t=\bar{i}/\lambda) \tag{5.36}$$

that is, the probability of the partial observation sequence from $t$+1 to $T_0$, given state $\bar{i}$ at time $t$ and the model $\lambda$. This equation can be solved through two steps:

1- Initialization

$$\beta_{T_o}(\bar{i})=1 \qquad\qquad \bar{i}=1,2,...,N \tag{5.37}$$

2- Induction

$$\beta_t(\bar{i})=\sum_{j=1}^{N}a_{\bar{i}\bar{j}}b_{\bar{j}}(o_{t+1})\beta_{t+1}(\bar{j}) \quad t=T_0-1,T_0-2,....1 \text{ and } \bar{i}=1,2,...,N \tag{5.38}$$

### 5.1.5 Training Difficulties

If the sequence of observations is long (bigger than ten as stated by Rabiner and Juang 1993), and as $\overline{A}$ and $\overline{B}$ coefficients are lower than one each term of forward parameters starts to head exponentially to zero. A scaling coefficient $c_t$ can be used to perform the computation.

The basic scaling procedures multiplies $\alpha_t(\bar{i})$ or $\beta_t(\bar{i})$ by a scaling coefficient that is independent of $\bar{i}$ (i.e. it depends only on $t$), with the goal of keeping the scaled $\alpha_t(\bar{i})$ within the dynamic range of the computer for $1\le t\le T_o$.

For forward procedure, $c_t$ can be chosen as follows:

$$c_t = \frac{1}{\sum_{i=1}^{N} \alpha_t(\bar{i})} \tag{5.39}$$

Similarly, for backward procedure,

$$c_t = \frac{1}{\sum_{i=1}^{N} \beta_t(\bar{i})} \tag{5.40}$$

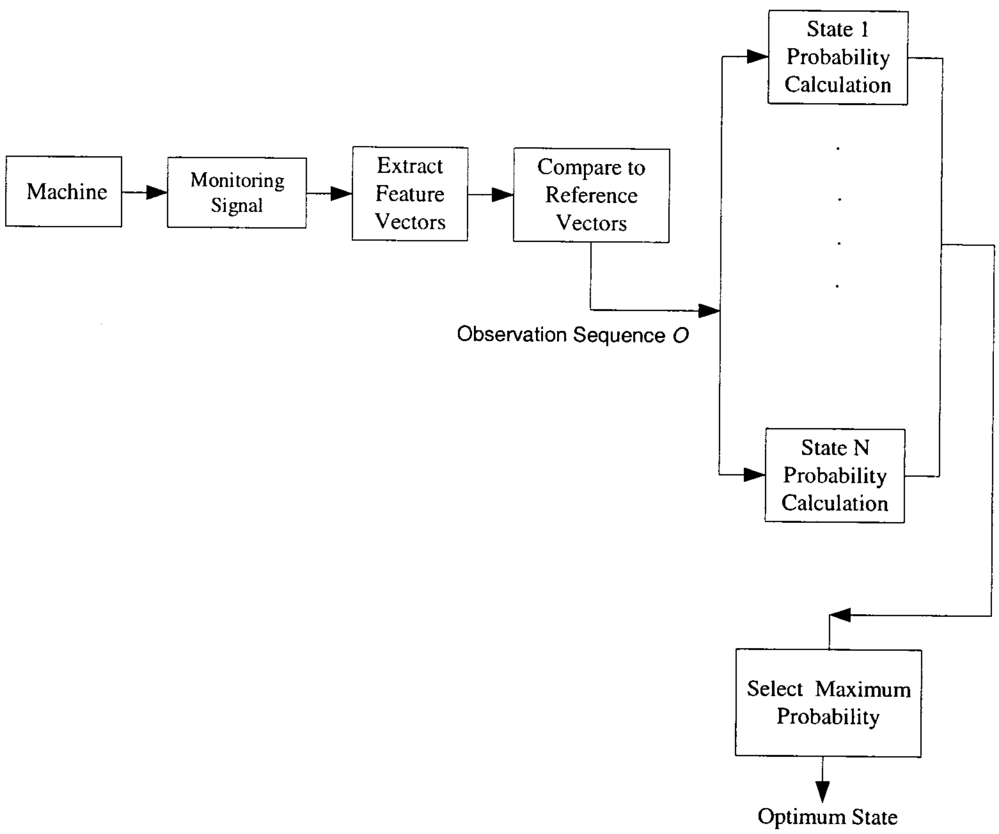



Figure 5.8 Diagnosis process using HMM.

## 5.2 Fault Localization

In the above sections, a HMM method was proposed to find the state or the condition of the monitored system (i.e. normal or faulty) and determine the severity of the fault. It is very important to take the investigation one step forward and identify the fault location. Nowadays, the technology is evolving rapidly and high cost equipments are widely used in several fields and applications. Stopping expensive equipment for a time period results certainly in loss of revenues and disturbances in the production plans. Therefore, any valuable information leading to a fast decision and quick repair becomes the main objective of the maintenance department manager.

As for example, bearings used in gas turbines are normally subject to wear or oil contaminations (fault). The decision to stop a gas turbine to check the bearings is very costly and requires approval from senior management. If the maintenance manager possesses information about the fault location (inner race, outer race, ball) in addition to the bearing state and fault severity, he or she may use these data with some other measurements such as oil analysis methods (e.g. spectrometry, X-Ray Fluorescence (XRF)) to increase the probability of a correct decision and reduce consequently the risk of unnecessary engine stoppage.

This explains the necessity to introduce the new concept of fault localization. This concept adds a different capability and some new dimensions to the equipment monitoring methodology.

### 5.2.1 Fault Location Determination

As described in the HMM training phase, the reference vectors or symbols form the codebook. These vectors are developed in such a manner to be unique and to well characterize the fault severity. Due to their uniqueness characteristic, these symbols can also be used to represent the fault locations. A representation index is therefore introduced. This index can take only two values: one (1) if the reference vector represents the fault location and zero (0) otherwise. Therefore, a Fault Localization Index Matrix (*FLIM*) is prepared during the training phase and is composed of binary values (i.e. 1 or 0). Assuming that the number of fault locations is *NFL* and the number of reference vectors (or symbols) is *M*, the *FLIM* dimension is $NFL \times M$.

This matrix is defined by checking during the training which symbol or reference vector is observed at each fault location. For example, if reference vector 2 (or symbol 2) is observed with a fault at location 1, $FLIM(1,2)$ which is the representation index of reference vector 2 with respect to fault location 1 is set to one (1).

As explained in Section 5.1.3, for a given sequence of observations a matching list of reference vectors is identified using the city block comparison technique. Utilizing the notation *ref* to represent this matching function, $ref\,(o_m)$ defines the reference vector number for observation $o_m$, i.e., it is a number varying between one (1) and $M$, the number of reference vectors or symbols in the codebook. In addition, taking $NOS$ as the number of observations in a sequence, the location index ($LI$) is computed for every fault location as follows:

$$LI(nfl) = \sum_{m=1}^{NOS} FLIM\,(nfl, ref\,(o_m)) \qquad 1 \le nfl \le NFL \tag{5.41}$$

where $nfl$ is the index for the number of fault locations ($NFL$).

The fault location is determined thereafter by finding the index corresponding to the maximum location index as follows:

$$i^* = \arg\max_i (LI(i)) \tag{5.42}$$

A numerical example illustrating the technique is presented in the experimental work.

## 5.3 Experimental Work

Similar to the previous chapters, the proposed HMM approach is applied to tool and bearing condition monitoring.

### 5.3.1 Tool Condition Monitoring

In this experiment, three hidden states ($N$=3) were chosen:

State 1: sharp tool

State 2: worn tool

State 3: broken tool

A reasonable question can be asked about the purpose of including a state for sharp tool (normal condition) at this stage as the fault detection has been already confirmed through Chapters 3 and 4 and the purpose of the HMM is to just diagnose the fault. However, as described in Chapter 3, the system is dynamic and anomaly can be transient. Therefore, some observations can show that the system is normal for a short duration of time and this can be reflected by the *Viterbi* technique, which tries to find the optimum sequence of states based on the sequence of observations.

The three states defined above suggest that the model should be left-right or Bakis as the transition from "worn" state to "sharp" state is not feasible in a perfect model. However, this is a dynamic problem, where observations can vary quickly as explained above depending upon machining characteristics (i.e. wear size, number of worn teeth, observation time length, etc.). As for example worn tool may generate the same observation as a sharp tool during machining time. This can be translated into a possible transition from worn to sharp by the *Viterbi* algorithm during the training phase. Therefore, an ergodic model was used (refer to Figure 5.9) allowing transitions to occur between all defined states despite the fact that probability to get from certain states to other ones could be quite small as shown later. In fact, an ergodic model was used by several researchers during similar tool wear monitoring (e.g. Ertunc and Oysu 2004, and Ertunc *et al* 2001)

Furthermore, a discrete HMM was used in this study. As explained previously, the observations in the discrete model can be defined based on real experimental data, making the computation and diagnosis as such more realistic.

#### 5.3.1.1 Training

The training activities were implemented using MATLAB signal processing toolbox. A *C* code program called *Training* is included in Appendix E. The training was performed using the data collected in Chapter 3 and as per below paragraphs

In order to determine the monitoring index vector, a bank of filters is defined based on the regions of interest specified in Chapter 3. This allows us as explained in Section 5.1.3.1.2.1 to extract the maximum information from the data in terms of frequency content. As such, three filters were used ($Z$ = 3) to obtain data within three desirable frequency ranges or bands:

Frequency band 1: 2500-3100 Hz

Frequency band 2: 3100-3700 Hz

Frequency band 3: 6500-7100 Hz

Regarding the optimum window size, unfortunately, there is no simple theoretical way to determine it as described in Section 5.1.3.1.2.1. Couvreur *et al* (1998) found that the best classification rate is obtained for a window size varying between 100 ms and 500 ms. In this experiment, a Hanning window with a size of 5000 samples (*L*=5000) was used. This window size is equivalent to 0.23 second based on a sampling frequency of 22050 Hz. This is in fact similar to the size of the sub-group used in Chapter 3.

The monitoring index vector is calculated based on the power. The training vectors were used in conjunction with the *K-mean clustering* method previously described to define the three reference vectors (or symbols) forming the "codebook". This codebook is illustrated in Figure 5.10.

Each observation symbol was compared to this "codebook" in order to assign the appropriate reference symbol (or vector). It is important to note that the number of reference vectors is small, reducing as such the diagnosis time. Figures 5.11-5.12 show how the number of the training vectors affects the convergence towards the reference vectors 1 and 2. The two figures demonstrate that the number of training vectors does not have an effect on the "codebook" after a sequence of 10 vectors or observations. This means that a period of approximately 2.3 seconds was enough to produce this "codebook".

Figures 5.13 and 5.14 present a sample of the training vectors obtained using sharp and worn tool respectively.

The following paragraph describes the preparation of the model parameters $(\overline{\pi}, \overline{A}, \overline{B})$. Before starting the machining process, the machinist usually conducts a visual inspection on the cutter to ensure "healthy" condition. This means that the probability of being at state 1 at time zero is equal to one ($\overline{\pi_1}=1$). The other model parameters $(\overline{A}, \overline{B})$ were defined using the *Viterbi algorithm* and *Welch* method as explained in Section 5.1.3.2. It is important to note that the time required to train the model was approximately 5 seconds. Furthermore, the difficulty of training the model as described in Section 5.1.5 was not experienced in this study. Therefore, the scaling method was not required, as the number of observations did not overpass ten.

The training process leaded to the following probability matrices:

$$\overline{A} = \left[a_{\overline{i}\,\overline{j}}\right] = \begin{array}{c} \begin{array}{ccc} \scriptstyle State\ 1 & \scriptstyle State\ 2 & \scriptstyle State\ 3 \end{array} \\ \begin{bmatrix} 0.80 & 0.10 & 0.10 \\ 0.15 & 0.80 & 0.05 \\ 0.05 & 0.05 & 0.90 \end{bmatrix} \end{array}$$

$$\overline{B} = \left[b_{\overline{j}}(\overline{k})\right] = \begin{array}{cc} \begin{array}{ccc} \scriptstyle Symbol\ 1 & \scriptstyle Symbol\ 2 & \scriptstyle Symbol\ 3 \end{array} & \\ \begin{bmatrix} 0.90 & 0.09 & 0.01 \\ 0.37 & 0.62 & 0.01 \\ 0.60 & 0.10 & 0.30 \end{bmatrix} & \begin{array}{l} State\ 1 \\ State\ 2 \\ State\ 3 \end{array} \end{array}$$

It is obvious from above matrices that $\sum_{\overline{j}=1}^{N} a_{\overline{i}\,\overline{j}} = 1 \quad \forall \overline{i}$ and $\sum_{\overline{k}=1}^{M} b_{\overline{j}}(\overline{k}) = 1 \quad \forall \overline{j}$ satisfying as such equation (5.3) and (5.5) respectively.

One can notice in the state transition probability matrix ($\overline{A}$) how the permission of transitions from worn to sharp or from breakage to sharp leads to a non-zero probabilities after training, although these values are relatively small. One note should be added with respect to tool breakage. The tool fracture occurs within a short time, during which it releases acoustic signal in a wide range of frequencies. This is why some observations may reflect a worn state during the breakage. Additionally, just before the breakage, the state is normal and observations represent that. This explains why the first two probabilities in the third row of $\overline{A}$ are not zero.

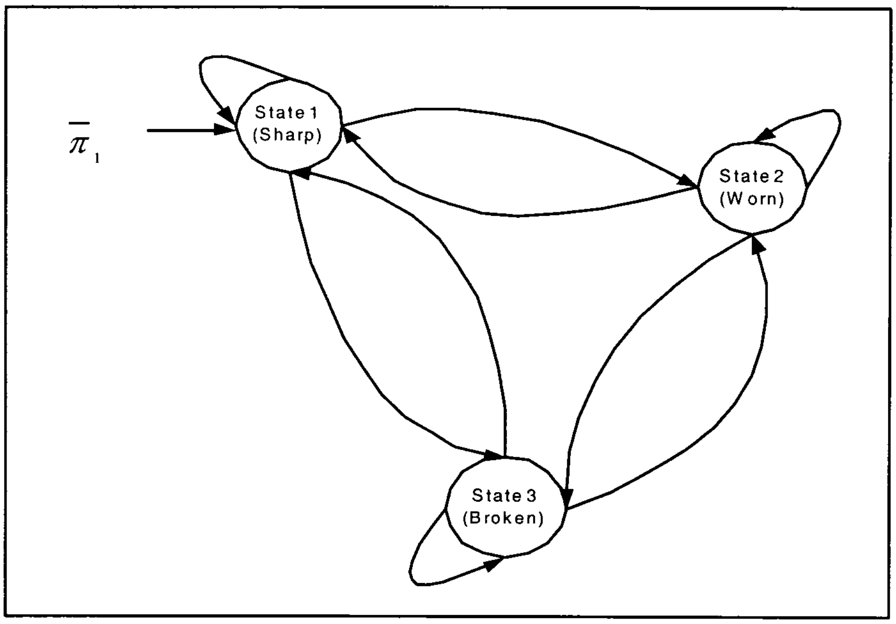



Figure 5.9 Tool ergodic model description

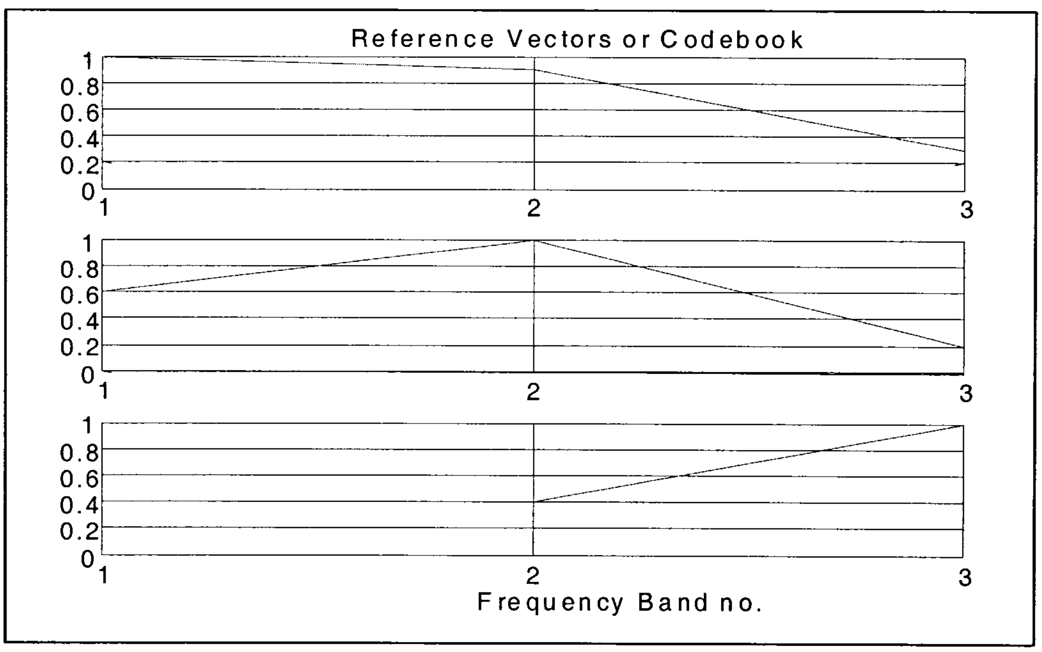



Figure 5.10 Tool codebook definition.

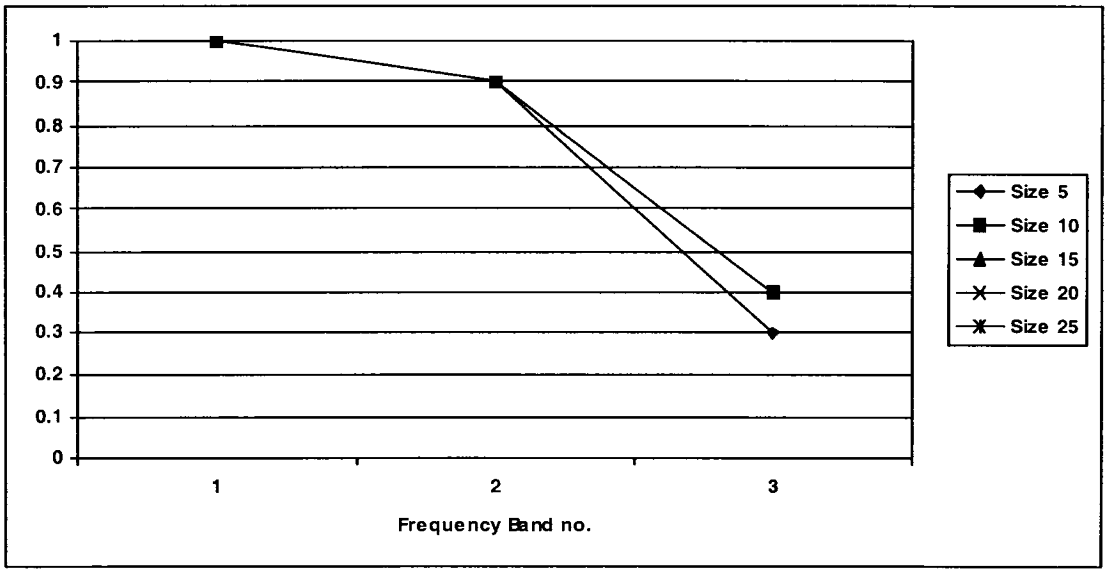



Figure 5.11 Effect of size of training vectors on determination of reference vector 1.

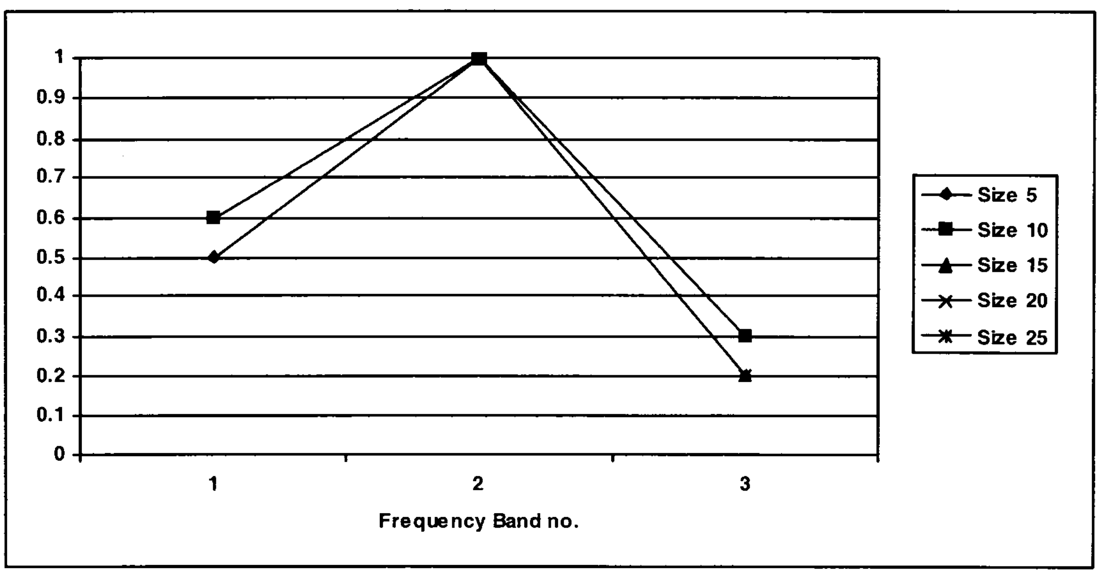



Figure 5.12 Effect of size of training vectors on determination of reference vector 2.

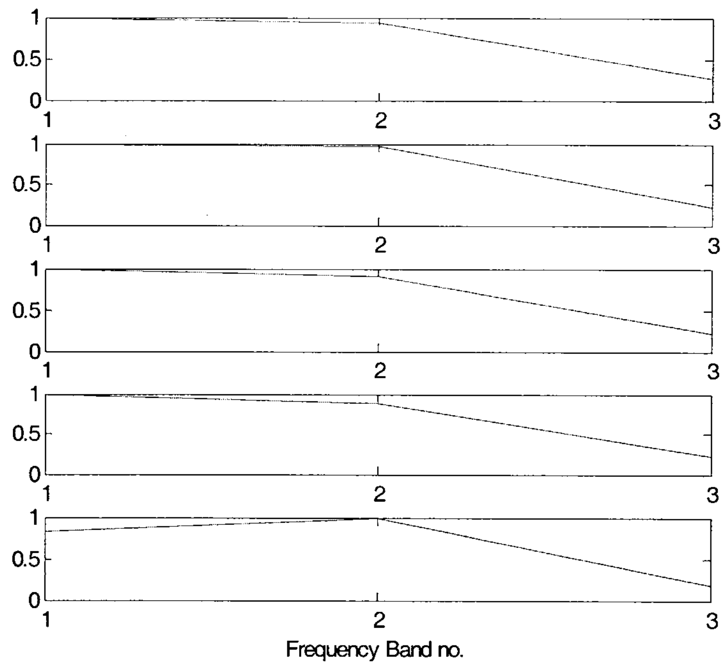



Figure 5.13 Samples of training vectors obtained using sharp tool at 210 rpm.

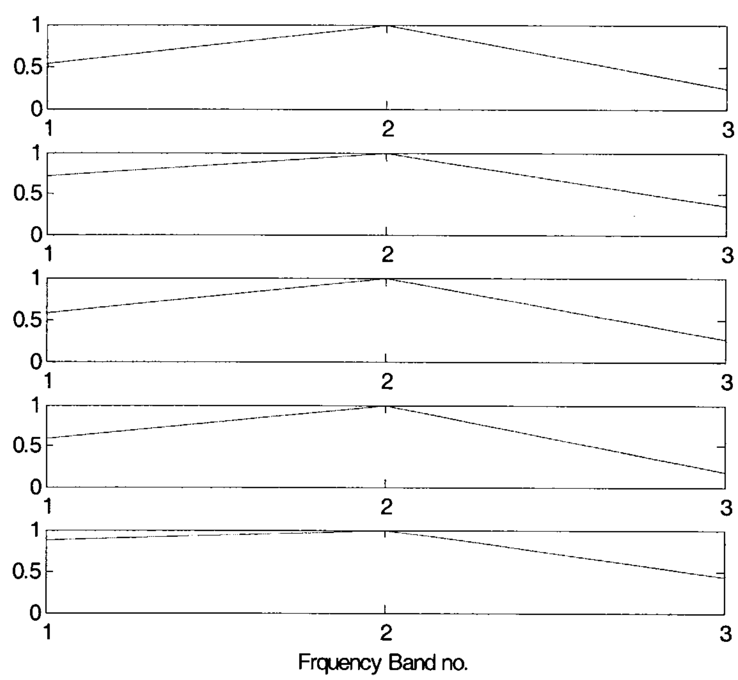



Figure 5.14 Samples of training vectors obtained using worn tool at 210 rpm.

#### 5.3.1.2 Diagnosis

Results were obtained using the MATLAB signal processing toolbox. A *C* code program called *Diagnosis* is included in Appendix E. Observations taken from data collected in Chapter 3 (but different than the ones considered for training) were fed into the three models laid out as per Figure 5.8. Five observations of each data file were utilized to determine the tool state. The results are shown in Table 5.1.

Table 5.1 Samples of the results obtained using the HMM approach.

| Reference File | No. of Observations | Probability of State 1 | Probability of State 2 | Probability of State 3 |
|---|---|---|---|---|
| Sharp at 135 rpm | 5 | **0.258** | 0.022 | 0.057 |
| Sharp at 210 rpm | 5 | **0.258** | 0.022 | 0.057 |
| Worn at 135 rpm | 5 | 3.7E-05 | **9.7E-04** | 1.5E-05 |
| Worn at 210 rpm | 5 | 3.7E-05 | **9.7E-04** | 1.5E-05 |
| Broken tool | 5 | 0.0004 | 0.0002 | **0.005** |

* Note: the predicted state is associated with the largest probability number (in bold).

By looking at the maximum probability highlighted in the above table, one can notice that all the five results match the state of the tool perfectly. Hence, the sequence of five observations was enough to produce accurate results. This is very encouraging because a decision can be made in a reasonably short time (approximately 1 second).

In order to determine the detection efficiency of this method based on a sequence of five observations, a series of twenty sequences was considered for each data file (except for tool breakage). For each sequence, the probability of each model was calculated and the tool state was identified based on the maximum probability value. Tables 5.2-4.5 show the calculation results and the corresponding success rate.

Table 5.2 Probability and success rate calculations for sharp tool at 135 rpm.

| Sequence No. | Probability of State 1 | Probability of State 2 | Probability of State 3 | Detected State |
|---|---|---|---|---|
| 1 | **0.258*** | 0.023 | 0.058 | 1 |
| 2 | **0.258** | 0.023 | 0.058 | 1 |
| 3 | **0.258** | 0.023 | 0.058 | 1 |
| 4 | **0.258** | 0.023 | 0.058 | 1 |
| 5 | **0.258** | 0.023 | 0.058 | 1 |
| 6 | **0.258** | 0.023 | 0.058 | 1 |
| 7 | **0.258** | 0.023 | 0.058 | 1 |
| 8 | **0.258** | 0.023 | 0.058 | 1 |
| 9 | **0.258** | 0.023 | 0.058 | 1 |
| 10 | **0.258** | 0.023 | 0.058 | 1 |
| 11 | **0.258** | 0.023 | 0.058 | 1 |
| 12 | **0.258** | 0.023 | 0.058 | 1 |
| 13 | **0.258** | 0.023 | 0.058 | 1 |
| 14 | **0.258** | 0.023 | 0.058 | 1 |
| 15 | **0.258** | 0.023 | 0.058 | 1 |
| 16 | **0.258** | 0.023 | 0.058 | 1 |
| 17 | **0.258** | 0.023 | 0.058 | 1 |
| 18 | **0.258** | 0.023 | 0.058 | 1 |
| 19 | **0.258** | 0.023 | 0.058 | 1 |
| 20 | **0.258** | 0.023 | 0.058 | 1 |
| Success Rate | | | | 100% |

* Note: the predicted state is associated with the largest probability number (in bold).

Table 5.3 Probability and success rate calculations for worn tool at 135 rpm.

| Sequence No. | Probability of State 1 | Probability of State 2 | Probability of State 3 | Detected State |
|---|---|---|---|---|
| 1 | 3.7E-05 | **9.7E-04*** | 1.5E-05 | 2 |
| 2 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 3 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 4 | 3.4E-04 | **8.8E-03** | 1.3E-04 | 2 |
| 5 | **4.0E-03** | 8.5E-04 | 1.0E-03 | 1 |
| 6 | 9.1E-04 | **1.7E-02** | 3.8E-04 | 2 |
| 7 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 8 | 3.4E-04 | **5.7E-04** | 8.7E-05 | 2 |
| 9 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 10 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 11 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 12 | 6.1E-04 | **7.4E-03** | 2.7E-04 | 2 |
| 13 | 6.8E-05 | **8.1E-04** | 3.0E-05 | 2 |
| 14 | 3.4E-04 | **5.7E-04** | 8.7E-05 | 2 |
| 15 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 16 | 3.4E-04 | **8.8E-03** | 1.3E-04 | 2 |
| 17 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 18 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 19 | 3.7E-05 | **8.5E-04** | 1.5E-05 | 2 |
| 20 | 3.7E-05 | **4.3E-03** | 1.6E-03 | 2 |
| Success Rate | | | | 95% |

* Note: the predicted state is associated with the largest probability number (in bold).

Table 5.4 Probability and success rate calculations for sharp tool at 210 rpm.

| Sequence No. | Probability of State 1 | Probability of State 2 | Probability of State 3 | Detected State |
|---|---|---|---|---|
| 1 | **0.258*** | 0.023 | 0.058 | 1 |
| 2 | **0.258** | 0.023 | 0.058 | 1 |
| 3 | **0.258** | 0.023 | 0.058 | 1 |
| 4 | **0.258** | 0.023 | 0.058 | 1 |
| 5 | **0.258** | 0.023 | 0.058 | 1 |
| 6 | **0.258** | 0.023 | 0.058 | 1 |
| 7 | **0.258** | 0.023 | 0.058 | 1 |
| 8 | **0.028** | 0.002 | 0.006 | 1 |
| 9 | **0.258** | 0.023 | 0.058 | 1 |
| 10 | **0.258** | 0.023 | 0.058 | 1 |
| 11 | **0.258** | 0.023 | 0.058 | 1 |
| 12 | **0.258** | 0.023 | 0.058 | 1 |
| 13 | **0.258** | 0.023 | 0.058 | 1 |
| 14 | **0.028** | 0.002 | 0.006 | 1 |
| 15 | **0.258** | 0.023 | 0.058 | 1 |
| 16 | **0.258** | 0.023 | 0.058 | 1 |
| 17 | **0.258** | 0.023 | 0.058 | 1 |
| 18 | **0.258** | 0.023 | 0.058 | 1 |
| 19 | **0.258** | 0.023 | 0.058 | 1 |
| 20 | **0.028** | 0.002 | 0.006 | 1 |
| Success Rate | | | | 100% |

* Note: the predicted state is associated with the largest probability number (in bold).

Table 5.5 Probability and success rate calculations for worn tool at 210 rpm.

| Sequence No. | Probability of State 1 | Probability of State 2 | Probability of State 3 | Detected State |
|---|---|---|---|---|
| 1 | 3.7E-05 | **9.7E-04*** | 1.5E-05 | 2 |
| 2 | 3.4E-04 | **8.8E-03** | 1.3E-04 | 2 |
| 3 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 4 | 4.2E-04 | **3.0E-03** | 1.8E-04 | 2 |
| 5 | 9.9E-05 | **6.3E-04** | 3.9E-05 | 2 |
| 6 | 3.4E-04 | **5.7E-04** | 8.7E-05 | 2 |
| 7 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 8 | 3.4E-04 | **8.8E-03** | 1.3E-04 | 2 |
| 9 | 6.8E-05 | **8.1E-04** | 3.0E-05 | 2 |
| 10 | 6.8E-05 | **8.1E-04** | 3.0E-05 | 2 |
| 11 | 9.9E-05 | **6.3E-04** | 3.9E-05 | 2 |
| 12 | 3.4E-04 | **8.8E-03** | 1.3E-04 | 2 |
| 13 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 14 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 15 | 4.4E-04 | **1.4E-03** | 1.7E-04 | 2 |
| 16 | 3.1E-04 | **4.8E-04** | 1.8E-04 | 2 |
| 17 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 18 | 3.4E-04 | **8.8E-03** | 1.3E-04 | 2 |
| 19 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| 20 | 3.7E-05 | **9.7E-04** | 1.5E-05 | 2 |
| Success Rate | | | | 100% |

* Note: the predicted state is associated with the largest number (in bold).

Based on Tables 5.2-5.5 a sequence of five observations was enough to provide at least 95% state detection efficiency. Increasing the size of the sequence would certainly increase the detection efficiency as explained by Rabiner and Juang (1993). However, the main objective of this study was to show that with a few number of observations, one could obtain accurate results.

In order to show the probability results with respect to time (to simulate an on-line monitoring process), and as probabilities are very small in general, a normalizing method is required. The normalization is performed in this case by dividing each state probability by the sum of all probabilities. This leads to values below one but much higher than the original probability values. For example, taking the first row of Table 5.2, the normalized probabilities will have the following values:

Normalized probability for State 1 = [0.258/ (0.258+0.023+0.058)] = 0.761

Normalized probability for State 2 = [0.023/(0.258+0.023+0.058)] = 0.068

Normalized probability for State 3 = [0.058/(0.258+0.023+0.058)] = 0.171

Applying the same principle to the values shown in Tables 5.2-5.5, one can plot the normalized probabilities with respect to time. Figures 5.15 and 5.17 show the trends for the sharp tool. The normalized probability for the sharp state has the highest value during both cases. Looking at Figure 5.16, one can note that one sequence of observations leads to an inaccurate decision. In this case, the normalized probability for the sharp state increases above the one for the worn state. In the same figure, few increases are remarked in the plot of the sharp normalized probability, however, values remain lower than the ones for the worn state. This means that at least one observation in the sequence reflects a normal state. The same remark can be made in Figure 5.18. It is interesting to observe bow both sharp and worn probabilities vary in opposite ways: when the first increases, the second decreases and vice versa.

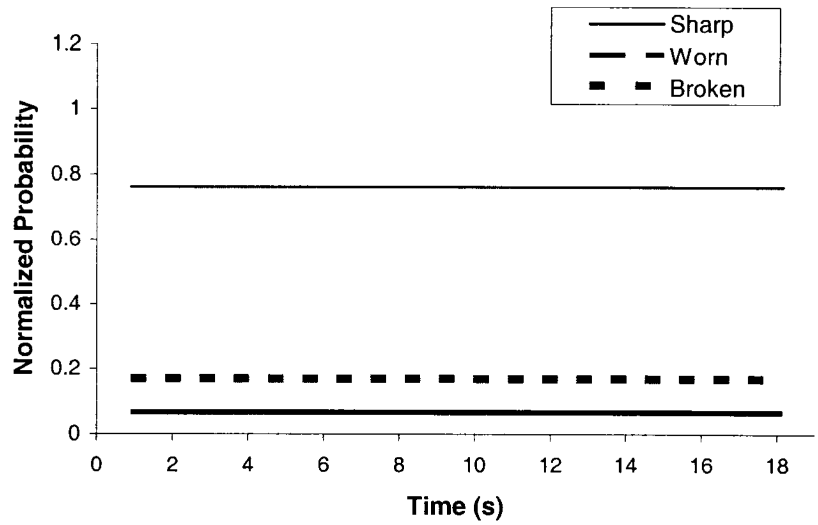



Figure 5.15 Normalized probabilities for sharp tool at 135 rpm.

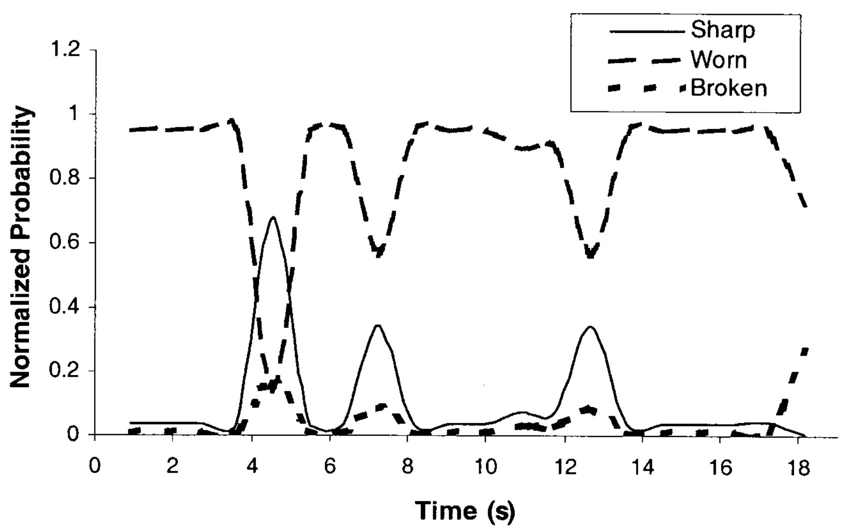



Figure 5.16 Normalized probabilities for worn tool at 135 rpm.

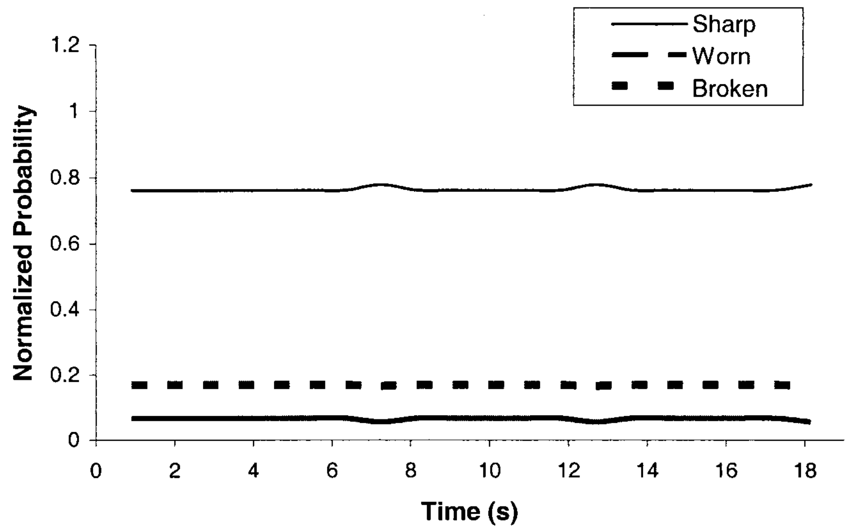



Figure 5.17 Normalized probabilities for sharp tool at 210 rpm.

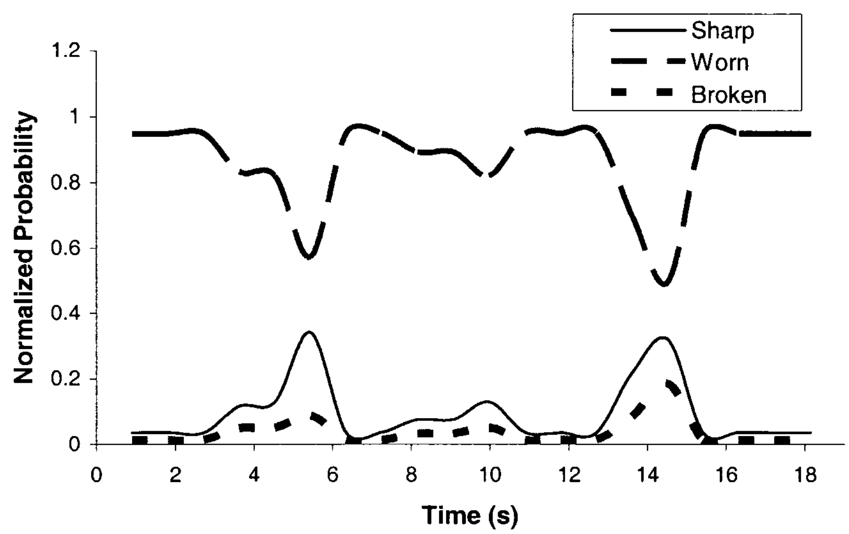



Figure 5.18 Normalized probabilities for worn tool at 210 rpm.

### 5.3.2 Bearing Monitoring

The states are defined based on the data provided for both drive end (DE) and fan end (FE) bearings. For the DE bearing the following five sates were chosen:

State 1: normal

State 2: low damage (fault diameter equal or less than 0.007")

State 3: medium damage (fault diameter bigger than 0.007" but equal or less than 0.014")

State 4: high damage (fault diameter bigger than 0.014" but equal or less than 0.021")

State 5: severe damage (fault diameter bigger than 0.0021 but equal or less than 0.028")

For the FE bearing the following four states were defined:

State 1: normal

State 2: low damage (fault diameter equal or less than 0.007")

State 3: medium damage (fault diameter bigger than 0.007" but equal or less than0.014")

State 4: high damage (fault diameter bigger than 0.014" but equal or less than 0.021")

As explained in the tool monitoring case, the use of the normal state is necessary as observations might reflect this condition during training, especially if anomaly is transient.

All the states defined in the previous paragraph suggest that the model should be a left-right or Bakis model which means that the transitions from a state to another are occurring in one direction, i.e., from left to right. In other words, the bearing state can degrade but cannot improve with time. For example, it is hard to imagine in a perfect world that a bearing can switch from state 3 (medium damage) to state 2 (low damage) or state 1 (normal). However, this is a dynamic model where observations can vary quickly within each state depending on the vibration characteristics of the system and the time length of the observation. For example, a "medium damage" bearing may exhibit the same observation as a low-damage bearing during a specific time. During the training, this can lead to a non-zero transition probability from state 3 to state 2. Although this probability is very small, it should be considered nevertheless to make the system more robust and closer to the reality. This is why an ergodic model is considered in this case allowing transitions to occur between all defined states despite the fact that some transitions have a very low probability. This type of model has been used by other researchers in similar applications (e.g., Lee *et al* 2004). Similar to the tool case, the model utilized in this study is discrete.

#### 5.3.2.1 Training

The training activities were implemented using MATLAB signal processing toolbox. The *C* code program *Training* (Appendix E) used in the tool case is used herein. The training was performed using the data collected in Chapter 3 (refer to Tables 3.4, 3.5 and 3.6) and as per below procedure.

The first step in the training is to define the codebook that contains the unique symbols characterizing the pre-defined states. In order to achieve that, a bank of filters was defined based on the regions of interest specified in Chapter 3. The main purpose of this step is to extract the most useful information from the signal that can lead to the reference vectors or symbols. Therefore, five bands covering the whole region of interest are selected in this analysis.

Frequency Band 1: 100-1100 Hz

Frequency Band 2: 1100-2200 Hz

Frequency Band 3: 2200-3300 Hz

Frequency Band 4: 3300-4400 Hz

Frequency Band 5: 4400-5500 Hz

As described earlier, there is no simple theoretical way to determine the optimum window size. The choice is dependent upon type of problem and signal characteristics (e.g. sampling frequency). In this experiment, a Hanning window with a size of 3000 samples ($L$=3000) was used (equivalent to five sub-groups). This window size is equivalent to 0.25 second based on a sampling frequency of 12000 Hz. This is in fact close to the window size used in Section 5.3.1.1.

In this experimental work, the power was again used to define the monitoring index vector. Each vector was normalized as per the instructions defined in Section 5.1.3.1.2.2.

The training vectors were used in conjunction with the *K-mean clustering* method described earlier to define the reference vectors or codebook. The reference vectors were created for both DE and FE bearings as shown in Figures 5.19 and 5.20. Seventeen reference vectors were developed for each bearing type based on the available data.

The number of observations needed to obtain the codebooks for both DE and FE bearings is illustrated in Figures 5.21 and 5.22 respectively. Looking at Figure 5.21, the maximum number of observations needed to train the DE bearing model is 10 (equivalent to 2.5 seconds). On the other hand, Figure 5.22 shows that a maximum of 12 observations (equivalent to 3 seconds) is required to train the FE bearing model. Each observation symbol was compared to these codebooks in order to assign the appropriate reference symbol. Based on these comparisons, the sequence of observations was defined.

Before starting the engine, it is assumed that a visual inspection is conducted to ensure adequate operation. This means that the probability of being at state one and at time zero is equal to one $(\overline{\pi}_1 = 1)$. The other model parameters ($\overline{A}$ and $\overline{B}$ matrices) were defined through training using the *Viterbi algorithm* and *Welch* method as explained in earlier sections.

For this purpose a total of 20 observations are taken (equivalent to 5 seconds of data) to define $\overline{A}$ and $\overline{B}$ matrices. The process was based mainly on a series of iterations performed until convergence was obtained as described earlier. In fact, the training phase was accomplished without experiencing difficulties as the number of observations used in the analysis was smaller than 10. However, the numbers equal to zero were replaced with small values (e.g. equal to 0.01) to facilitate the training as proposed by Rabiner and Juang (1993).

For DE bearing, the following probability matrices were defined:

$$\overline{A}_{DE} = \left[a_{\overline{ij}}\right] = \begin{array}{c} \begin{array}{ccccc} \scriptstyle State\ 1 & \scriptstyle State\ 2 & \scriptstyle State\ 3 & \scriptstyle State\ 4 & \scriptstyle State\ 5 \end{array} \\ \begin{bmatrix} 0.750 & 0.100 & 0.050 & 0.050 & 0.050 \\ 0.009 & 0.731 & 0.180 & 0.040 & 0.040 \\ 0.010 & 0.100 & 0.750 & 0.100 & 0.040 \\ 0.001 & 0.141 & 0.125 & 0.624 & 0.100 \\ 0.010 & 0.010 & 0.010 & 0.010 & 0.960 \end{bmatrix} \end{array}$$

$$\bar{B}_{DE}=\left[b_{\bar{j}}(\bar{k})\right]=\begin{array}{c}
\begin{array}{ccccccccccccccccc} \scriptstyle Sym1 & \scriptstyle Sym2 & & & & & & & & & & & & & & & \scriptstyle Sym17 \end{array} \\
\begin{bmatrix}
0.920 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 & 0.005 \\
0.010 & 0.250 & 0.083 & 0.483 & 0.010 & 0.054 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 \\
0.010 & 0.270 & 0.010 & 0.010 & 0.010 & 0.110 & 0.182 & 0.076 & 0.145 & 0.289 & 0.010 & 0.010 & 0.130 & 0.068 & 0.010 & 0.010 & 0.010 \\
0.010 & 0.010 & 0.010 & 0.155 & 0.013 & 0.108 & 0.010 & 0.010 & 0.010 & 0.201 & 0.135 & 0.138 & 0.150 & 0.010 & 0.010 & 0.010 & 0.010 \\
0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.010 & 0.250 & 0.125 & 0.125 & 0.370
\end{bmatrix}
\end{array}$$

On the other hand, the probability matrices for FE bearing were determined as follows:

$$\bar{A}_{FE}=\left[a_{\bar{i}\bar{j}}\right]=\begin{array}{c}
\begin{array}{cccc} \scriptstyle State\ 1 & \scriptstyle State\ 2 & \scriptstyle State\ 3 & \scriptstyle State\ 4 \end{array} \\
\begin{bmatrix}
0.800 & 0.100 & 0.005 & 0.005 \\
0.010 & 0.810 & 0.100 & 0.008 \\
0.010 & 0.010 & 0.940 & 0.040 \\
0.010 & 0.080 & 0.010 & 0.900
\end{bmatrix}
\end{array}$$

$$\bar{B}_{FE}=\left[b_{\bar{j}}(\bar{k})\right]=\begin{array}{c}
\begin{array}{ccccccccccccccccc} \scriptstyle Sym1 & \scriptstyle Sym2 & & & & & & & & & & & & & & & \scriptstyle Sym17 \end{array} \\
\begin{bmatrix}
0.84 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 \\
0.01 & 0.20 & 0.10 & 0.11 & 0.05 & 0.03 & 0.15 & 0.21 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.04 & 0.01 & 0.03 & 0.01 \\
0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.32 & 0.13 & 0.11 & 0.17 & 0.14 & 0.02 & 0.01 & 0.01 & 0.01 \\
0.01 & 0.01 & 0.01 & 0.02 & 0.01 & 0.01 & 0.01 & 0.05 & 0.01 & 0.01 & 0.01 & 0.01 & 0.01 & 0.41 & 0.20 & 0.10 & 0.11
\end{bmatrix}
\end{array}$$

It is obvious from above matrices that $\sum_{\bar{j}=1}^{N} a_{\bar{i}\bar{j}} = 1 \ (\forall \bar{i})$ and $\sum_{\bar{k}=1}^{M} b_{\bar{j}}(\bar{k}) = 1 (\forall \bar{j})$ satisfying as such equations (5.3) and (5.5) respectively.

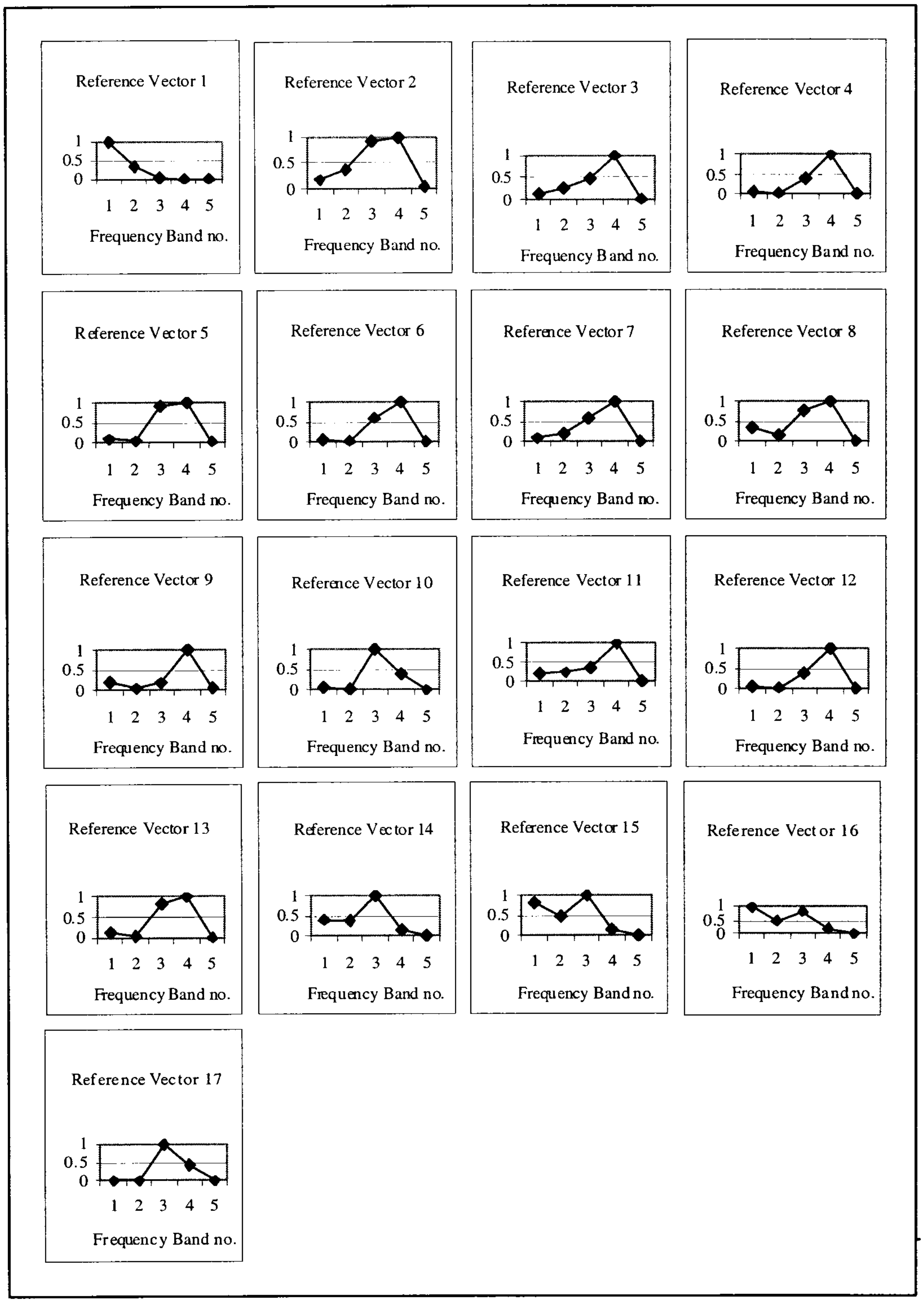



Figure 5.19 Drive end bearing codebook.

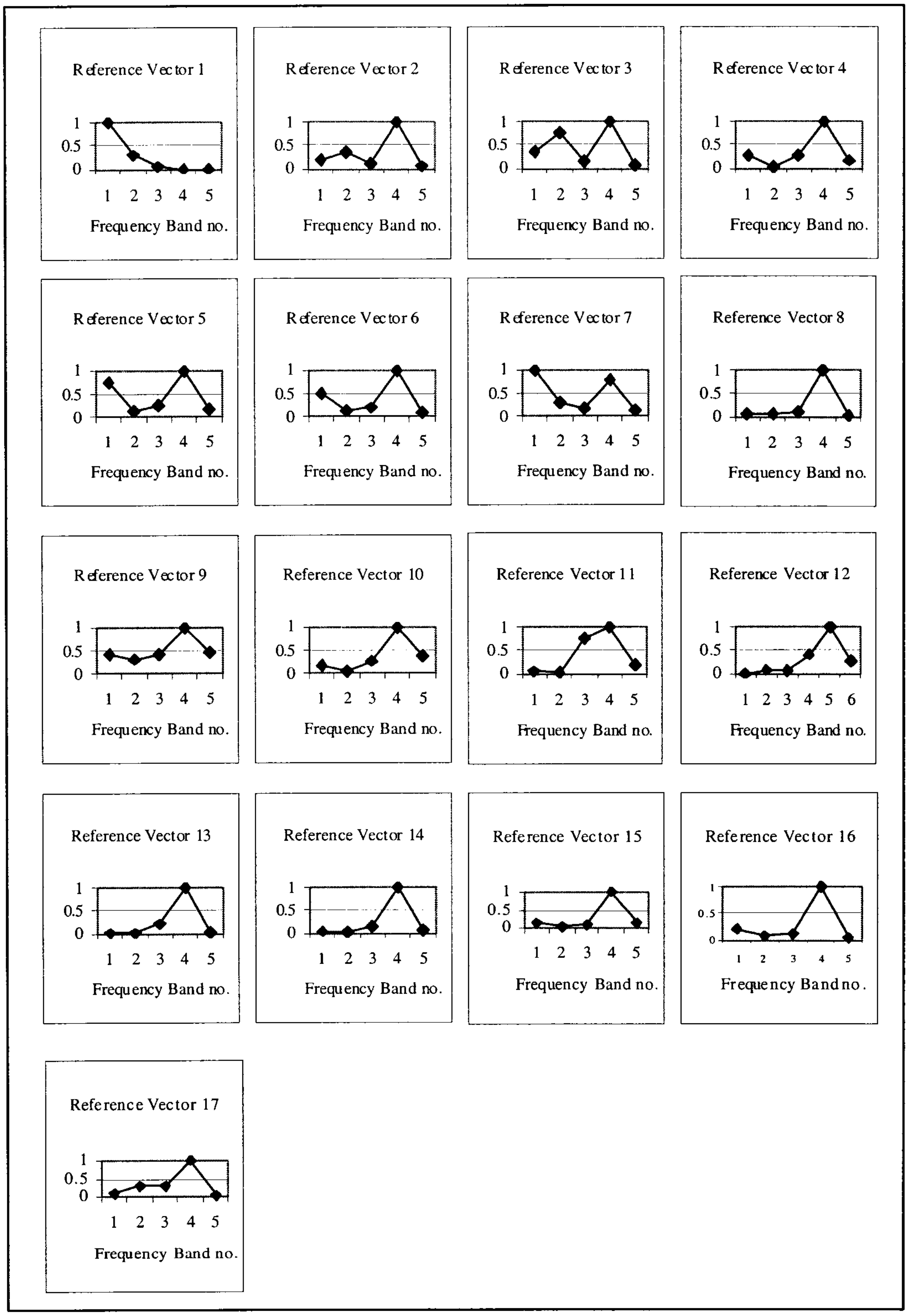



Figure 5.20 Fan end bearing codebook.

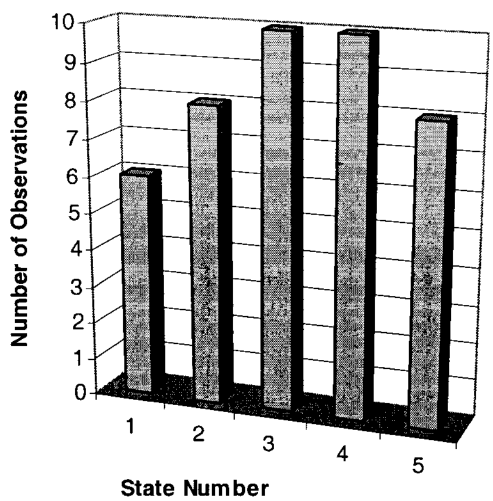



Figure 5.21 Number of observations required for drive end bearing codebook convergence.

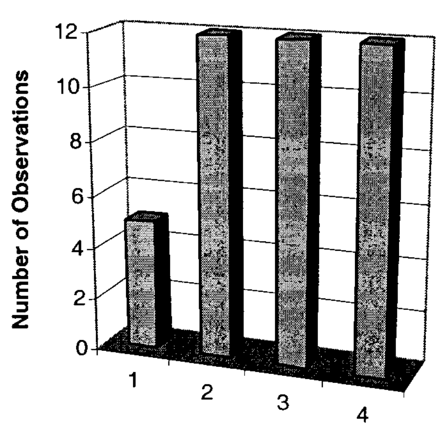



Figure 5.22 Number of observations required for fan end bearing codebook convergence.

#### 5.3.2.2 Diagnosis

Results were obtained using MATLAB signal processing toolbox. A program called *Diagnosis* is attached to Appendix E. Observations taken from data collected in Chapter 3 (but different from the ones used for training) were fed into the model as illustrated in Figure 5.8.

It is important to understand the effect of the observations sequence length on the analysis. As described in previous sections, as the number of observations increases the detection efficiency increases simultaneously. However, as the model probability is a chain of probability multiplications, considerations should be given to the fact that result may be close to zero. This is why it is recommended to use the scaling technique to overcome such problems especially for a sequence length of 10 or more observations which often leads to very small model probability.

To examine this issue, a total of thirty-six (36) observations were used and grouped into sequences of three (3), four (4) and six (6) observations. Based on these sequence lengths, the state detection efficiency was computed and presented in Tables 5.6 and 5.7 for both DE and FE bearings respectively.

As shown in Tables 5.6 and 5.7, the difference in terms of detection efficiency between using a sequence of three, four and five observations is not significant. The detection efficiency based on a sequence of three observations is as expected the lowest followed by the one for four and five observations respectively. Therefore, a sequence of four observations can be used as a baseline for the diagnosis calculation. In fact, the size of four observations was enough to provide at least 95% state detection efficiency. This detection efficiency could be much higher if some doubtful measurements for the DE bearing were discarded. For example one can see in Table 5.6 that for fault diameter of 0.021" (cases no. 37-39), seeded at the outer race of the bearing, the state detection efficiency was very low reaching a minimum value of 17%. The cause of this low efficiency is that the bearing, under these conditions, generated observations similar to the ones seen in states 2 and 3. This means that the vibration signal was continuously varying between low, medium and high levels. For this reason, the author thinks that the measurements collected in that special case were inaccurate. In fact, looking at Table 5.7, one can notice that the state detection for the same case and conditions was 100%.

Using a sequence of four observations the model probabilities were calculated based on the technique described previously. Tables 5.8 and 5.9 represent a sample of the results obtained for the DE and FE bearings respectively. The maximum probability in each case led to the state of the monitored system. The results shown below illustrate clearly the capability of the model to identify the fault size. For both DE and FE bearings data samples, the model detected every time the state correctly.

Another important observation to make is that the probability results shown in Tables 5.8 and 5.9 are as expected very small. For this reason, it has been recommended in the above sections to use the scaling method if the number of observations is above 10. One can also note that the HMM can be used to confirm indirectly the fault detection. However, the method described in the previous chapter is faster and can provide other useful information for maintainers such as type of anomaly ("transient" or "continuous") and index of deterioration.

To plot the state probabilities with respect to time, the same normalization technique proposed in the tool monitoring is applied herein. As such, each state probability is normalized with respect to the sum of all probabilities. Several typical examples are illustrated in this case, covering all fault sizes at the inner race of the DE bearing and the two extremities of speed (1797-1730 rpm) and load (0-3 hp).

Figures 5.23 and 5.24 show the normalized probabilities for normal condition. In both cases, the normalized probability for the normal state is higher than the other probabilities. The low probability values of the other states indicate that all the observations are clearly representing a normal condition. Figures 5.25-5.32 describe the trends of the normalized probabilities when the DE bearing is seeded at the inner race with a fault diameter varying between 0.007" and 0.028". In all the illustrated examples, the highest normalized probability matches the real state. One quick note can be made when examining Figure 5.28: the low damage normalized probability increases at one instant and approaches the medium damage probability. This means that at that time at least one observation is driving the decision towards the low damage state.

Table 5.6 DE bearing state detection efficiency.

| Case No. | Condition Description | Real State | 3 Observations | 4 Observations | 6 Observations |
|---|---|---|---|---|---|
| 1 | Normal, 0 hp, 1797 rpm | 1 | 100% | 100% | 100% |
| 2 | Normal, 1 hp, 1772 rpm | 1 | 100% | 100% | 100% |
| 3 | Normal, 2 hp, 1750 rpm | 1 | 100% | 100% | 100% |
| 4 | Normal, 3 hp, 1730 rpm | 1 | 100% | 100% | 100% |
| 5 | Fault Diameter 0.007", at Inner Race, 0 hp, 1797 rpm | 2 | 100% | 100% | 100% |
| 6 | Fault Diameter 0.007", at Inner Race, 1 hp, 1772 rpm | 2 | 100% | 100% | 100% |
| 7 | Fault Diameter 0.007", at Inner Race, 2 hp, 1750 rpm | 2 | 100% | 100% | 100% |
| 8 | Fault Diameter 0.007", at Inner Race, 3 hp, 1730 rpm | 2 | 100% | 100% | 100% |
| 9 | Fault Diameter 0.007", at Ball, 0 hp, 1797 rpm | 2 | 75% | 100% | 100% |
| 10 | Fault Diameter 0.007", at Ball, 1 hp, 1772 rpm | 2 | 100% | 100% | 100% |
| 11 | Fault Diameter 0.007", at Ball, 2 hp, 1750 rpm | 2 | 100% | 100% | 100% |
| 12 | Fault Diameter 0.007", at Ball, 3 hp, 1730 rpm | 2 | 100% | 100% | 100% |
| 13 | Fault Diameter 0.007", at Outer Race @ 6:00, 0 hp, 1797 rpm | 2 | 100% | 100% | 100% |
| 14 | Fault Diameter 0.007", at Outer Race @ 6:00, 1 hp, 1772 rpm | 2 | 100% | 100% | 100% |
| 15 | Fault Diameter 0.007", at Outer Race @ 6:00, 2 hp, 1750 rpm | 2 | 100% | 100% | 100% |
| 16 | Fault Diameter 0.007", at Outer Race @ 6:00, 3 hp, 1730 rpm | 2 | 100% | 100% | 100% |
| 17 | Fault Diameter 0.014", at Inner Race, 0 hp, 1797 rpm | 3 | 100% | 100% | 100% |
| 18 | Fault Diameter 0.014", at Inner Race, 1 hp, 1772 rpm | 3 | 100% | 100% | 100% |
| 19 | Fault Diameter 0.014", at Inner Race, 2 hp, 1750 rpm | 3 | 83% | 89% | 100% |
| 20 | Fault Diameter 0.014", at Inner Race, 3 hp, 1730 rpm | 3 | 100% | 100% | 100% |
| 21 | Fault Diameter 0.014", at Ball, 0 hp, 1797 rpm | 3 | 100% | 100% | 100% |
| 22 | Fault Diameter 0.014", at Ball, 1 hp, 1772 rpm | 3 | 100% | 100% | 100% |
| 23 | Fault Diameter 0.014", at Ball, 2 hp, 1750 rpm | 3 | 50% | 89% | 100% |
| 24 | Fault Diameter 0.014", at Outer Race @ 6:00, 0 hp, 1797 rpm | 3 | 100% | 100% | 100% |
| 25 | Fault Diameter 0.014", at Outer Race @ 6:00, 1 hp, 1772 rpm | 3 | 100% | 100% | 100% |
| 26 | Fault Diameter 0.014", at Outer Race @ 6:00, 2 hp, 1750 rpm | 3 | 100% | 100% | 100% |
| 27 | Fault Diameter 0.014", at Outer Race @ 6:00, 3 hp, 1730 rpm | 3 | 100% | 100% | 100% |
| 28 | Fault Diameter 0.021", at Inner Race, 0 hp, 1797 rpm | 4 | 100% | 100% | 100% |
| 29 | Fault Diameter 0.021", at Inner Race, 1 hp, 1772 rpm | 4 | 100% | 100% | 100% |
| 30 | Fault Diameter 0.021", at Inner Race, 2 hp, 1750 rpm | 4 | 100% | 100% | 100% |
| 31 | Fault Diameter 0.021", at Inner Race, 3 hp, 1730 rpm | 4 | 100% | 100% | 100% |
| 32 | Fault Diameter 0.021", at Ball, 0 hp, 1797 rpm | 4 | 83% | 100% | 100% |
| 33 | Fault Diameter 0.021", at Ball, 1 hp, 1772 rpm | 4 | 83% | 89% | 100% |
| 34 | Fault Diameter 0.021", at Ball, 2 hp, 1750 rpm | 4 | 100% | 100% | 100% |
| 35 | Fault Diameter 0.021", at Ball, 3 hp, 1730 rpm | 4 | 100% | 100% | 100% |
| 36 | Fault Diameter 0.021", at Outer Race @ 6:00, 0 hp, 1797 rpm | 4 | 41% | 89% | 100% |
| 37 | Fault Diameter 0.021", at Outer Race @ 6:00, 1 hp, 1772 rpm | 4 | 33% | 44% | 33% |
| 38 | Fault Diameter 0.021", at Outer Race @ 6:00, 2 hp, 1750 rpm | 4 | 41% | 55% | 33% |
| 39 | Fault Diameter 0.021", at Outer Race @ 6:00, 3 hp, 1730 pm | 4 | 17% | 22% | 17% |
| 40 | Fault Diameter 0.028", at Inner Race, 0 hp, 1797 rpm | 5 | 100% | 100% | 100% |
| 41 | Fault Diameter 0.028", at Inner Race, 1 hp, 1772 pm | 5 | 100% | 100% | 100% |
| 42 | Fault Diameter 0.028", at Inner Race, 2 hp, 1750 rpm | 5 | 100% | 100% | 100% |
| 43 | Fault Diameter 0.028", at Inner Race, 3 hp, 1730 rpm | 5 | 100% | 100% | 100% |
| 44 | Fault Diameter 0.028", at Ball, 0 hp, 1797 rpm | 5 | 100% | 100% | 100% |
| 45 | Fault Diameter 0.028", at Ball, 1 hp, 1772 rpm | 5 | 100% | 100% | 100% |
| 46 | Fault Diameter 0.028", at Ball, 2 hp, 1750 rpm | 5 | 100% | 100% | 100% |
| 47 | Fault Diameter 0.028", at Ball, 3 hp, 1730 rpm | 5 | 100% | 100% | 100% |
| **Average** | | | **91.81%** | **95.32%** | **95.38%** |

Table 5.7 FE bearing state detection efficiency.

| Case No. | Condition Description | Real State | 3 Observations | 4 Observations | 6 Observations |
|---|---|---|---|---|---|
| 1 | Normal, 0 hp, 1797 rpm | 1 | 100% | 100% | 100% |
| 2 | Normal, 1 hp, 1772 rpm | 1 | 100% | 100% | 100% |
| 3 | Normal, 2 hp, 1750 rpm | 1 | 100% | 100% | 100% |
| 4 | Normal, 3 hp, 1730 rpm | 1 | 100% | 100% | 100% |
| 5 | Fault Diameter 0.007", at Inner Race, 0 hp, 1797 rpm | 2 | 75% | 89% | 90% |
| 6 | Fault Diameter 0.007", at Inner Race, 1 hp, 1772 rpm | 2 | 100% | 100% | 100% |
| 7 | Fault Diameter 0.007", at Inner Race, 2hp, 1750 rpm | 2 | 83% | 89% | 100% |
| 8 | Fault Diameter 0.007", at Inner Race, 3 hp, 1730 rpm | 2 | 100% | 100% | 100% |
| 9 | Fault Diameter 0.007", at Ball, 0 hp, 1797 rpm | 2 | 83% | 89% | 100% |
| 10 | Fault Diameter 0.007", at Ball, 1 hp, 1772 rpm | 2 | 100% | 100% | 100% |
| 11 | Fault Diameter 0.007", at Ball, 2 hp, 1750 rpm | 2 | 100% | 100% | 100% |
| 12 | Fault Diameter 0.007", at Ball, 3 hp, 1730 rpm | 2 | 100% | 100% | 100% |
| 13 | Fault Diameter 0.007", at Outer Race @ 6:00, 0 hp, 1797 rpm | 2 | 100% | 100% | 100% |
| 14 | Fault Diameter 0.007", at Outer Race @ 6:00,1 hp, 1772 rpm | 2 | 100% | 100% | 100% |
| 15 | Fault Diameter 0.007", at Outer Race @ 6:00, 2 hp 1750 rpm | 2 | 100% | 100% | 100% |
| 16 | Fault Diameter 0.007", at Outer Race @ 6:00, 3 hp, 1730 rpm | 2 | 100% | 100% | 100% |
| 17 | Fault Diameter 0.014", at Inner Race, 0 hp , 1797 rpm | 3 | 100% | 100% | 100% |
| 18 | Fault Diameter 0.014", at Inner Race, 1 hp, 1772 rpm | 3 | 100% | 100% | 100% |
| 19 | Fault Diameter 0.014", at Inner Race, 3 hp, 1730 rpm | 3 | 100% | 100% | 100% |
| 20 | Fault Diameter 0.014", at Ball, 0 hp, 1797 rpm | 3 | 100% | 100% | 100% |
| 21 | Fault Diameter 0.014", at Ball, 2 hp, 1750 rpm | 3 | 100% | 100% | 100% |
| 22 | Fault Diameter 0.014", at Ball, 3 hp, 1730 rpm | 3 | 100% | 100% | 100% |
| 23 | Fault Diameter 0.014", at Outer Race @ 6:00, 0 hp, 1797 rpm | 3 | 100% | 100% | 100% |
| 24 | Fault Diameter 0.021", at Inner Race, 0 hp, 1797 rpm | 4 | 100% | 100% | 100% |
| 25 | Fault Diameter 0.021", at Inner Race, 1 hp, 1772 rpm | 4 | 100% | 100% | 100% |
| 26 | Fault Diameter 0.021", at Inner Race, 2 hp, 1750 rpm | 4 | 100% | 100% | 100% |
| 27 | Fault Diameter 0.021", at Inner Race, 3 hp, 1730 rpm | 4 | 100% | 100% | 100% |
| 28 | Fault Diameter 0.021", at Ball, 0 hp 1797 rpm | 4 | 100% | 100% | 100% |
| 29 | Fault Diameter 0.021", at Ball, 1 hp, 1772 rpm | 4 | 100% | 100% | 100% |
| 30 | Fault Diameter 0.021", at Ball, 2 hp, 1750 rpm | 4 | 83% | 89% | 100% |
| 31 | Fault Diameter 0.021", at Ball, 3 hp, 1730 rpm | 4 | 100% | 100% | 100% |
| 32 | Fault Diameter 0.021", at Outer Race @ 6:00, 0 hp, 1797 rpm | 4 | 100% | 100% | 100% |
| **Average** | | | **97.63%** | **98.66%** | **99.69%** |

Table 5.8 DE bearing state determination.

| Case No. | Condition Description | Real State | Probability | | | | | Detected State |
|---|---|---|---|---|---|---|---|---|
| | | | State 1 | State 2 | State 3 | State 4 | State 5 | |
| 1 | Normal, 0 hp, 1797 rpm | 1 | **0.302*** | 1E-16 | 1E-16 | 1E-16 | 1E-16 | 1 |
| 2 | Normal, 1 hp, 1772 rpm | 1 | **0.302** | 1E-16 | 1E-16 | 1E-16 | 1E-16 | 1 |
| 3 | Normal, 2 hp, 1750 rpm | 1 | **0.302** | 1E-16 | 1E-16 | 1E-16 | 1E-16 | 1 |
| 4 | Normal, 3 hp, 1730 rpm | 1 | **0.302** | 1E-16 | 1E-16 | 1E-16 | 1E-16 | 1 |
| 5 | Fault Diameter 0.007", at Inner Race, 0 hp, 1797 rpm | 2 | 1.4E-09 | **4.3E-06** | 1.3E-07 | 1.1E-08 | 1.1E-08 | 2 |
| 6 | Fault Diameter 0.007", at Inner Race, 1 hp, 1772 rpm | 2 | 1.4E-09 | **4.3E-06** | 1.3E-07 | 1.1E-08 | 1.1E-08 | 2 |
| 7 | Fault Diameter 0.007", at Inner Race, 2 hp, 1750 rpm | 2 | 1.4E-09 | **4.3E-06** | 1.3E-07 | 1.1E-08 | 1.1E-08 | 2 |
| 8 | Fault Diameter 0.007", at Inner Race, 3 hp, 1730 rpm | 2 | 4.0E-10 | **1.7E-07** | 5.8E-09 | 1.5E-09 | 1.7E-09 | 2 |
| 9 | Fault Diameter 0.007", at Ball, 0 hp, 1797 rpm | 2 | 4.4E-09 | **3.2E-05** | 1.7E-07 | 1.1E-06 | 4.3E-08 | 2 |
| 10 | Fault Diameter 0.007", at Ball, 1 hp, 1772 rpm | 2 | 4.4E-09 | **3.2E-05** | 1.7E-07 | 1.1E-06 | 4.3E-08 | 2 |
| 11 | Fault Diameter 0.007", at Ball, 2 hp, 1750 rpm | 2 | 4.4E-09 | **3.2E-05** | 1.7E-07 | 1.1E-06 | 4.3E-08 | 2 |
| 12 | Fault Diameter 0.007", at Ball, 3 hp, 1730 rpm | 2 | 4.4E-09 | **3.2E-05** | 1.7E-07 | 1.1E-06 | 4.3E-08 | 2 |
| 13 | Fault Diameter 0.007", at Outer Race @ 6:00, 0 hp, 1797 | 2 | 4.4E-09 | **3.2E-05** | 1.7E-07 | 1.1E-06 | 4.3E-08 | 2 |
| 14 | Fault Diameter 0.007", at Outer Race @ 6:00, 1 hp, 1772 | 2 | 4.4E-09 | **3.2E-05** | 1.7E-07 | 1.1E-06 | 4.3E-08 | 2 |
| 15 | Fault Diameter 0.007", at Outer Race @ 6:00, 2 hp, 1750 | 2 | 4.4E-09 | **3.2E-05** | 1.7E-07 | 1.1E-06 | 4.3E-08 | 2 |
| 16 | Fault Diameter 0.007", at Outer Race @ 6:00, 3 hp, 1730 | 2 | 4.4E-09 | **3.2E-05** | 1.7E-07 | 1.1E-06 | 4.3E-08 | 2 |
| 17 | Fault Diameter 0.014", at Inner Race, 0 hp, 1797 rpm | 3 | 3.9E-10 | 3.6E-09 | **1.8E-07** | 5.7E-09 | 2.0E-09 | 3 |
| 18 | Fault Diameter 0.014", at Inner Race, 1 hp, 1772 rpm | 3 | 6.0E-10 | 1.3E-07 | **6.6E-07** | 4.1E-07 | 9.1E-09 | 3 |
| 19 | Fault Diameter 0.014", at Inner Race, 2 hp, 1750 rpm | 3 | 9.7E-10 | 6.0E-07 | **6.2E-07** | 3.2E-07 | 1.1E-08 | 3 |
| 20 | Fault Diameter 0.014", at Inner Race, 3 hp, 1730 rpm | 3 | 3.4E-10 | 1.6E-09 | **9.6E-09** | 1.6E-09 | 1.0E-09 | 3 |
| 21 | Fault Diameter 0.014", at Ball, 0 hp, 1797 rpm | 3 | 4.4E-09 | 6.6E-07 | **4.9E-06** | 7.1E-08 | 4.3E-08 | 3 |
| 22 | Fault Diameter 0.014", at Ball, 1 hp, 1772 rpm | 3 | 4.9E-10 | 5.1E-09 | **4.7E-07** | 4.7E-09 | 2.3E-09 | 3 |
| 23 | Fault Diameter 0.014", at Ball, 2 hp, 1750 rpm | 3 | 3.6E-10 | 8.8E-09 | **9.2E-08** | 1.7E-09 | 1.4E-09 | 3 |
| 24 | Fault Diameter 0.014", at Outer Race @ 6:00, 0 hp, 1797 rpm | 3 | 1.1E-09 | 1.7E-08 | **3.5E-06** | 1.7E-08 | 7.3E-09 | 3 |
| 25 | Fault Diameter 0.014", at Outer Race @ 6:00, 1 hp, 1772 rpm | 3 | 1.1E-09 | 1.7E-08 | **3.5E-06** | 1.7E-08 | 7.3E-09 | 3 |
| 26 | Fault Diameter 0.014", at Outer Race @ 6:00, 2 hp, 1750 rpm | 3 | 1.1E-09 | 1.7E-08 | **3.5E-06** | 1.7E-08 | 7.3E-09 | 3 |
| 27 | Fault Diameter 0.014", at Outer Race @ 6:00, 3 hp, 1730 rpm | 3 | 1.1E-09 | 1.7E-08 | **3.5E-06** | 1.7E-08 | 7.3E-09 | 3 |
| 28 | Fault Diameter 0.021", at Inner Race, 0 hp, 1797 rpm | 4 | 3.1E-10 | 1.0E-08 | 9.3E-09 | **8.5E-07** | 7.6E-09 | 4 |
| 29 | Fault Diameter 0.021", at Inner Race, 1 hp, 1772 rpm | 4 | 3.1E-10 | 1.0E-08 | 9.3E-09 | **8.5E-07** | 7.6E-09 | 4 |
| 30 | Fault Diameter 0.021", at Inner Race, 2 hp, 1750 rpm | 4 | 3.1E-10 | 1.0E-08 | 9.3E-09 | **8.5E-07** | 7.6E-09 | 4 |
| 31 | Fault Diameter 0.021", at Inner Race, 3 hp, 1730 rpm | 4 | 3.1E-10 | 1.0E-08 | 9.3E-09 | **8.5E-07** | 7.6E-09 | 4 |
| 32 | Fault Diameter 0.021", at Ball, 0 hp, 1797 rpm | 4 | 1.1E-09 | 1.7E-09 | 3.5E-09 | **1.7E-08** | 7.3E-09 | 4 |
| 33 | Fault Diameter 0.021", at Ball, 1 hp, 1772 rpm | 4 | 4.0E-10 | 3.7E-09 | 2.5E-08 | **5.6E-08** | 2.0E-09 | 4 |
| 34 | Fault Diameter 0.021", at Ball, 2 hp, 1750 rpm | 4 | 3.9E-10 | 3.5E-09 | 5.1E-10 | **3.8E-09** | 1.9E-09 | 4 |
| 35 | Fault Diameter 0.021", at Ball, 3 hp, 1730 rpm | 4 | 2.9E-10 | 5.2E-09 | 4.7E-09 | **2.6E-07** | 3.8E-09 | 4 |
| 36 | Fault Diameter 0.021", at Outer Race @ 6:00, 0 hp, 1797 rpm | 4 | 1.2E-09 | 4.2E-07 | 9.3E-08 | **5.8E-07** | 1.3E-08 | 4 |
| 37 | Fault Diameter 0.021", at Outer Race @ 6:00, 1 hp, 1772 rpm | 4 | 4.0E-10 | 1.6E-08 | 1.9E-08 | **4.8E-08** | 1.6E-09 | 4 |
| 38 | Fault Diameter 0.021", at Outer Race @ 6:00, 2 hp, 1750 rpm | 4 | 4.4E-09 | 3.2E-07 | 1.7E-07 | **1.1E-06** | 4.3E-08 | 4 |
| 39 | Fault Diameter 0.021", at Outer Race @ 6:00, 3 hp, 1730 rpm | 4 | 5.8E-10 | 5.0E-08 | 1.2E-07 | **4.3E-07** | 7.4E-09 | 4 |
| 40 | Fault Diameter 0.028", at Inner Race, 0 hp, 1797 rpm | 5 | 1.1E-09 | 2.2E-09 | 2.1E-09 | 1.9E-09 | **3.7E-06** | 5 |
| 41 | Fault Diameter 0.028", at Inner Race, 1 hp, 1772 pm | 5 | 1.1E-09 | 2.2E-09 | 2.1E-09 | 1.9E-09 | **3.7E-06** | 5 |
| 42 | Fault Diameter 0.028", at Inner Race, 2 hp, 1750 rpm | 5 | 4.7E-10 | 1.0E-09 | 9.1E-10 | 7.1E-10 | **4.7E-07** | 5 |
| 43 | Fault Diameter 0.028", at Inner Race, 3 hp, 1730 rpm | 5 | 4.7E-10 | 1.0E-09 | 9.1E-10 | 7.1E-10 | **4.7E-07** | 5 |
| 44 | Fault Diameter 0.028", at Ball, 0 hp, 1797 rpm | 5 | 2.0E-09 | 4.0E-09 | 3.9E-09 | 3.7E-09 | **1.2E-05** | 5 |
| 45 | Fault Diameter 0.028", at Ball, 1 hp, 1772 rpm | 5 | 2.0E-09 | 4.0E-09 | 3.9E-09 | 3.7E-09 | **1.2E-05** | 5 |
| 46 | Fault Diameter 0.028", at Ball, 2 hp, 1750 rpm | 5 | 2.0E-09 | 4.0E-09 | 3.9E-09 | 3.7E-09 | **1.2E-05** | 5 |
| 47 | Fault Diameter 0.028", at Ball, 3 hp, 1730 rpm | 5 | 2.0E-09 | 4.0E-09 | 3.9E-09 | 3.7E-09 | **1.2E-05** | 5 |

Table 5.9 FE bearing state determination.

| Case No. | Condition Description | Real State | Probability | | | | Detected State |
|---|---|---|---|---|---|---|---|
| | | | State 1 | State 2 | State 3 | State 4 | |
| 1 | Normal, 0 hp, 1797 rpm | 1 | **0.255*** | 0.000 | 0.000 | 0.000 | 1 |
| 2 | Normal, 1 hp, 1772 rpm | 1 | **0.255** | 0.000 | 0.000 | 0.000 | 1 |
| 3 | Normal, 2 hp, 1750 rpm | 1 | **0.255** | 0.000 | 0.000 | 0.000 | 1 |
| 4 | Normal, 3 hp, 1730 rpm | 1 | **0.255** | 0.000 | 0.000 | 0.000 | 1 |
| 5 | Fault Diameter 0.007", at Inner Race, 0 hp, 1797 rpm | 2 | 6.0E-09 | **1.2E-06** | 8.8E-09 | 1.1E-08 | 2 |
| 6 | Fault Diameter 0.007", at Inner Race, 1 hp, 1772 rpm | 2 | 8.8E-09 | **5.5E-06** | 3.7E-08 | 3.0E-08 | 2 |
| 7 | Fault Diameter 0.007", at Inner Race, 2hp, 1750 rpm | 2 | 8.8E-09 | **5.5E-06** | 3.7E-08 | 3.0E-08 | 2 |
| 8 | Fault Diameter 0.007", at Inner Race, 3 hp, 1730 rpm | 2 | 6.1E-09 | **7.3E-07** | 1.1E-08 | 9.1E-09 | 2 |
| 9 | Fault Diameter 0.007", at Ball, 0 hp, 1797 rpm | 2 | 5.8E-09 | **3.5E-08** | 2.5E-08 | 1.9E-08 | 2 |
| 10 | Fault Diameter 0.007", at Ball, 1 hp, 1772 rpm | 2 | 5.4E-09 | **1.0E-07** | 4.1E-09 | 3.5E-09 | 2 |
| 11 | Fault Diameter 0.007", at Ball, 2 hp, 1750 rpm | 2 | 7.2E-09 | **2.4E-06** | 2.2E-08 | 1.8E-08 | 2 |
| 12 | Fault Diameter 0.007", at Ball, 3 hp, 1730 rpm | 2 | 5.7E-09 | **6.1E-07** | 6.4E-09 | 5.3E-09 | 2 |
| 13 | Fault Diameter 0.007", at Outer Race @ 6:00, 0 hp, 1797 rpm | 2 | 6.1E-09 | **3.0E-08** | 1.1E-09 | 5.0E-09 | 2 |
| 14 | Fault Diameter 0.007", at Outer Race @ 6:00,1 hp, 1772 rpm | 2 | 7.5E-09 | **2.2E-06** | 1.5E-08 | 4.8E-07 | 2 |
| 15 | Fault Diameter 0.007", at Outer Race @ 6:00, 2 hp 1750 rpm | 2 | 9.3E-09 | **6.5E-06** | 4.1E-08 | 2.5E-07 | 2 |
| 16 | Fault Diameter 0.007", at Outer Race @ 6:00, 3 hp, 1730 rpm | 2 | 9.3E-09 | **6.5E-06** | 4.1E-08 | 2.5E-07 | 2 |
| 17 | Fault Diameter 0.014", at Inner Race, 0 hp , 1797 rpm | 3 | 9.3E-09 | **6.5E-06** | 4.1E-08 | 2.5E-07 | 3 |
| 18 | Fault Diameter 0.014", at Inner Race, 1 hp, 1772 rpm | 3 | 7.3E-09 | 4.2E-09 | **2.6E-06** | 1.0E-08 | 3 |
| 19 | Fault Diameter 0.014", at Inner Race, 3 hp, 1730 rpm | 3 | 1.0E-08 | 7.2E-09 | **1.5E-05** | 2.2E-08 | 3 |
| 20 | Fault Diameter 0.014", at Ball, 0 hp, 1797 rpm | 3 | 1.0E-08 | 7.2E-09 | **1.5E-05** | 2.2E-08 | 3 |
| 21 | Fault Diameter 0.014", at Ball, 2 hp, 1750 rpm | 3 | 6.3E-09 | 3.3E-09 | **1.4E-06** | 6.0E-09 | 3 |
| 22 | Fault Diameter 0.014", at Ball, 3 hp, 1730 rpm | 3 | 5.9E-09 | 2.9E-09 | **7.6E-07** | 4.4E-09 | 3 |
| 23 | Fault Diameter 0.014", at Outer Race @ 6:00, 0 hp, 1797 rpm | 3 | 5.8E-09 | 2.7E-09 | **6.5E-07** | 4.0E-09 | 3 |
| 24 | Fault Diameter 0.021", at Inner Race, 0 hp, 1797 rpm | 4 | 6.2E-09 | 3.1E-09 | **1.3E-06** | 5.5E-09 | 4 |
| 25 | Fault Diameter 0.021", at Inner Race, 1 hp, 1772 rpm | 4 | 7.9E-09 | 2.9E-07 | 3.0E-08 | **6.8E-06** | 4 |
| 26 | Fault Diameter 0.021", at Inner Race, 2 hp, 1750 rpm | 4 | 1.3E-08 | 3.3E-07 | 2.8E-08 | **2.9E-05** | 4 |
| 27 | Fault Diameter 0.021", at Inner Race, 3 hp, 1730 rpm | 4 | 7.9E-09 | 2.9E-07 | 3.0E-08 | **6.8E-06** | 4 |
| 28 | Fault Diameter 0.021", at Ball, 0 hp 1797 rpm | 4 | 1.3E-08 | 3.3E-07 | 2.8E-08 | **2.9E-05** | 4 |
| 29 | Fault Diameter 0.021", at Ball, 1 hp, 1772 rpm | 4 | 7.1E-09 | 5.3E-08 | 3.5E-09 | **1.7E-06** | 4 |
| 30 | Fault Diameter 0.021", at Ball, 2 hp, 1750 rpm | 4 | 6.2E-09 | 4.6E-08 | 6.9E-09 | **9.4E-07** | 4 |
| 31 | Fault Diameter 0.021", at Ball, 3 hp, 1730 rpm | 4 | 5.8E-09 | 4.1E-08 | 3.1E-09 | **4.8E-07** | 4 |
| 32 | Fault Diameter 0.021", at Outer Race @ 6:00, 0 hp, 1797 rpm | 4 | 5.8E-09 | 7.1E-09 | 2.1E-09 | **6.0E-07** | 4 |

* Note: the predicted state in Tables 5.8 and 5.9 is associated with the largest probability number (in bold).

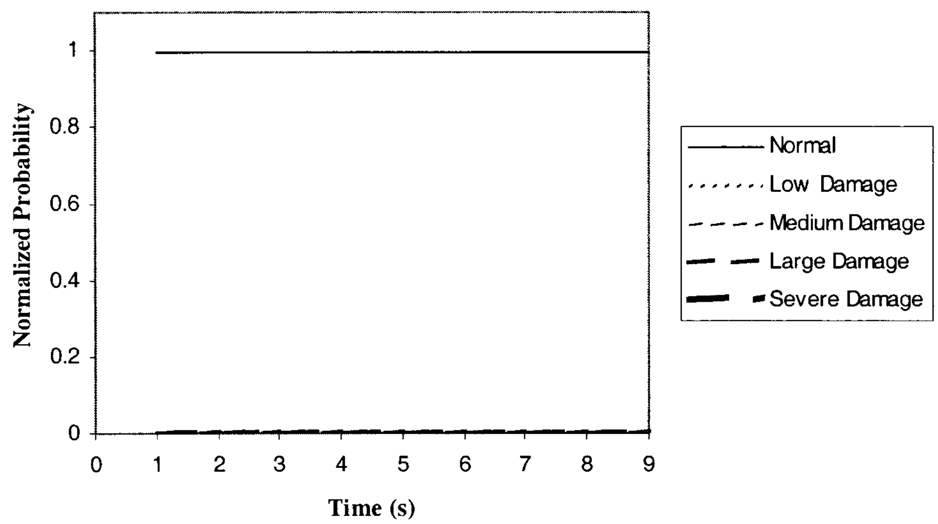



Figure 5.23 Normalized probability for drive end bearing, normal condition at 1797 rpm and 0 hp motor load.

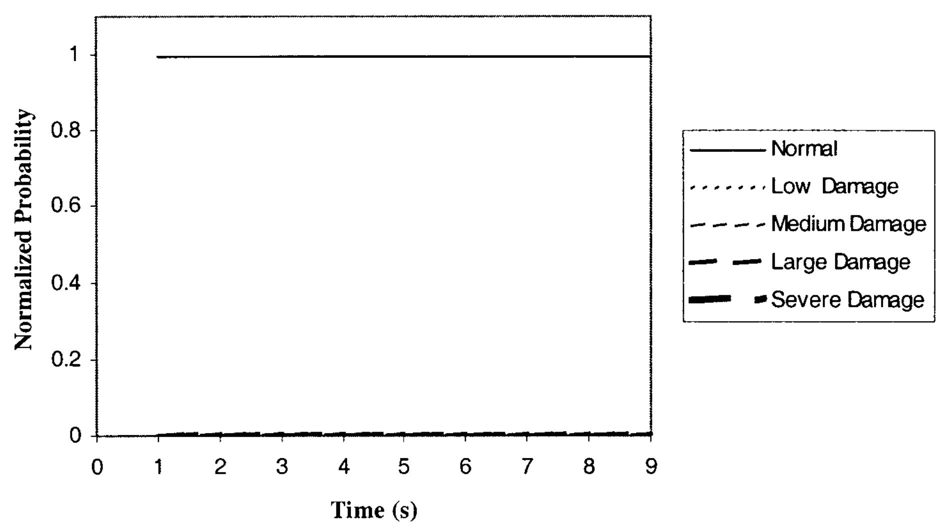



Figure 5.24 Normalized probability for drive end bearing, normal condition at 1730 rpm and 3 hp motor load.

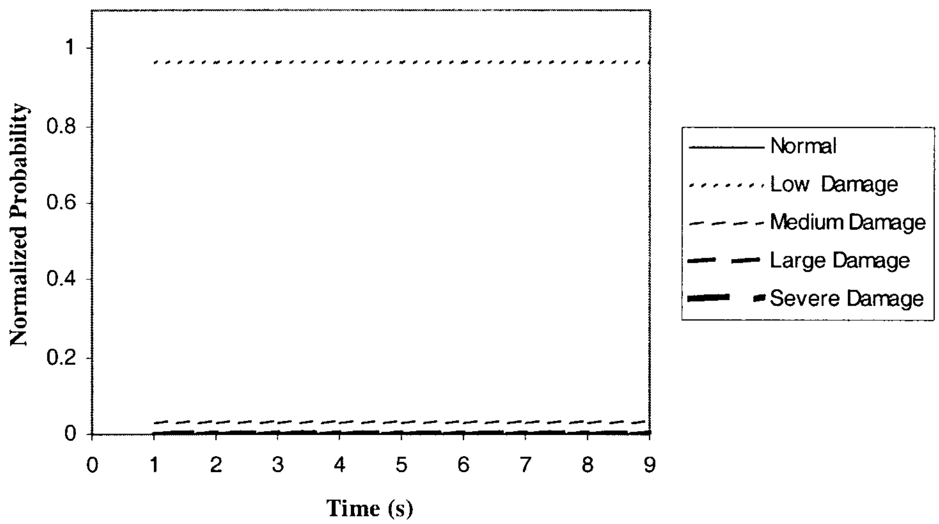



Figure 5.25 Normalized probability for drive end bearing, 0.007” fault diameter at inner race, 1797 rpm and 0 hp motor load.

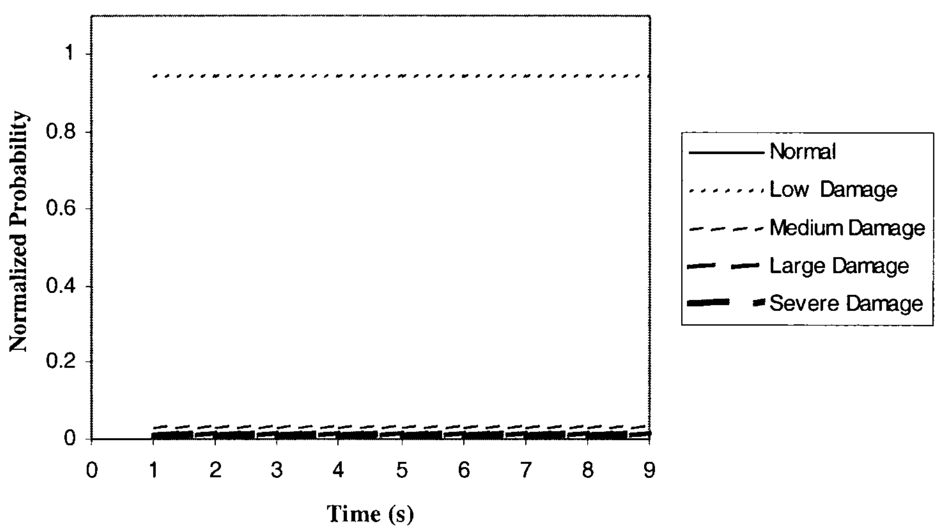



Figure 5.26 Normalized probability for drive end bearing, 0.007” fault diameter at inner race, 1730 rpm and 3 hp motor load.

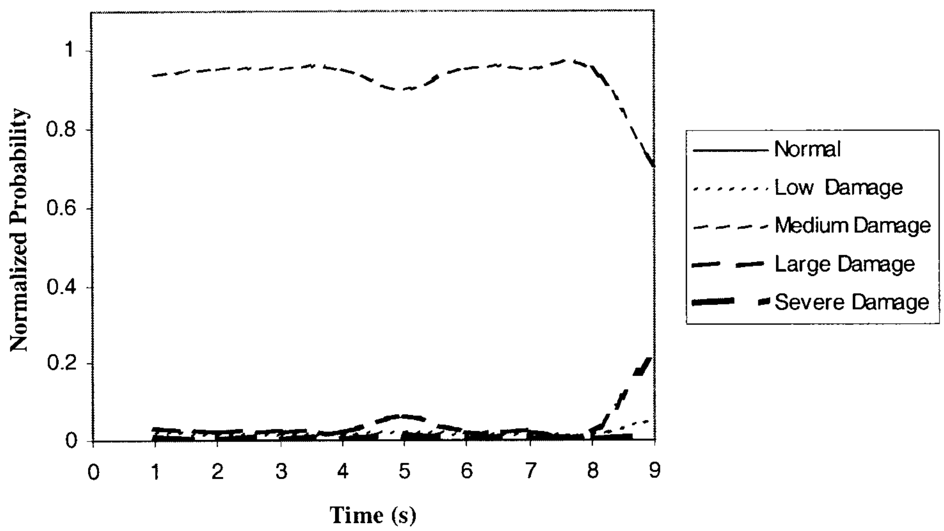



Figure 5.27 Normalized probability for drive end bearing, 0.014" fault diameter at inner race, 1797 rpm and 0 hp motor load.

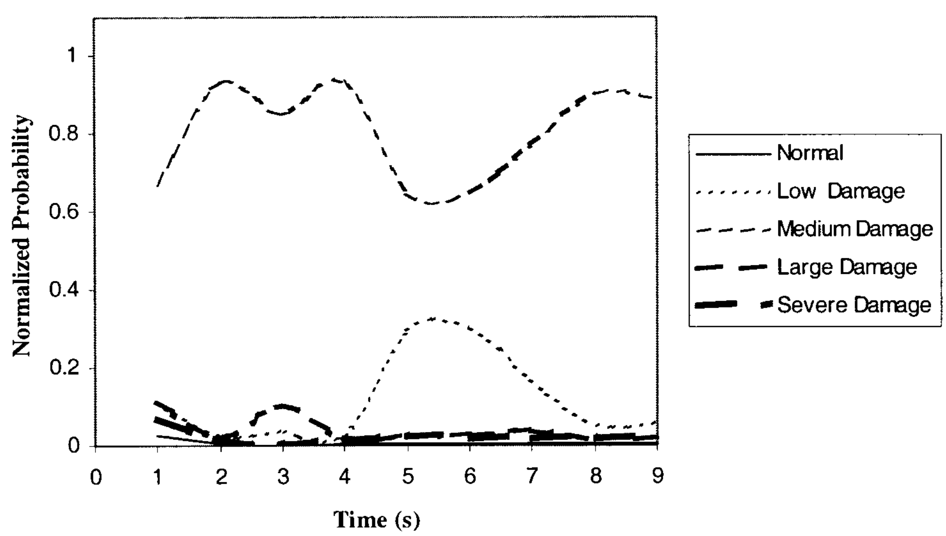



Figure 5.28 Normalized probability for drive end bearing, 0.014" fault diameter at inner race, 1730 rpm and 3 hp motor load.

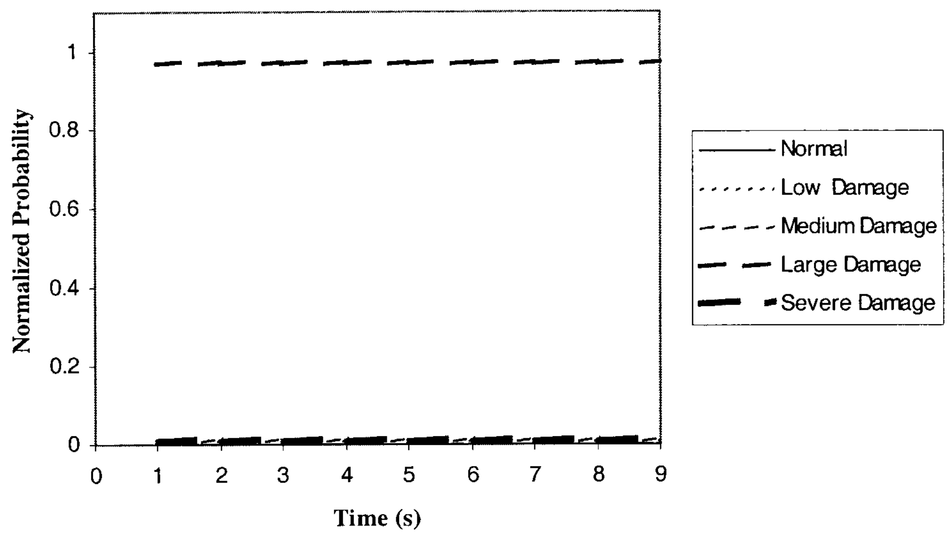



Figure 5.29 Normalized probability for drive end bearing, 0.021" fault diameter at inner race, 1797 rpm and 0 hp motor load.

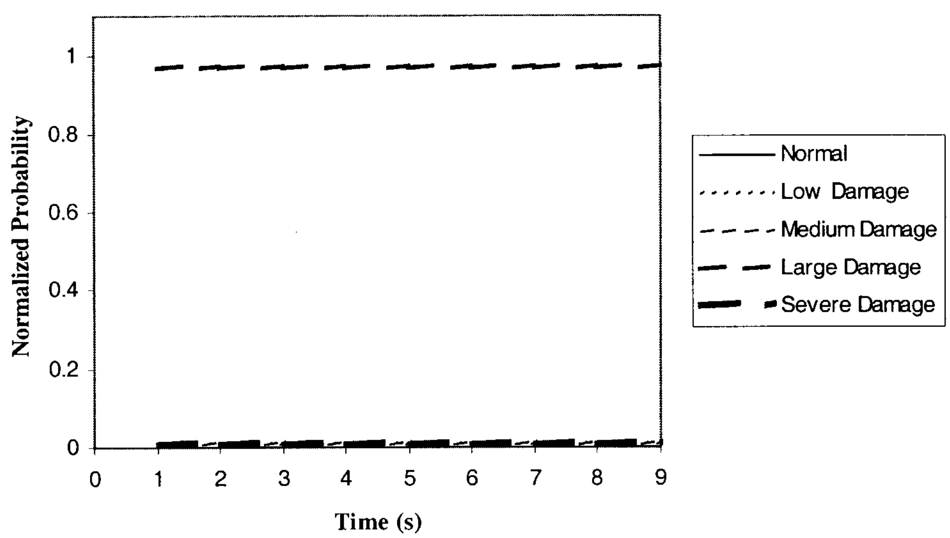



Figure 5.30 Normalized probability for drive end bearing, 0.021" fault diameter at inner race, 1730 rpm and 3 hp motor load.

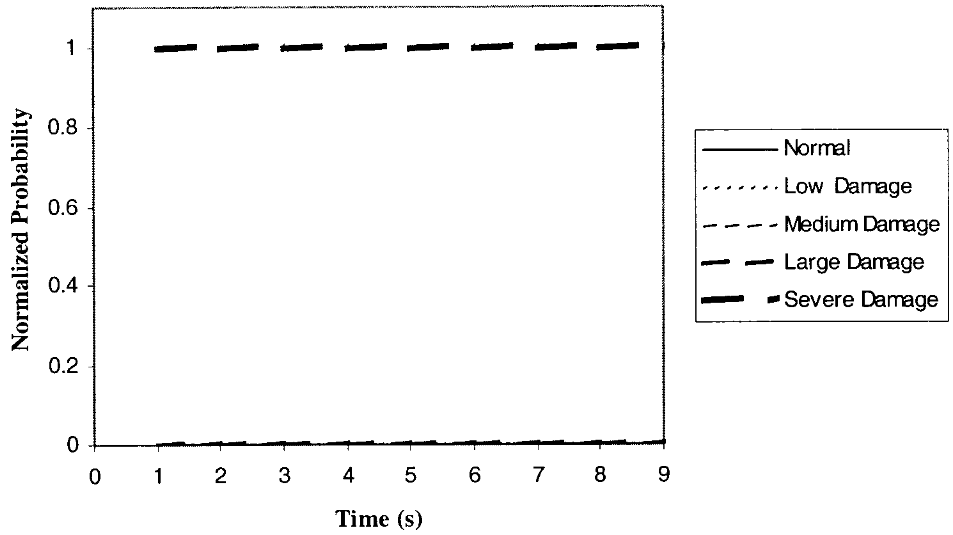



Figure 5.31 Normalized probability for drive end bearing, 0.028" fault diameter at inner race, 1797 rpm and 0 hp motor load.

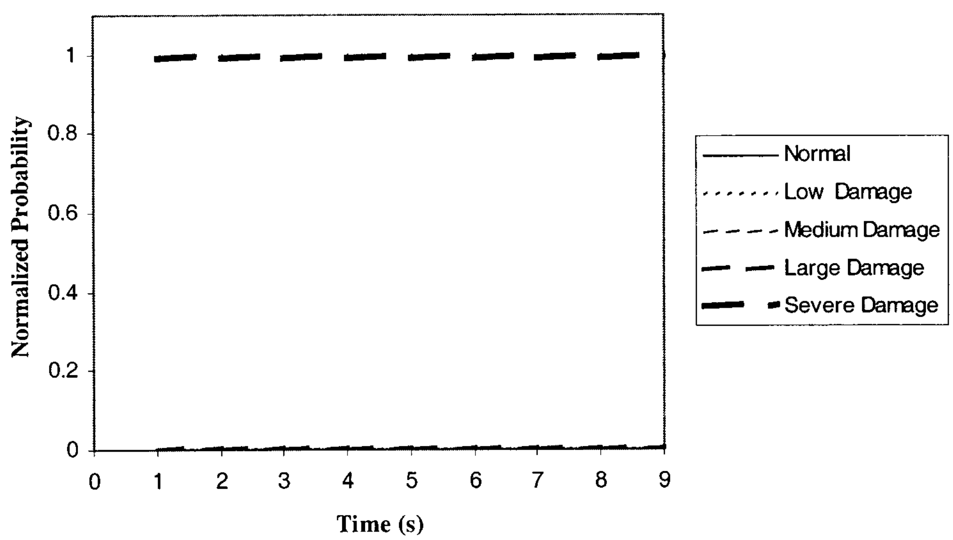



Figure 5.32 Normalized probability for drive end bearing, 0.028" fault diameter at inner race, 1730 rpm and 3 hp motor load.

### 5.3.3 Fault Localization Implementation

The fault localization concept is applied in this case to both DE and FE bearings. Based on the given data, the number of fault locations (*NFL*) is three:

Location1: inner race (i.e. $nfl = 1$)

Location 2: ball (i.e. $nfl = 2$)

Location 3: outer race (i.e. $nfl = 3$)

Furthermore, as the number of symbols in the DE and FE codebooks is seventeen, the *FLIM* dimension for each bearing is ($3 \times 17$).

During the training exercise, the *FLIM* matrices for both DE and FE bearings are prepared and shown below.

$$[FLIM]_{DE} = \begin{array}{c} \begin{array}{ccccccccccccccccc} sym1 & sym2 & . & . & & & & & & & & & & . & . & sym16 & sym17 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$[FLIM]_{FE} = \begin{array}{c} \begin{array}{ccccccccccccccccc} sym1 & sym2 & . & . & . & & & & & & & & & . & . & sym16 & sym17 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

To explain the meaning of these matrices, let's take for example the first row of the DE *FLIM* matrix. It means that fault location 1 (inner race) can be represented by the following symbols (or reference vectors): 2,3,5,6,7,10,14,15 and 16 which correspond to non-zero positions in the first row of the matrix. It is important also to note that some symbols can represent sometimes more than one location. For example symbol 4 represents location 2 (ball) and location 3 (outer race). This means that if the sequence of observations is formed only by symbol 4, the result can lead to two locations (2 and 3). Although the solution in this case might not look conclusive, it is still very useful for monitoring purposes because the domain of focus has been narrowed down.

For each sequence of four observations ($NOS = 4$), and using equation (5.41), the location index ($LI$) is computed for the pre-defined locations. Samples of the results are shown in Tables 5.10 and 5.11 for the DE and FE bearings respectively.

To illustrate the method, assume that a sequence of four observations taken for the DE bearing matches the following series of symbols (or reference vector) (symbol 2, symbol 3, symbol 2, symbol 4). Hence,

$$ref(o_1) = 2\,;\ ref(o_2) = 3\,;\ ref(o_3) = 2\,;\ ref(o_4) = 4$$

Using equation (5.41), the location index can be calculated as follows:

$$LI(1) = \sum_{m=1}^{4} FLIM(1, ref(o_m)) = FLIM(1,2) + FLIM(1,3) + FLIM(1,2) + FLIM(1,4) = 1 + 1 + 0 + 1 = 3$$

$$LI(2) = \sum_{m=1}^{4} FLIM(2, ref(o_m)) = FLIM(2,2) + FLIM(2,3) + FLIM(2,2) + FLIM(2,4) = 0 + 0 + 0 + 1 = 1$$

$$LI(3) = \sum_{m=1}^{4} FLIM(3, ref(o_m)) = FLIM(3,2) + FLIM(3,3) + FLIM(3,2) + FLIM(3,4) = 0 + 0 + 0 + 1 = 1$$

Utilizing equation (5.42), the location is determined based on the maximum value.

$$i^* = \arg\max(LI(1), LI(2), LI(3)) = \arg\max(3,1,1) = 1$$

hence, the fault is placed at location 1 (inner race).

Looking at Table 5.10, the location index for the ball and outer race leads in few cases to a dual solution (e.g. condition 17-20). This is due to the fact that some symbols (e.g. 4,12) represent both locations as shown in the DE *FLIM* matrix. However, the inner race location is singly identified in all cases. Along this line, this type of dual solutions is not observed in Table 5.11. All the fault locations are correctly identified by the location index.

Table 5.10 Location index for drive end bearing.

| Condition Description | Motor Load (hp) | Motor Speed (rpm) | Location Index (*LI*) | | | Detected Fault Location |
|---|---|---|---|---|---|---|
| | | | Inner Race | Ball | Outer Race | |
| 1- Fault Diameter 0.007", at Inner Race | 0 | 1797 | **4*** | 0 | 0 | Inner |
| 2- Fault Diameter 0.007", at Inner Race | 1 | 1772 | **4** | 0 | 0 | Inner |
| 3- Fault Diameter 0.007", at Inner Race | 2 | 1750 | **4** | 0 | 0 | Inner |
| 4- Fault Diameter 0.007", at Inner Race | 3 | 1730 | **4** | 0 | 0 | Inner |
| 5- Fault Diameter 0.014", at Inner Race | 0 | 1797 | **3** | 0 | 1 | Inner |
| 6- Fault Diameter 0.014", at Inner Race | 1 | 1772 | **3** | 1 | 1 | Inner |
| 7- Fault Diameter 0.014", at Inner Race | 2 | 1750 | **4** | 0 | 0 | Inner |
| 8- Fault Diameter 0.014", at Inner Race | 3 | 1730 | **4** | 0 | 0 | Inner |
| 9- Fault Diameter 0.021", at Inner Race | 0 | 1797 | **4** | 0 | 0 | Inner |
| 10- Fault Diameter 0.021", at Inner Race | 1 | 1772 | **4** | 0 | 0 | Inner |
| 11- Fault Diameter 0.021", at Inner Race | 2 | 1750 | **4** | 0 | 0 | Inner |
| 12- Fault Diameter 0.021", at Inner Race | 3 | 1730 | **4** | 0 | 0 | Inner |
| 13- Fault Diameter 0.028", at Inner Race | 0 | 1797 | **4** | 0 | 0 | Inner |
| 14- Fault Diameter 0.028", at Inner Race | 1 | 1772 | **4** | 0 | 0 | Inner |
| 15- Fault Diameter 0.028", at Inner Race | 2 | 1750 | **4** | 0 | 0 | Inner |
| 16- Fault Diameter 0.028", at Inner Race | 3 | 1730 | **4** | 0 | 0 | Inner |
| 17- Fault Diameter 0.007", at Ball | 0 | 1797 | 1 | **3** | **3** | Ball or Outer |
| 18- Fault Diameter 0.007", at Ball | 1 | 1772 | 1 | **3** | **3** | Ball or Outer |
| 19- Fault Diameter 0.007", at Ball | 2 | 1750 | 0 | **4** | **4** | Ball or Outer |
| 20- Fault Diameter 0.007", at Ball | 3 | 1730 | 0 | **4** | **4** | Ball or Outer |
| 21- Fault Diameter 0.014", at Ball | 0 | 1797 | 0 | **3** | 1 | Ball |
| 22- Fault Diameter 0.014", at Ball | 1 | 1772 | 0 | **4** | 0 | Ball |
| 23- Fault Diameter 0.014", at Ball | 2 | 1750 | 1 | **3** | 0 | Ball |
| 24- Fault Diameter 0.021", at Ball | 0 | 1797 | 0 | **4** | **4** | Ball or Outer |
| 25- Fault Diameter 0.021", at Ball | 1 | 1772 | 0 | **4** | **4** | Ball or Outer |
| 26- Fault Diameter 0.021", at Ball | 2 | 1750 | 0 | **4** | 0 | Ball |
| 27- Fault Diameter 0.021", at Ball | 3 | 1730 | 0 | **4** | 0 | Ball |
| 28- Fault Diameter 0.028", at Ball | 0 | 1797 | 0 | **4** | 0 | Ball |
| 29- Fault Diameter 0.028", at Ball | 1 | 1772 | 0 | **4** | 0 | Ball |
| 30- Fault Diameter 0.028", at Ball | 2 | 1750 | 0 | **4** | 0 | Ball |
| 31- Fault Diameter 0.028", at Ball | 3 | 1730 | 0 | **4** | 0 | Ball |
| 32- Fault Diameter 0.007", at Outer Race | 0 | 1797 | 0 | **4** | **4** | Ball or Outer |
| 33- Fault Diameter 0.007", at Outer Race | 1 | 1772 | 0 | **4** | **4** | Ball or Outer |
| 34- Fault Diameter 0.007", at Outer Race | 2 | 1750 | 0 | **4** | **4** | Ball or Outer |
| 35- Fault Diameter 0.007", at Outer Race | 3 | 1730 | 0 | **4** | **4** | Ball or Outer |
| 36- Fault Diameter 0.014", at Outer Race | 0 | 1797 | 0 | 0 | **4** | Outer Race |
| 37- Fault Diameter 0.014", at Outer Race | 1 | 1772 | 0 | 0 | **4** | Outer Race |
| 38- Fault Diameter 0.014", at Outer Race | 2 | 1750 | 0 | 0 | **4** | Outer Race |
| 39- Fault Diameter 0.014", at Outer Race | 3 | 1730 | 0 | 0 | **4** | Outer Race |
| 40- Fault Diameter 0.021", at Outer Race | 0 | 1797 | 0 | 2 | **4** | Outer Race |
| 41- Fault Diameter 0.021", at Outer Race | 1 | 1772 | 1 | 0 | **3** | Outer Race |
| 42- Fault Diameter 0.021", at Outer Race | 2 | 1750 | 0 | **4** | **4** | Ball or Outer |
| 43- Fault Diameter 0.021", at Outer Race | 3 | 1730 | 0 | **4** | **4** | Ball or Outer |

* Note: the predicted fault location(s) is associated with the largest number (in bold).

Table 5.11Location index for fan end bearing.

| Condition Description | Motor Load (hp) | Motor Speed (rpm) | Location Index (*LI*) | | | Detected Fault Location |
|---|---|---|---|---|---|---|
| | | | Inner Race | Ball | Outer Race | |
| 1-Fault Diameter 0.007", at Inner Race | 0 | 1797 | **4*** | 0 | 0 | Inner |
| 2- Fault Diameter 0.007", at Inner Race | 1 | 1772 | **4** | 0 | 0 | Inner |
| 3- Fault Diameter 0.007", at Inner Race | 2 | 1750 | **4** | 0 | 0 | Inner |
| 4- Fault Diameter 0.007", at Inner Race | 3 | 1730 | **4** | 0 | 0 | Inner |
| 5- Fault Diameter 0.014", at Inner Race | 0 | 1797 | **4** | 0 | 1 | Inner |
| 6- Fault Diameter 0.014", at Inner Race | 1 | 1772 | **4** | 1 | 1 | Inner |
| 7- Fault Diameter 0.014", at Inner Race | 3 | 1730 | **4** | 0 | 0 | Inner |
| 8- Fault Diameter 0.021", at Inner Race | 0 | 1797 | **4** | 0 | 0 | Inner |
| 9- Fault Diameter 0.021", at Inner Race | 1 | 1772 | **4** | 0 | 0 | Inner |
| 10- Fault Diameter 0.021", at Inner Race | 2 | 1750 | **4** | 0 | 0 | Inner |
| 11- Fault Diameter 0.021", at Inner Race | 3 | 1730 | **4** | 0 | 0 | Inner |
| 12- Fault Diameter 0.007", at Ball | 0 | 1797 | 0 | **4** | 0 | Ball |
| 13- Fault Diameter 0.007", at Ball | 1 | 1772 | 0 | **4** | 0 | Ball |
| 14- Fault Diameter 0.007", at Ball | 2 | 1750 | 0 | **4** | 0 | Ball |
| 15- Fault Diameter 0.007", at Ball | 3 | 1730 | 0 | **4** | 0 | Ball |
| 16- Fault Diameter 0.014", at Ball | 0 | 1797 | 0 | **4** | 0 | Ball |
| 17- Fault Diameter 0.014", at Ball | 2 | 1750 | 0 | **4** | 0 | Ball |
| 18- Fault Diameter 0.014", at Ball | 3 | 1730 | 0 | **4** | 0 | Ball |
| 19- Fault Diameter 0.021", at Ball | 0 | 1797 | 0 | **4** | 0 | Ball |
| 20- Fault Diameter 0.021", at Ball | 1 | 1772 | 0 | **4** | 0 | Ball |
| 21- Fault Diameter 0.021", at Ball | 2 | 1750 | 0 | **4** | 0 | Ball |
| 22- Fault Diameter 0.021", at Ball | 3 | 1730 | 0 | **3** | 1 | Ball |
| 23- Fault Diameter 0.007", at Outer Race | 0 | 1797 | 1 | 0 | **3** | Outer Race |
| 24- Fault Diameter 0.007", at Outer Race | 1 | 1772 | 0 | 0 | **4** | Outer Race |
| 25- Fault Diameter 0.007", at Outer Race | 2 | 1750 | 0 | 0 | **4** | Outer Race |
| 26- Fault Diameter 0.007", at Outer Race | 3 | 1730 | 0 | 0 | **4** | Outer Race |
| 27- Fault Diameter 0.014", at Outer Race | 0 | 1797 | 0 | 0 | **4** | Outer Race |
| 28- Fault Diameter 0.021", at Outer Race | 0 | 1797 | 0 | 0 | **4** | Outer Race |

* Note: the predicted fault location(s) is associated with the largest number (in bold).

To determine the detection efficiency of the fault localization method described above, a total of 36 observations are used in every case. Each decision is based on a series of four (4) successive observations. Tables 5.12 and 5.13 summarize the results for both DE and FE bearings. The efficiency is defined as the ratio of the number of times the location is correctly identified to the total number of tests. In both tables, the average efficiency is above 96%. In Table 5.12, the lowest efficiency is experienced for the outer race location (condition 40 – 56%). Very similar results were observed in Table 5.6 for the DE bearing with 0.021" fault at outer race. The reason behind this low efficiency (as explained above) is that the observations are continuously varying suggesting as such that the vibration signal is not consistently collected.

Table 5.12 Localization method efficiency for drive end bearing.

| Condition Description | Motor Load (hp) | Motor Speed (rpm) | Detected Fault Location | Detection Efficiency |
|---|---|---|---|---|
| 1- Fault Diameter 0.007", at Inner Race | 0 | 1797 | Inner | 100% |
| 2- Fault Diameter 0.007", at Inner Race | 1 | 1772 | Inner | 100% |
| 3- Fault Diameter 0.007", at Inner Race | 2 | 1750 | Inner | 100% |
| 4- Fault Diameter 0.007", at Inner Race | 3 | 1730 | Inner | 100% |
| 5- Fault Diameter 0.014", at Inner Race | 0 | 1797 | Inner | 100% |
| 6- Fault Diameter 0.014", at Inner Race | 1 | 1772 | Inner | 100% |
| 7- Fault Diameter 0.014", at Inner Race | 2 | 1750 | Inner | 100% |
| 8- Fault Diameter 0.014", at Inner Race | 3 | 1730 | Inner | 100% |
| 9- Fault Diameter 0.021", at Inner Race | 0 | 1797 | Inner | 100% |
| 10- Fault Diameter 0.021", at Inner Race | 1 | 1772 | Inner | 100% |
| 11- Fault Diameter 0.021", at Inner Race | 2 | 1750 | Inner | 100% |
| 12- Fault Diameter 0.021", at Inner Race | 3 | 1730 | Inner | 100% |
| 13- Fault Diameter 0.028", at Inner Race | 0 | 1797 | Inner | 100% |
| 14- Fault Diameter 0.028", at Inner Race | 1 | 1772 | Inner | 100% |
| 15- Fault Diameter 0.028", at Inner Race | 2 | 1750 | Inner | 100% |
| 16- Fault Diameter 0.028", at Inner Race | 3 | 1730 | Inner | 100% |
| 17- Fault Diameter 0.007", at Ball | 0 | 1797 | Ball or Outer Race | 78% |
| 18- Fault Diameter 0.007", at Ball | 1 | 1772 | Ball or Outer Race | 100% |
| 19- Fault Diameter 0.007", at Ball | 2 | 1750 | Ball or Outer Race | 100% |
| 20- Fault Diameter 0.007", at Ball | 3 | 1730 | Ball or Outer Race | 100% |
| 21- Fault Diameter 0.014", at Ball | 0 | 1797 | Ball | 78% |
| 22- Fault Diameter 0.014", at Ball | 1 | 1772 | Ball | 78% |
| 23- Fault Diameter 0.014", at Ball | 2 | 1750 | Ball | 67% |
| 24- Fault Diameter 0.021", at Ball | 0 | 1797 | Ball or Outer Race | 78% |
| 25- Fault Diameter 0.021", at Ball | 1 | 1772 | Ball or Outer Race | 100% |
| 26- Fault Diameter 0.021", at Ball | 2 | 1750 | Ball | 100% |
| 27- Fault Diameter 0.021", at Ball | 3 | 1730 | Ball | 100% |
| 28- Fault Diameter 0.028", at Ball | 0 | 1797 | Ball | 100% |
| 29- Fault Diameter 0.028", at Ball | 1 | 1772 | Ball | 100% |
| 30- Fault Diameter 0.028", at Ball | 2 | 1750 | Ball | 100% |
| 31- Fault Diameter 0.028", at Ball | 3 | 1730 | Ball | 100% |
| 32- Fault Diameter 0.007", at Outer Race | 0 | 1797 | Ball or Outer Race | 100% |
| 33- Fault Diameter 0.007", at Outer Race | 1 | 1772 | Ball or Outer Race | 89% |
| 34- Fault Diameter 0.007", at Outer Race | 2 | 1750 | Ball or Outer Race | 100% |
| 35- Fault Diameter 0.007", at Outer Race | 3 | 1730 | Ball or Outer Race | 100% |
| 36- Fault Diameter 0.014", at Outer Race | 0 | 1797 | Outer Race | 100% |
| 37- Fault Diameter 0.014", at Outer Race | 1 | 1772 | Outer Race | 100% |
| 38- Fault Diameter 0.014", at Outer Race | 2 | 1750 | Outer Race | 100% |
| 39- Fault Diameter 0.014", at Outer Race | 3 | 1730 | Outer Race | 100% |
| 40- Fault Diameter 0.021", at Outer Race | 0 | 1797 | Outer Race | 56% |
| 41- Fault Diameter 0.021", at Outer Race | 1 | 1772 | Ball or Outer Race | 79% |
| 42- Fault Diameter 0.021", at Outer Race | 2 | 1750 | Ball or Outer Race | 67% |
| 43- Fault Diameter 0.021", at Outer Race | 3 | 1730 | Ball or Outer Race | 78% |
| **Average Detection Efficiency** | | | | **96.4%** |

Table 5.13 Localization method efficiency for fan end bearing.

| Condition Description | Motor Load (hp) | Motor Speed (rpm) | Detected Fault Location | Detection Efficiency |
|---|---|---|---|---|
| 1-Fault Diameter 0.007", at Inner Race | 1 | 1772 | Inner | 100% |
| 2- Fault Diameter 0.007", at Inner Race | 2 | 1750 | Inner | 100% |
| 3- Fault Diameter 0.007", at Inner Race | 3 | 1730 | Inner | 100% |
| 4- Fault Diameter 0.007", at Inner Race | 0 | 1797 | Inner | 100% |
| 5- Fault Diameter 0.014", at Inner Race | 1 | 1772 | Inner | 100% |
| 6- Fault Diameter 0.014", at Inner Race | 3 | 1730 | Inner | 100% |
| 7- Fault Diameter 0.014", at Inner Race | 0 | 1797 | Inner | 100% |
| 8- Fault Diameter 0.021", at Inner Race | 1 | 1772 | Inner | 100% |
| 9- Fault Diameter 0.021", at Inner Race | 2 | 1750 | Inner | 100% |
| 10- Fault Diameter 0.021", at Inner Race | 3 | 1730 | Inner | 100% |
| 11- Fault Diameter 0.021", at Inner Race | 0 | 1797 | Ball | 89% |
| 12- Fault Diameter 0.007", at Ball | 1 | 1772 | Ball | 100% |
| 13- Fault Diameter 0.007", at Ball | 2 | 1750 | Ball | 100% |
| 14- Fault Diameter 0.007", at Ball | 3 | 1730 | Ball | 100% |
| 15- Fault Diameter 0.007", at Ball | 0 | 1797 | Ball | 89% |
| 16- Fault Diameter 0.014", at Ball | 2 | 1750 | Ball | 100% |
| 17- Fault Diameter 0.014", at Ball | 3 | 1730 | Ball | 100% |
| 18- Fault Diameter 0.014", at Ball | 0 | 1797 | Ball | 78% |
| 19- Fault Diameter 0.021", at Ball | 1 | 1772 | Ball | 89% |
| 20- Fault Diameter 0.021", at Ball | 2 | 1750 | Ball | 67% |
| 21- Fault Diameter 0.021", at Ball | 3 | 1730 | Ball | 100% |
| 22- Fault Diameter 0.021", at Ball | 0 | 1797 | Outer Race | 78% |
| 23- Fault Diameter 0.007", at Outer Race | 1 | 1772 | Outer Race | 100% |
| 24- Fault Diameter 0.007", at Outer Race | 2 | 1750 | Outer Race | 100% |
| 25- Fault Diameter 0.007", at Outer Race | 3 | 1730 | Outer Race | 100% |
| 26- Fault Diameter 0.007", at Outer Race | 0 | 1797 | Outer Race | 100% |
| 27- Fault Diameter 0.014", at Outer Race | 0 | 1797 | Outer Race | 100% |
| **Average Detection Efficiency** | | | | **99.6%** |

## 5.4 Conclusions

In this chapter, a discrete HMM was developed to identify the fault severity and location. The approach was successfully applied to classify tool conditions and bearing faults size. With just a sequence of four observations (approximately 1 second), this method showed a minimum fault classification and localization efficiency of 95%.

The vector extraction technique seems efficient in terms of providing the necessary information leading to accurate results. The HMM technique was used successfully to monitor the status of two different engine bearings. The accurate results obtained during the analysis demonstrate the effectiveness of this technique and reflect its robustness.

# Chapter 6

# Conclusions and Future Research

## 6.1 Conclusions

A fault detection and diagnosis approach for machining processes and rotating machinery was presented in this study. The monitoring method consists of two stages: the first one defines the fault detection techniques that can be employed to determine the condition of the monitored system (i.e. healthy or not) and the second identifies the severity and the location of the detected fault. In the light of the results obtained during this study, the following conclusions can be drawn:

(1) A new generic monitoring technique and four condition indicators were developed to detect transient and continuous anomalies. This approach was applied successfully to tool and bearing monitoring. The power and standard deviation condition indicators have shown a high degree of effectiveness in detecting transient anomalies, whereas the power and standard deviation correlation factors have demonstrated a high sensitivity to continuous type of faults.

(2) To enhance the detection efficiency, the four condition indicators were successfully fused using two different methods: fuzzy-set based fusion and fuzzy-rule based fusion. A condition index (*CI*) was proposed in the first method. The results obtained through experimental work suggest that the fuzzy-set based fusion provides a quick and acceptable solution when the thresholds for monitoring indices are well defined and condition indicators have equal weights. The method shows a lack of robustness and flexibility in the case where some of the monitoring indices are not sensitive enough to faults. Within the first approach an index of deterioration (*IoD*) was also proposed to monitor the performance deterioration of the system in question. The effectiveness of this index was

evaluated using two engine bearings. The *IoD* showed in general a good sensitivity to fault size. On the other hand, the second fusion technique provides a weighted and more robust solution as the expert domain knowledge can be emulated through the use of a set of rules. A fuzzy fused index (*FFI*) was developed based on Sugeno-style inference engine. This fusion index was tested similarly in tool and bearing condition monitoring. The *FFI* has demonstrated the following characteristics:

- robustness with respect to load and speed
- robustness with respect to fault location
- sensitivity to fault size

(3) As part of the second monitoring stage, a discrete hidden Markov model was proposed to determine the size and location of the detected fault. The approach was applied successfully to tool and bearing monitoring. In the first case the model detected correctly the state of the tool (i.e. sharp, worn, or broken) whereas in the second application, the model classified the severity of the fault seeded in two different engine bearings. The success rate obtained during testing for fault severity classification was minimum 95%. In addition to the fault severity, a location index was developed to determine the fault location. Three different locations (inner race, ball and outer race) were identified in the bearing monitoring cases with a success rate of 96%.

## 6.2 Contributions

The contributions made through this study can be summarized as follows:

(1) The monitoring technique and the four condition indicators developed throughout this work are original. This approach is flexible, robust, and versatile so it can be utilized generically in various applications such as monitoring of machining processes and rotating machinery.

(2) The use of the Sugeno-style inference engine in the fuzzy-rule based fusion method leaded to the development of an attractive fused index (i.e. *FFI*). This index was found sensitive to fault severity and not susceptible to load and speed changes, giving as such some originality to this technique.

(3) The application of the discrete hidden Markov model to fault diagnosis (fault classification and isolation) is original.

## 6.3 Future Research

One of the most important tasks in the condition-based maintenance is the development and use of effective signal processing techniques. The main objective of this task is to improve the signal to noise ratio. This is required to enhance the fault detection and facilitate the development of condition indicators. Certainly, the problem can be partially alleviated through the use of a high quality equipment/sensor that provide a clean and rich signal. However, decoupling the data of interest from the noise remains one of the most difficult and crucial parts of this phase. Despite the availability of advanced filtering techniques, in the case of weak signals the useful data is embedded in the noise. Additionally, filtering may cause the loss of some useful information when the signal frequency content matches in some ranges the ones for the noise. Therefore, it will be interesting to develop techniques that allow the extraction of the features from heavily contaminated signals without significant loss or distortion of information

On the front of the fault diagnosis, the selection of the number of observations in the hidden Markov model approach affects the solution considerably. Increasing number of observations often leads to better diagnosis. However, this requires more computation time and causes a longer decision-making process. In this study the selection of the number of observations was based on some experimental investigations. It will be very useful to develop, as part of the model, an adaptive and intelligent module that can determine the length of the observations sequence based on the actual observations and a set of rules prepared through training, to improve the performance of the model. This module boosts the efficiency of the approach and compensates for lack of robustness that could be experienced in some dynamic and complex systems where observations vary at a rate higher than the one observed in this study.

# References


Baillie, D.C., and Mathew, J., 1996, A Comparison of Autoregressive Modeling Techniques for Fault Diagnosis of Rolling Element Bearings, *Mechanical Systems and Signal Processing*, **10**(1), 1-17.

Baydar, N., and Ball, A., 2001, A Comparative Study of Acoustic and Vibration Signals in Detection of Gear Failures Using Wigner-Ville Distribution, *Mechanical Systems and Signal Processing*, **15**(6), 1091-1107.

Baydar, N., and Ball, A., 2003, Detection of Gear Failure Via Vibration and Acoustic Signal Using Wavelet Transform, *Mechanical Systems and Signal Processing*, **17**(4), 787-804.

Beggan, C., Woulfe, M., Young, P., and Byrne, G., 1999, Using Acoustic Emission to Predict Surface Quality, *International Journal of Advanced Manufacturing and Technology*, **15**(10), 737-742.

Beranek, L. L., 1971, *Noise and Vibration Control*, 1st edition, (New York: McGraw-Hill).

Bunks, C., McCathy, D., and Al-Ani, T., 2000, Condition based maintenance of machines using hidden Markov models, *Mechanical Systems and signal Processing*, **14**(4), 597-612.

Chakrabarti, B., Gupta, K.N., and Yadva, G.S., Diagnosing turbo-generator faults with a rule based expert system, *Maintenance Journal Conference Communication*, **10**(5), 12-18.

Cho, W., Lee, S.W., and Kim, J.H., 1995, Modeling and Recognition of Cursive Words with Hidden Markov Models, *Pattern Recognition*, **28**(12), 1941-1953.

Choi, D., Kwon, W.T., and Chu, C. N., 1999, Real Time Monitoring of Tool Fracture in Turning Using Sensor Fusion, *International Journal of Advanced Manufacturing and Technology*, **15**(5), 305-310.

Choi, M., Mangum, P.M., and Yee, S.O., 1991, A neural Network Approach to Real-Time Condition Monitoring of Induction Motors, *IEEE Transactions on Industrial Electronics*, **38**(6), 448-453.

Choudhury, S.K., and Rath, S., 2000, In process tool wear estimation in milling using cutting force model, *Journal of Materials Processing Technology*, **99**(1-3), 113-119.

Choudhury, S.K., Jain, V.K., and Rama Rao, C.V.V., 1999, On-line monitoring of tool wear in turning using neural network, *International Journal of Machine Tools and Manufacture*, **39**(3), 489-504

Chow, M., Sharpe, R.N., and Hung, J.C., 1993, On the Application and Design of Artificial Neural Networks for Motor Fault Detection – Part I, *IEEE Transactions on Industrial Electronics*, **40**(2), 181-188.

Chow, M., Sharpe, R.N., and Hung, J.C., 1993, On the Application and Design of Artificial Neural Networks for Motor Fault Detection – Part II, *IEEE Transactions on Industrial Electronics*, **40**(2), 189-196.

Collacott, R.A., 1977, *Mechanical fault diagnosis*, 1st edition, (London: Chapman and Hall).

Couvreur, C., Fontaine, V., Gaunard, P., and Mubikangiey, C.G., 1998, Automatic Classification of Environmental Noise Events by Hidden Markov Modes, *Applied Acoustics*, **54**(3), 187-206.

Crupi, V., Guglielmino, E., and Milazzo, G., 2004, Neural-Network-Based system for Novel Fault Detection in Rotating Machinery, *Journal of Vibration and Control*, **10**(8), 1137-1150.

Dimla, D.E., 2002, The Correlation of Vibration Signal Features to Cutting Tool Wear in a Metal Turning Operation, *International Journal of Advanced Manufacturing and Technology*, **19**(10), 705-713.

Dornfeld, D., and Rangwala, S., 1990, Sensor Integration Using Neural Networks for Intelligent Tool Condition Monitoring, *ASME Transactions Journal of Engineering for Industry*, **112**, August, 219-228.

Du, R.X., Elbestawi, M.A., and Li, S., 1992, Tool Condition Monitoring in Turning Using Fuzzy Set Theory, *International Journal of Machine Tools and Manufacture*, **32**(6), 781-796

Dyer, D., Stewart, R.M., 1978, Detection of rolling element bearing damage by statistical vibration analysis, *Journal of Mechanical Design*, **100**(2), 229-235.

Elbestawi, M.A., Papazafiriou, T.A., and Du, R.X., 1991, In-Process Monitoring of Tool Wear in Milling Using Cutting Force Signature, *International Journal of Machine Tools and Manufacture*, **31**(1), 55-73.

Ertunc, H.M., and Loparo, K.A., 2001, Tool wear condition monitoring in drilling operations using hidden Markov models, *International Journal of Machine Tools and Manufacture*, **41**(9), 1347-1363.

Ertunc, H.M., Loparo, K.A., and Ocak, H., 2001, Tool wear condition monitoring in drilling operations using hidden Markov models (HMMs), *International Journal of Machine Tools and Manufacture*, **41**(9), 1363-1384

Ertunc, H.M., and Oysu, C., 2004, Drill wear monitoring using cutting force signals, *Mechatronics*, **14**(5), 533-548.

Everson, C.E., and Cheraghi, S.H., 1999, The application of acoustic emission for precision drilling process monitoring, *International Journal of Machine Tools and Manufacture*, **39**(3), 371-387.

Filippetti, F., Franceschini, G., Tassoni, C., and Vas, P., 2000, Recent Developments of Induction Motor Drives Fault Diagnosis Using AI Techniques, *IEEE Transactions on Industrial Electronics*, **47**(5), 994- 1004.

Forester, B.D., 1989, Use of Wigner-Ville distribution in helicopter transmission fault detection, *Proceedings of the Australian Symposium on Signal Processing and Applications*, Adelaide, Australia.

Frank, P.M., 1990, Fault Diagnosis in Dynamic Systems Using Analytical and Knowledge-Based Redundancy – A Survey and Some New Results, *Automatica*, **26**(3), 459-474.

Gertler, J.J., 1988, Survey of Model-Based Failure Detection and Isolation in complex Plants, *IEE Control Systems Magazine*, **8**(6), 3-11.

Ghasempoor, A., Jeswiet, J., and Moore, T.N., 1999, Real time implementation of on-line tool condition monitoring in turning, *International Journal of Machine Tools and Manufacture*, **39**(12), 1883-1902.

Goode, P.V., and Chow, M., 1995, Using a Neural/Fuzzy System to Extract Heuristic Knowledge of Incipient Faults in Induction Motors: Part I – Methodology, *IEEE Transactions on Industrial Electronics*, **42**(2), 131-138.

Goode, P.V., and Chow, M., 1995, Using a Neural/Fuzzy System to Extract Heuristic Knowledge of Incipient Faults in Induction Motors: Part II – Methodology, *IEEE Transactions on Industrial Electronics*, **42**(2), 139-146.

Grimmelius, H.T., 1995, Modeling of a compression refrigeration plant for fault simulation, *in Proceedings 19th International Congress on Refrigeration*, **IIIa**, 299-306.

Grimmelius, H.T., Meiler, P.P., Maas, H.L., Bonnier, B., Grevink, J.S., and Kuilenburg, R.F., 1999, Three State-of the-Art Methods for Condition Monitoring, *IEEE Transactions on Industrial Electronics*, **46**(2), 407-416.

Han, Y., and Song, Y.H., 2003, Condition Monitoring Techniques for Electrical Equipment- A Literature Review, *IEEE Transactions on Power Delivery*, **18**(1), 4-13.

Harris, T.A., 1991, *Rolling Bearing Analysis*, 3rd edition, (New York: John Wiley and Sons).

Heng, R.B.W., and Nor, M.J.M., 1998, Statistical Analysis of Sound and Vibration Signals for Monitoring Rolling Element Bearing Condition, *Applied Acoustics*, **53**(1-3), 211-226.

Himmelblau, D.M., 1978, *Fualt Detection and Diagnosis in Chemical and Petrochemical Processes*, 1st edition, (Amesterdam: Elsevier).

Hoskins, J.C., and Himmelblau, D.M., 1988, Artificial Neural Network Model of Knowledge Presentation in Chemical Engineering, *Computers and Chemical Engineering*, **12**(9-10), 881-890.

Hou, R., Hunt, A., and Williams, R.A., 1999, Acoustic monitoring of pipeline flows: particulate slurries, *Powder Technology*, **106**(1-2), 30-36.

Houshmand, A.A., 1995, A Dynamic Model for Tool Wear Detection Using Acoustic Emission, *Mechanical Systems and Signal Processing*, **9**(4), 415-428.

Inasaki, I., 1999, Sensor Fusion for Monitoring and Controlling Grinding Processes, *International Advanced Manufacturing and Technology*, **15**(10), 730-736.

Jack, L.B., and Nandi, A.K., 2000, Genetic algorithms for feature selection in machine condition monitoring with vibration signals, *IEE Proceedings. Vision, Image and Signal Processing*, **147**(3), 205-212.

Jelinek, F., 1976, Continuous speech recognition by statistical method, *Proceedings of the IEEE*, **64**(4), 532-536.

Khajavi, A.N., and Komanduri, R., 1995, Frequency and time Domain Analyses of Sensor Signals in Drilling – II. Investigation on some Problems Associated With Sensor Integration, *International Journal of Machine Tools and Manufacture*, **35**(6), 795-815.

Klir, G.J., and Folger, T.A., 1988, *Fuzzy Sets, Uncertainty and Information*, 1st edition, (New Jersey: Prentice Hall).

Kwon, W.T., and Ehmann, K.F., 1994, Tool Wear Monitoring by Using the Imaginary Part of the Transfer Function of the Cutting Dynamics, *International Journal of Machine Tools and Manufacture*, **34**(3), 393-406.

Kwon, K.C., and Kim, J.H., 1999, Accident identification in nuclear power plants using hidden Markov models, *Engineering Applications of Artificial Intelligence*, **12**(4), 491-501.

Lee, J.M., Kim, S.J., Hwang, Y., and Song, C.S, 2004, Diagnosis of mechanical fault signals using continuous hidden Markov model, *Journal of Sound and Vibration*, **276**(3-5), 1065-1080.

Lee, J.M., Kim, S.J., and Hwang, Y., 2002, Mechanical signal analysis using hidden Markov model, *Ninth International Congress on Sound and Vibration*, Orlando, Florida.

Lee, C.H., Soong, F.K., and Paliwai, K.K., 1996, *Automatic speech and speaker recognition: advanced topics*, 1st edition, (Boston: Kluwer Academic).

Levinson, S.E., Rabiner, L.R., and Sondhi, M.M., 1983, An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition, *Bell System Technology Journal*, **62** (4), 1035-1074.

Levi, R., and Rossetto, S., 1975, Some considerations on tool-life scatter and its implications, *ASME Transactions Journal of Engineering for Industry*, **97**,August, 945-950.

Li, X., Dong, S., and Venuvinod, P.K., 2000, Hybrid Learning for Tool Wear Monitoring, *International Journal of Advanced Manufacturing and Technology*, **16**(5), 303-307.

Liang, M., Yeap, T., Hermansyah, A., and Rahmati, S., 2003, Fuzzy control of spindle torque for industrial CNC machining, *International Journal of Machine Tools and Manufacture*, **43**(14), 1497-1508.

Liang, M., Yeap, T., and Hermansyah, A., 2004, A fuzzy system for chatter suppression in end milling, *Proceedings of the Institution of Mechanical Engineers Part B Journal of Engineering Manufacture*, **218**(4), 403-417.

Liu, B., 2005, Selection of wavelet packet basis for rotating machinery fault diagnosis, *Journal of sound and Vibration*, **284**(3-5), 567-582.

Liu, M., and Liang, S.Y., 1991, Analytical Modeling of Acoustic Emission for Monitoring of Peripheral Milling Process, *International Journal of Machine Tools and Manufacture*, **31**(4), 589-606.

Liu, T.I., and Mengel, J.M., 1992, Intelligent Monitoring of Ball Bearing Conditions, *Mechanical Systems and Signal Processing*, **6**(5), 419-431.

Loparo, K.A., Adams, M.L., Lin, W., Abdel-Magied, M.F., and Afshari, N., 2000, Fault Detection and Diagnosis of Rotating Machinery, *IEEE Transactions on Industrial Electronics*, **47**(5), 1005-1014.

Martin, H.R., 1996, Comparison between Fourier and wavelet methods for bearing damage monitoring, *JASTED International Conference on Signal and Image Processing*, Orlando, USA.

McFadden, P.D., and Wang, W.J., 1993, Early detection of gear failure by vibration analysis –I. Calculation of the time-frequency distribution, *Mechanical System and Signal Processing*, **7**(3), 193-203.

Mechefske, C.K., 1998, Objective Machinery fault Diagnosis Using Fuzzy Logic, *Mechanical Systems and Signal Processing*, **12**(6), 855-862.

Mesina, O.S., and Langari, R., 2001, A Neuro-Fuzzy System for Tool Condition Monitoring in Metal Cutting, *ASME Transactions Journal of Manufacturing Science and Engineering*, **123**(2), 312-318.

Miyachi, T., and Seki, K., 1986, An investigation of the early detection of defects in ball bearings using vibration monitoring – practical limit of detectability and growth speed of defects, *Proceedings of the International Conference on Rotordynamics*, JSME-IftoMM, Tokyo, Japan, 403-408.

Ming, L., Xiahong, Y., and Shuzi, Y., 1999, Tool Wear Length Estimation with a Self-Learning Fuzzy Inference Algorithm in Finish Milling, *International Journal of Advanced Manufacturing and Technology*, **15**(8), 537-545.

Morando, L.E., 1988, Measuring shock pulses is ideal for bearing condition monitoring, *Pulp and Paper*, **62**(12), 96-98.

Owsley, L.M.D, Atlas, L.E., and Bernard, L.E., 1997, Self-Organizing Feature Maps and Hidden Markov Models for Machine Tool Monitoring, *IEEE Transactions on Signal Processing*, **45**(11), 2787-2798.

Pao, Y., 1989, *Adaptive Pattern Recognition and Neural Networks*, 1st edition, (New York: Addison Wesley).

Paya, B.A., Esat, I.I., and Badi, M.N.M., 1997, Artificial Neural Network Based Fault Diagnostics of Rotating Machinery using Wavelet Transforms as a Pre processor, *Mechanical Systems and Signal Processing*, **11**(5), 751-765.

Pham, D.T., and Pham, P.T.N., 1999, Artificial Intelligence Engineering, *International Journal of Machine Tools and Manufacture*, **39**(6), 937-949.

Peng, Z., and Goodwin, S., 2001, Wear-debris analysis in expert systems, *Tribology Letters*, **11**(3-4), 177-183.

Poulezos, A.D., and Stavrakakis, G.S., 1994, *Real Time Fault Monitoring of Industrial Processes*, 1st edition, (Dordrecht: Kluwer Academic Publishers).

Proakis, J. G., and Manolakis, D.J., 1996, *Digital Signal Processing*, 3rd edition, (New Jersey: Prentice Hall).

Rabiner, L.R., 1989, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, *Proceedings on the IEEE*, **77**(2), 257-286.

Rabiner, L.R., and Gold, B., 1975, *Theory and Application of Digital Signal Processing*, 1st edition, (New Jersey: Prentice Hall)

Rabiner, L.R., and Juang, B.H., 1993, *Fundamentals of Speech Recognition*, 1st edition, (New Jersey: Prentice Hall).

Rabiner, L.R., Schafer, R.W., 1978, *Digital Processing of Speech Signals*, 1st edition, (New Jersey: Prentice Hall).

Rao, B.K.N., 1996, *Handbook of Condition Monitoring*, 1st edition, (Oxford: Elsevier Advanced Technology).

Rich, S.H., and Venkatasubramanian, V., 1987, Model Based Reasoning in Diagnostic Expert Systems for Chemical Processes Plants, *Computers and Chemical Engineering*, **11**(2), 111-122.

Roth, J.T., and Pandit, S.M., 1999, Monitoring end-mill Wear and Predicting Tool Failure using Accelerometers, *ASME Transactions Journal of Manufacturing Science and Engineering*, **121**(4), 559-567.

Sadat, A.B., and Roman, S., 1987, Detection of Tool Flank Wear Using Acoustic Signature Analysis, *Wear*, **115**(3), 265-272.

Scheaffer, J. L., and McClave, J. T., 1990, *Probability and Statistics for Engineers*, 3rd edition, (California: Duxbury Press).

Shapiro, S.C., Srihari, S.N., Taie, M.R., and Celler, J., 1986, VMES: A network Versatile Maintenance Expert System, in Sriram, D., and Adey, R. (eds), *Application of Artificial Intelligence in Engineering Problems*, (London: Springer-Verlag), 925-936.

Shigley, J.E., 1977, *Mechanical Engineering Design*, 3rd edition, (New York: McGraw-Hill).

Siegel, D., and Roedersheimer, M., 1990, An expert system for rotating equipment vibration diagnosis, *Predictive/Preventive Maintenance Technology*, **3**(1).

Singh, G.K., and Al Kazzaz, S.A.S., 2003, Induction machine drive condition monitoring and diagnostic research – a survey, *Electric Power Systems Research*, **64**(2), 145-158.

Singh, G.K., and Al Kazzaz, S.A.S., 2004, Vibration signal analysis using wavelet transform for isolation and identification of electrical faults in induction machine, *Electric Power Systems Research*, **68**(2), 119-136.

Smyth, P., 1994, Hidden Markov models for fault detection in dynamic systems, *Pattern Recognition*, **27**(1), 149-164.

Sokolowski, A., 2004, One some aspects of fuzzy logic application in machine monitoring and diagnostics, *Engineering Applications of Artificial Intelligence*, **17**(4), 429-437.

Sorensen, O., and Suraland, P.V., 1988, Using the right tools for the right job: condition based monitoring pay's off, *Profile*, **6**(1), 12-16.

Sorsa, T., Koivo, H.N., and Koivisto, H., 1991, Neural Network in Process Fault Diagnosis, *IEEE Transactions on Systems, Man, and Cybernetics*, **21**(4), 815-825.

Stander, C.J., and Heyns, P.S., 2002, Using Vibration Monitoring for Local Fault Detection on Gears Operating Under Fluctuating Load Conditions, *Mechanical Systems and Signal Processing*, **16**(6), 1005-1024.

Su, Y.T., Lin, M.H., and Lee, M.S., 1993, The Effects of surface Irregularities on roller Bearing Vibrations, *Journal of Sound and Vibration*, **165**(3), 455-466.

Susanto, V., and chen, J.C., 2003, Fuzzy Logic Based In-Process Tool Wear Monitoring System in Face Milling Operations, *International Journal of Advanced Manufacturing and Technology*, **21**(3), 186-192.

Tandon, N., 1994, A comparison of some vibration parameters for the condition monitoring of rolling element bearings, *Measurement*, **12**(3), 285-289.

Tandon, N., and Choudhury, A., 1999, A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings, *Tribology International*, **32**(8), 469-480.

Tansel, I.N., 1994, Identification of the Prefailure Phase in Microdrilling operations Using Multiple Sensors, *International Journal of Machine Tools and Manufacture*, **34**(3), 351-364.

Tansel, I.N., Wagiman, A., and Tziranis, A., 1991, Recognition of Chatter with Neural Networks, *International Journal of Machine Tools and Manufacture*, **31**(4), 539-552.

Tarng, Y.S., 1990, Study of milling Cutting Force Pulsation Applied to the Detection of Tool Breakage, *International Journal of Machine Tools and Manufacture*, **30**(4), 651-660.

Tarng, Y.S., Young, H.T., and Lee, B.Y., 1994, An Analytical Model of Chatter Vibration in Metal Cutting, *International Journal of Machine Tools and Manufacture*, **34**(2), 183-197.

Tsai, Y.H., Chen, J.C., and Lou, S.J., 1999, An in-process surface recognition system based on neural networks in end milling cutting operations, *International Journal of Machine Tools and Manufacture*, **34**(4), 583-605.

Tzafestas, S.G., 1989, System Fault Diagnosis Using the Knowledge-Based Methodology, in Patton, R., Frank, P., and Clark, R. (eds), *Fault Diagnosis in Dynamic Systems Theory and Applications*, (UK: Prentice Hall), 509-572.

Weck, M., Verhaag, E., and Gather, M., 1975, Adaptive control for face milling operations with strategies for avoiding chatter vibrations and for automatic cut distribution, *Annals of CIRP*, **24**, 405-409.

Xiaoli, L., 1999, On-line detection of the breakage of small diameter drills using current signature wavelet transform, *International Journal of Machine Tools and Manufacture*, **39**(1), 157-164.

Yadav, G.S., Chakrabarti, B., 1996, Condition monitoring of machines by vibration analysis, *US Symposium on Emerging Trends in Vibration and Noise Engineering*, Delhi, India, 333-343.

Yao, Y.L., and Fang, X.D., 1993, Assessment of Chip Forming Patterns with Tool Wear Progression in Machining Via Neural Networks, *International Journal of Machine Tools and Manufacture*, **33**(1), 89-102.

Yesilyurt, I., 1997, *Gearbox fault detection and severity assessment using vibration analysis*, Ph.D., thesis, Department of Mechanical Engineering, University of Manchester, USA.

Zhang, J., and Morris, J., 1996, Process modelling and fault diagnosis using fuzzy neural networks, *Fuzzy Sets and Systems*, **79**(1), 127-140.

Zheng, G.T., and McFadden P.D., 1993, The Detection of Failure in Gears by the Parameter Identification Technique, in Rao, R.B.K.N., and Trmal, G.J. (eds), *Proceedings of 5$^{th}$ International Congress on Condition Monitoring and Diagnostic Engineering Management* (Bristol: UWE), 85-90.

# Appendix A

## MATLAB Program *Monitoring*

```
%-----------------------------------------------------------------------
% File called Monitoring.m
% This program processes the signal and generates generic condition
% indicators as per technique defined in Chapter 3
% ----------------------------------------------------------------------
% Clear values from memory.
clear d
clear b
clear e
clear size
clear f
clear p11
clear power_CI
clear mean_CI
clear mean
clear power
clear std
clear std_CI
clear p2
clear logp11
clear length(f)
clear length(p11)
clear length(logp11)
clear time
clear mm
clear cfp
clear cfs

'run normal files first to collect average reference values for both
power and standard'
% Read File and store data into matrix d
load('a209.mat');
d=X209_DE_time;
%%fid=fopen('tool_br.dat','r');
%Read data file.
%[d,count]=fread(fid,800000);
%status=fclose(fid);
% dd represents the number of runs in the loop
dd=1;
for dd=1:15;
% fs is the sampling frequency and fss is the Nyquisit frequency
%fs=1*22050;
fs=12000;
fss=fs/2;
% Filtering the noise using a high pass Butterworth filter where w1 is
% the cut-off frequency
w1=100;
%w1=2000
[c,a]=butter(3,w1/fss,'high');
% 'e' is the filtered vector
e=filter(c,a,d);
% Filter the noise using a low-pass Butterworth filter where w2 is the
% cut-off frequency
w2= 5000;
[m,n]=butter(3,w2/fss);
% 'b' is the final filtered vector
b=filter(m,n,e);
```

```
% Specifying the number of samples in a mini group.
%space=200 % for tool monitoring
space=60; % for bearing monitoring
% Define number of mini-groups in a sub-group
nsg=10;
%nsg=25;
% Specifying the starting and the end points for each sub-group
%(composed of 10 mini-groups--- the number of mini group for tool wear
% is 25)
k1=10+nsg*dd;
k2=10+(dd-1)*nsg;
size=k1*space;
start=k2*space;
% Nbs is the number of mini groups considered in a sub-group
Nbs=k1-k2;
% Initialize the max and min values for the condition indicators
maxpower=0;
minpower=100000000;
maxmean=0;
minmean=100000000;
maxstd=0;
minstd=100000000;
for i=1:Nbs;
lim=(i-1)*space+start+1;
limit=i*space+start;
% Calculate the power using only the amplitudes
power(i)=sum(b(lim:limit).^2)/space;
p2(i)=power(i);
% Calculate the mean and standard deviation values
mean(i)=sum(b(lim:limit))/space;
std(i)=sqrt((sum((b(lim:limit)-mean(i)).^2))/(space-1));
% Define max and min values
if power(i)>maxpower
maxpower=power(i);
end;
if power(i)<minpower
minpower=power(i);
end;
if mean(i)>maxmean
maxmean=mean(i);
end;
if mean(i)<minmean
minmean=mean(i);
end;
if std(i)>maxstd;
maxstd=std(i);
end;
if std(i)<minstd;
minstd=std(i);
end;
end;
% Initialize the sum for the condition indicators
sum_power=0;
sum_std=0;
sum_std22=0;
sum_power=sum(power);
sum_std=sum(std);
```

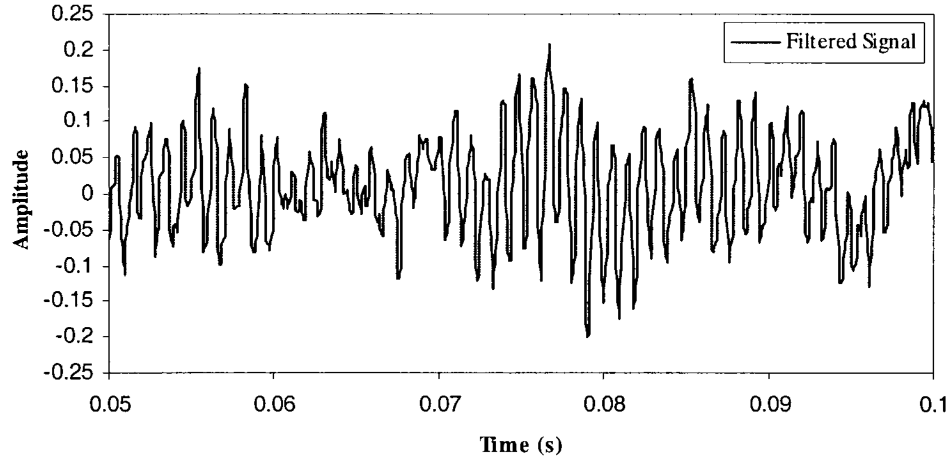
```
% To define the reference average for both power and standard
% deviation: that reference average could be obtained by running the
% healthy files and saving the average
% values (for 2-5 seconds)...when saved, make the 2 below lines as
% comments ...take samples of mini groups 100-1300
%refp=sum_power/Nbs
%refstd=sum_std/Nbs
% Compute average values
average_power=(sum_power)/Nbs;
average_std=(sum_std)/Nbs;
% SSEP is the sum square error for power and SSES for standard
deviation
SSEP=0;
SSES=0;
SS2P=0;
SS2S=0;
sum_p22=0;
sum_std22=0;
for i=1:Nbs;
sum_p22=sum_p22+p2(i)^2;
sum_std22=sum_std22+std(i)^2;
SSEP=SSEP+(power(i)-refp)^2;
SSES=SSES+(std(i)-refstd)^2;
end;
% sump2 is the sum of all power values
sump2=(sum_power)^2;
SS2P=(sum_p22-(sump2)/Nbs);
% Repeat for standard deviation
sum_std2=(sum_std)^2;
SS2S=(sum_std22-(sum_std2)/Nbs);
% Calculate the power correlation factor
coefp_det=SS2P/SSEP;
corfp=sqrt(coefp_det);
% Calculate the standard deviation correlation factor
coefs_det=SS2S/SSES;
corfs=sqrt(coefs_det);
sum_power_CI=0;
sum_std_CI=0;
% Calculate the power an standard deviation condition indicators
for i=1:Nbs
power_CI(i)=(abs(power(i)-average_power))/(maxpower-minpower);
sum_power_CI=sum_power_CI+power_CI(i);
std_CI(i)=(abs(std(i)-average_std))/(maxstd-minstd);
sum_std_CI=sum_std_CI+std_CI(i);
end;
%Initialize counts
count1=0;
count2=0;
count3=0;
count4=0;
% Count the number of anomalies in both power and standard
for i=1:Nbs;
if power_CI(i)>0.70; % to be modified according to application and
thresholds described in Chap 3
count1=count1+1;
end;
end;
```

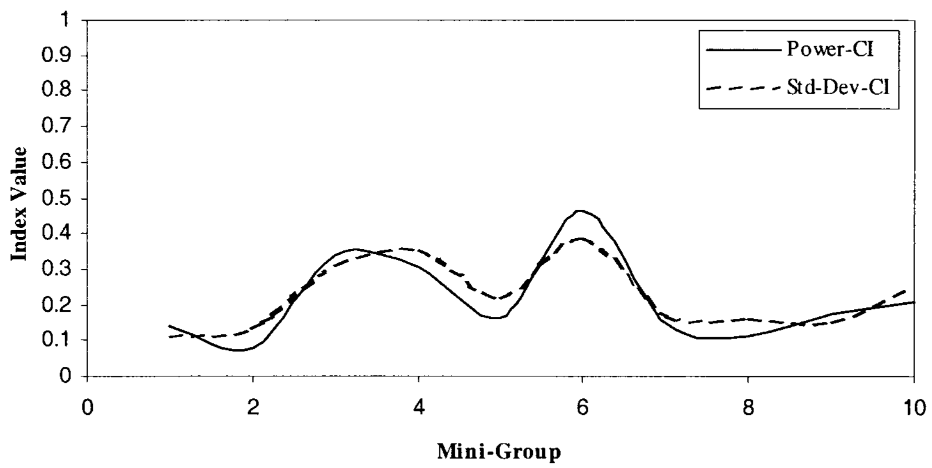
```
for i=1:Nbs;
if std_CI(i)>0.65;
count2=count2+1;
end;
end;
if corfp<0.7
    count3=count3+1;
end;
if corfs<0.65
    count4=count4+1;
end;
% Plot power Condition Indicator
subplot(3,1,1),plot(power_CI,'r-')
xlabel('Samples')
ylabel('Power-CI')
% Plot Standad Deviation Condition Indicator
subplot(3,1,2),plot(std_CI,'r-')
xlabel('Samples')
ylabel('Std-Dev-CI')
% Compute power spectrum density
[pll,f]=psd(d,256,fs);
time=(start/fs:1/fs:size/fs);
% mm used to transfer the raw data into files
for i=1:size-start+1;
mm(i)=d(start-1+i);
end;
mm(i+1)=start;
mm(i+2)=size;
length(time);
length(d(start:size))
subplot(3,1,3),plot(f,10*log10(pll),'r-')
%%xlabel('Frequency (Hz)')
%%ylabel('Power Spectrum Density (db)')
axis([0 8000 -50 15])
%%%%%axis([0 5000 -10 200])
plot (time,d(start:size))
axis([start/fs size/fs -1.5 1.5])
% Copy values of correlation factors into matrices
cfp(dd)=corfp;
cfs(dd)=corfs;
end;% to close the main loop

%-------------------------------------------------------------
```

# Appendix B

## Fault Detection Results for Drive End and Fan End Bearings



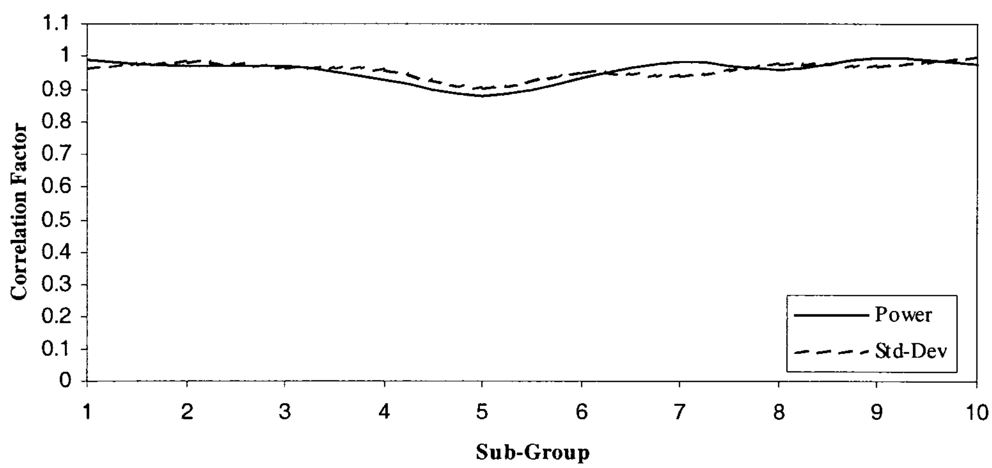




Figure B.1 Drive end bearing, normal data representation at 0 hp motor load and 1797 rpm.

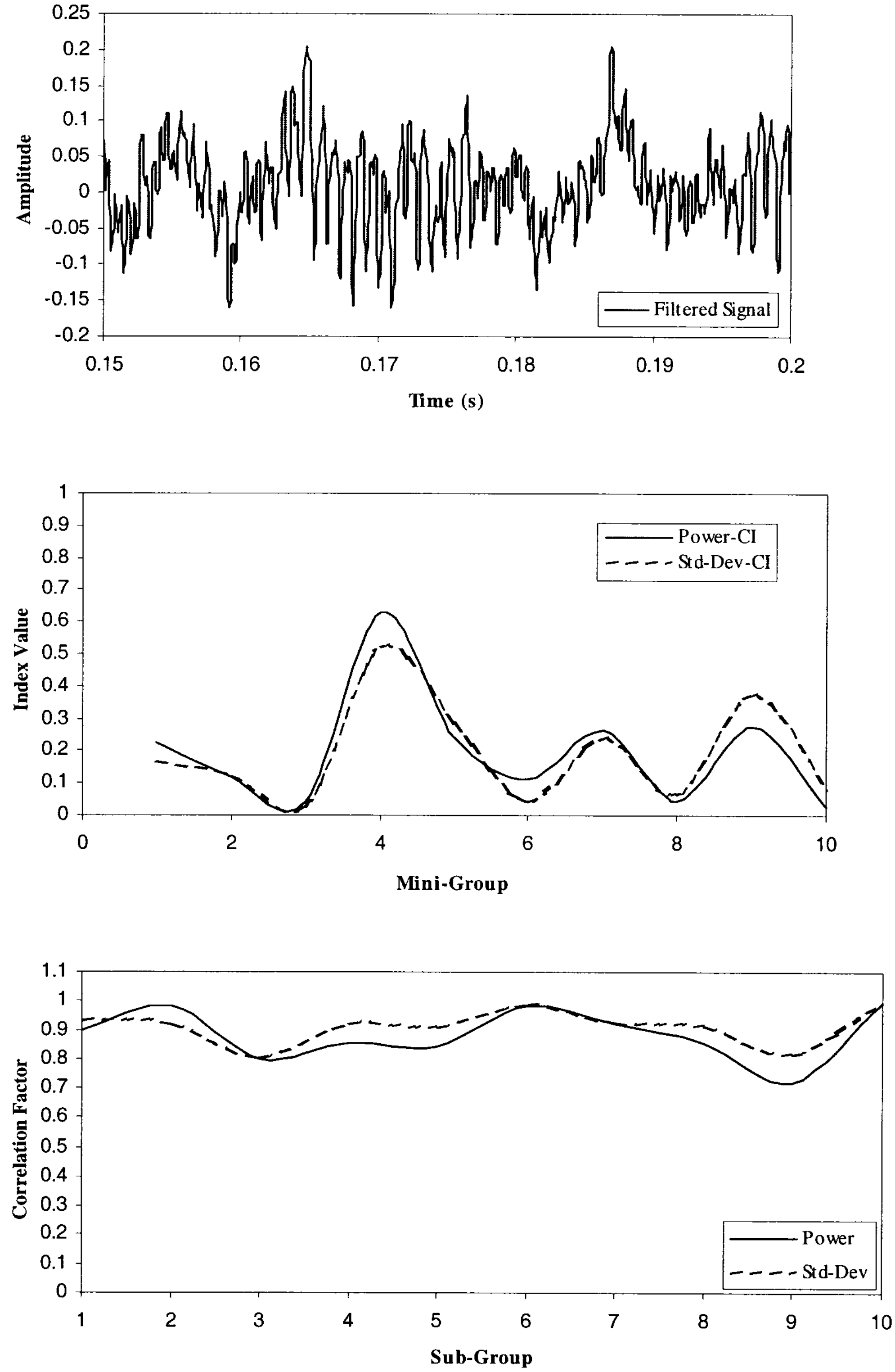



Figure B.2 Drive end bearing, normal data representation at 1 hp motor load and 1772 rpm.

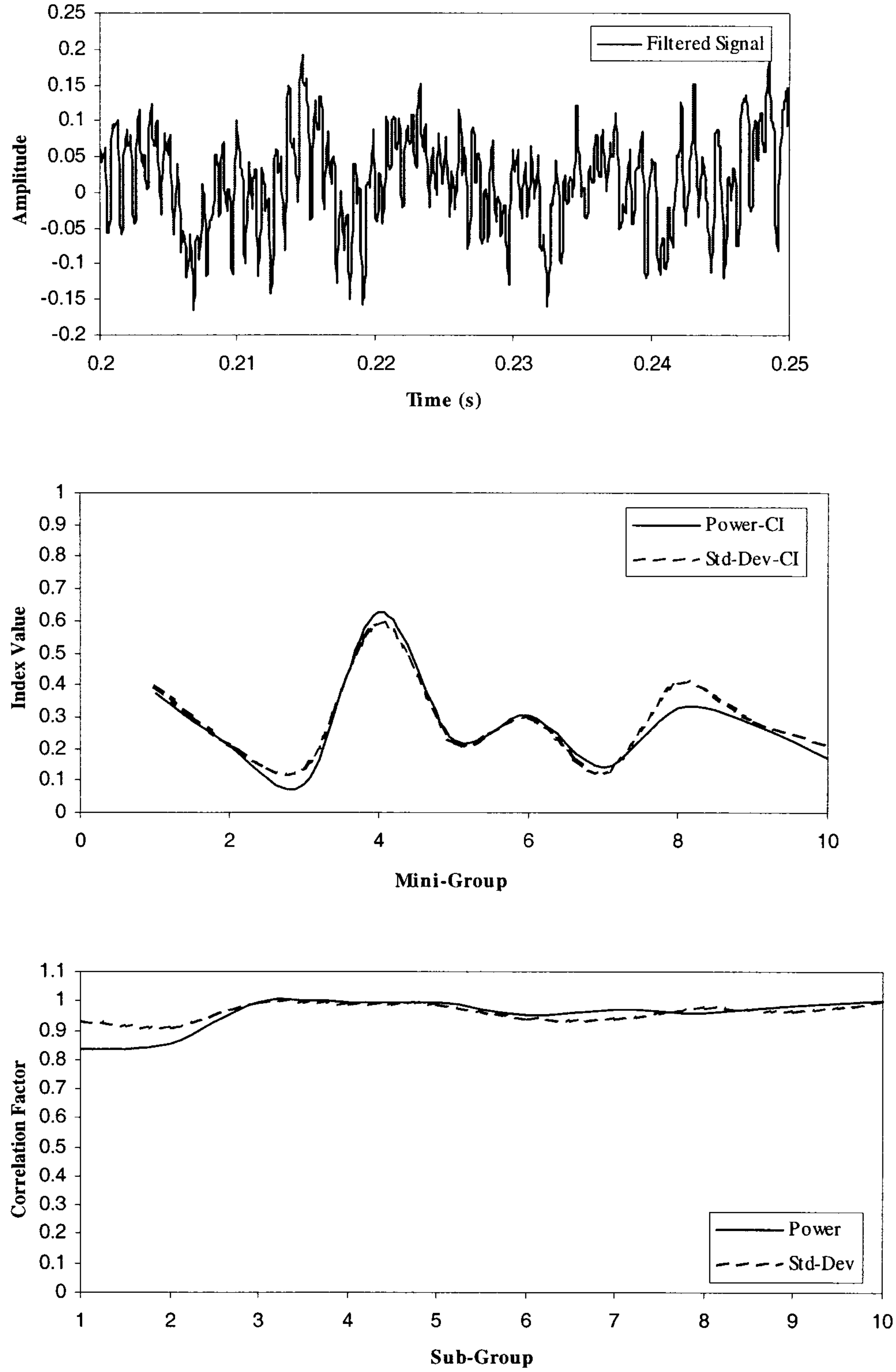



Figure B.3 Drive end bearing, normal data representation at 2 hp motor load and 1750 rpm.

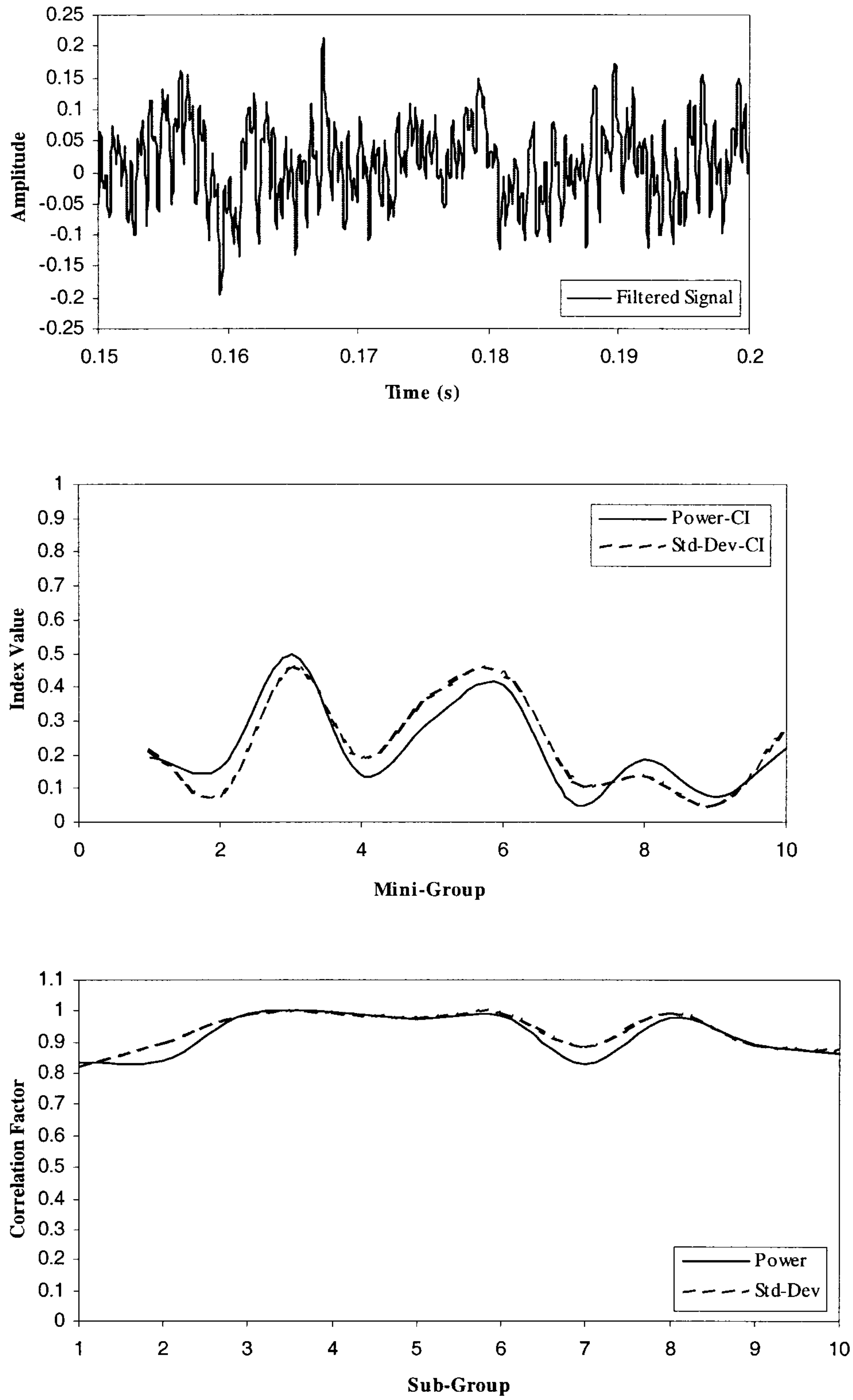



Figure B.4 Drive end bearing, normal data representation at 3 hp motor load and 1730 rpm.

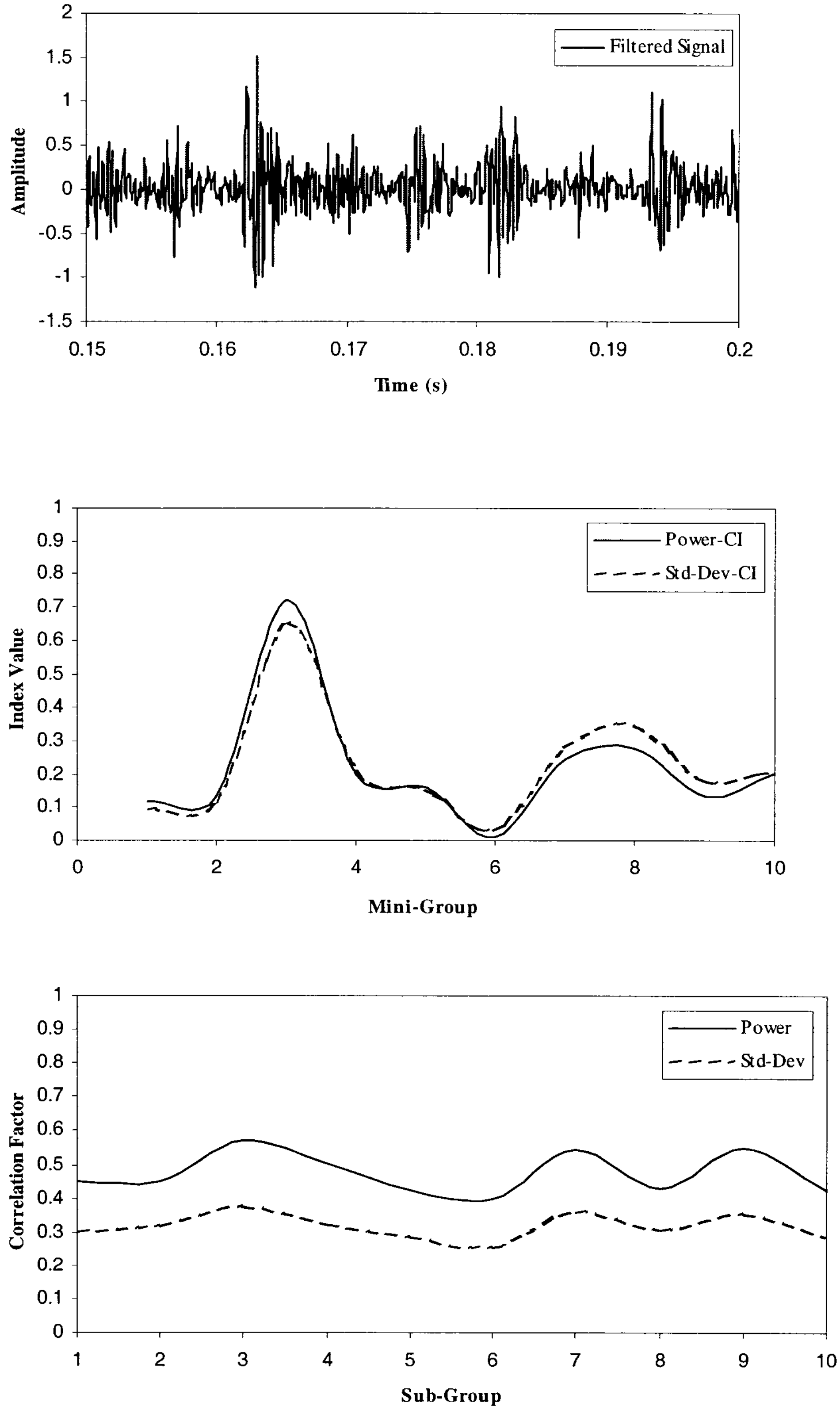



Figure B.5 Drive end bearing, fault detection – 0.007" fault diameter at inner race, 0 hp motor load and 1797 rpm.

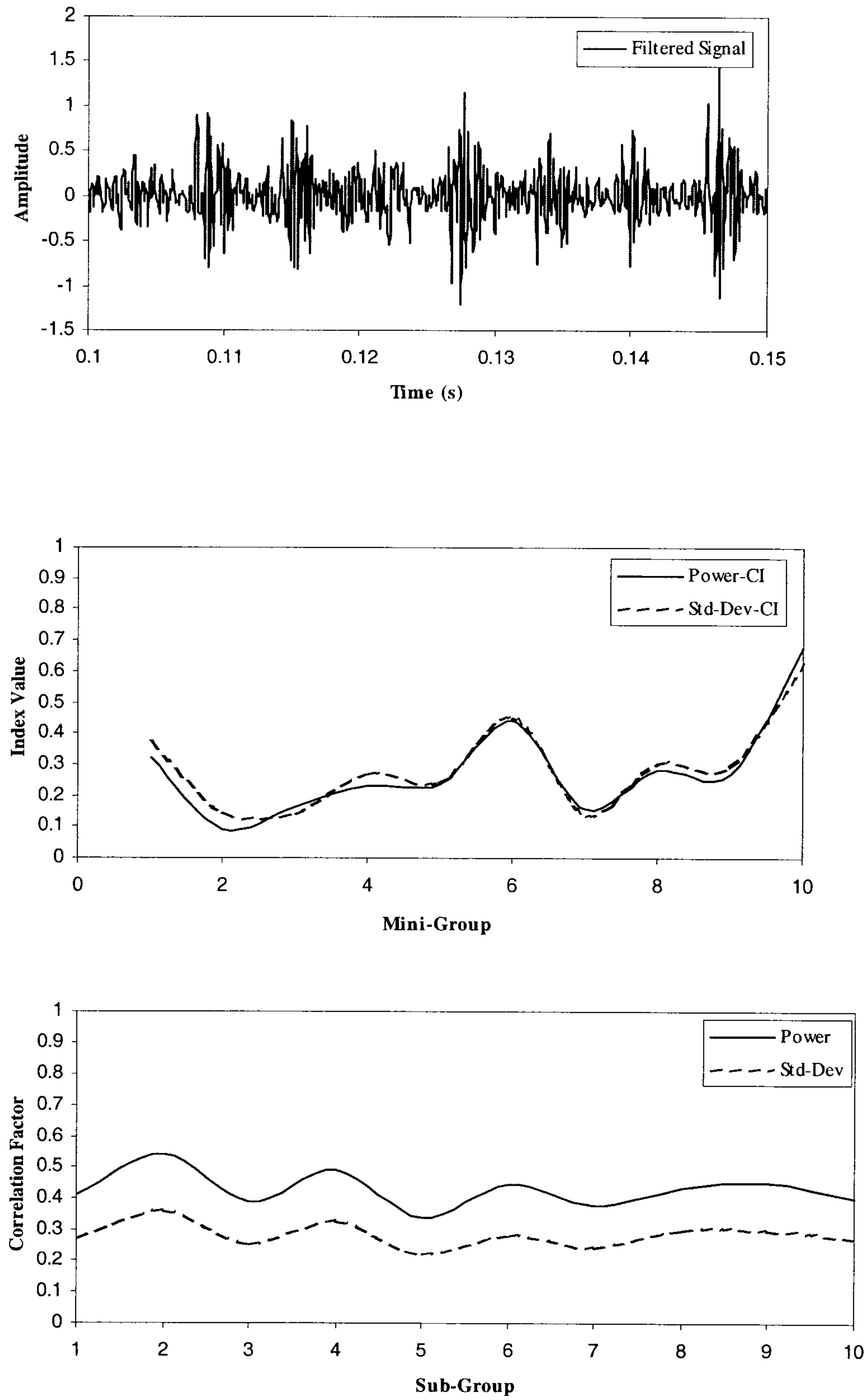



Figure B.6 Drive end bearing, fault detection – 0.007" fault diameter at inner race, 1 hp motor load and 1772 rpm.

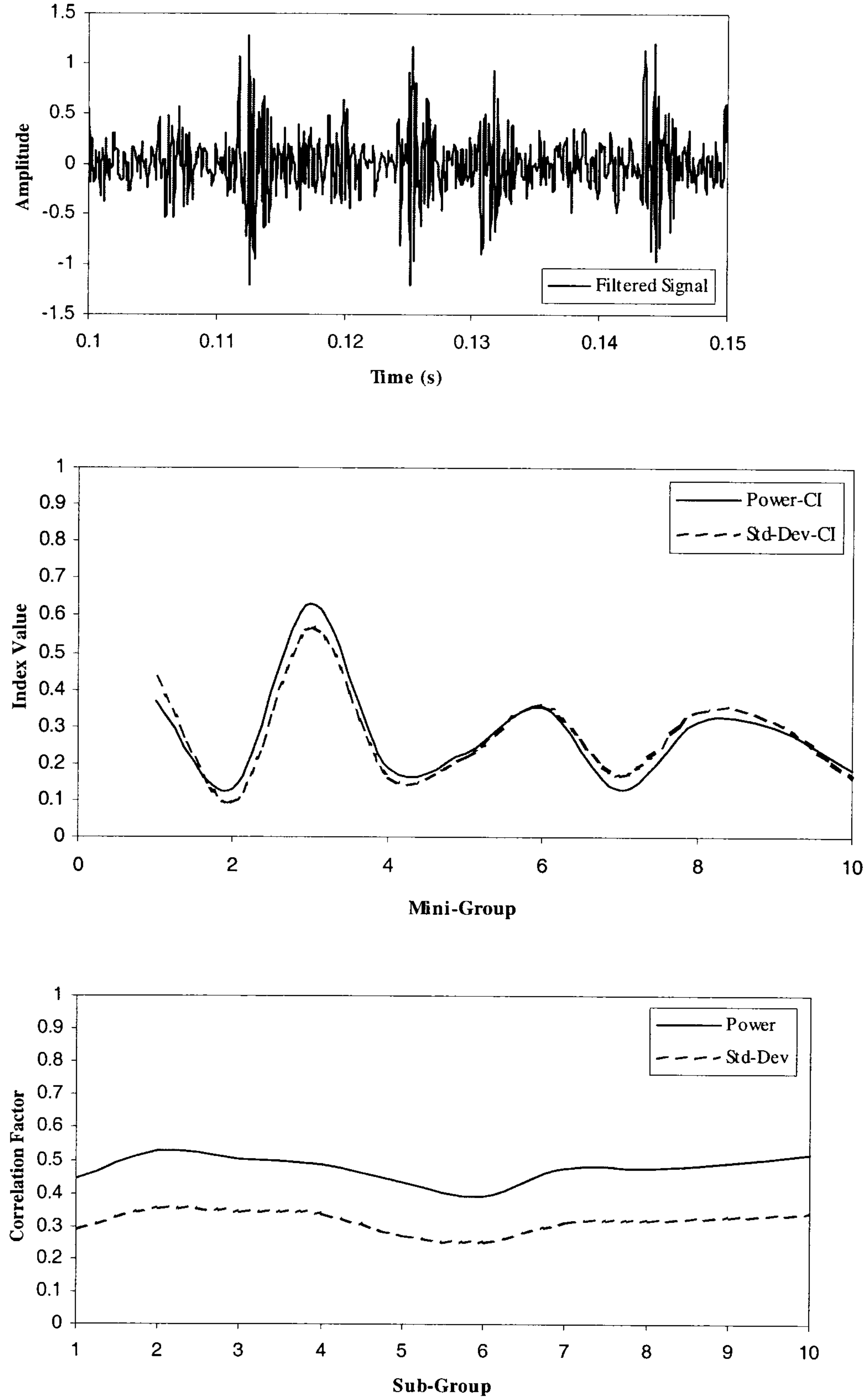



Figure B.7 Drive end bearing, fault detection – 0.007" fault diameter at inner race, 2 hp motor load and 1750 rpm.

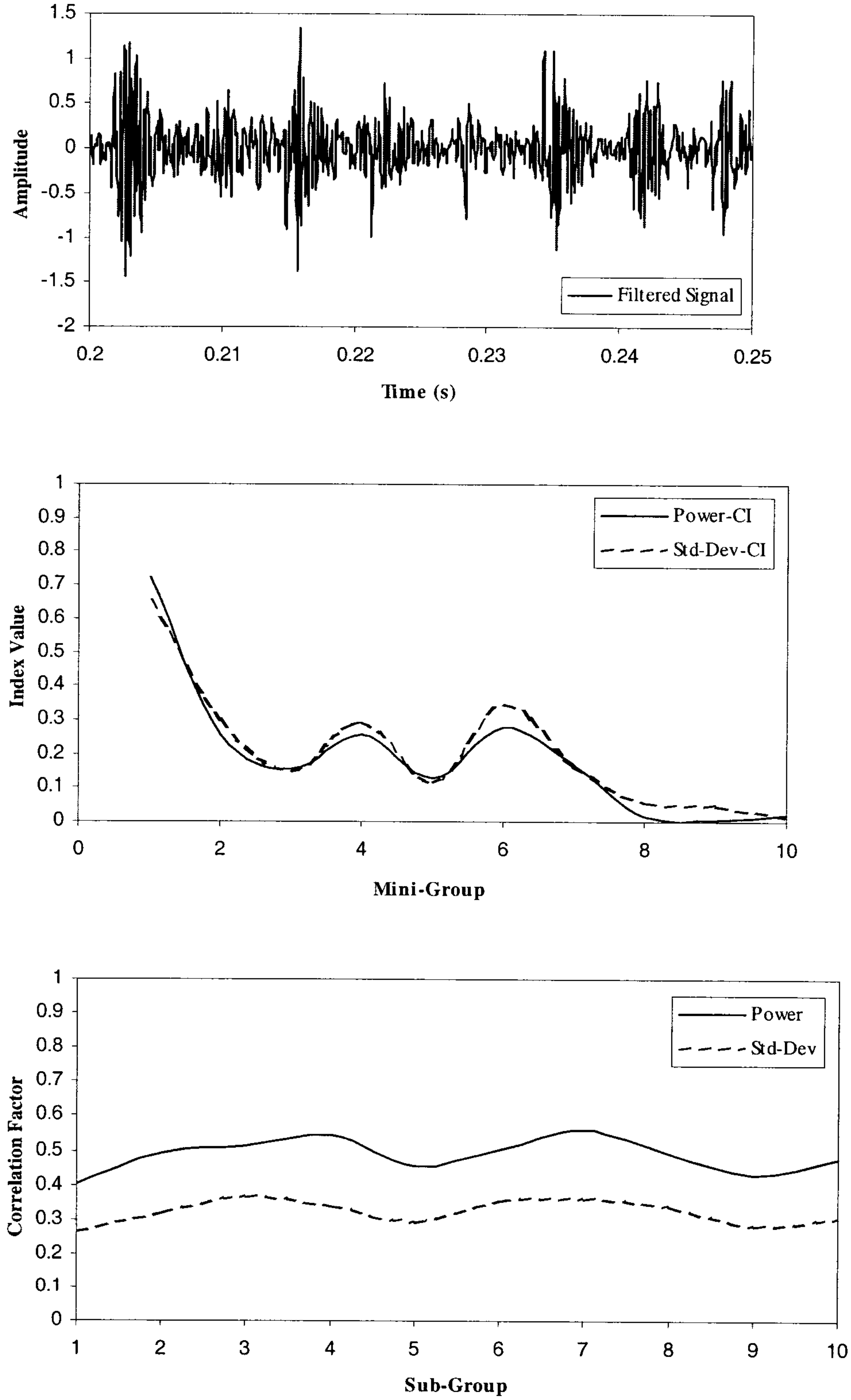
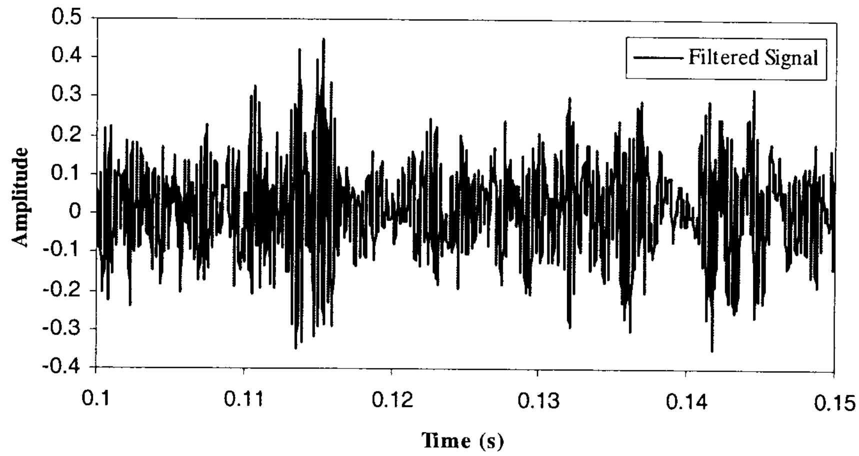
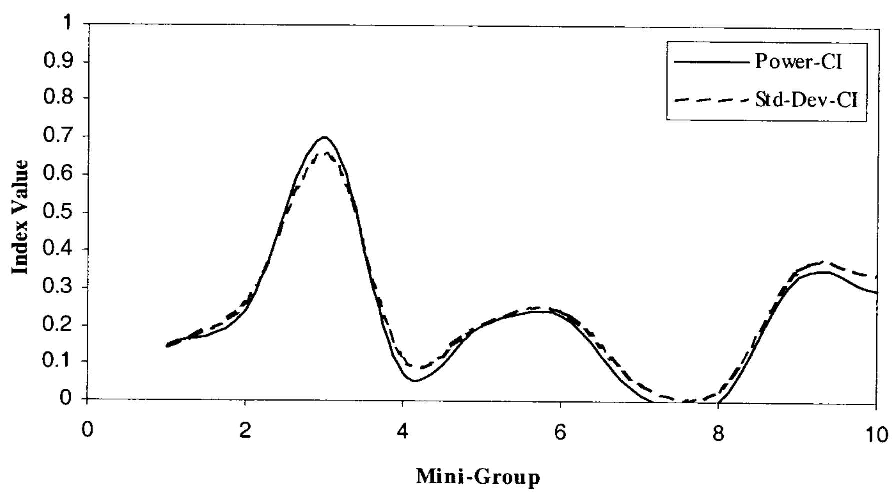



Figure B.8 Drive end bearing, fault detection – 0.007" fault diameter at inner race, 3 hp motor load and 1730 rpm.



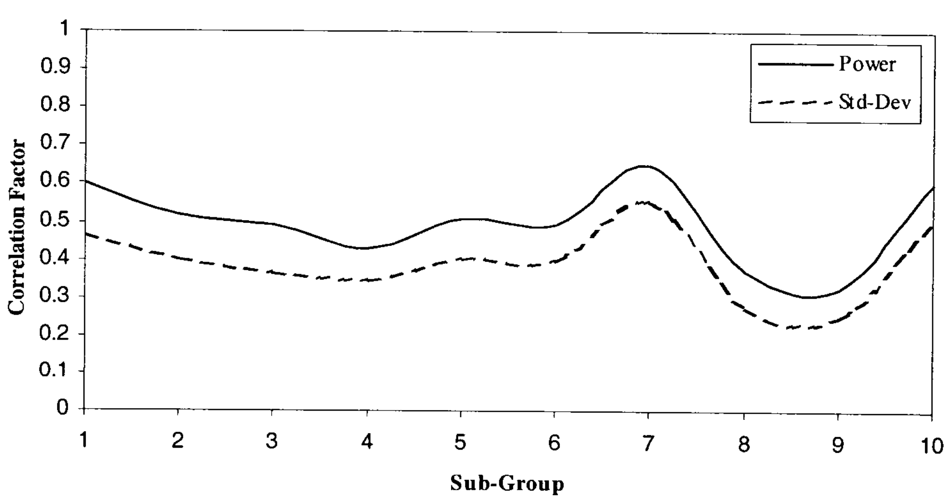
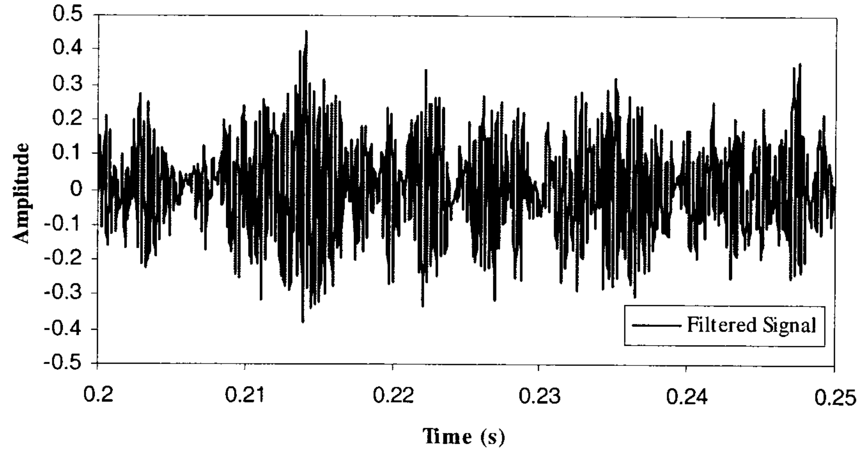
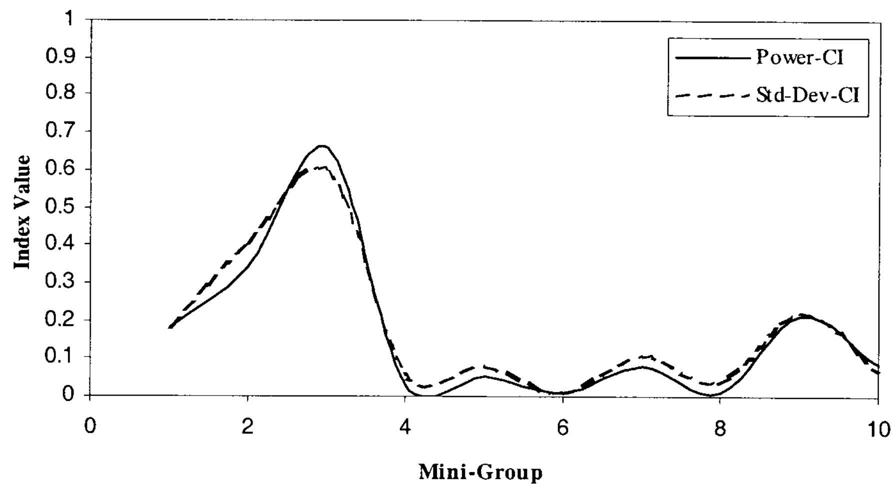




Figure B.9 Drive end bearing, fault detection – 0.007" fault diameter at ball, 0 hp motor load and 1797 rpm.



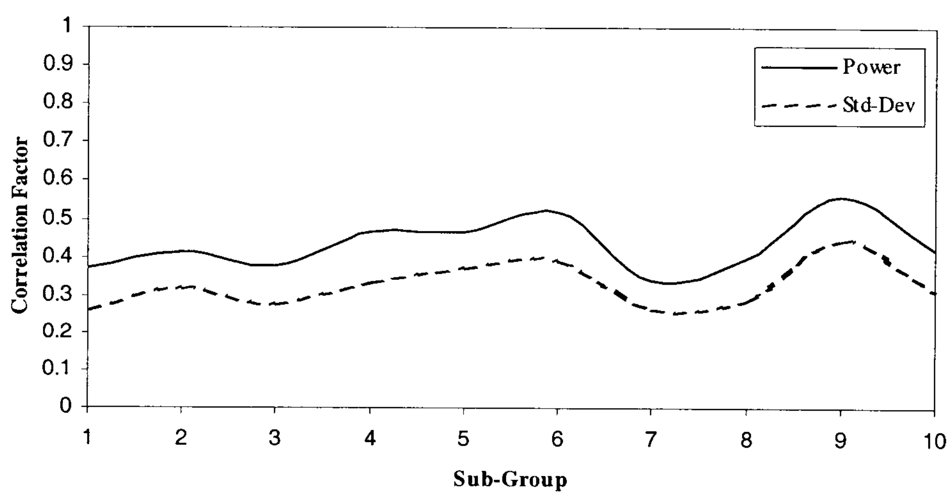




Figure B.10 Drive end bearing, fault detection – 0.007" fault diameter at ball, 1 hp motor load and 1772 rpm.

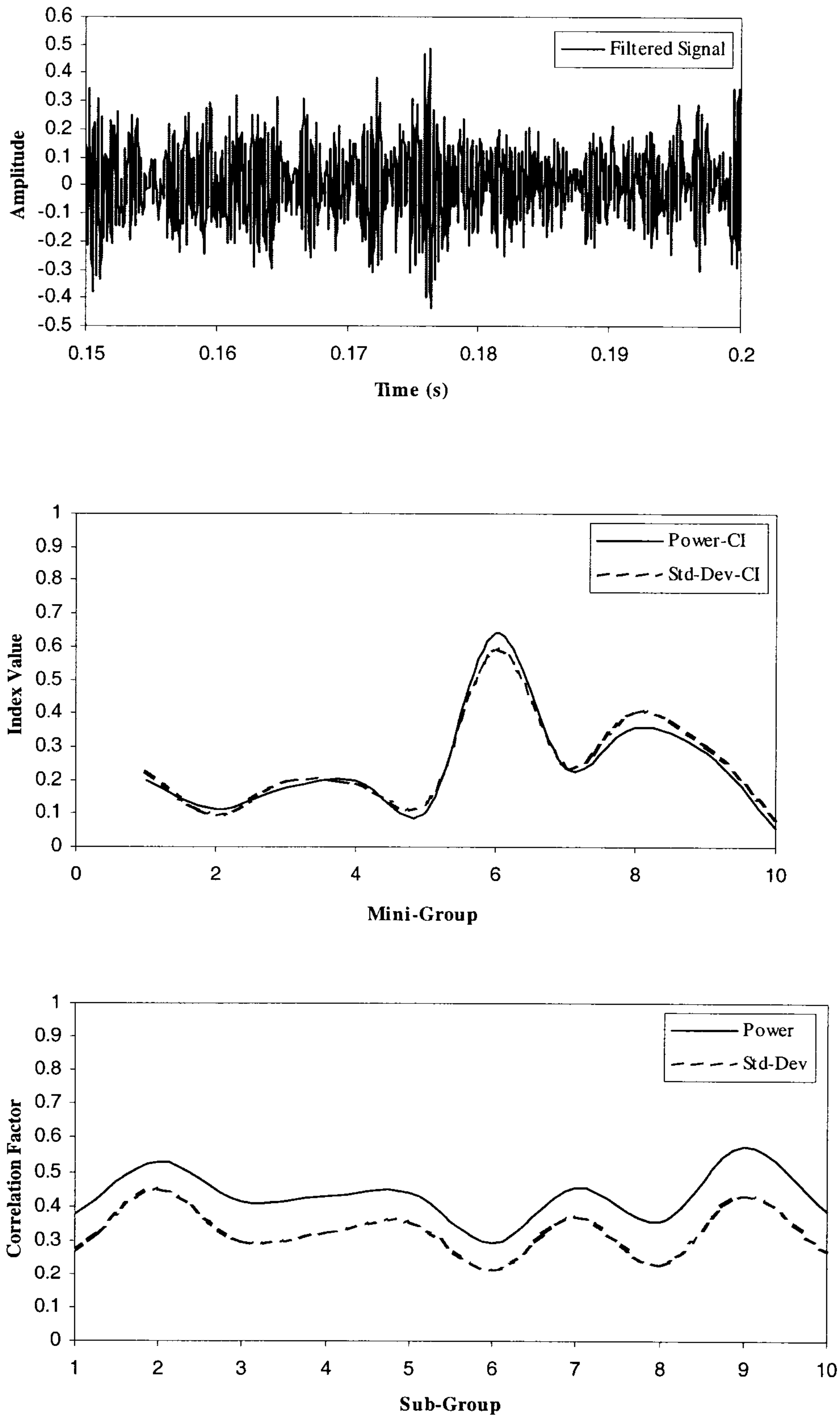



Figure B.11 Drive end bearing, fault detection – 0.007" fault diameter at ball, 2 hp motor load and 1750 rpm.

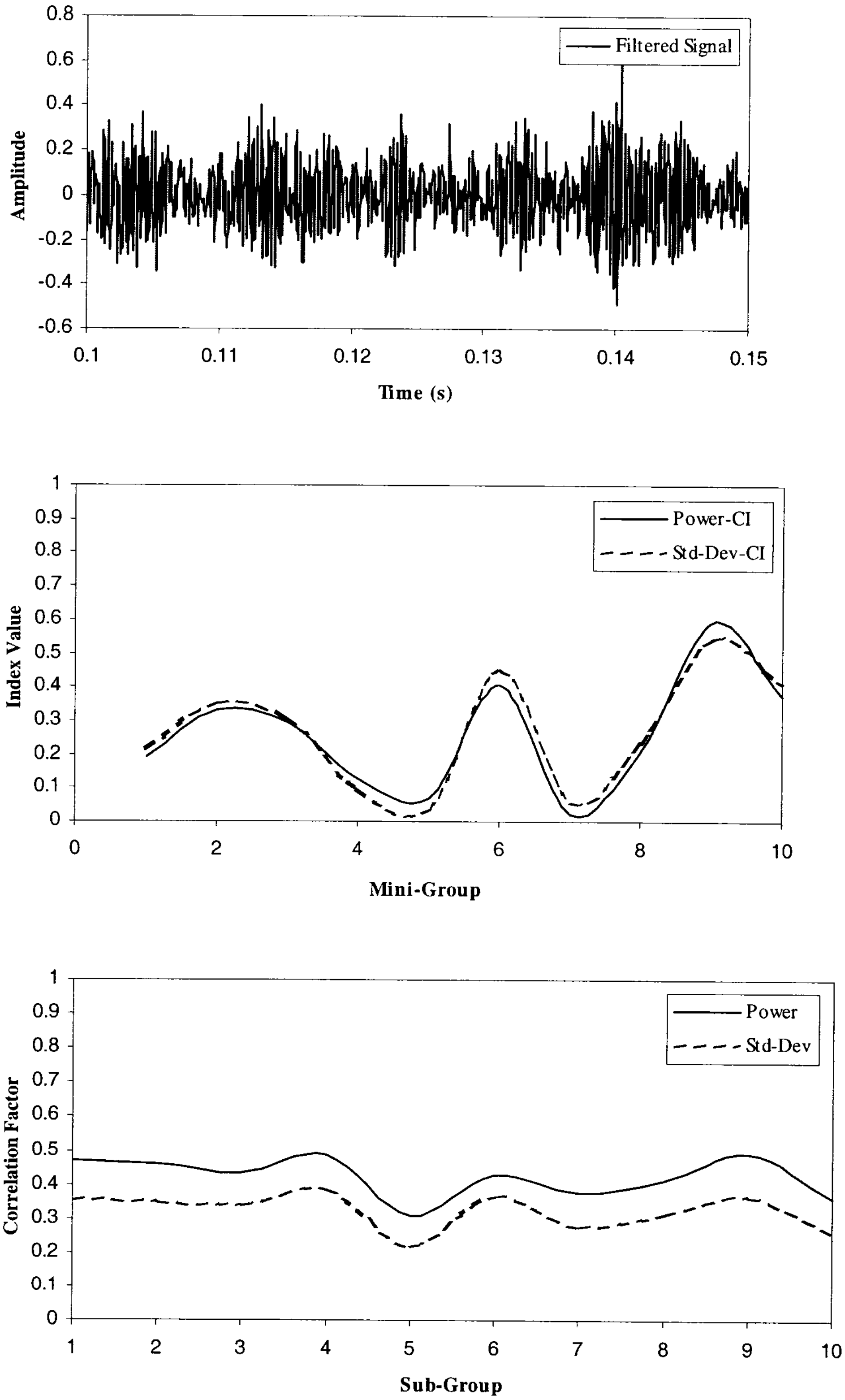
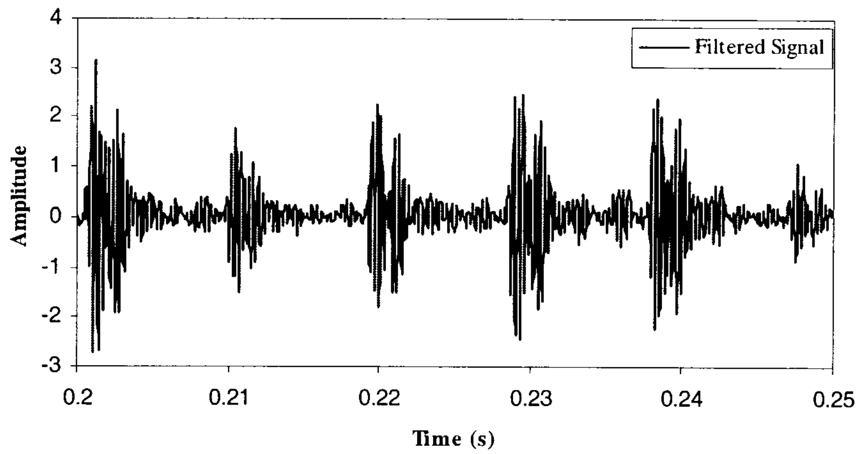
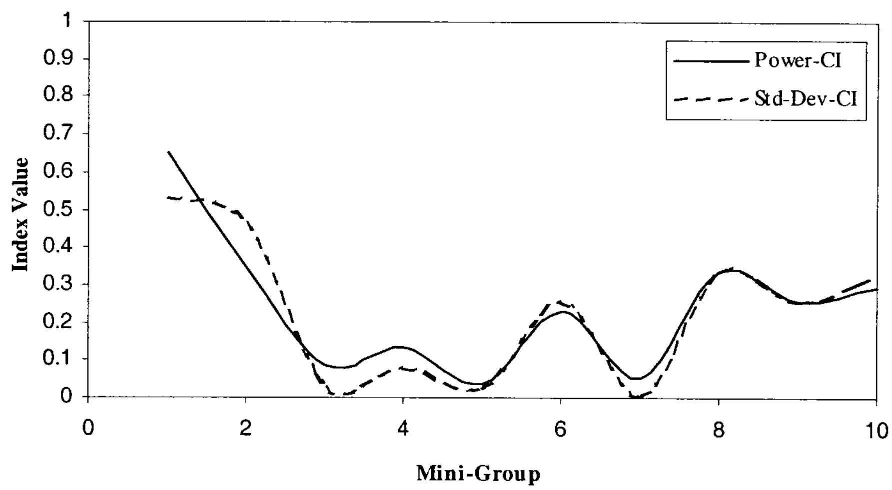



Figure B.12 Drive end bearing, fault detection – 0.007" fault diameter at ball, 3 hp motor load and 1730 rpm.



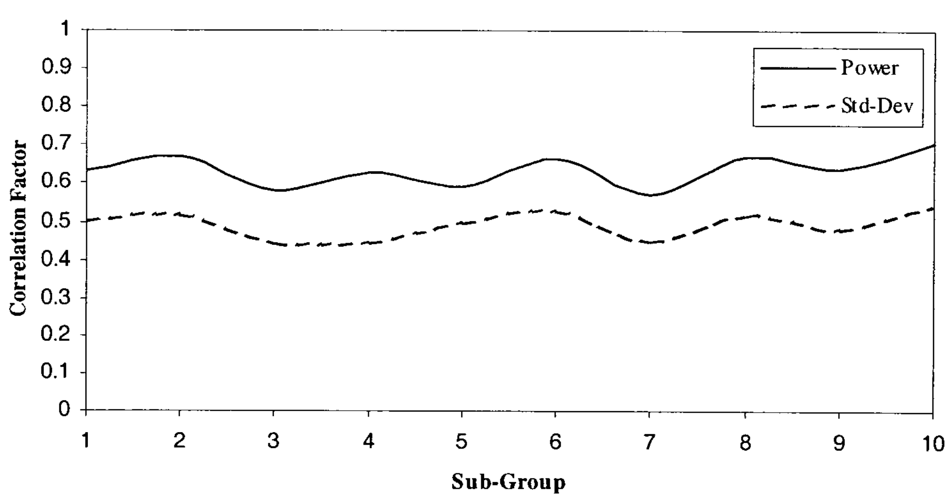




Figure B.13 Drive end bearing, fault detection – 0.007" fault diameter at outer race, 0 hp motor load and 1797 rpm.

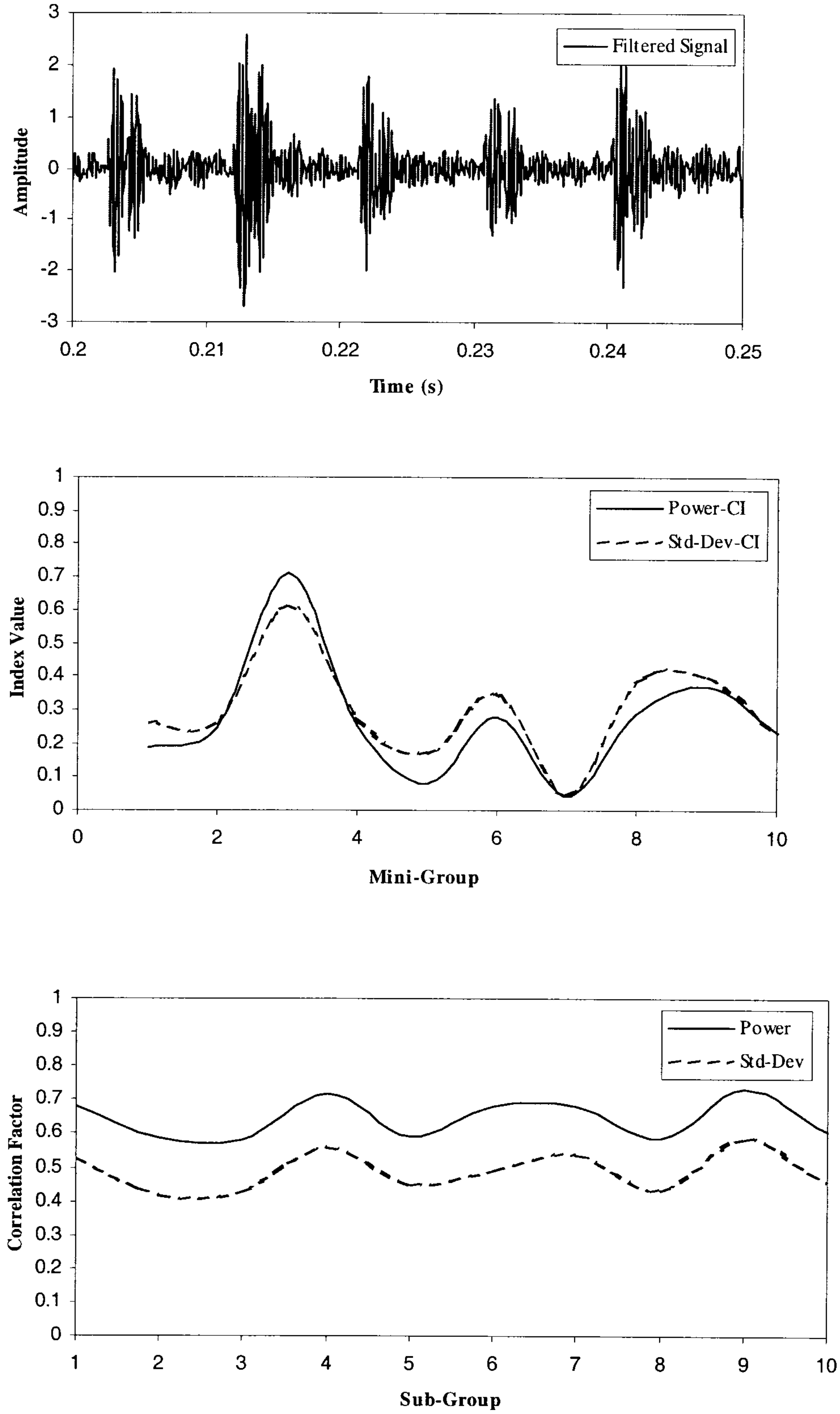



Figure B.14 Drive end bearing, fault detection – 0.007" fault diameter at outer race, 1 hp motor load and 1772 rpm.

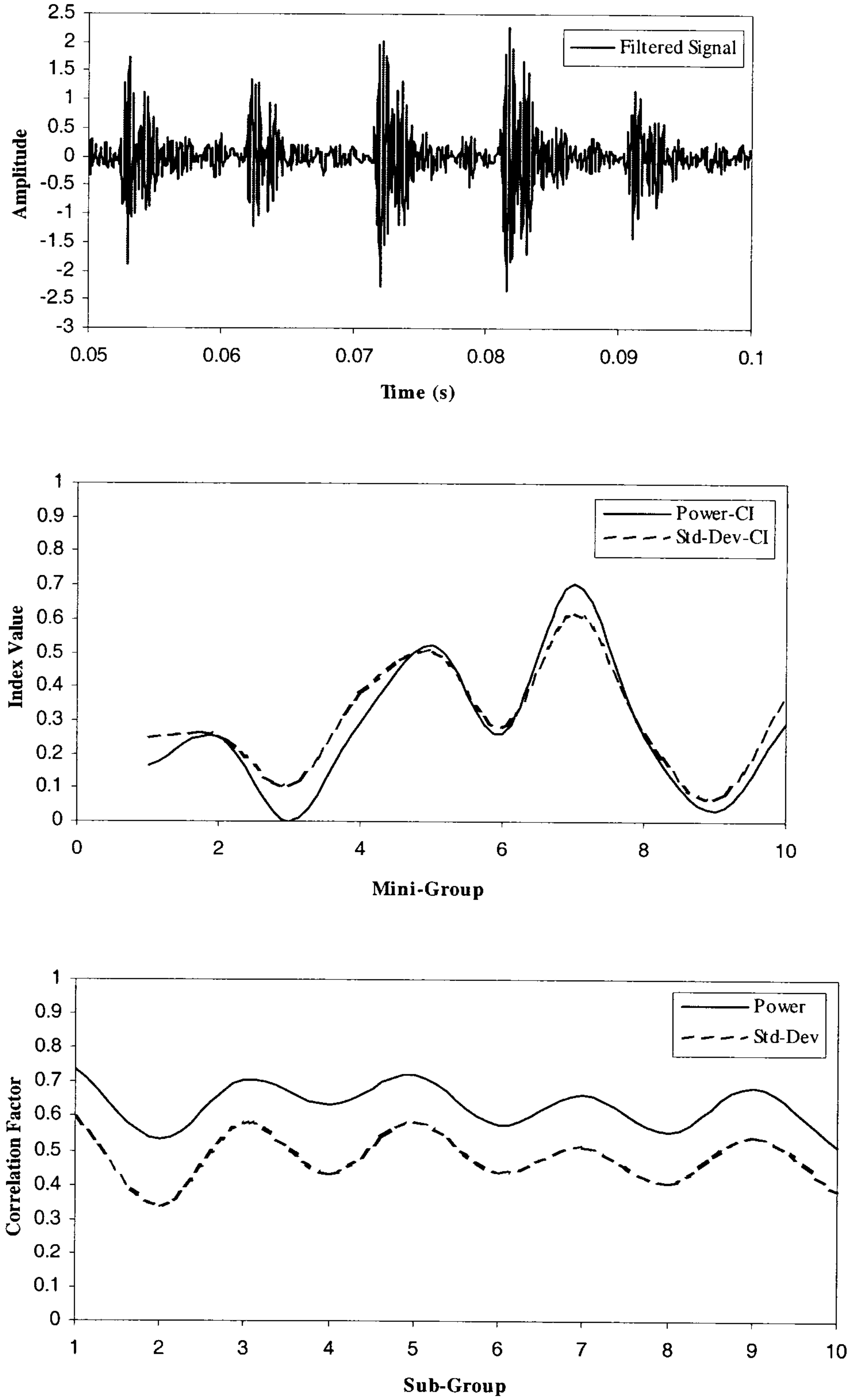



Figure B.15 Drive end bearing, fault detection – 0.007" fault diameter at outer race, 2 hp motor load and 1750 rpm.

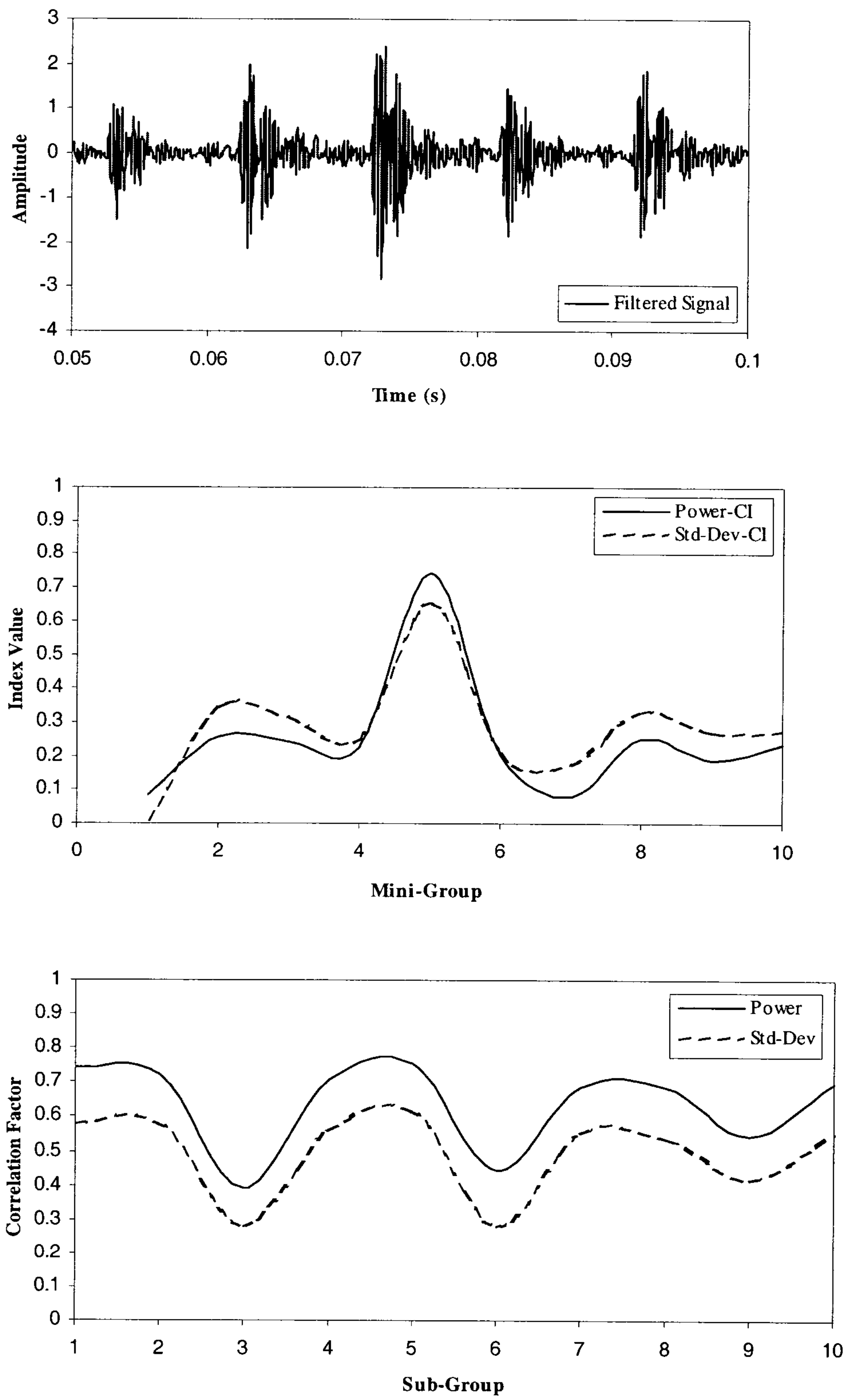
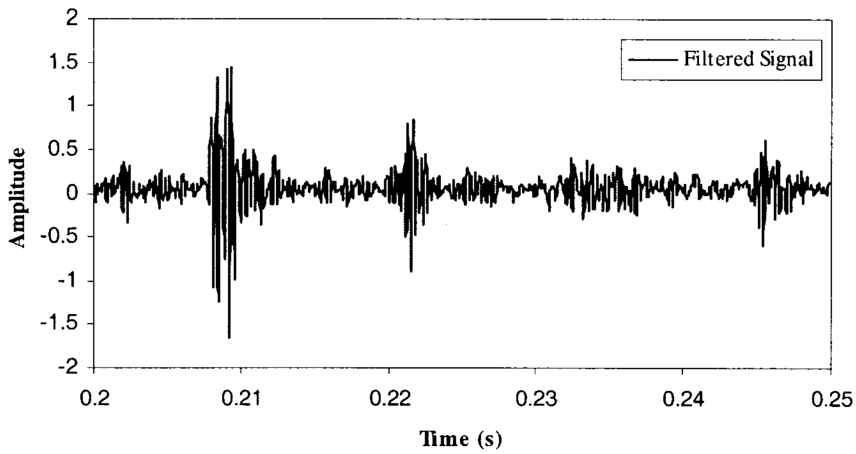
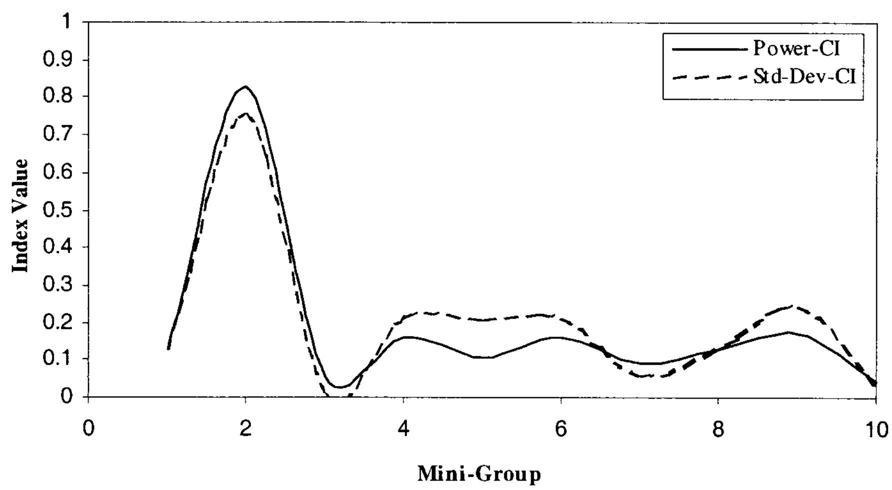



Figure B.16 Drive end bearing, fault detection – 0.007" fault diameter at outer race, 3 hp motor load and 1730 rpm.



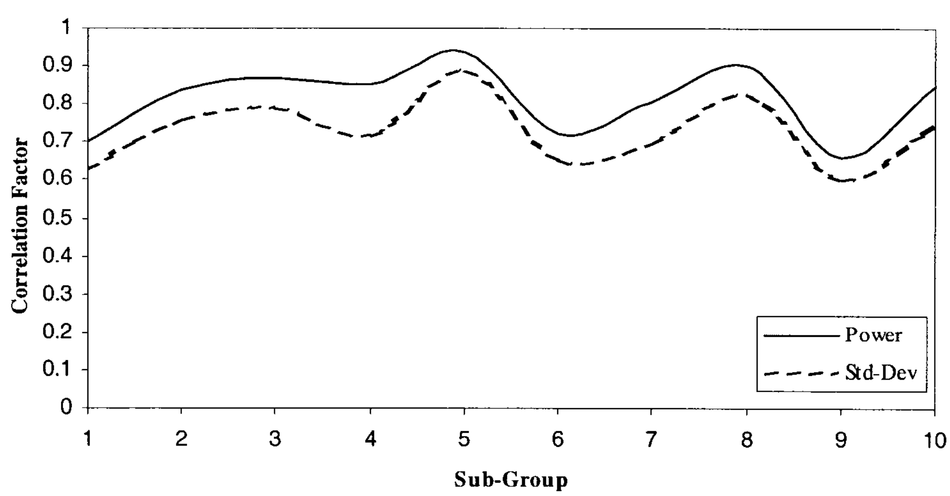




Figure B.17 Drive end bearing, fault detection – 0.014" fault diameter at inner race, 0 hp motor load and 1797 rpm.

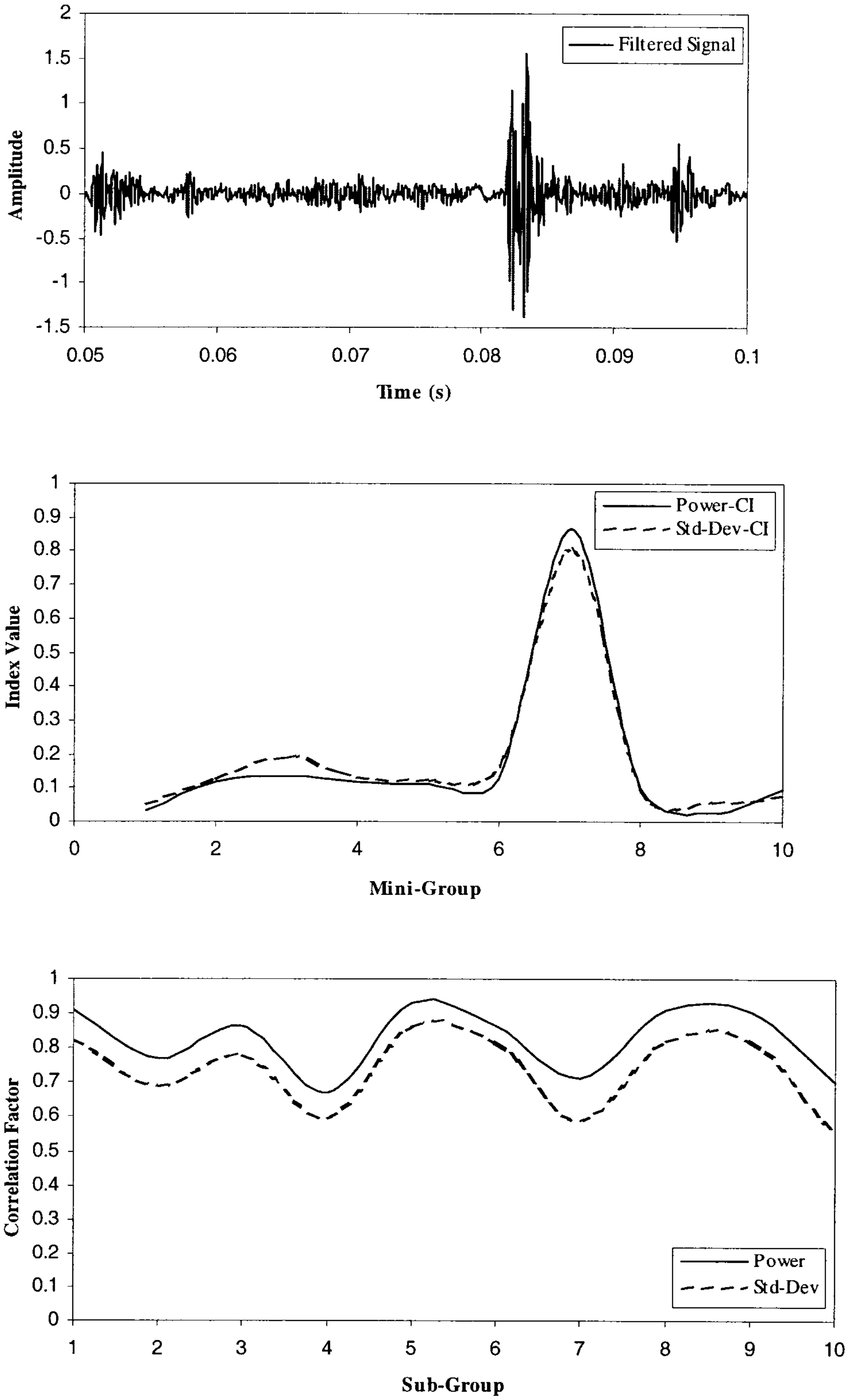
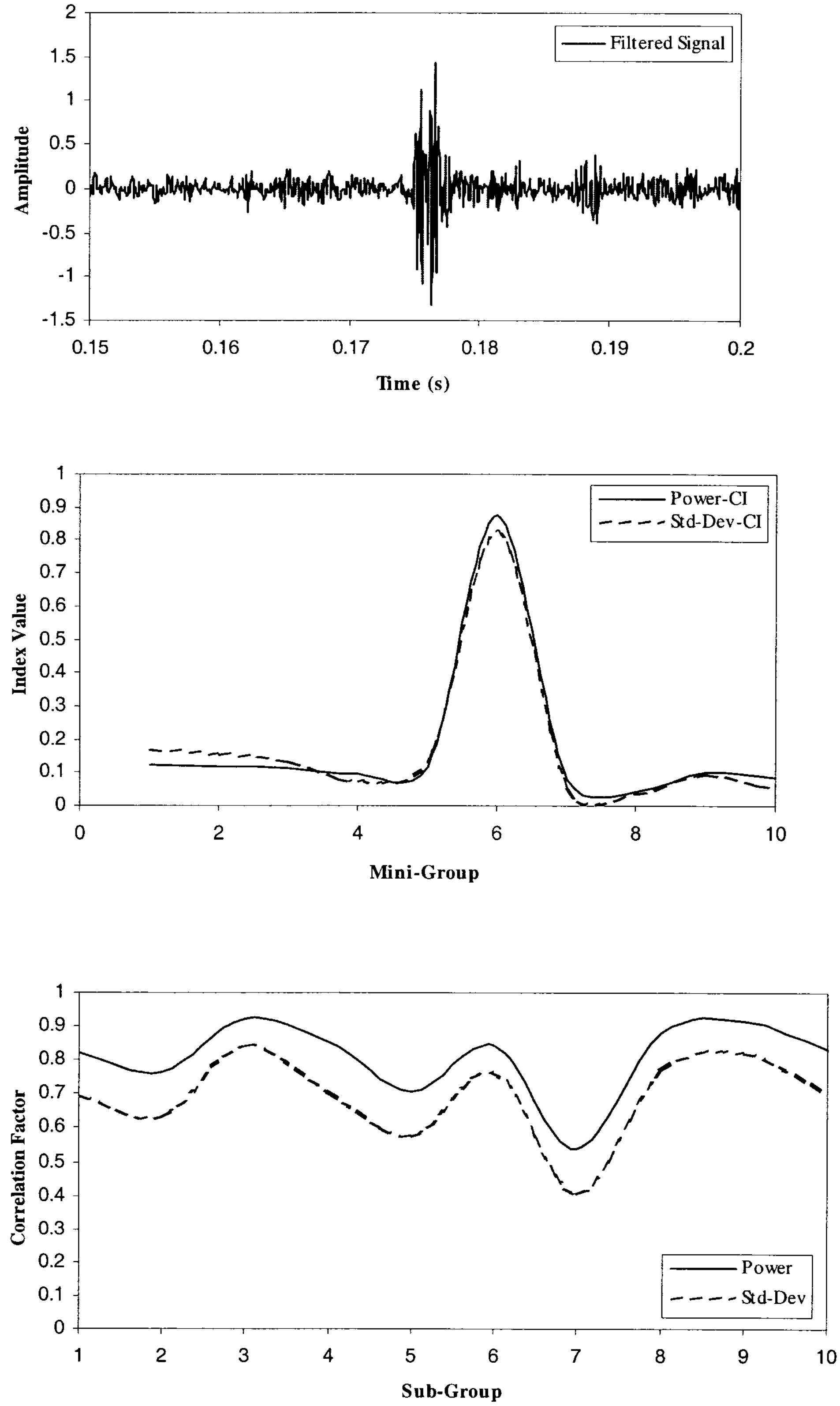
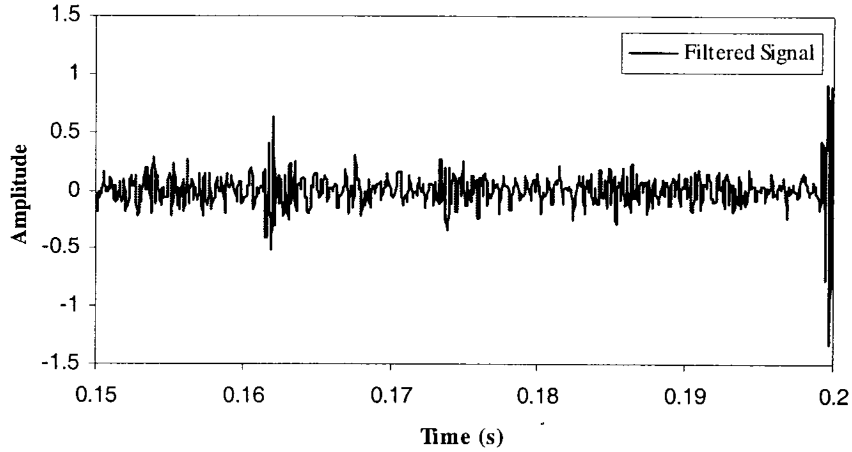
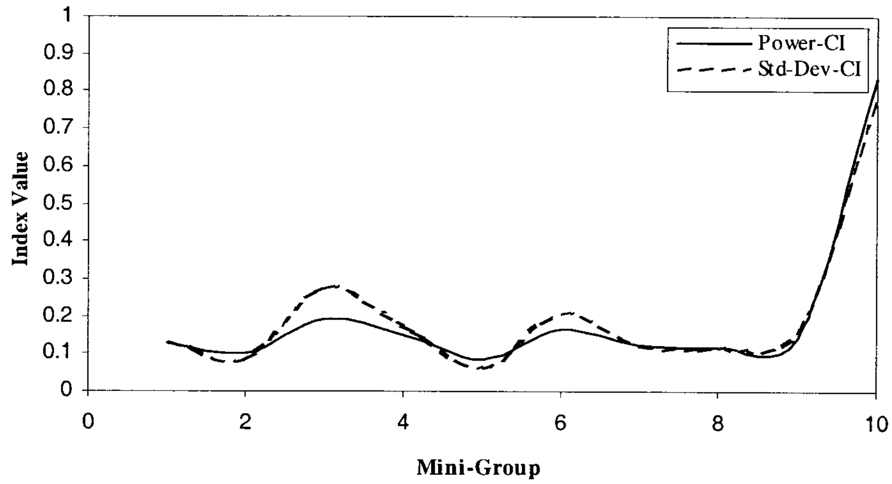



Figure B.18 Drive end bearing, fault detection – 0.014" fault diameter at inner race, 1 hp motor load and 1772 rpm.




FigureB.19 Drive end bearing, fault detection – 0.014" fault diameter at inner race, 2 hp motor load and 1750 rpm.



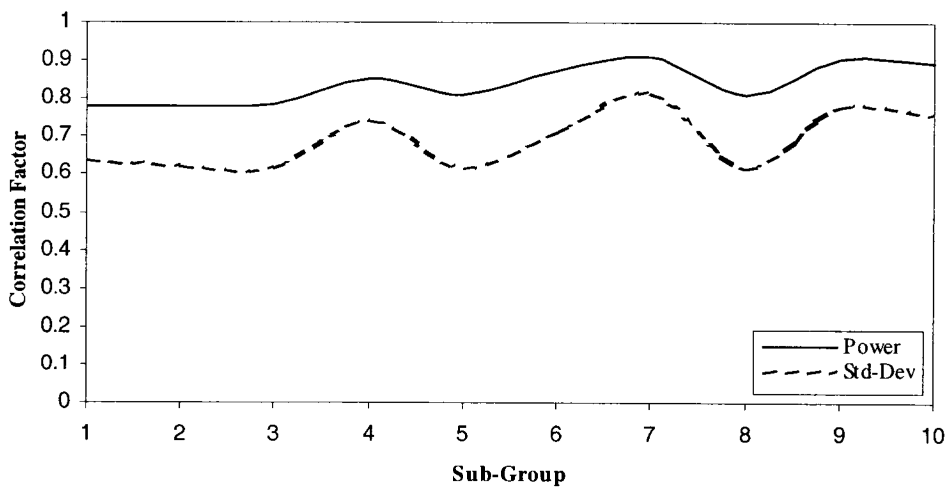
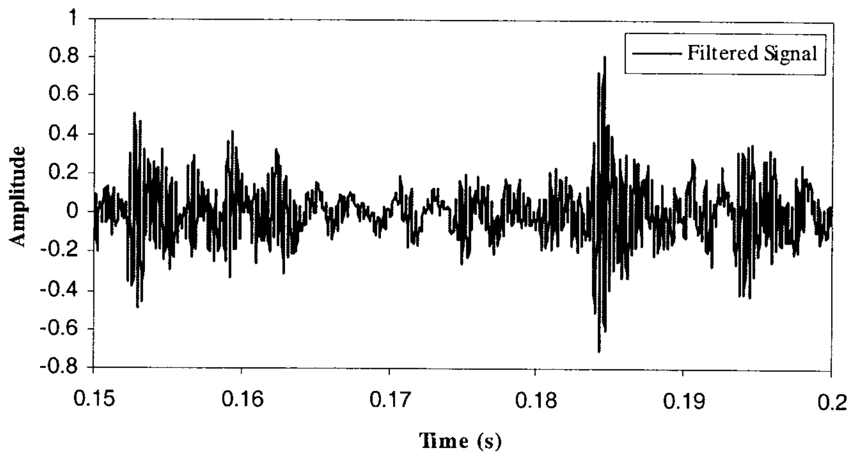
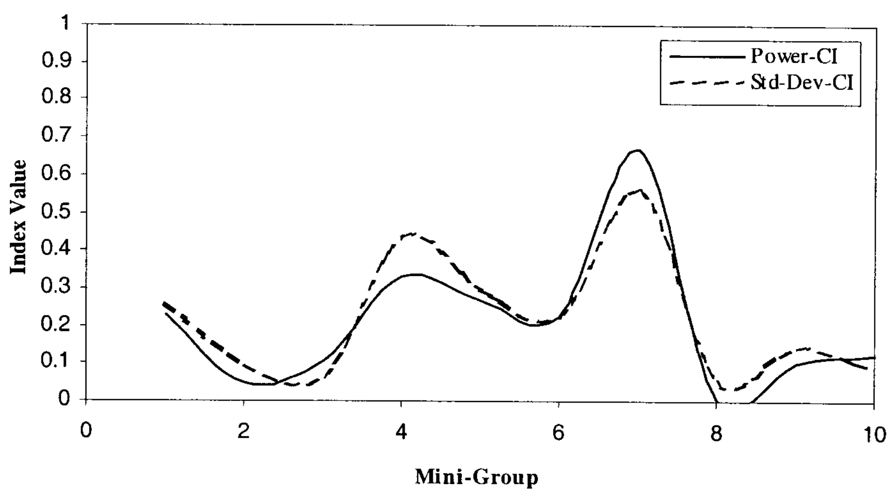




Figure B.20 Drive end bearing, fault detection – 0.014" fault diameter at inner race, 3 hp motor load and 1730 rpm



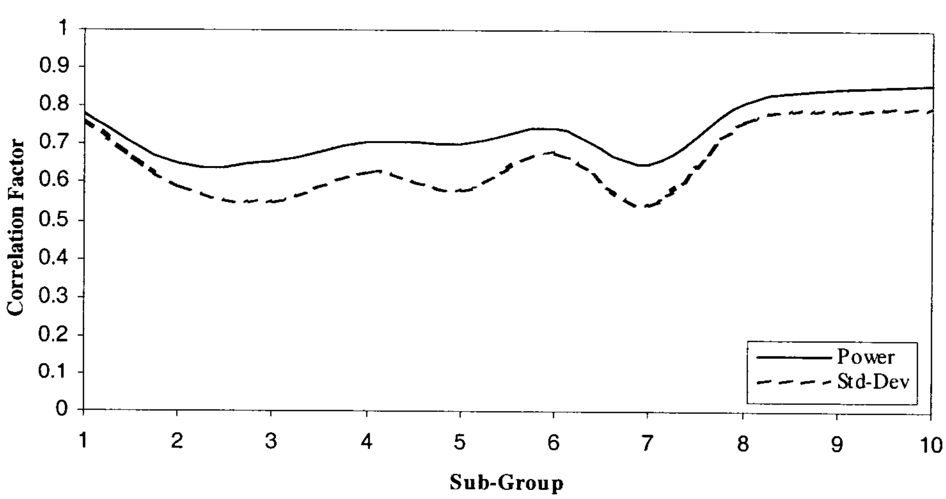




Figure B.21 Drive end bearing, fault detection – 0.014" fault diameter at ball, 0 hp motor load and 1797 rpm

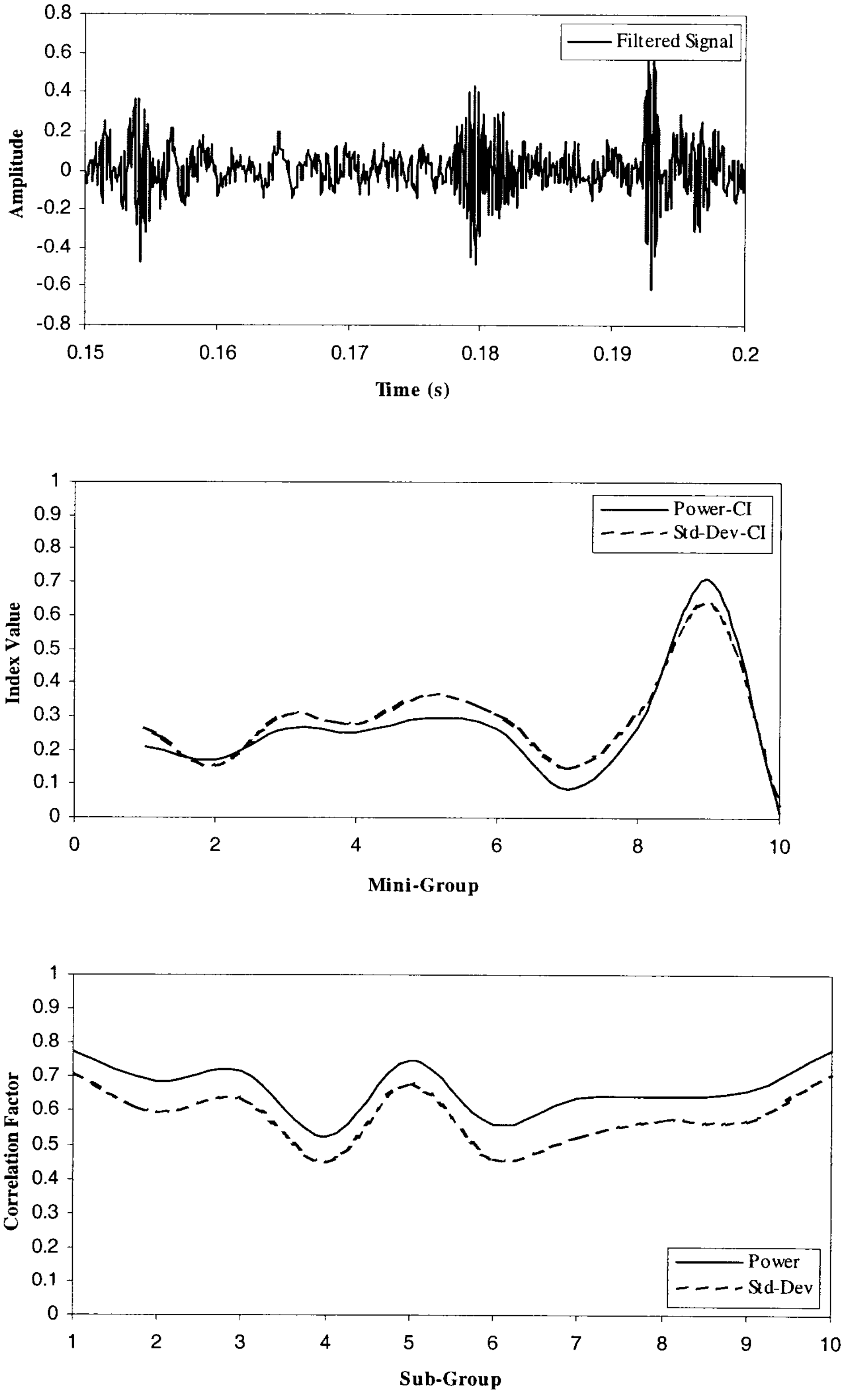



Figure B.22 Drive end bearing, fault detection – 0.014" fault diameter at ball, 1 hp motor load and 1772 rpm

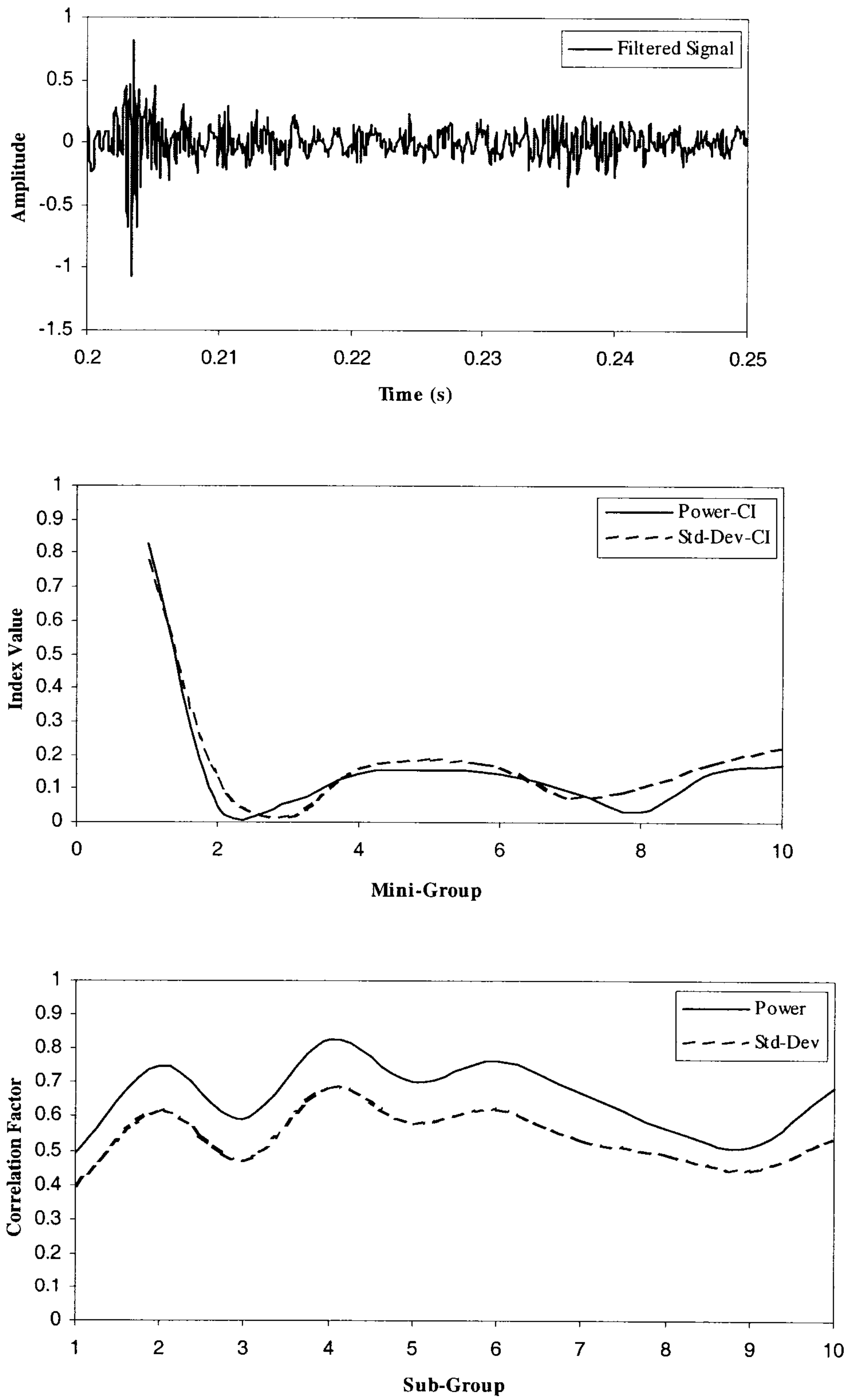
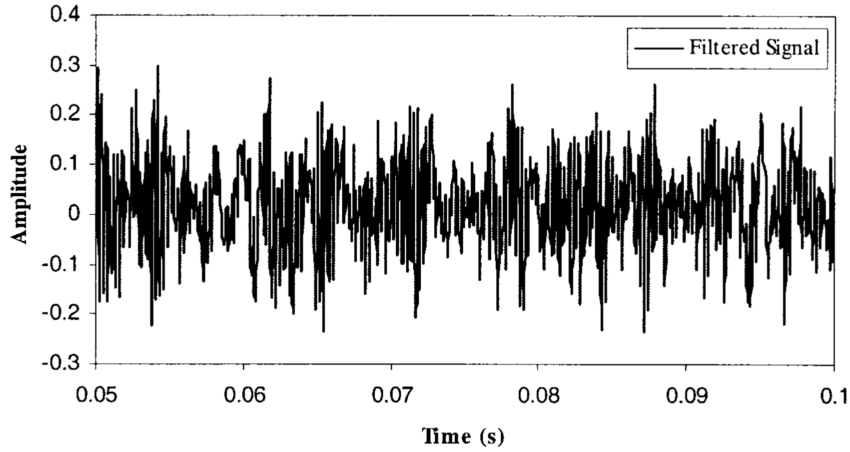
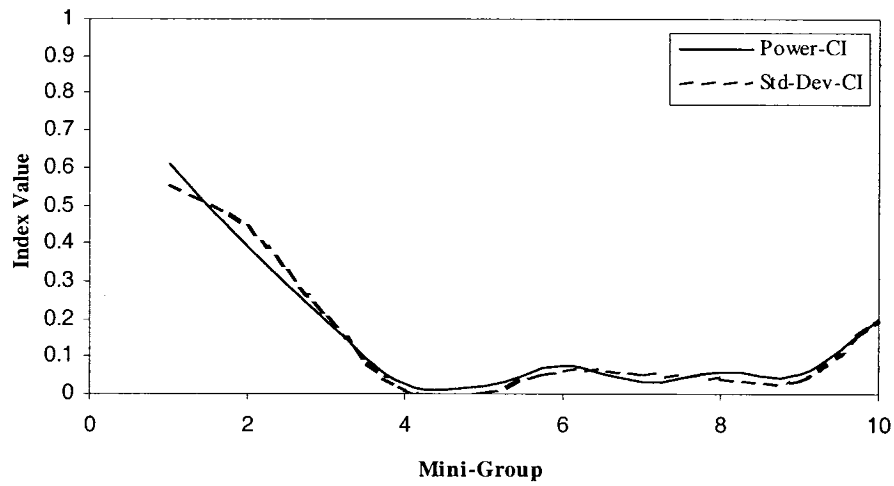



Figure B.23 Drive end bearing, fault detection – 0.014" fault diameter at ball, 2 hp motor load and 1750 rpm



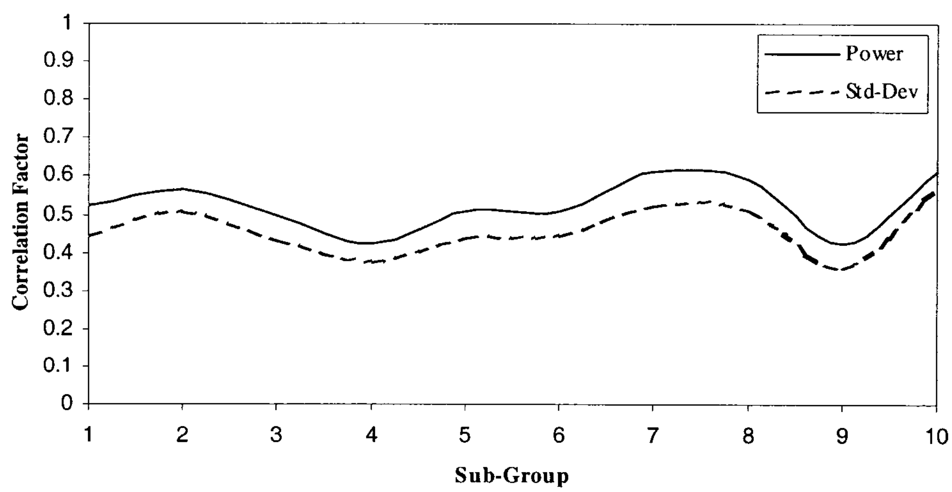




Figure B.24 Drive end bearing, fault detection – 0.014" fault diameter at outer race, 0 hp motor load and 1797 rpm.

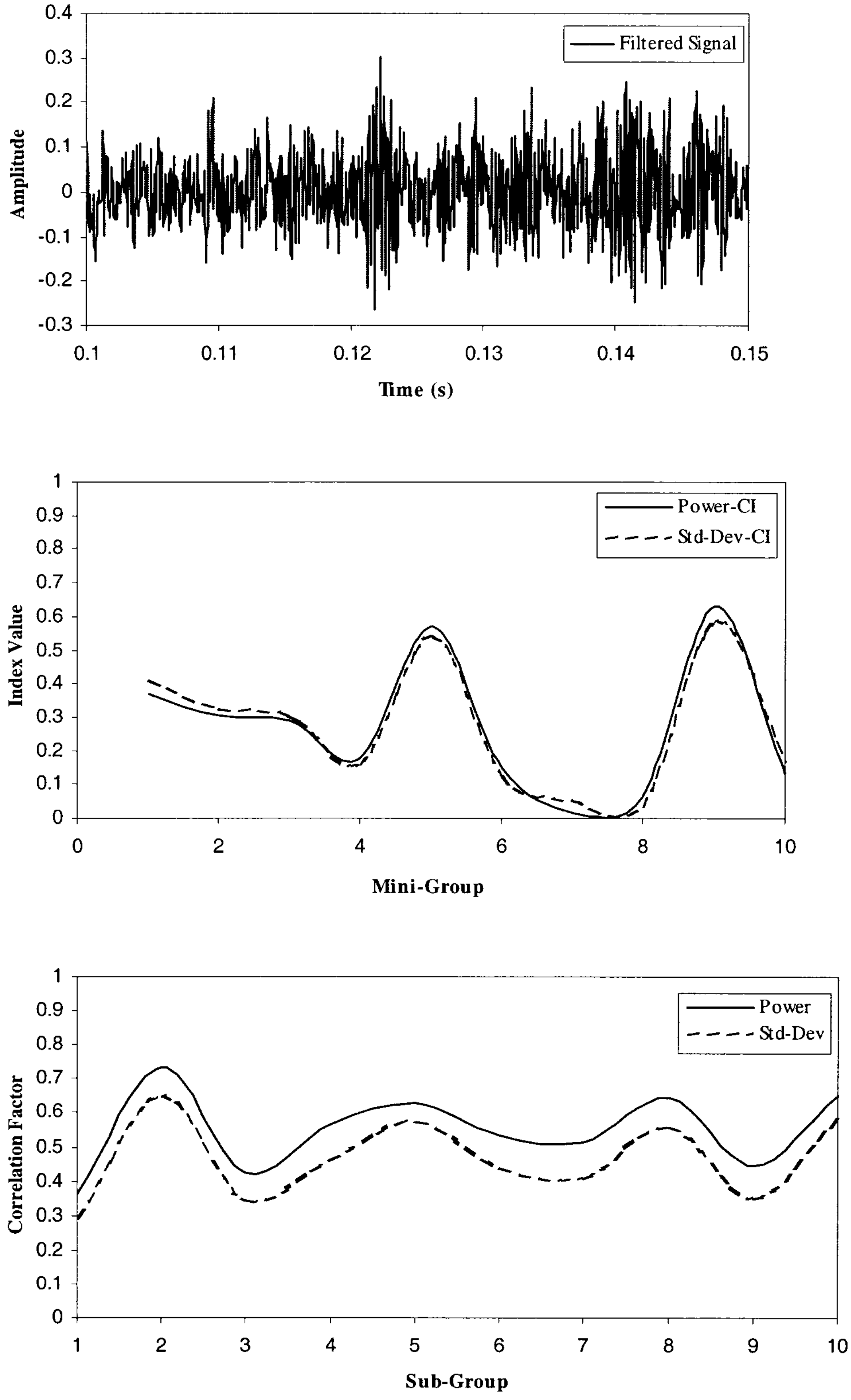



Figure B.25 Drive end bearing, fault detection – 0.014” fault diameter at outer, 1 hp motor load and 1772 rpm.

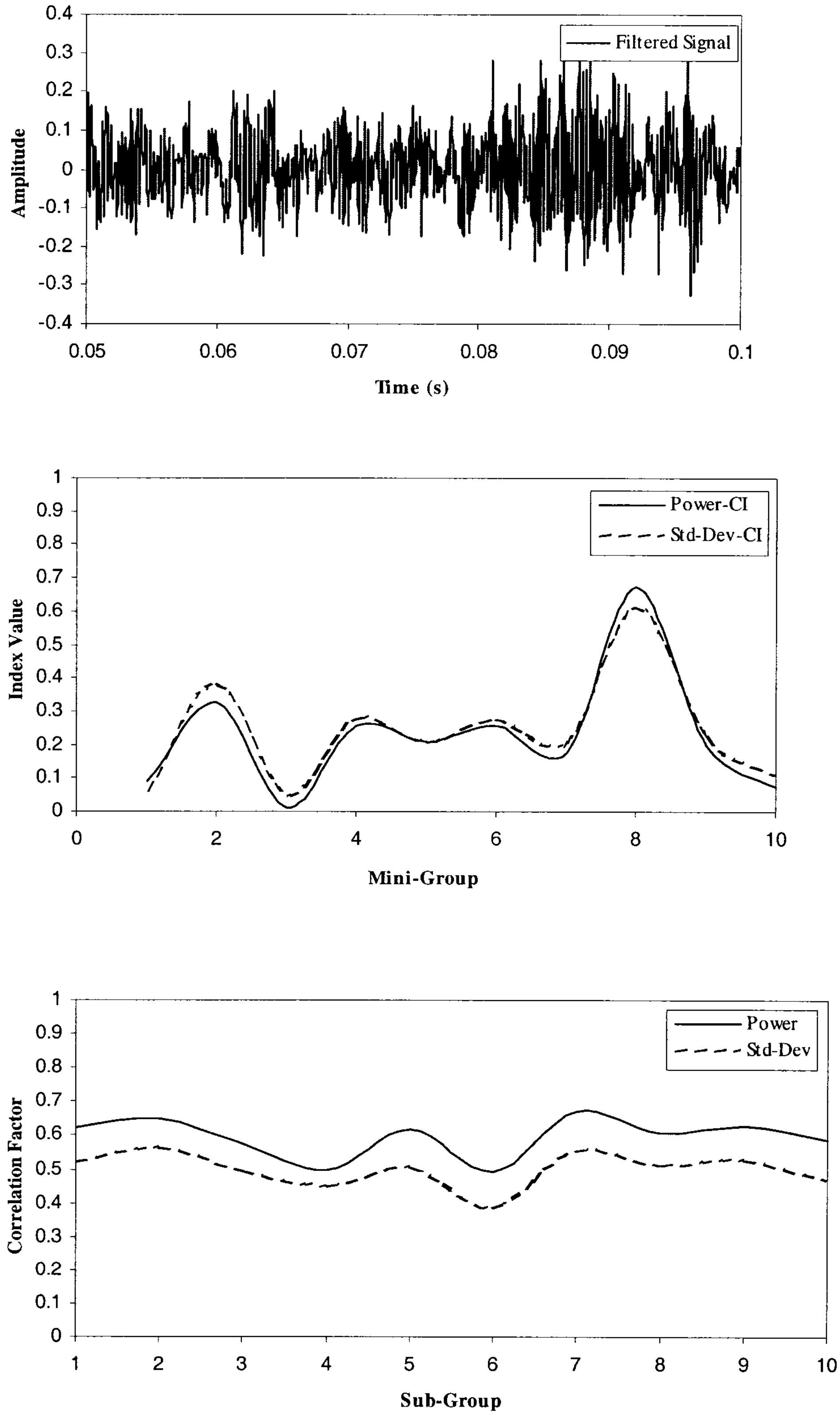



Figure B.26 Drive end bearing, fault detection – 0.014" fault diameter at outer race, 2 hp motor load and 1750 rpm.

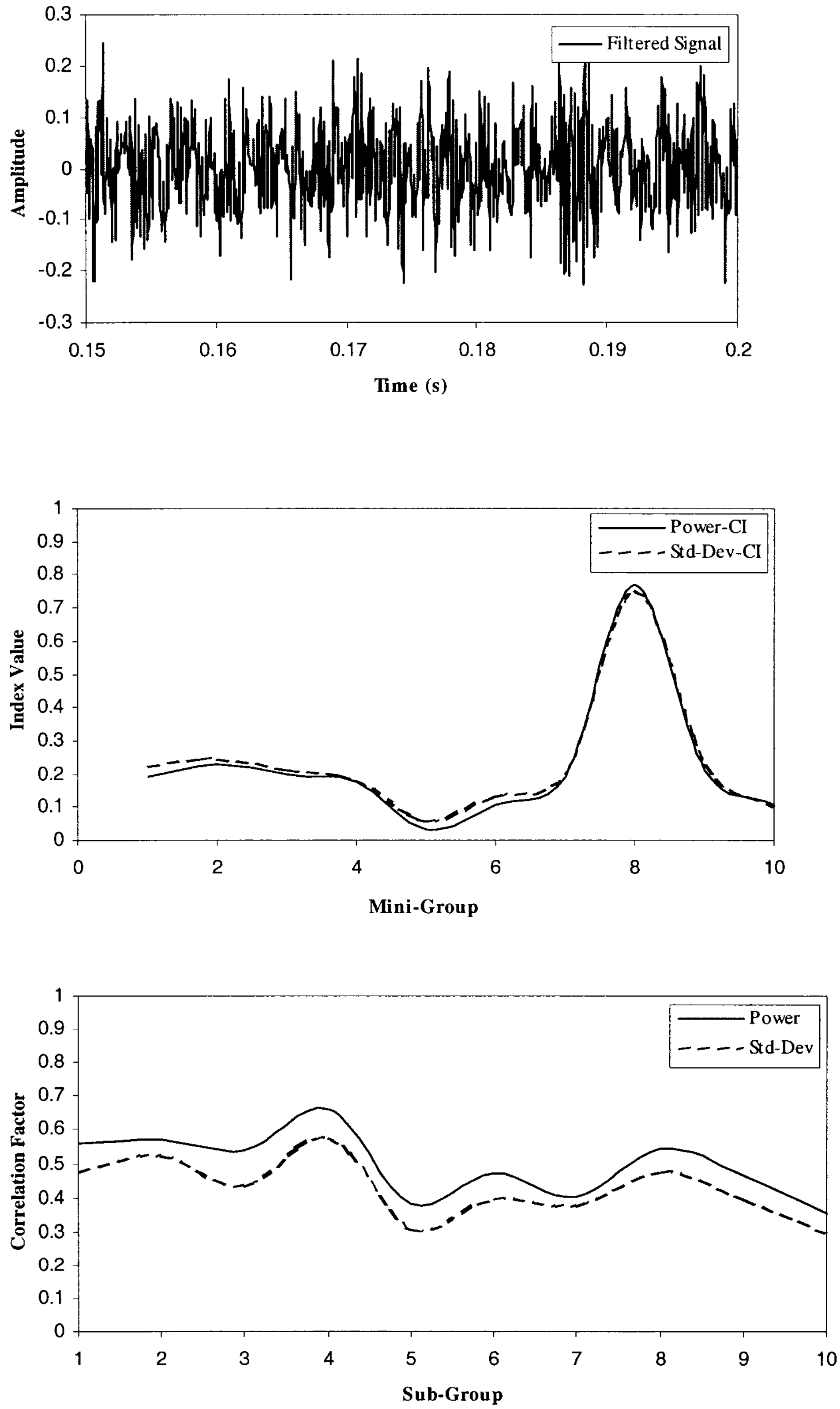



Figure B.27 Drive end bearing, fault detection – 0.014" fault diameter at outer, 3 hp motor load and 1730 rpm.

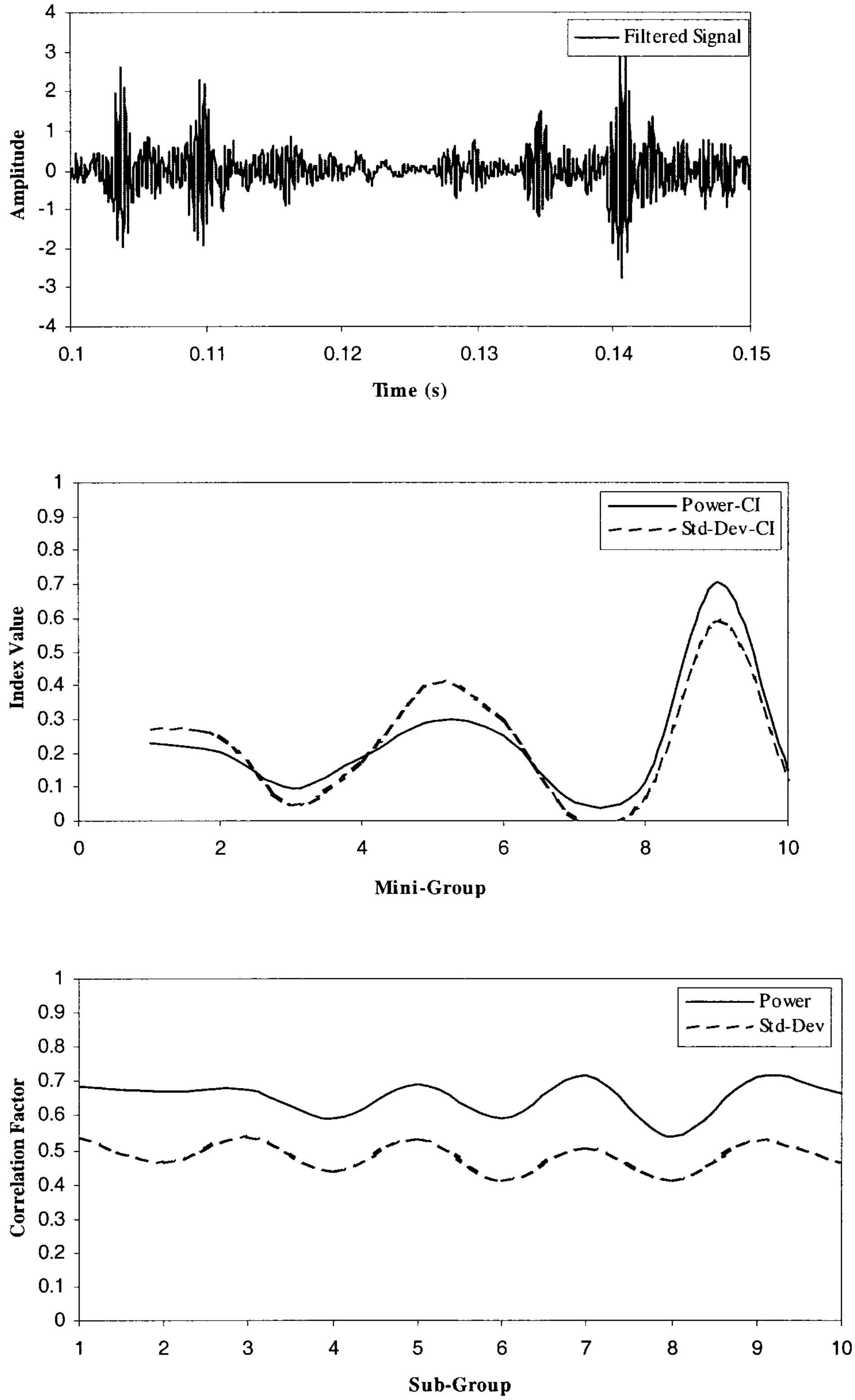



Figure B.28 Drive end bearing, fault detection – 0.021" fault diameter at inner, 0 hp motor load and 1797 rpm

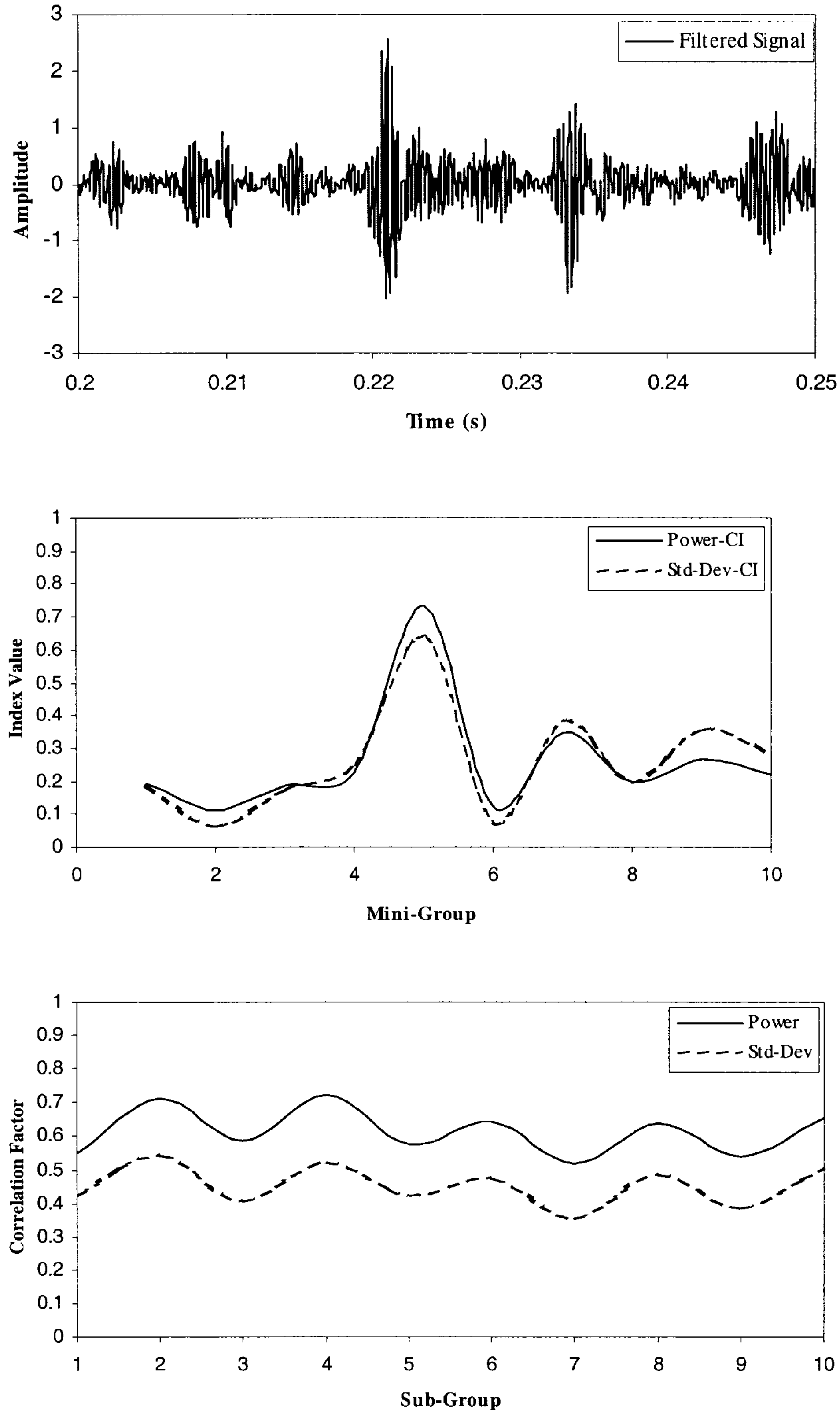
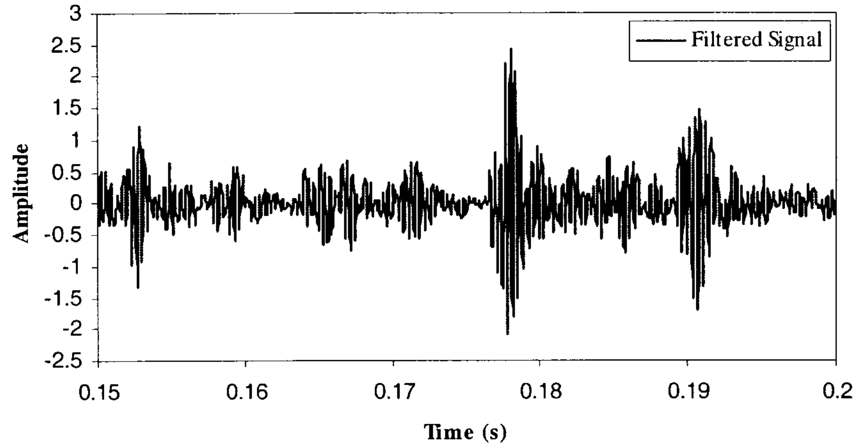
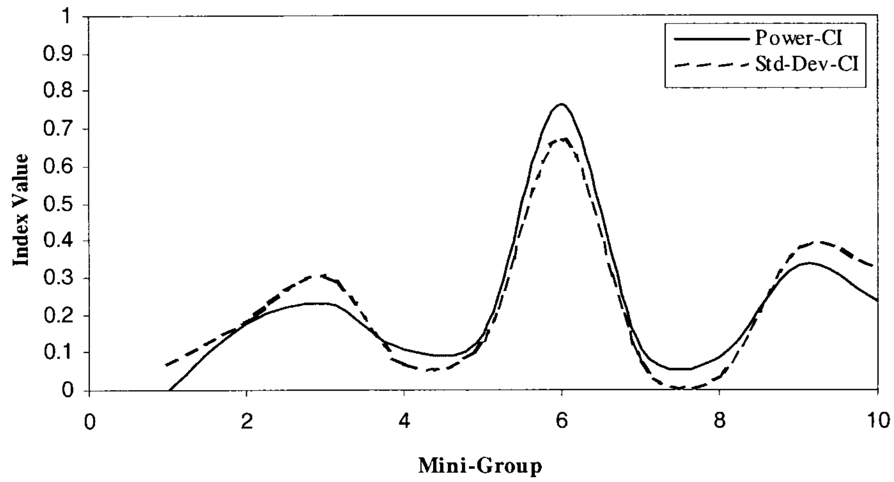



Figure B.29 Drive end bearing, fault detection – 0.021" fault diameter at inner, 1 hp motor load and 1772 rpm



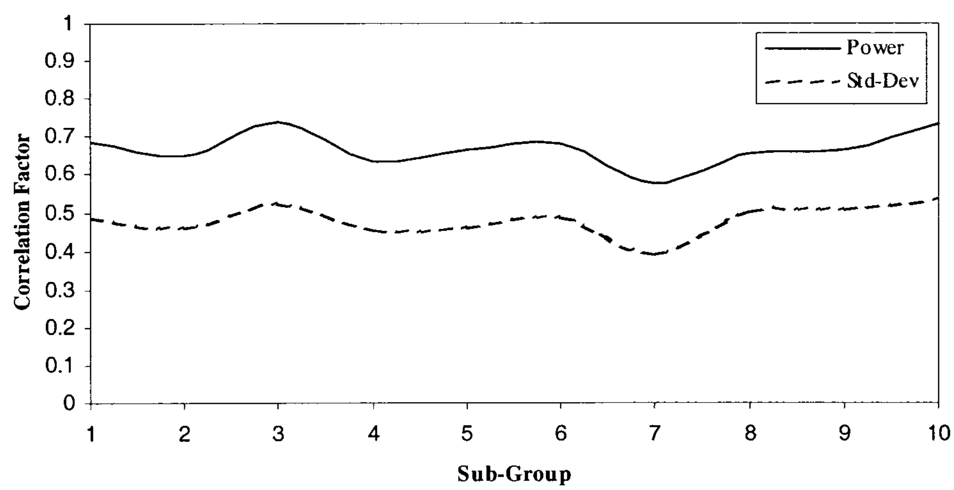




Figure B.30 Drive end bearing, fault detection – 0.021" fault diameter at inner, 2 hp motor load and 1750rpm.

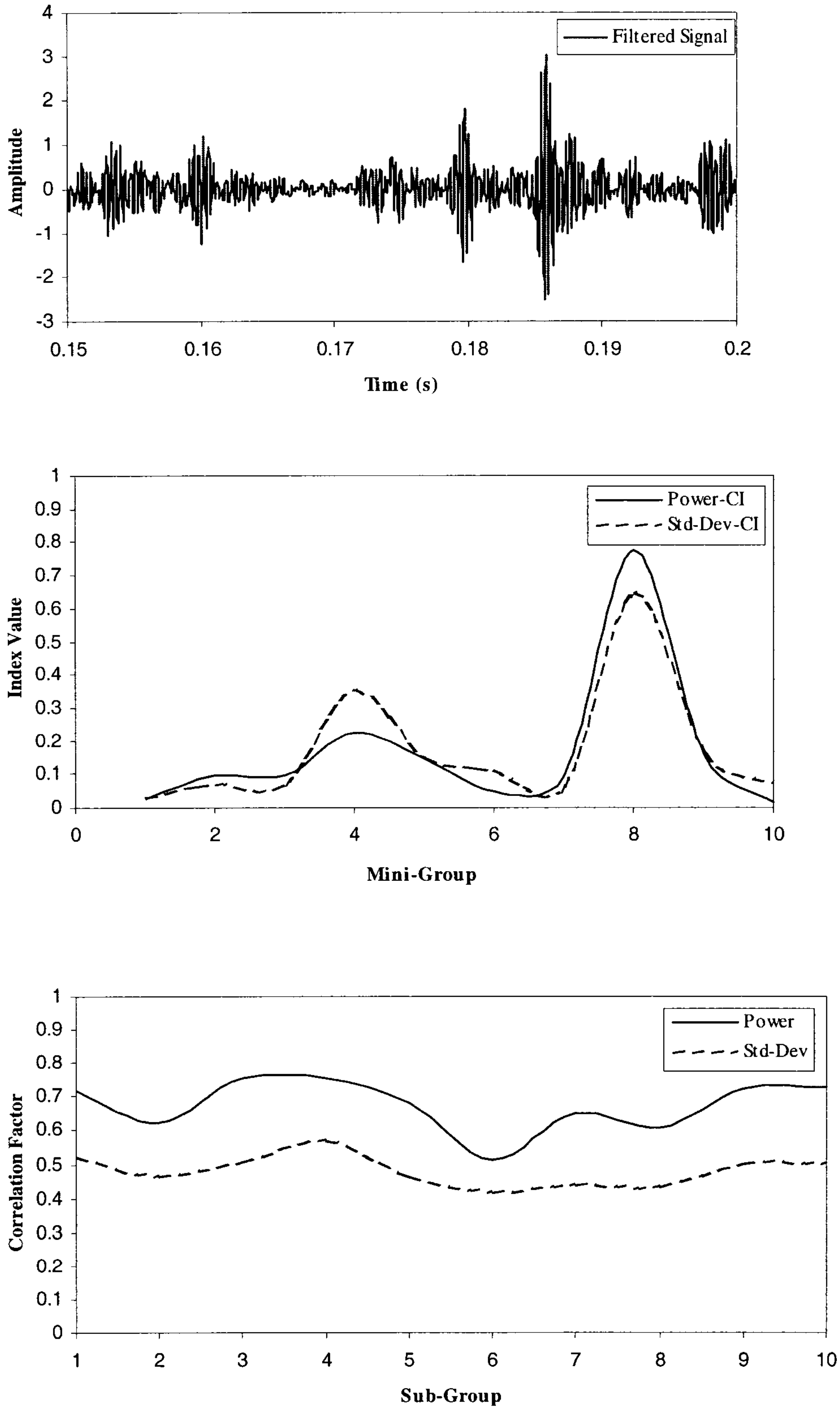
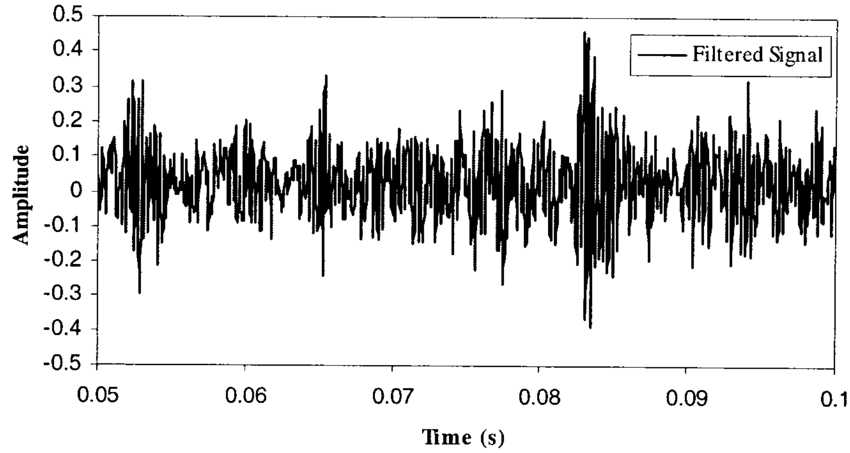
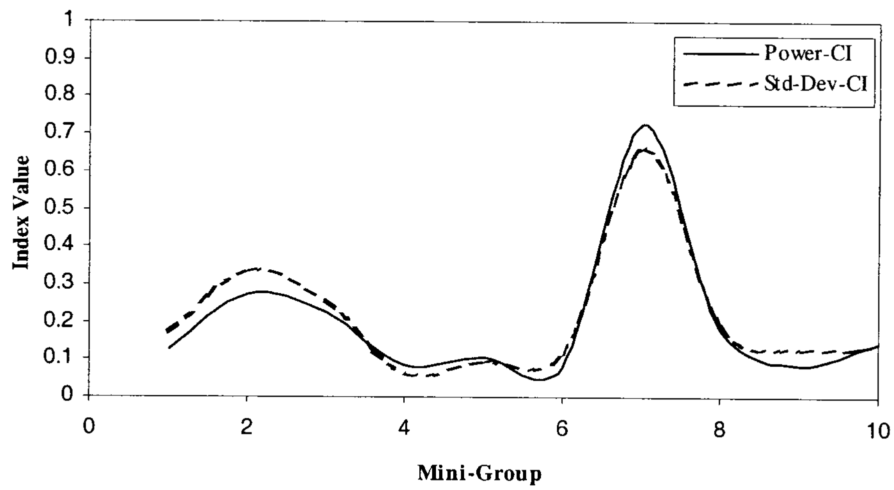



Figure B.31 Drive end bearing, fault detection – 0.021" fault diameter at inner, 3 hp motor load and 1730rpm



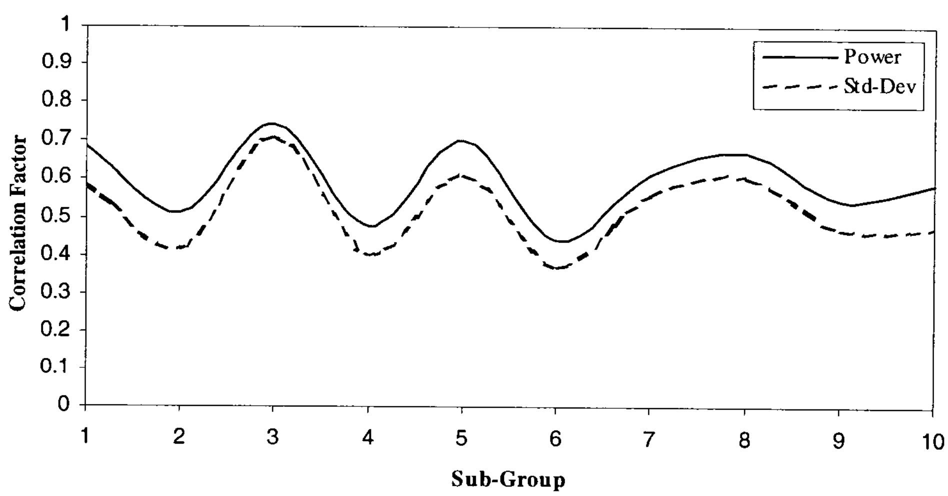
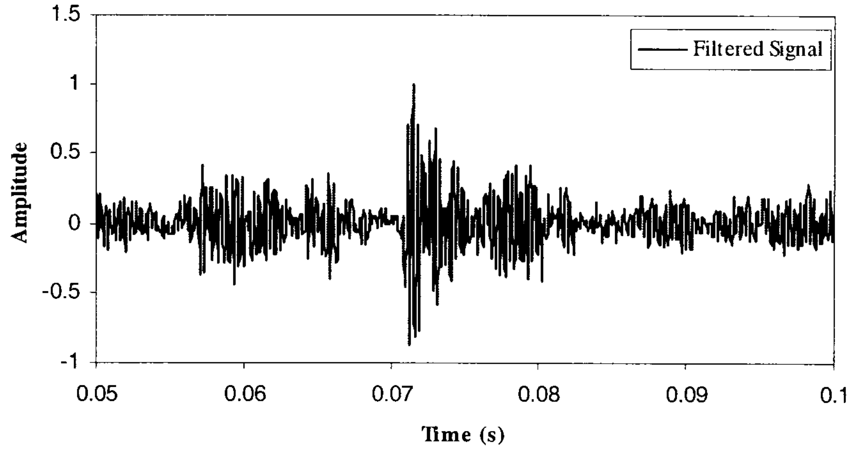
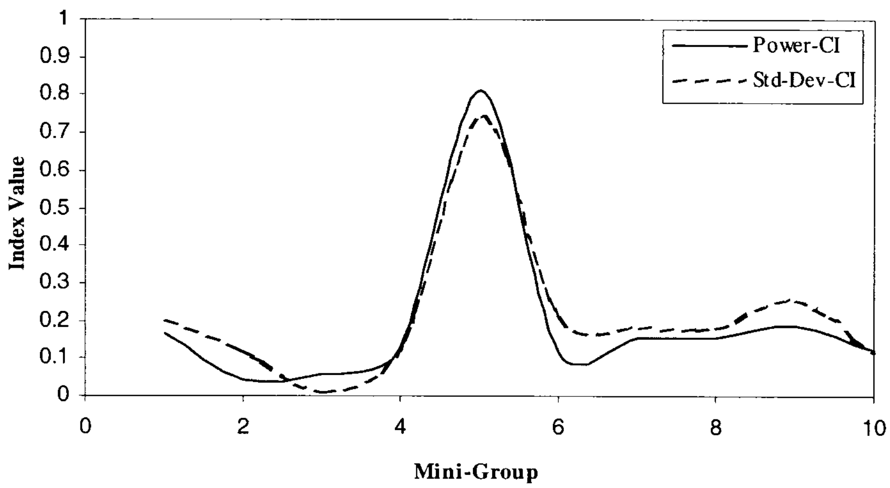




Figure B.32 Drive end bearing, fault detection – 0.021" fault diameter at ball, 0 hp motor load and 1797rpm.



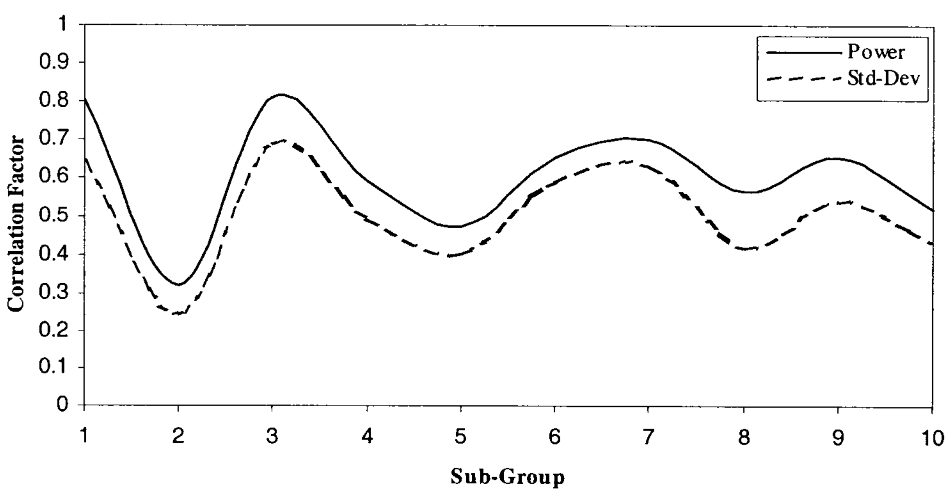
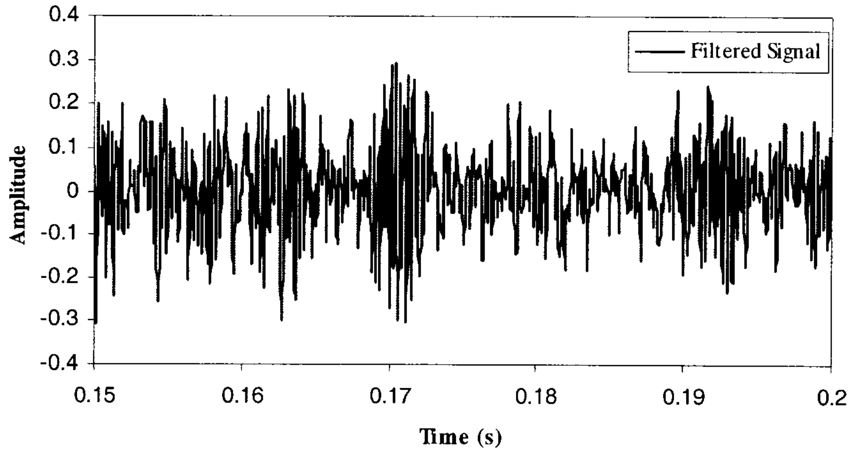
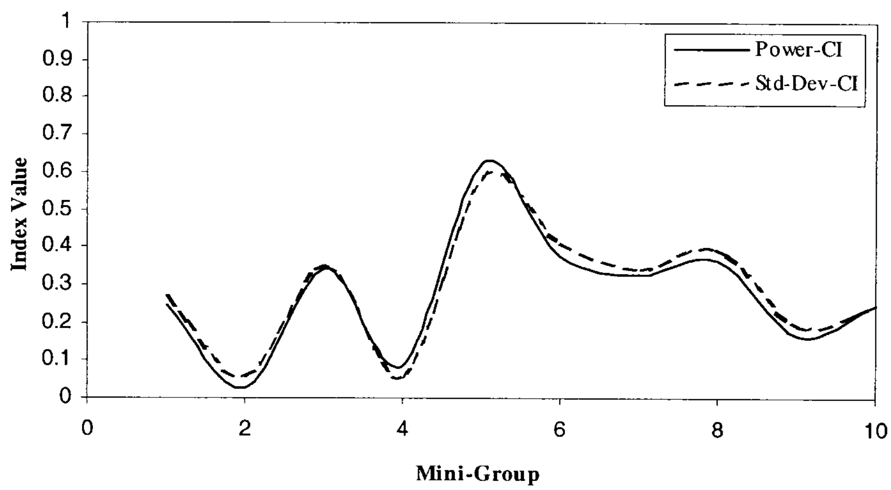




Figure B.33 Drive end bearing, fault detection – 0.021" fault diameter at ball, 1 hp motor load and 1772 rpm.



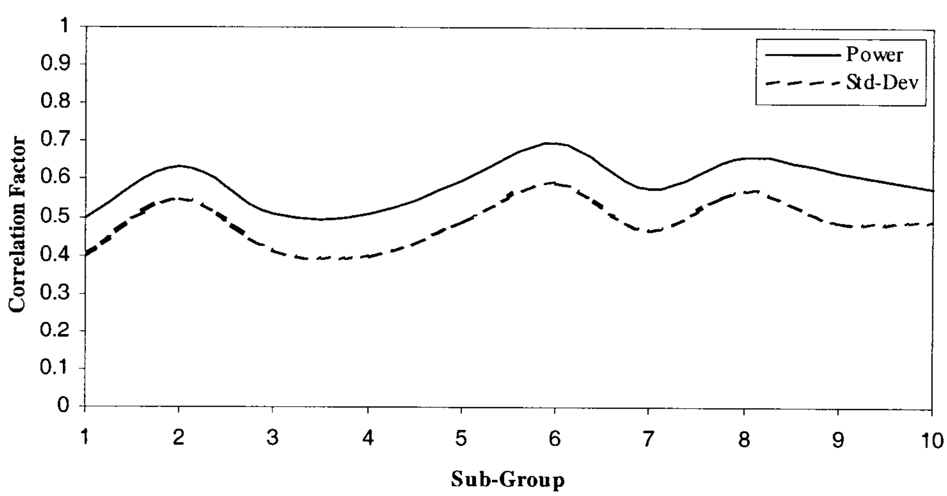
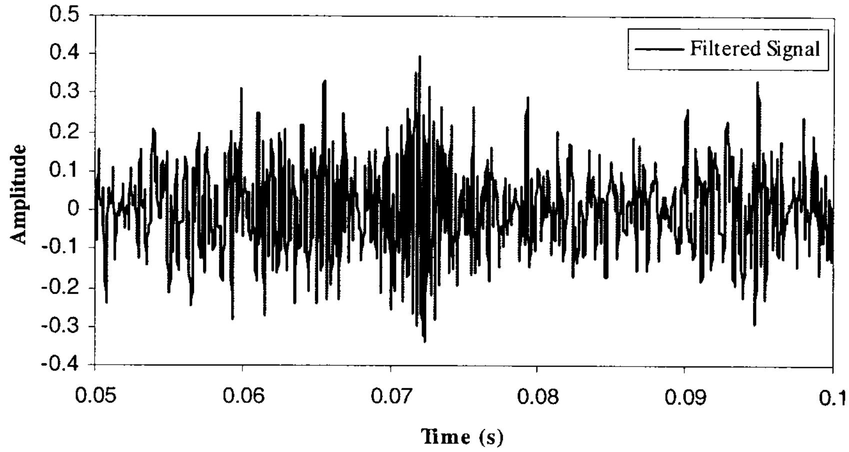
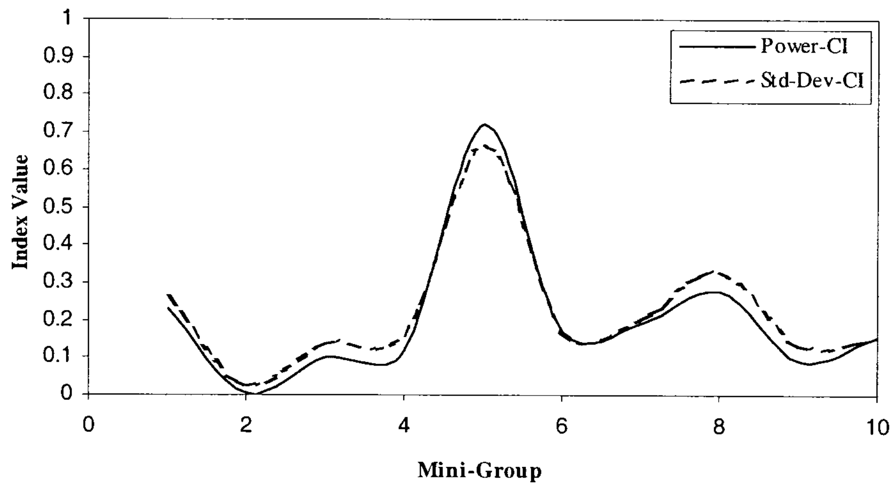




Figure B.34 Drive end bearing, fault detection – 0.021" fault diameter at ball, 2 hp motor load and 1750 rpm.



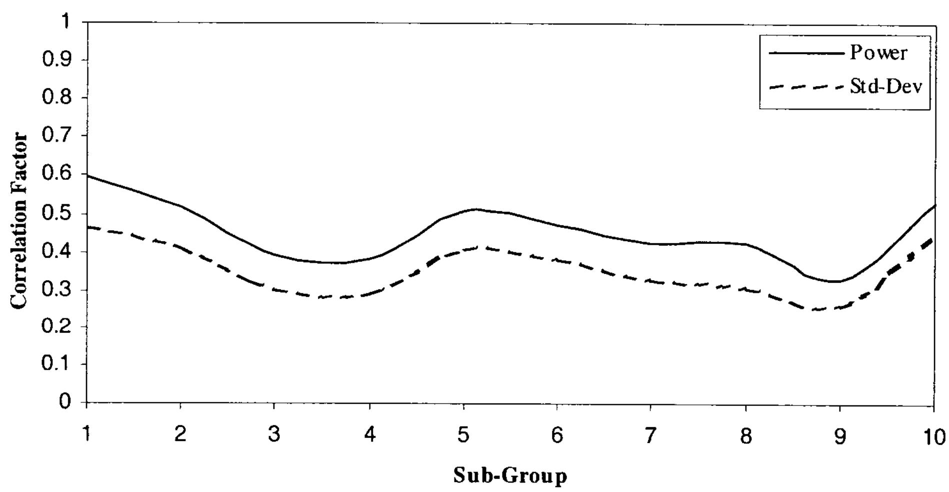
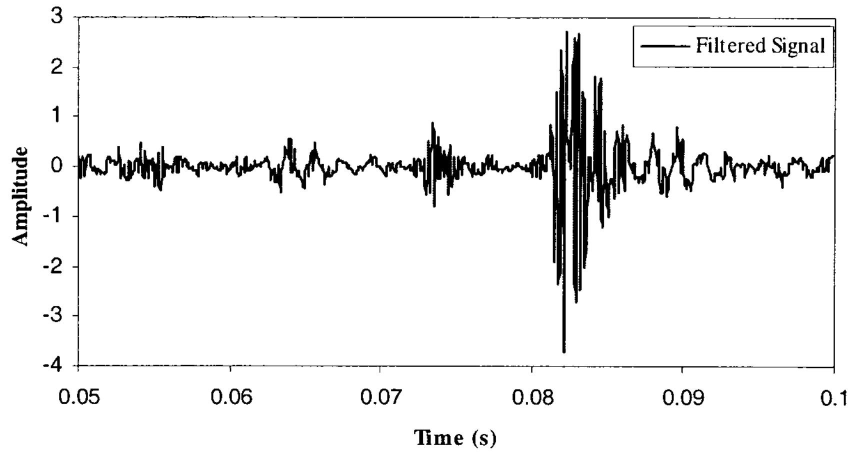
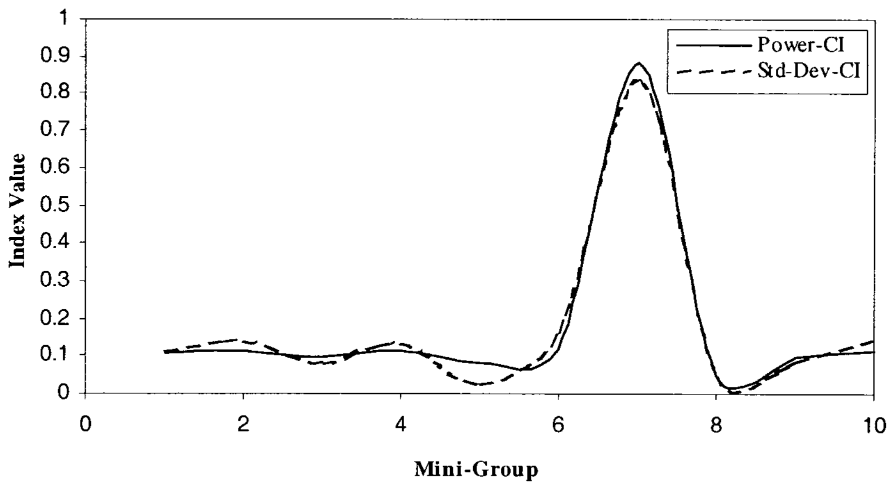




Figure B.35 Drive end bearing, fault detection – 0.021" fault diameter at ball, 3 hp motor load and 1730 rpm.



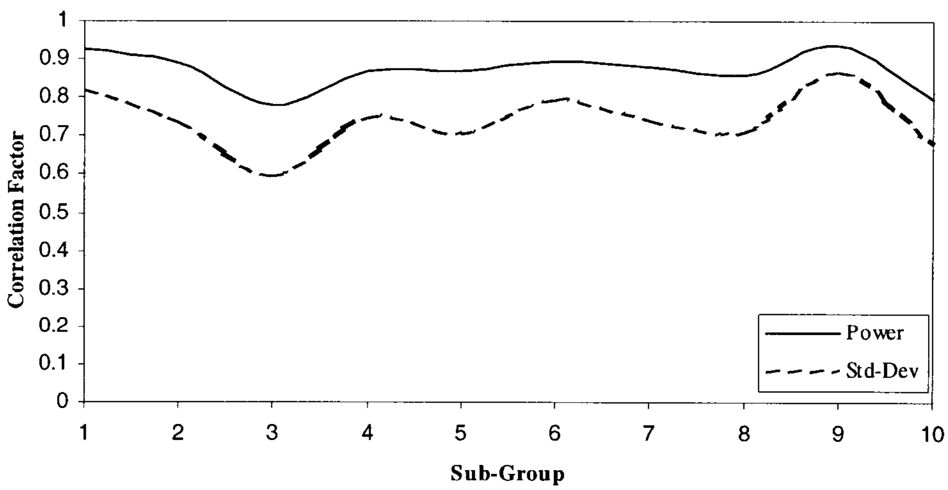




Figure B.36 Drive end bearing, fault detection – 0.021" fault diameter at outer race, 0 hp motor load and 1797 rpm.

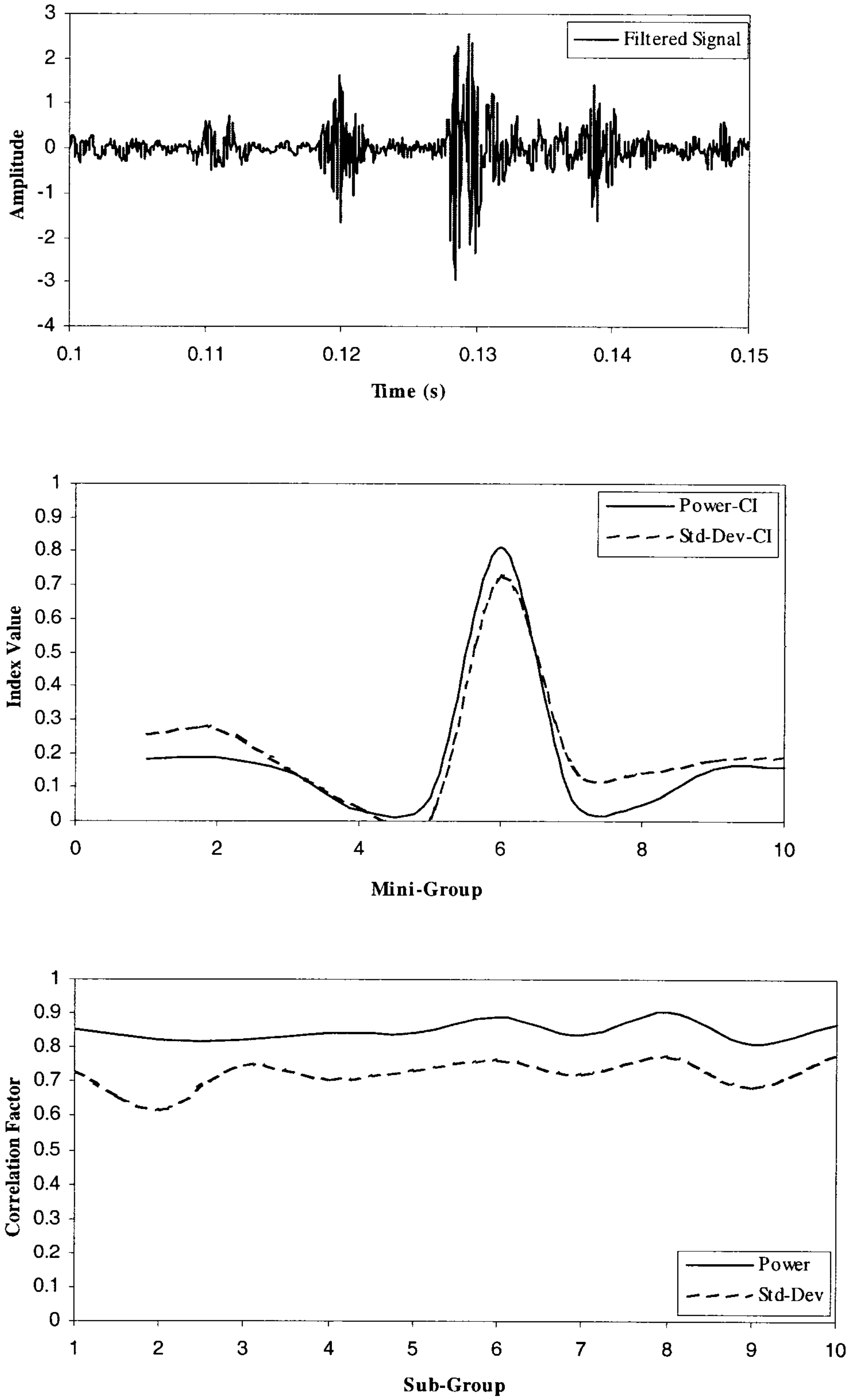



Figure B.37 Drive end bearing, fault detection – 0.021" fault diameter at outer, 1 hp motor load and 1772 rpm.

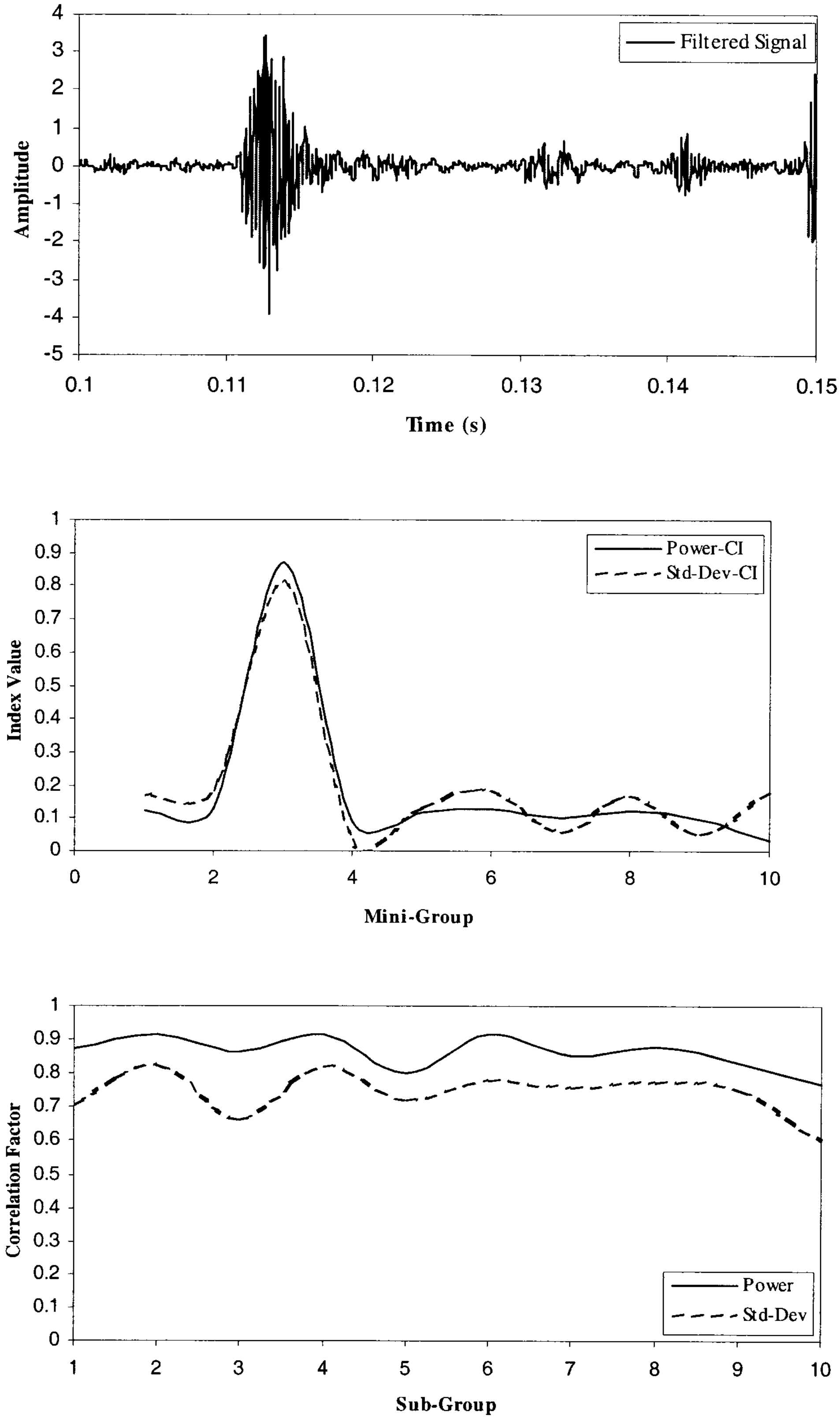



Figure B.38 Drive end bearing, fault detection – 0.021" fault diameter at outer, 2 hp motor load and 1750 rpm.

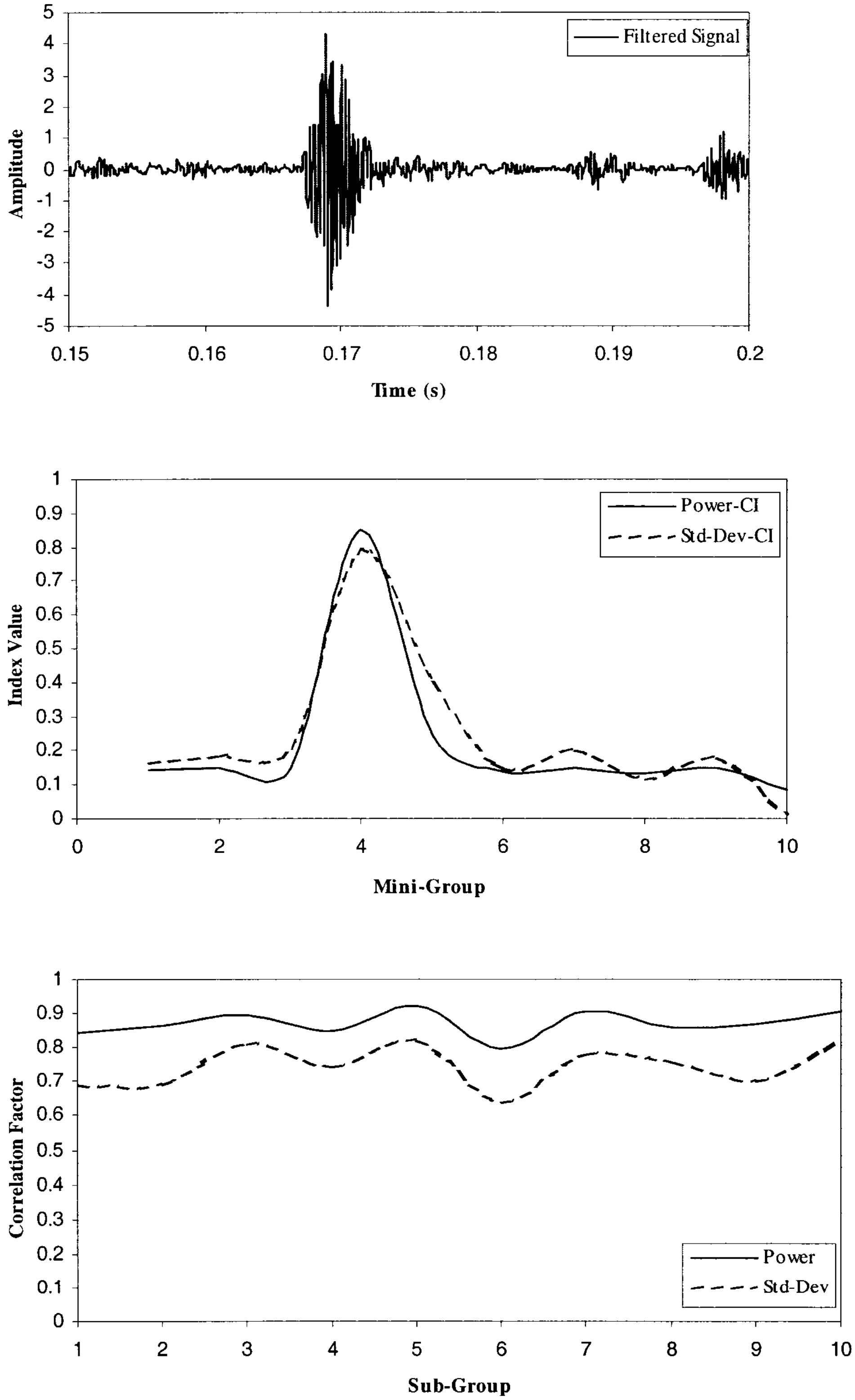
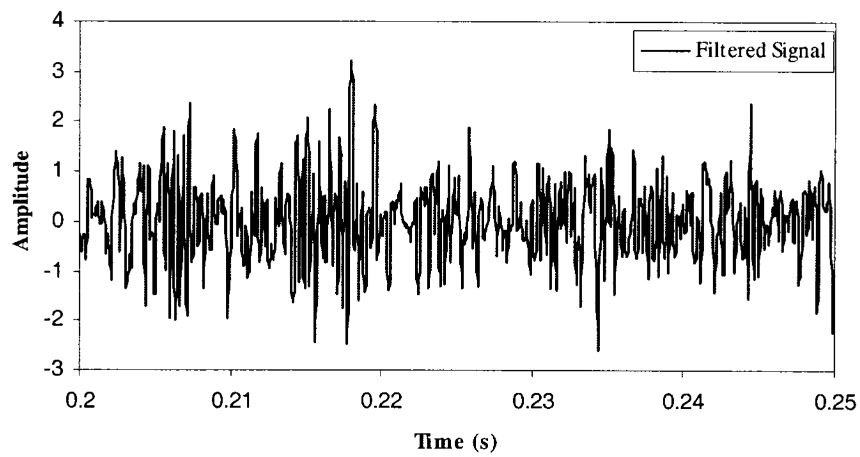
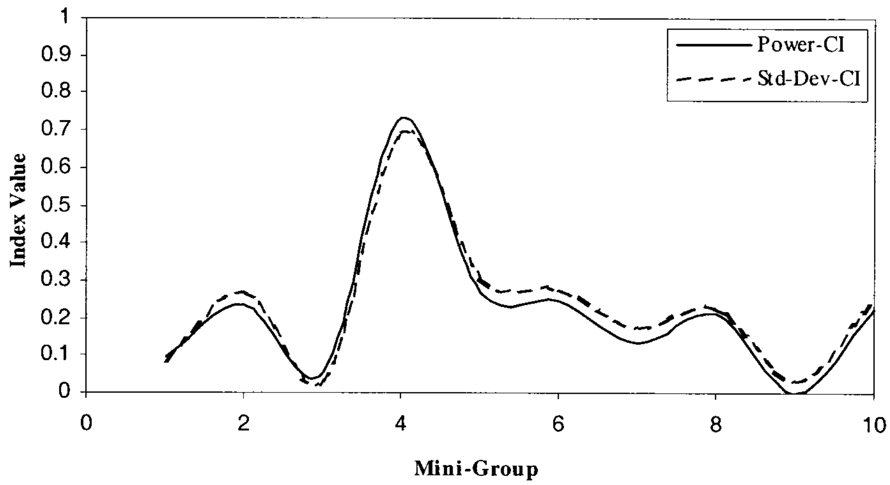



Figure B.39 Drive end bearing, fault detection – 0.021" fault diameter at outer race, 3 hp motor load and 1730 rpm.



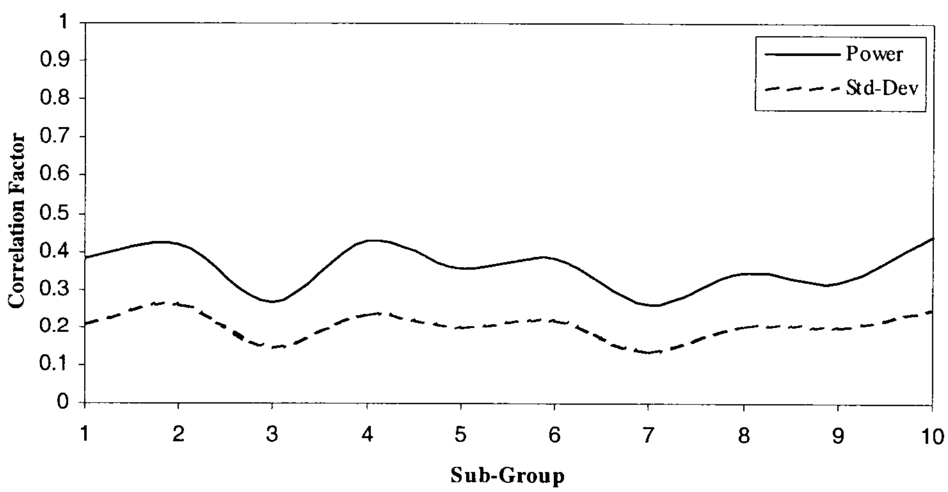




Figure B.40 Drive end bearing, fault detection – 0.028" fault diameter at inner race, 0 hp motor load and 1797 rpm.

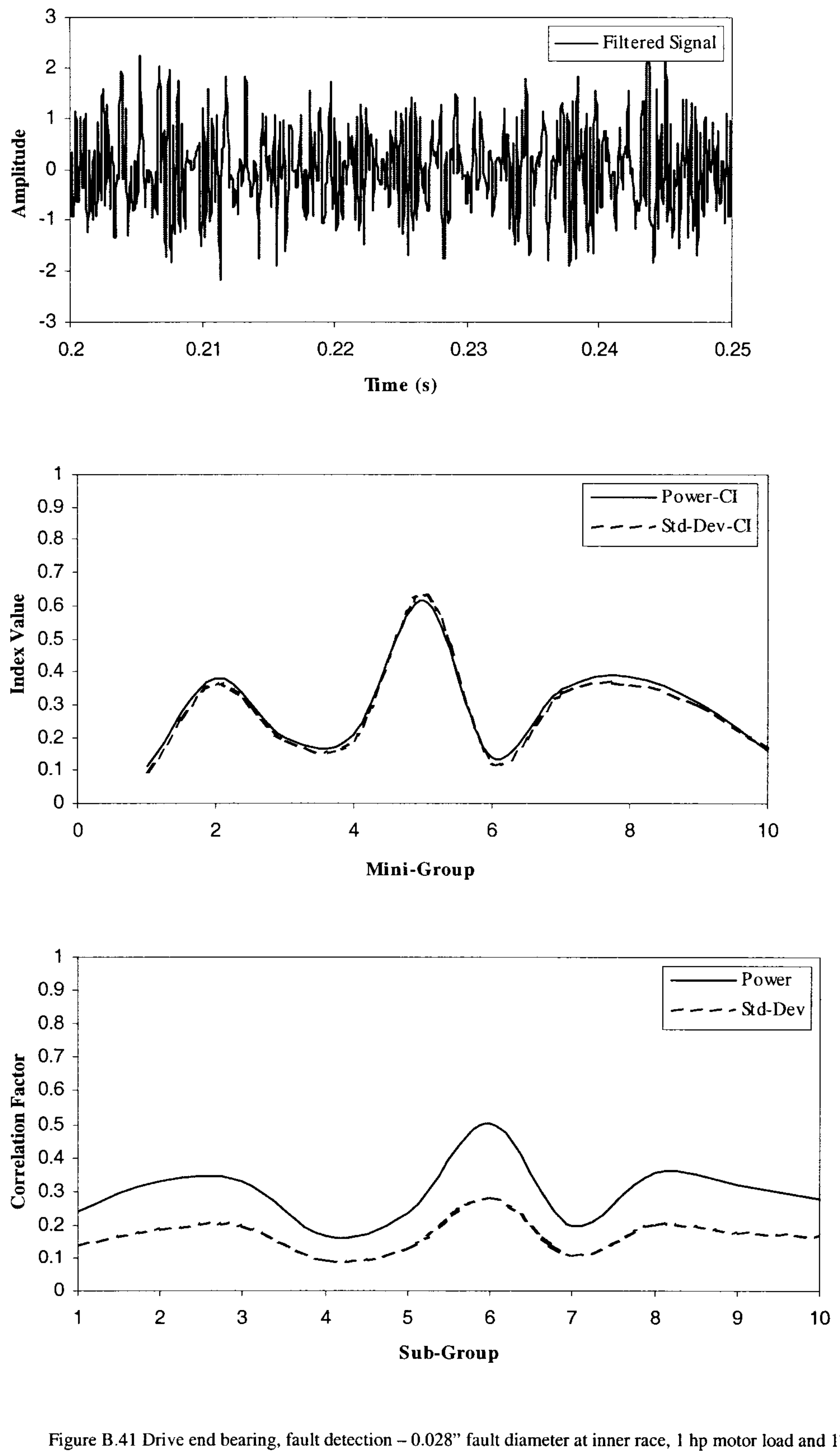



Figure B.41 Drive end bearing, fault detection – 0.028" fault diameter at inner race, 1 hp motor load and 1772 rpm.

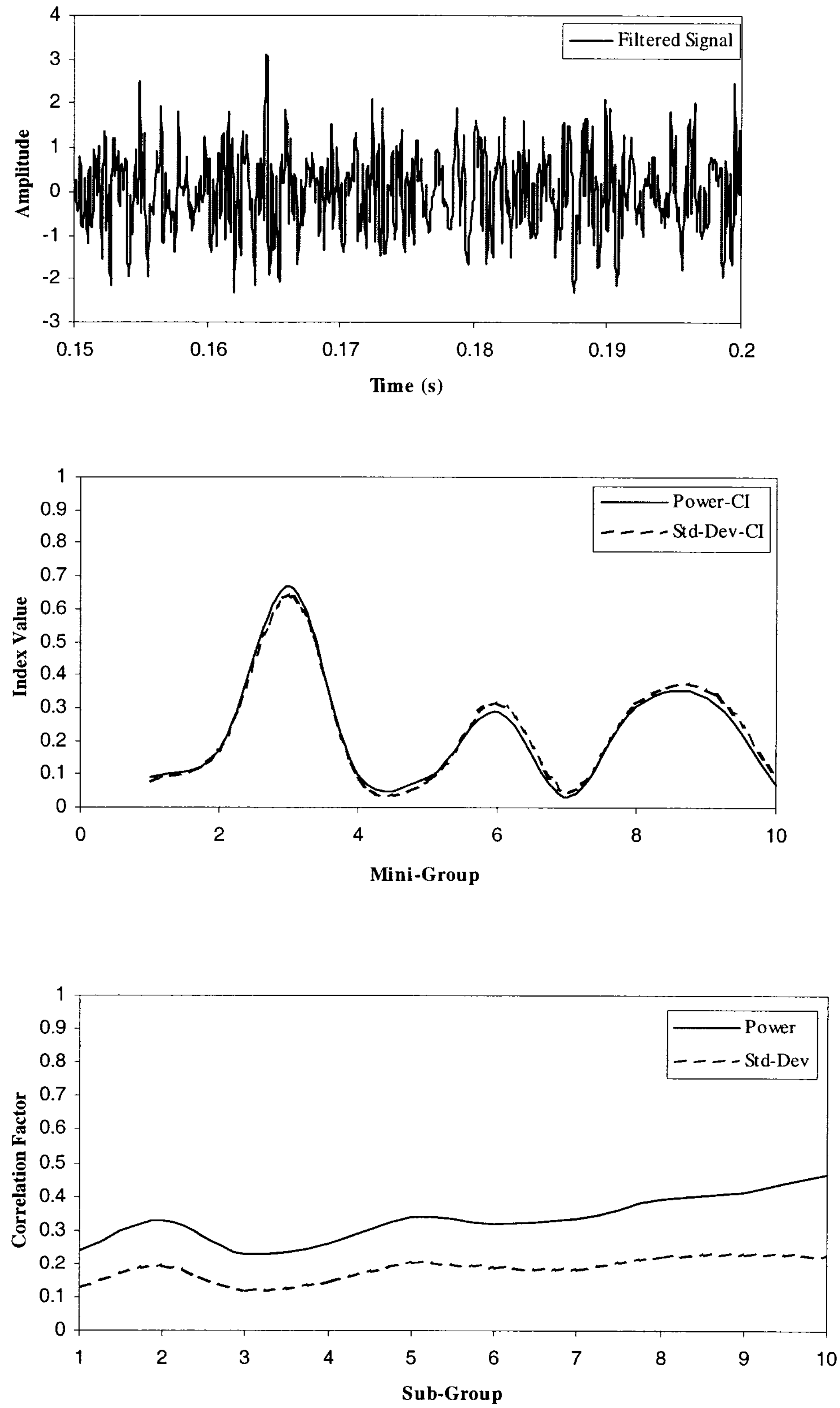



Figure B.42 Drive end bearing, fault detection – 0.028" fault diameter at inner race, 2 hp motor load and 1750 rpm.

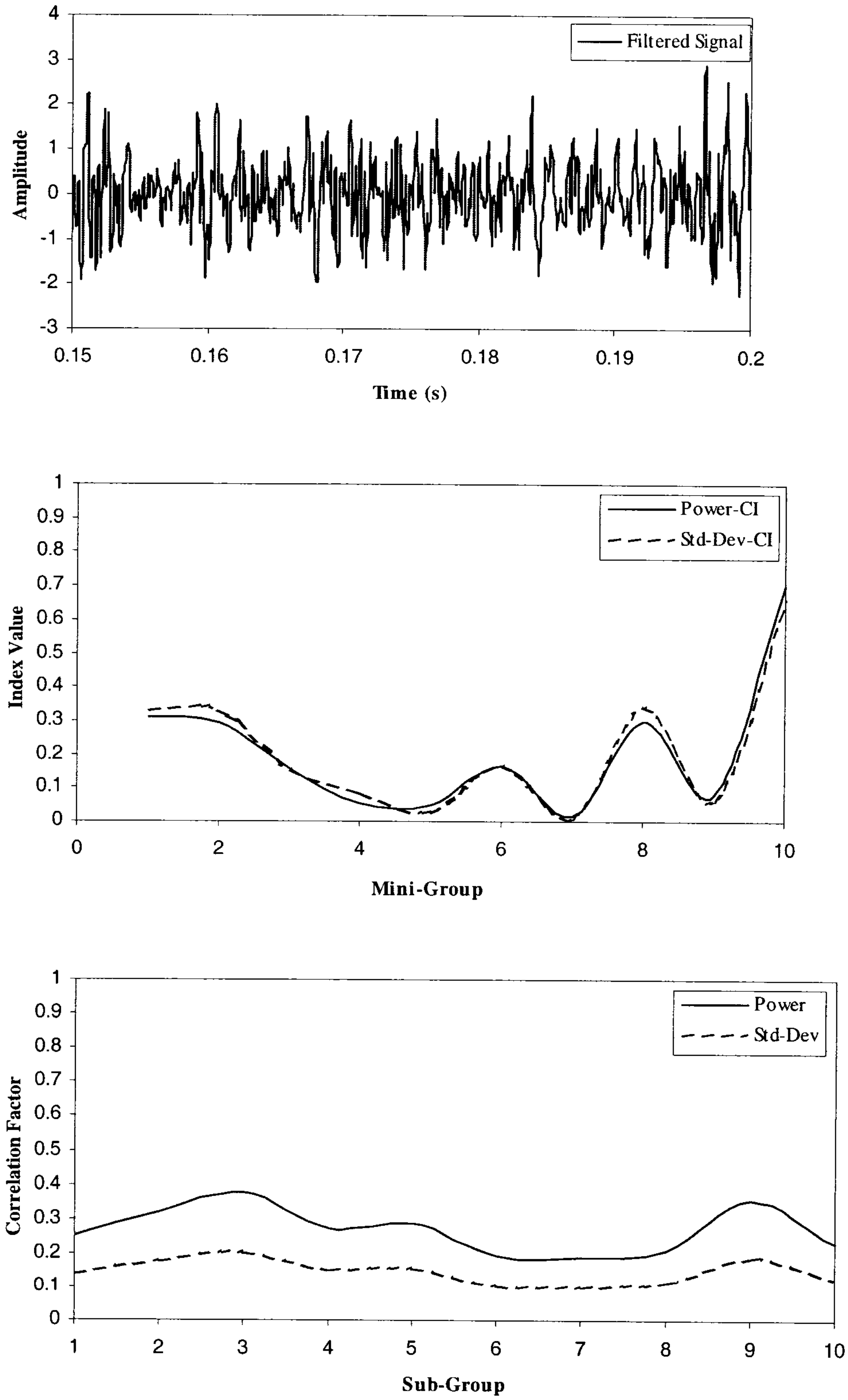



Figure B.43 Drive end bearing, fault detection – 0.028" fault diameter at inner race, 3 hp motor load and 1730 rpm.

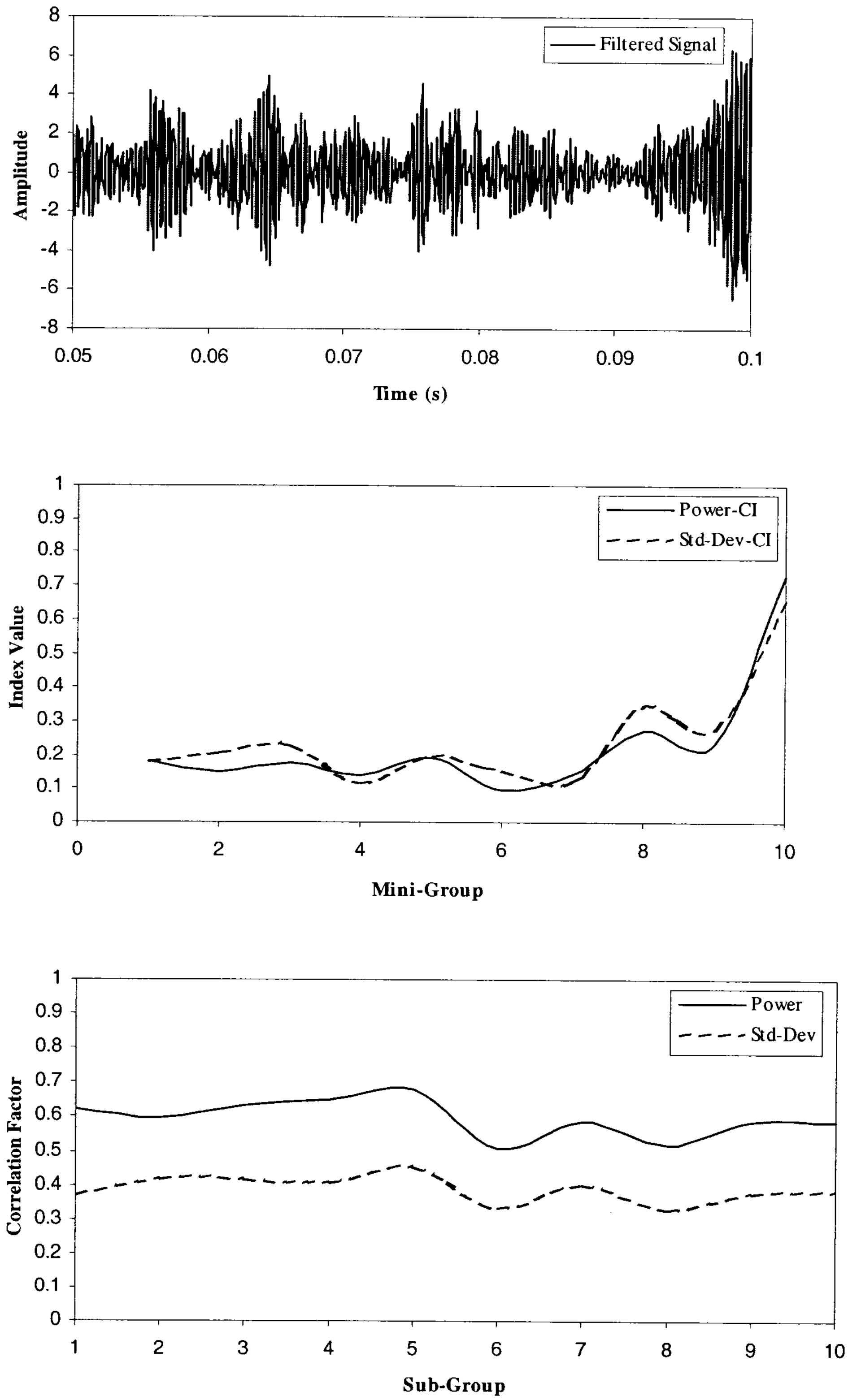



Figure B.44 Drive end bearing, fault detection – 0.028" fault diameter at ball, 0 hp motor load and 1797 rpm.

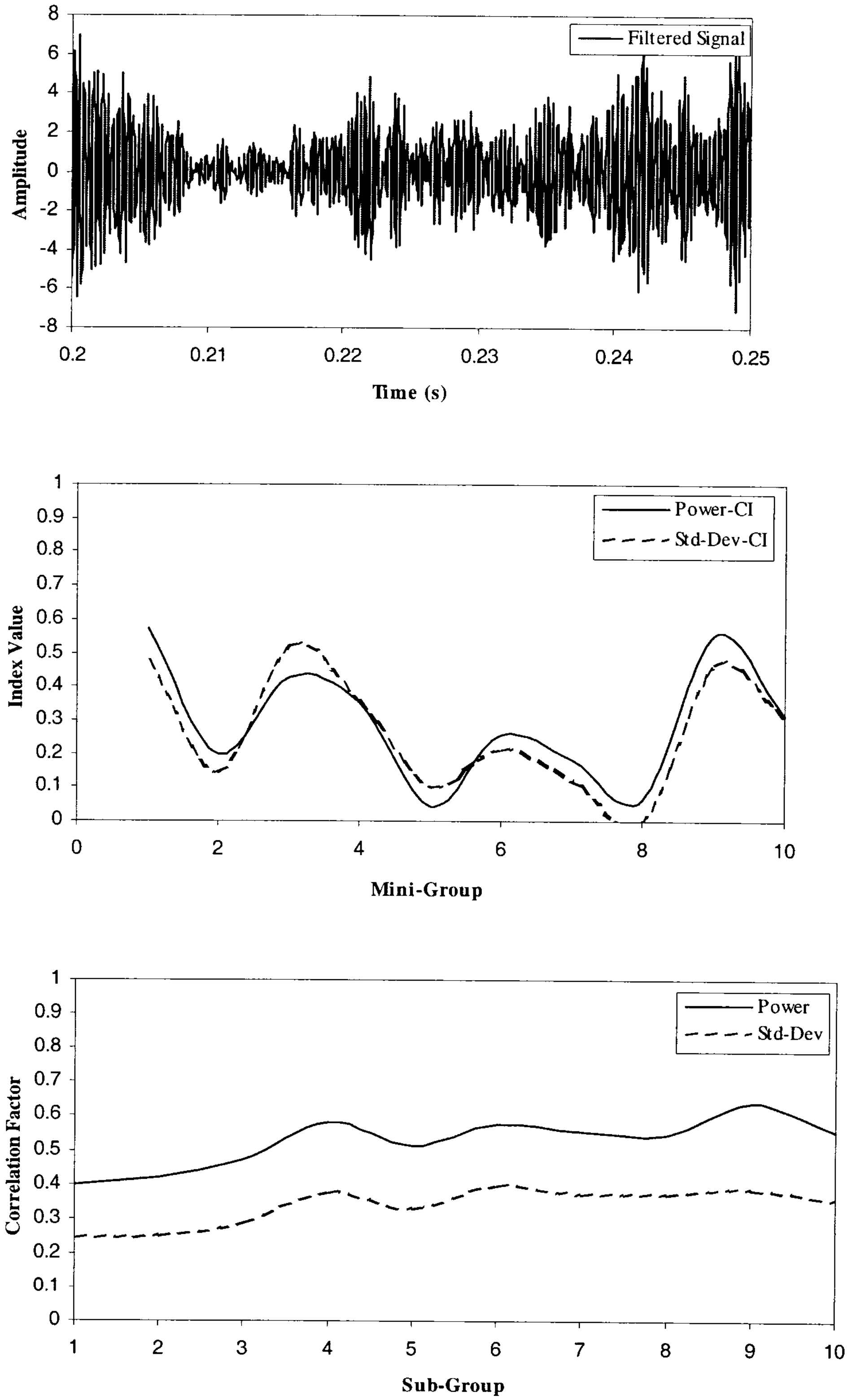



Figure B.45 Drive end bearing, fault detection – 0.028" fault diameter at ball, 1 hp motor load and 1772 rpm

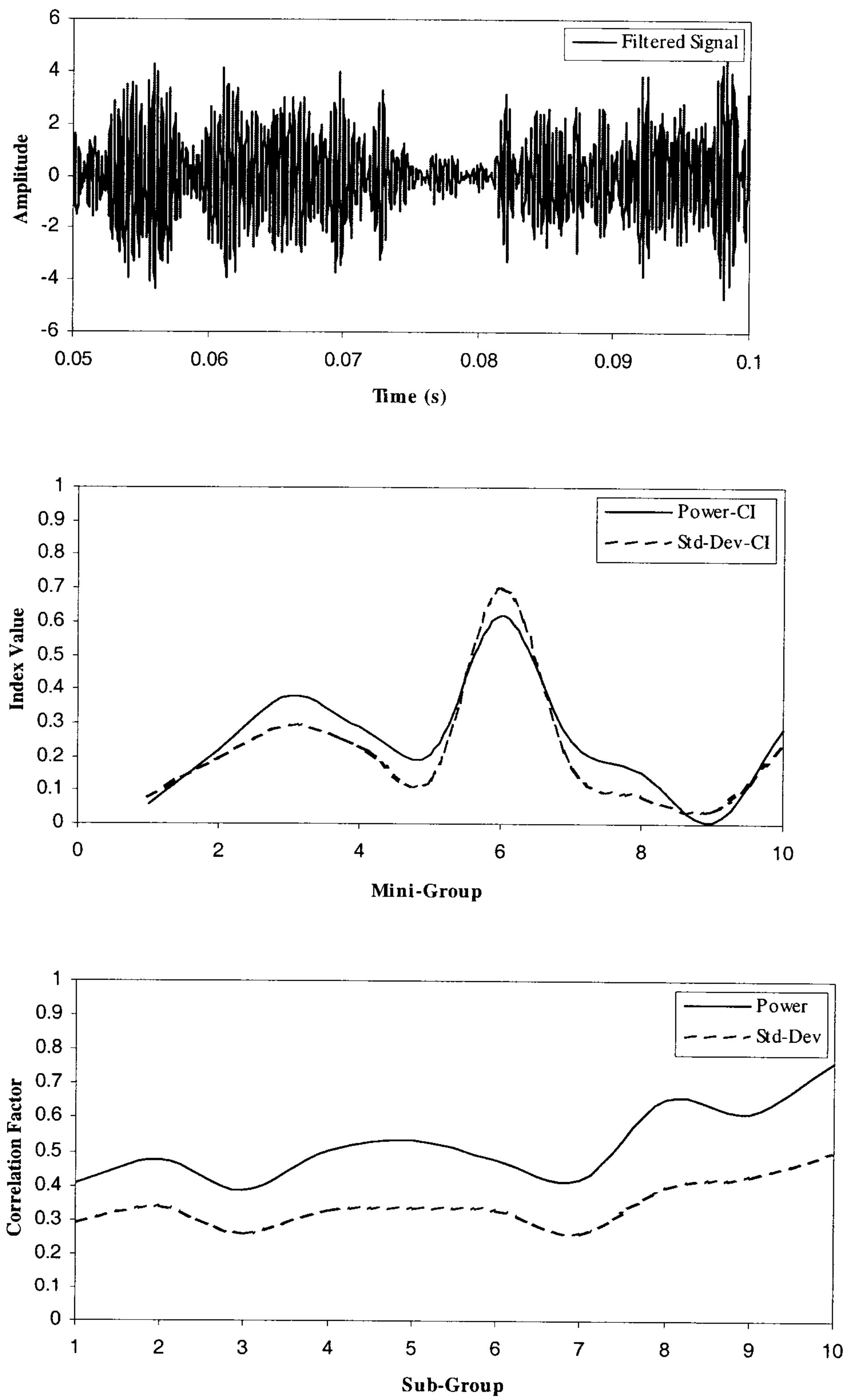
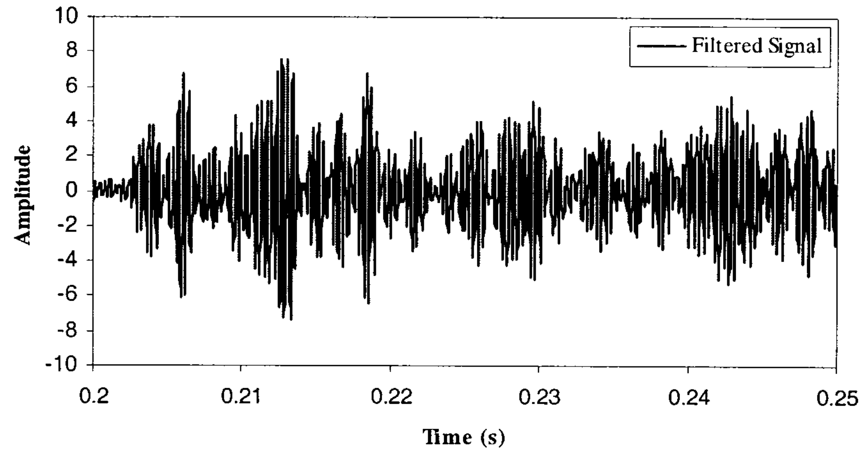
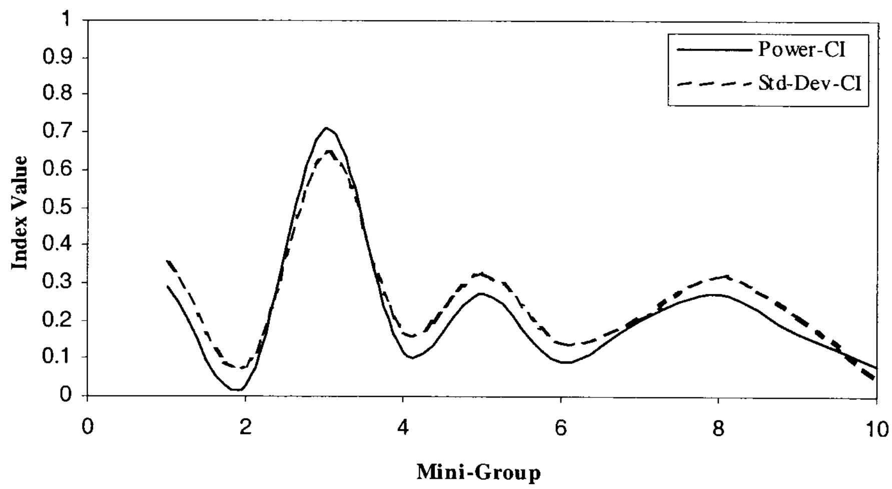



Figure B.46 Drive end bearing, fault detection – 0.028" fault diameter at ball, 2 hp motor load and 1750 rpm.



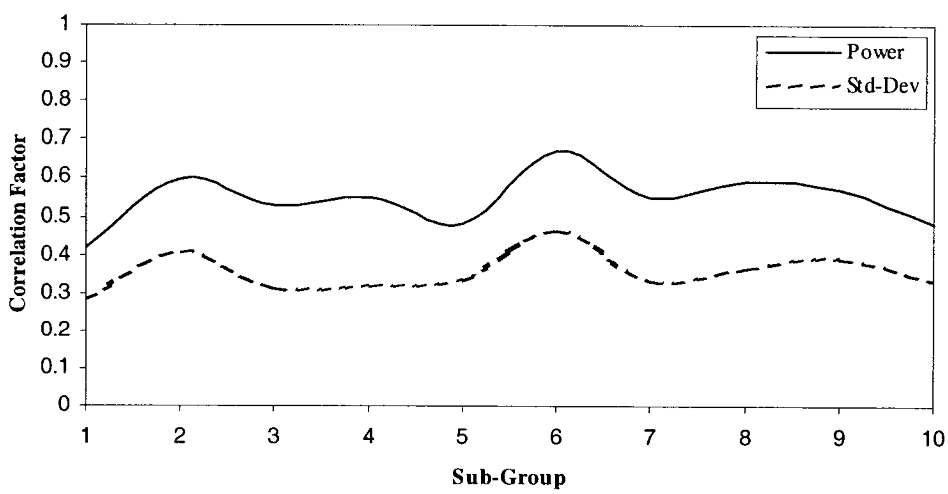




Figure B.47 Drive end bearing, fault detection – 0.028" fault diameter at ball, 3 hp motor load and 1730 rpm.

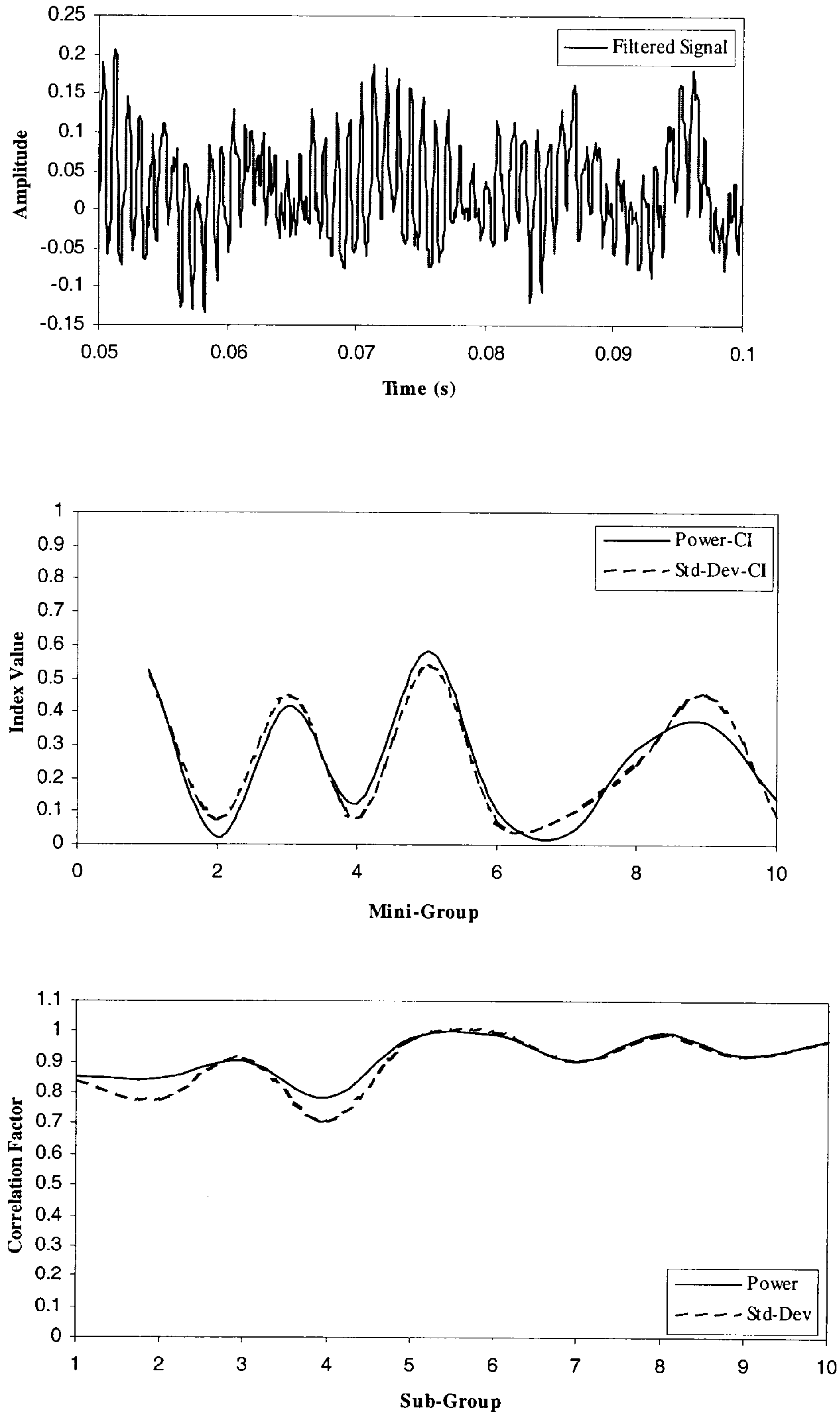



Figure B.48 Fan end bearing, normal data presentation at 0 hp and 1797 rpm.

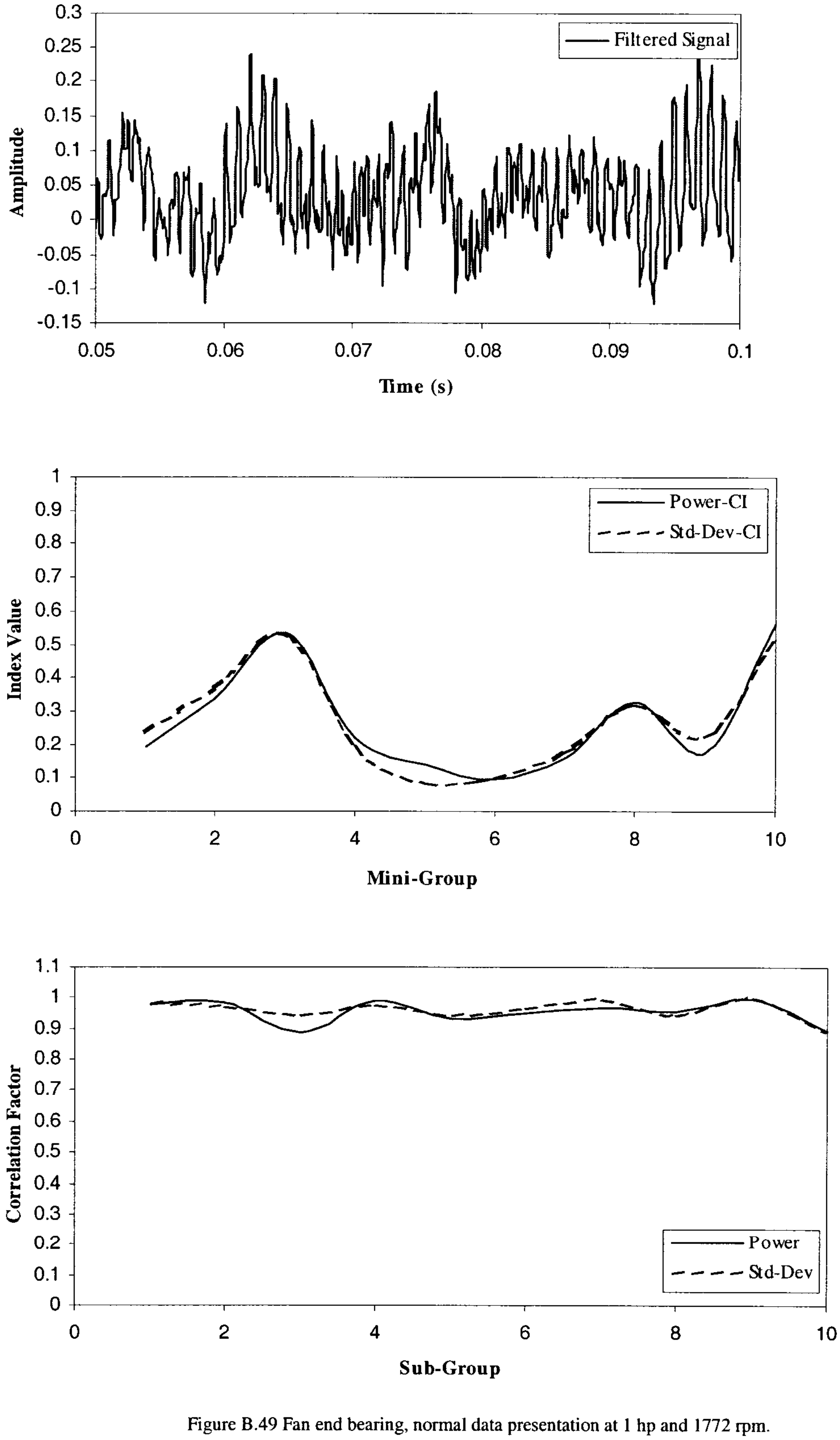



Figure B.49 Fan end bearing, normal data presentation at 1 hp and 1772 rpm.

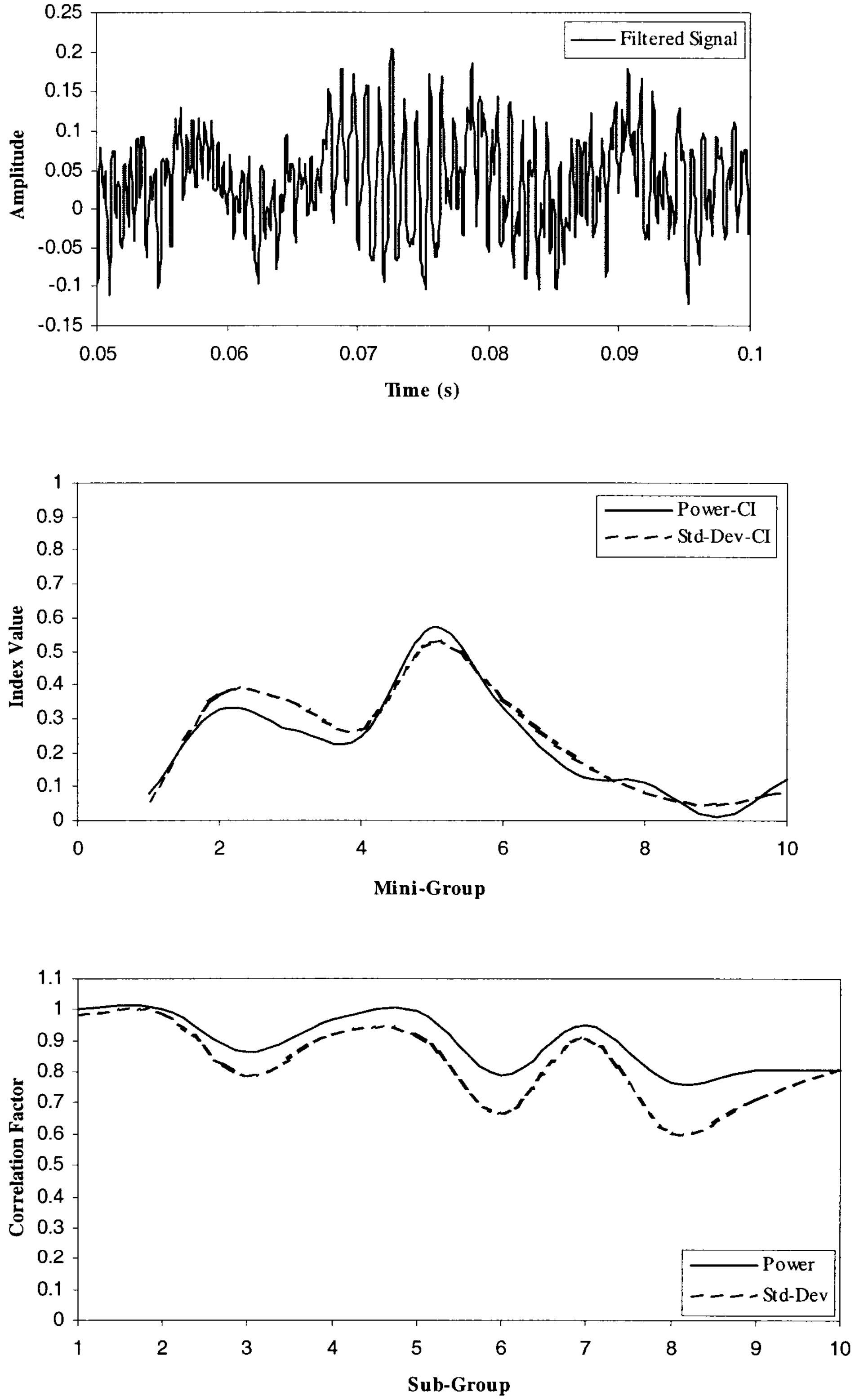



Figure B.50 Fan end bearing, normal data presentation at 2 hp and 1750 rpm.

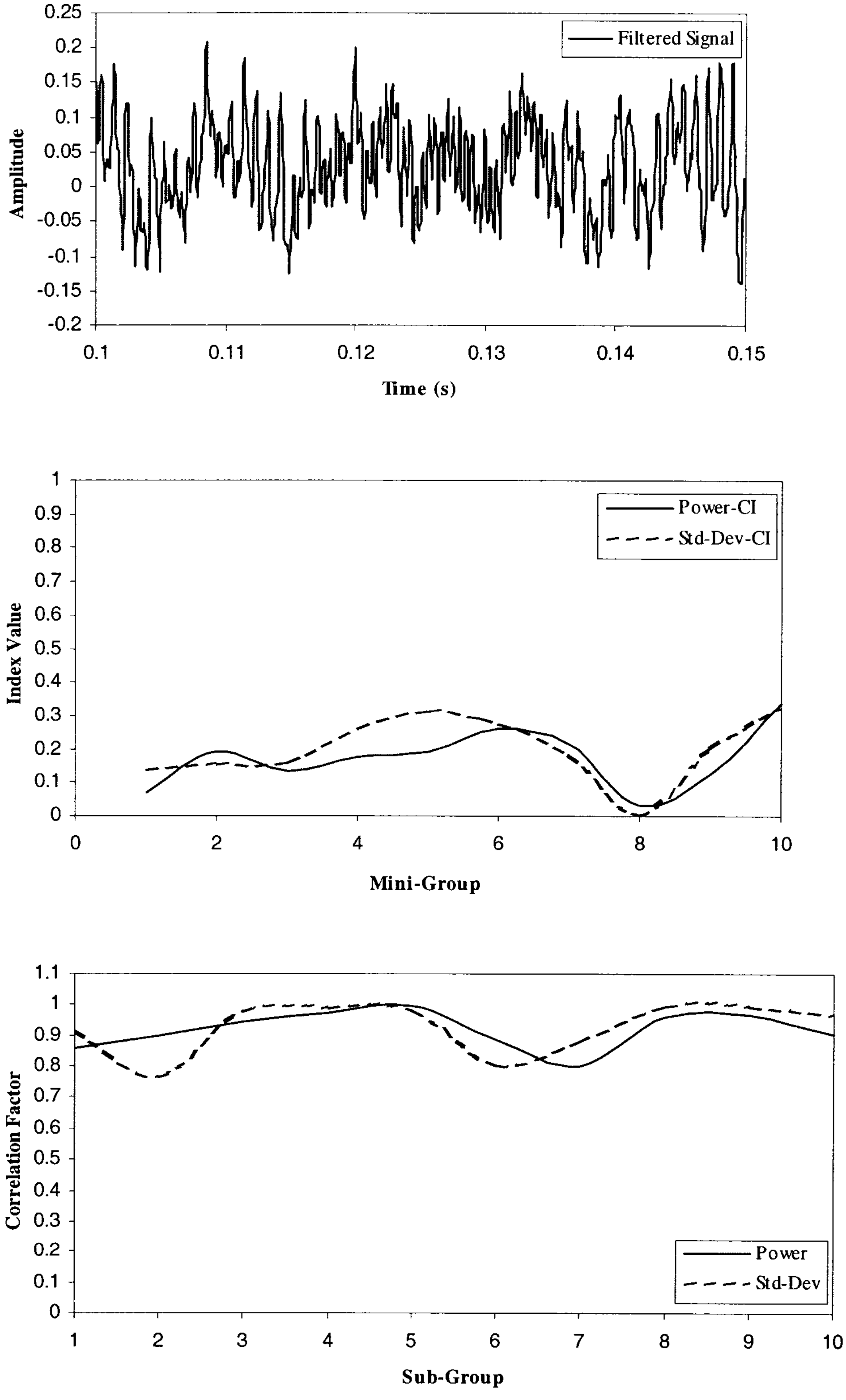



Figure B.51 Fan end bearing, normal data presentation at 3 hp and 1730 rpm

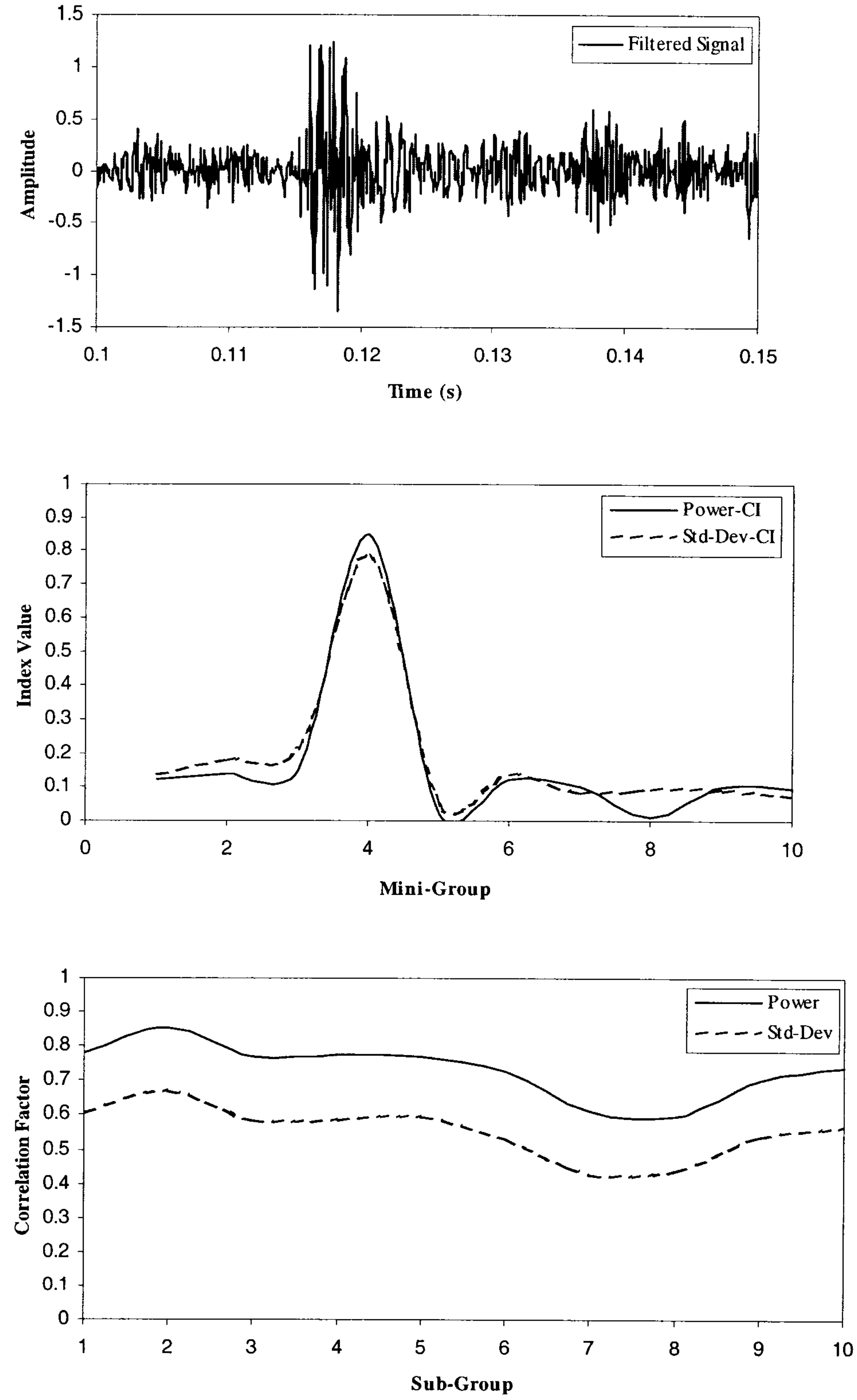
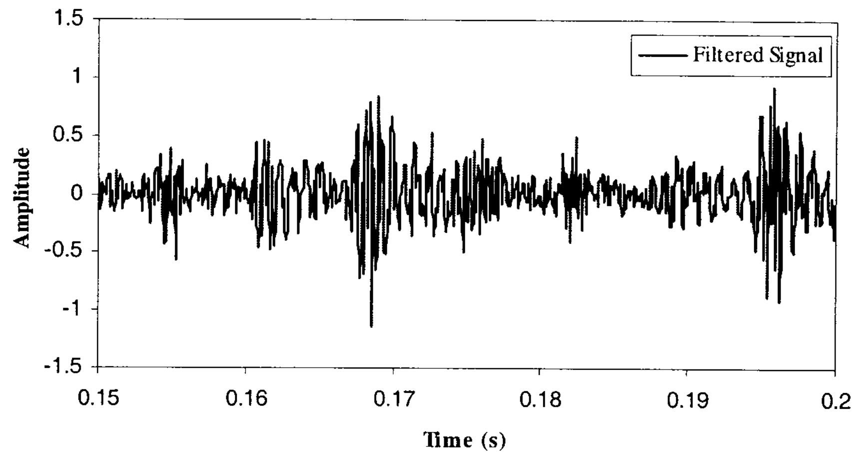
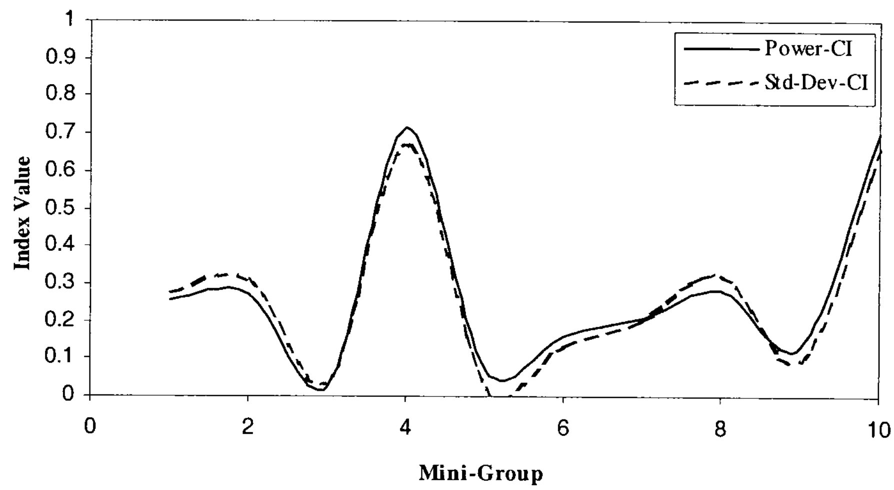



Figure B.52 Fan end bearing, fault detection – 0.007" fault diameter at inner race, 0 hp and 1797 rpm.



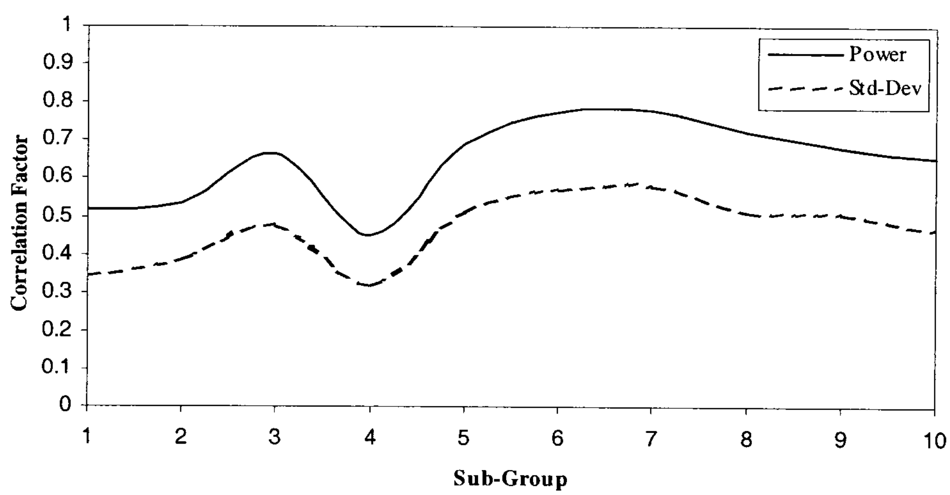




Figure B.53 Fan end bearing, fault detection – 0.007" fault diameter at inner race, 1 hp and 1772 rpm.

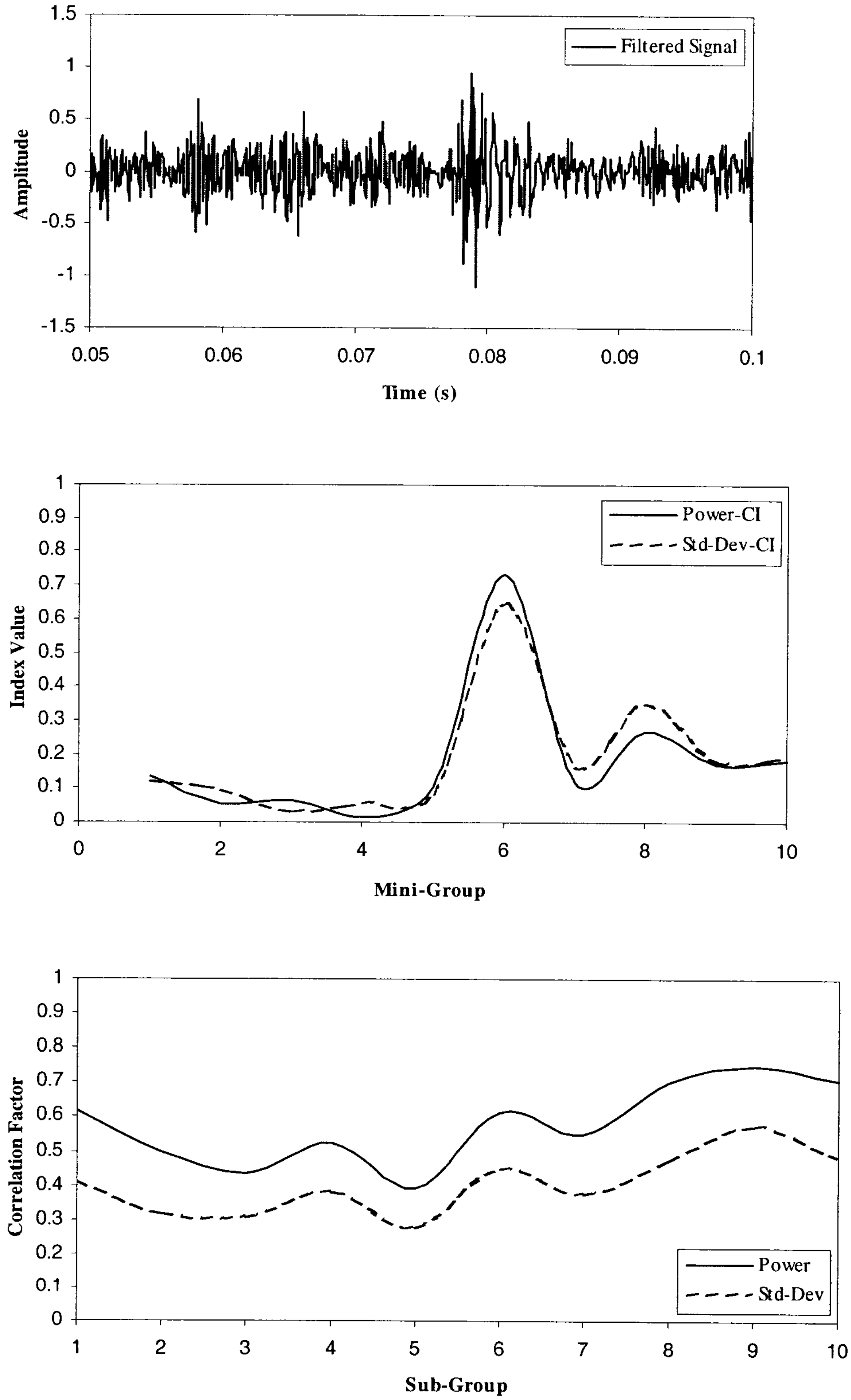
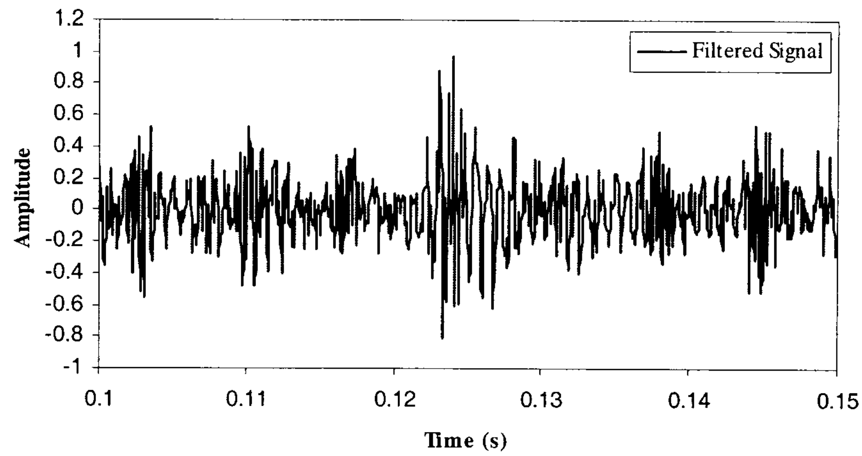
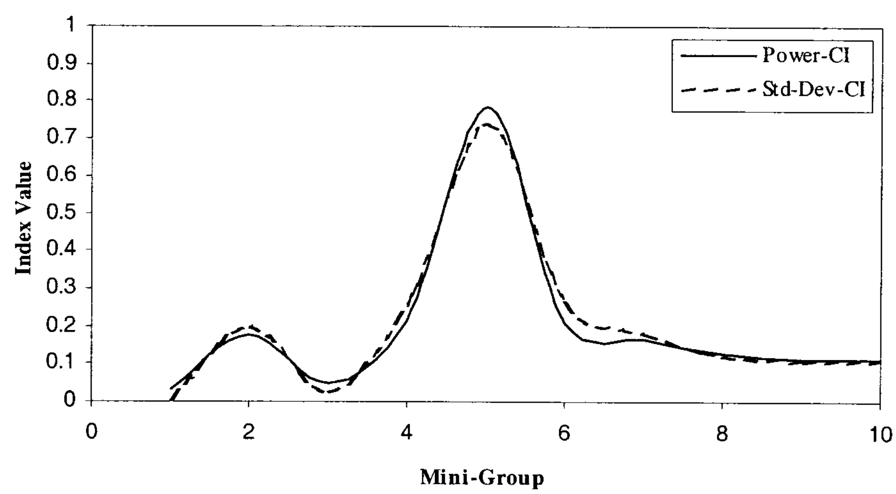



Figure B.54 Fan end bearing, fault detection – 0.007" fault diameter at inner race, 2 hp and 1750 rpm.



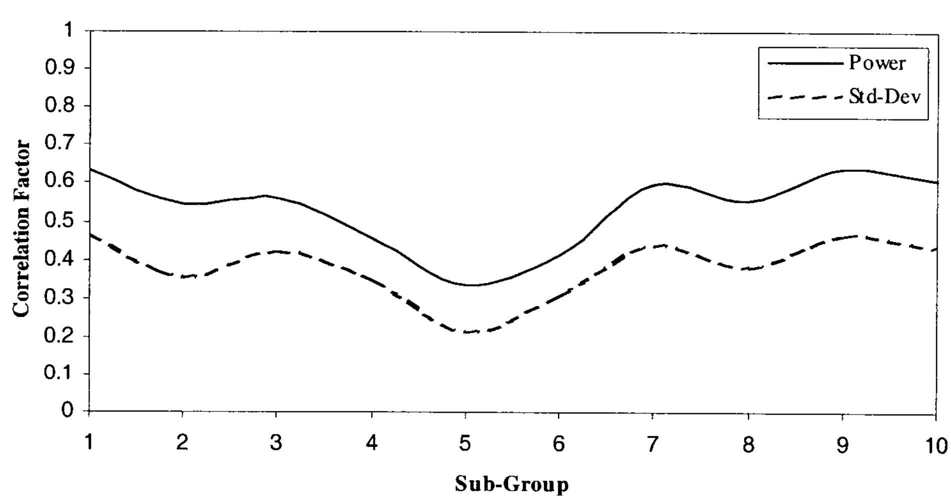




Figure B.55 Fan end bearing, fault detection – 0.007" fault diameter at inner race, 3 hp and 1730 rpm.

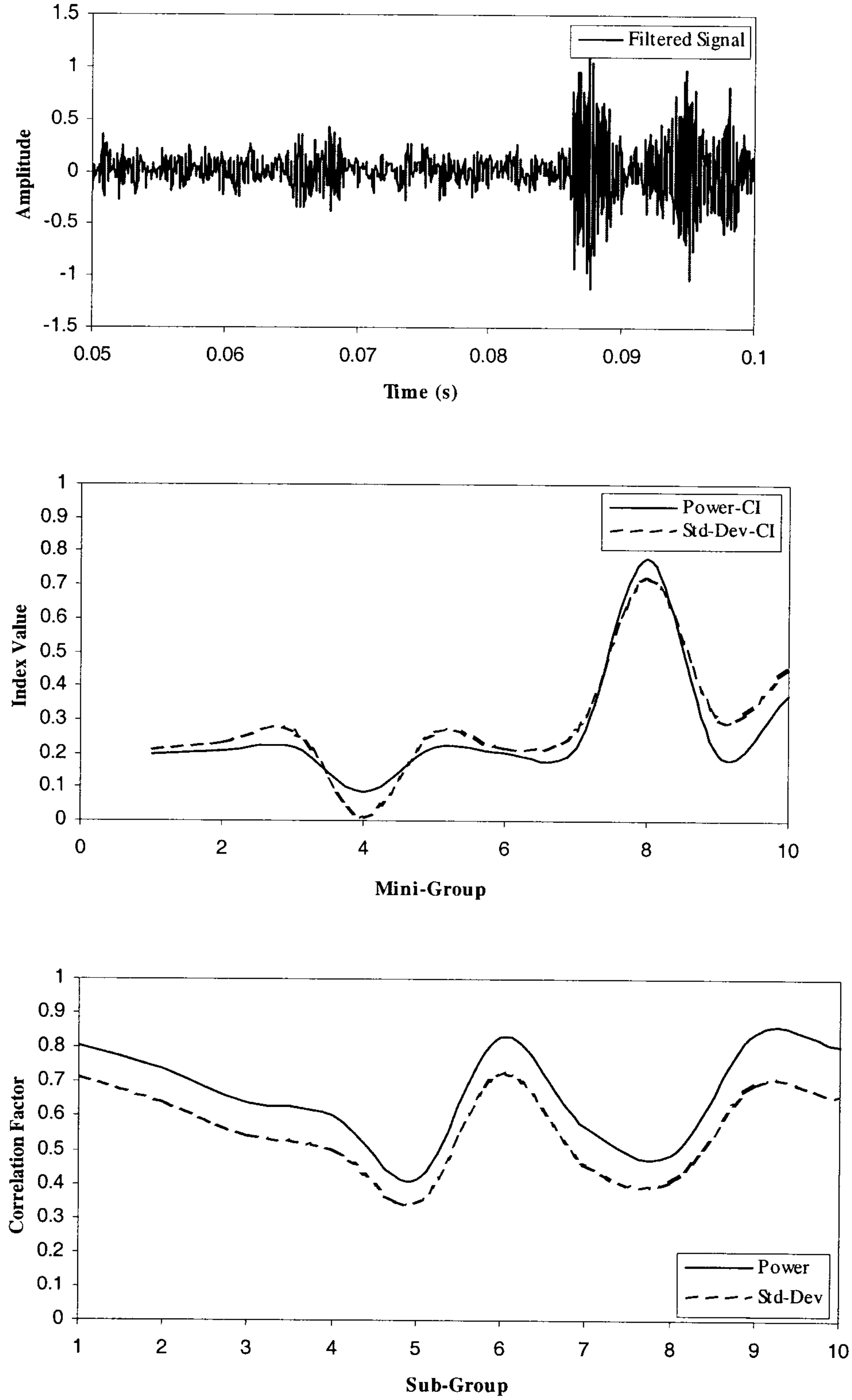



Figure B.56 Fan end bearing, fault detection – 0.007" fault diameter at ball, 0 hp and 1797 rpm.

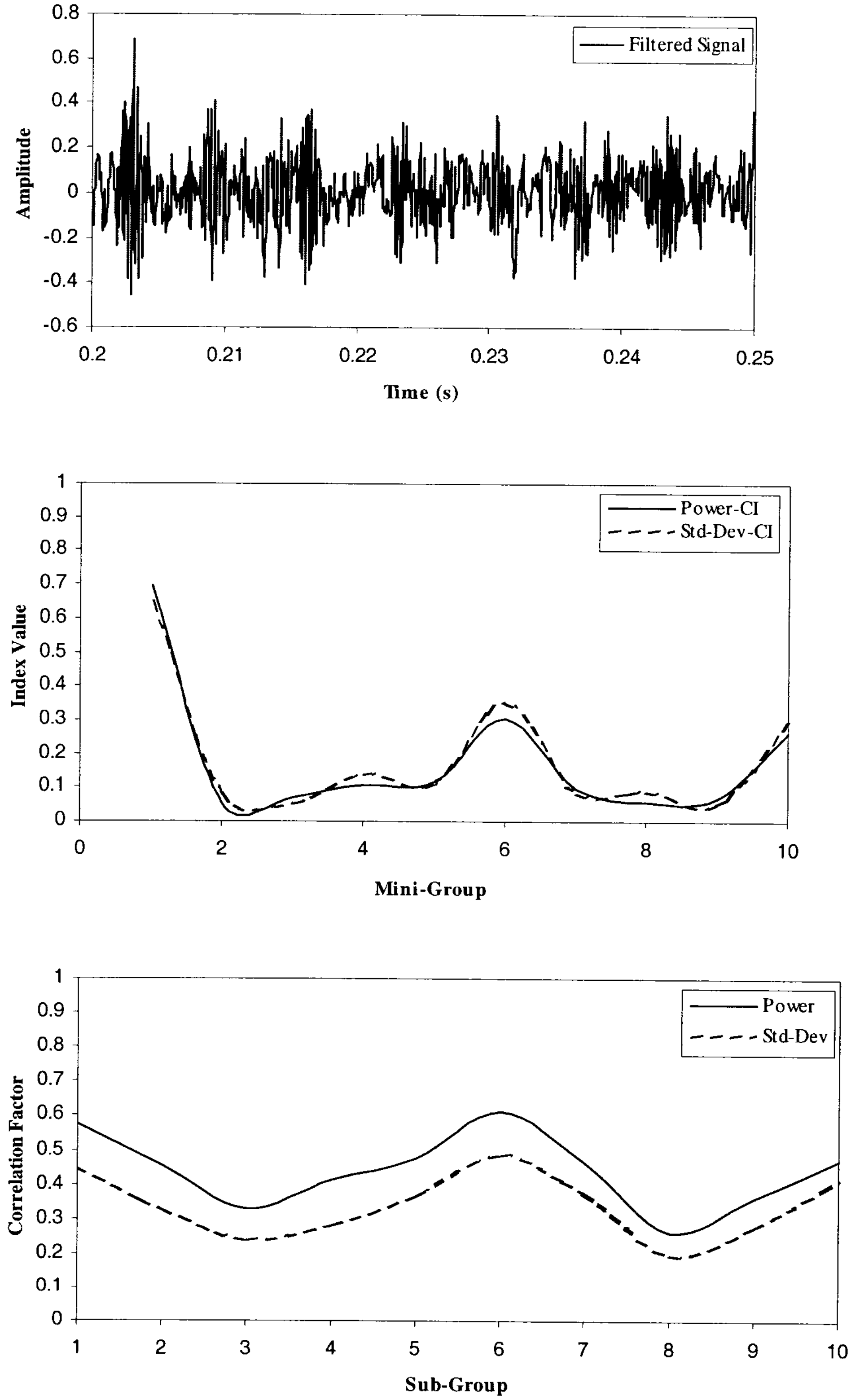



Figure B.57 Fan end bearing, fault detection – 0.007" fault diameter at ball, 1 hp and 1772 rpm.

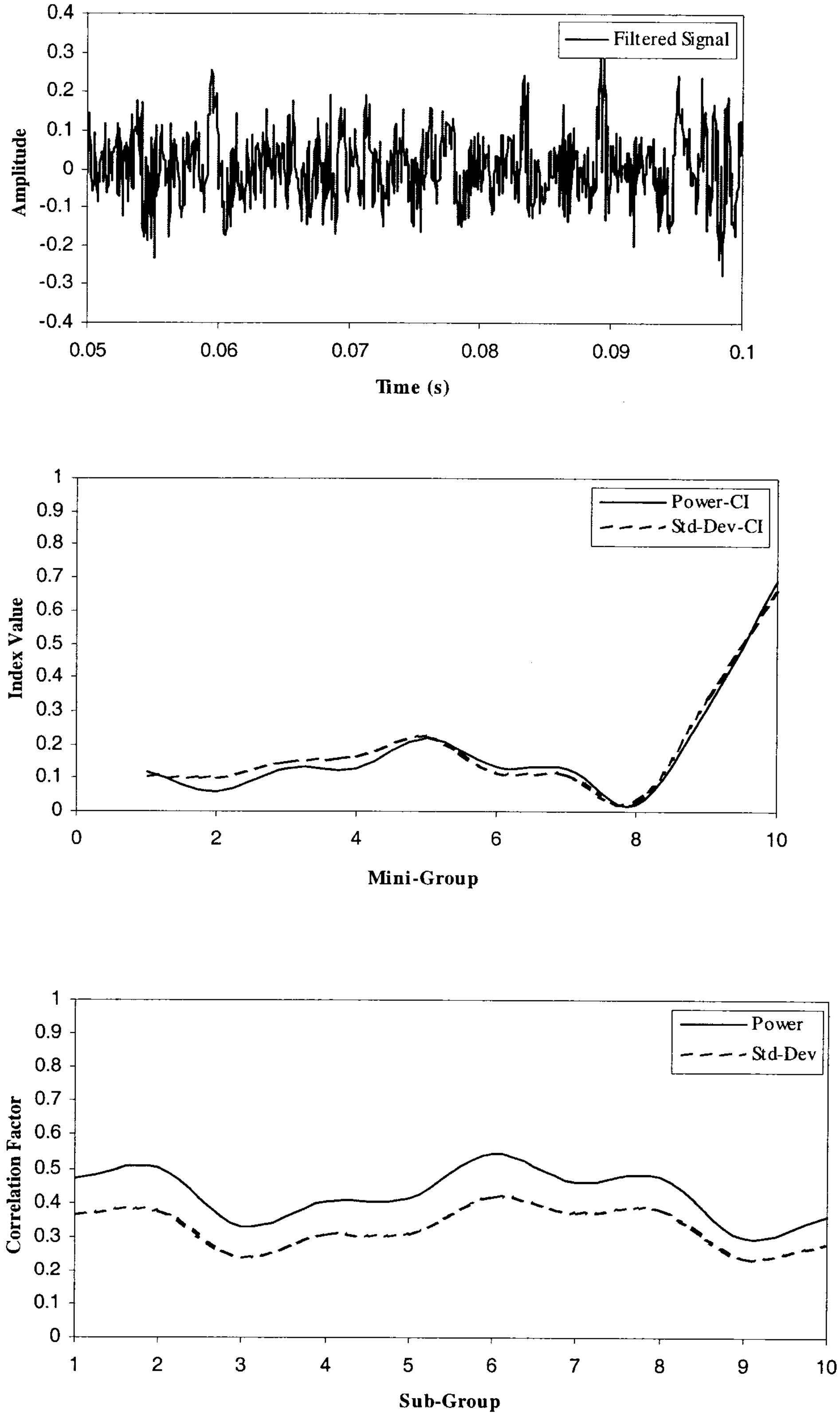
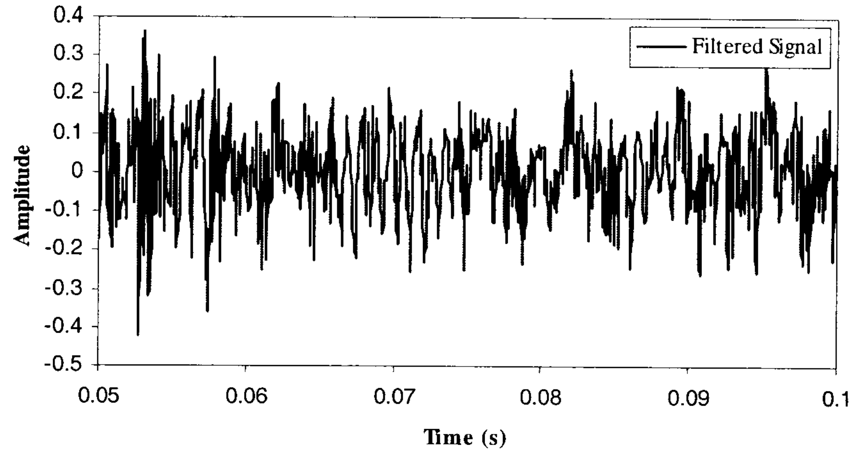
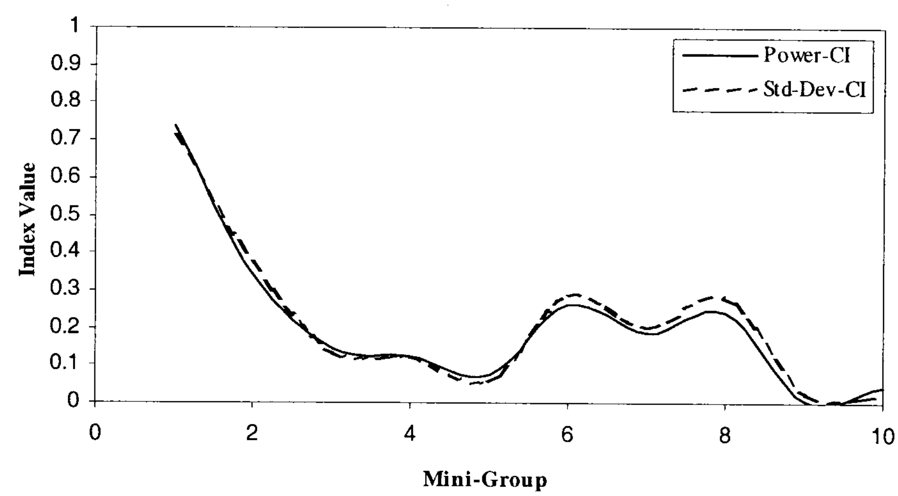



Figure B.58 Fan end bearing, fault detection – 0.007" fault diameter at ball, 2 hp and 1750 rpm.



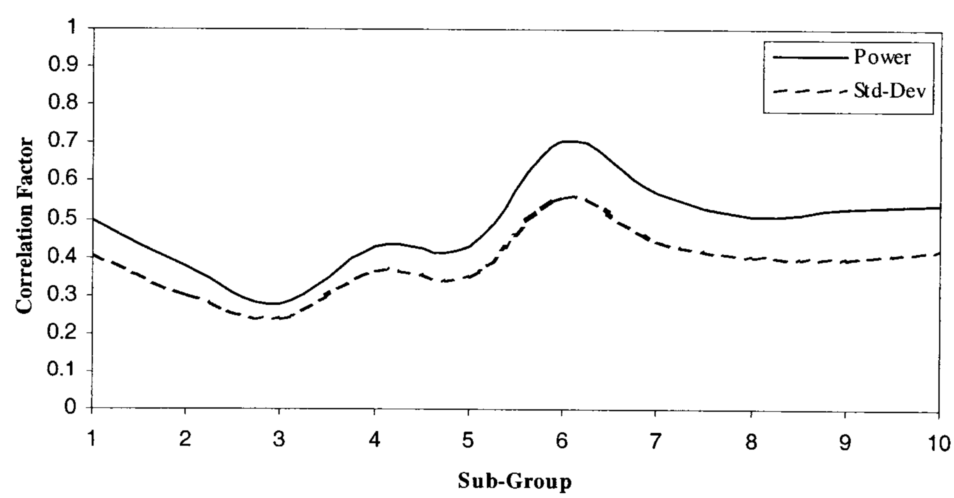




Figure B.59 Fan end bearing, fault detection – 0.007" fault diameter at ball, 3 hp and 1730 rpm.

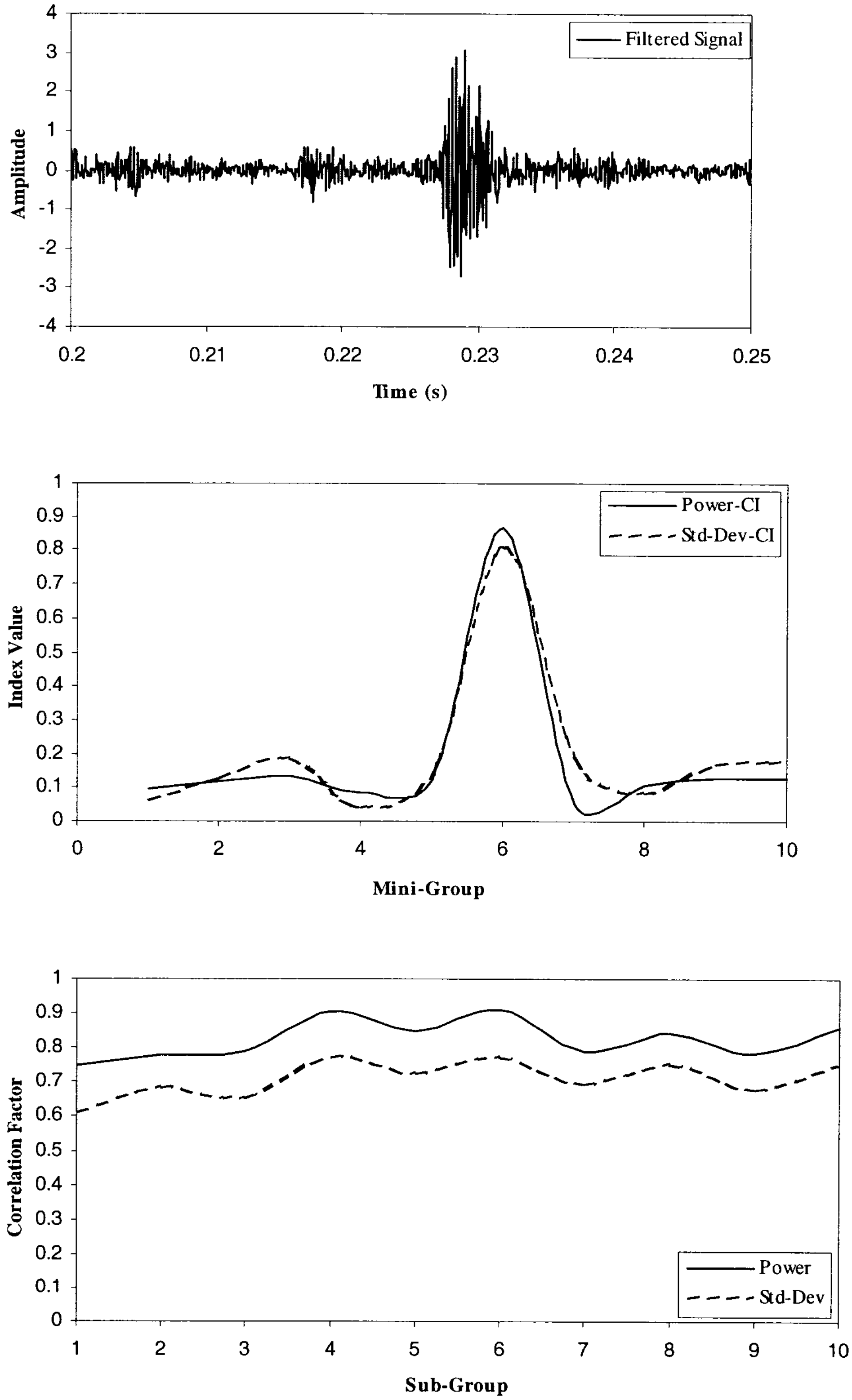



Figure B.60 Fan end bearing, fault detection – 0.007" fault diameter at outer race, 0 hp and 1797 rpm.

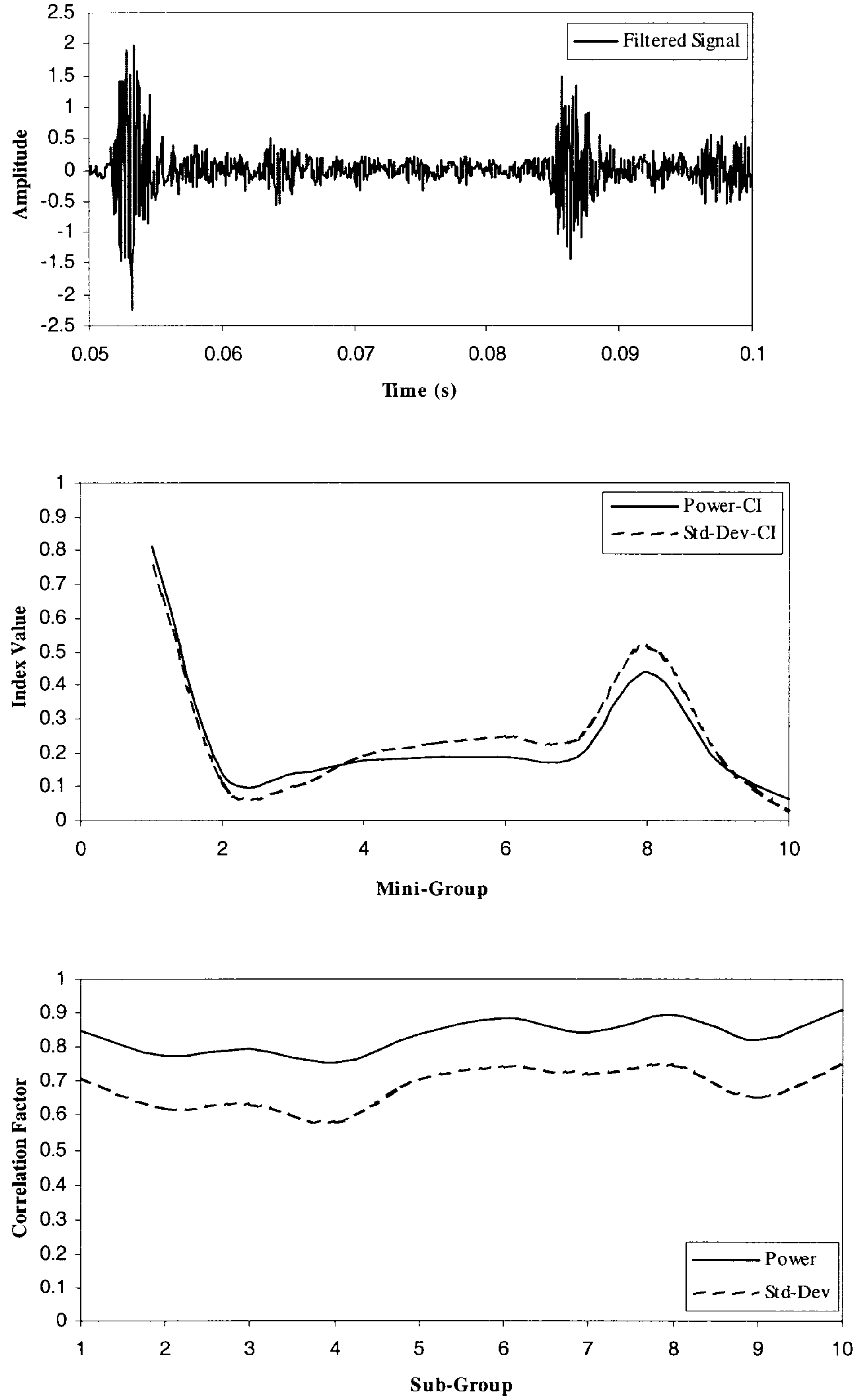
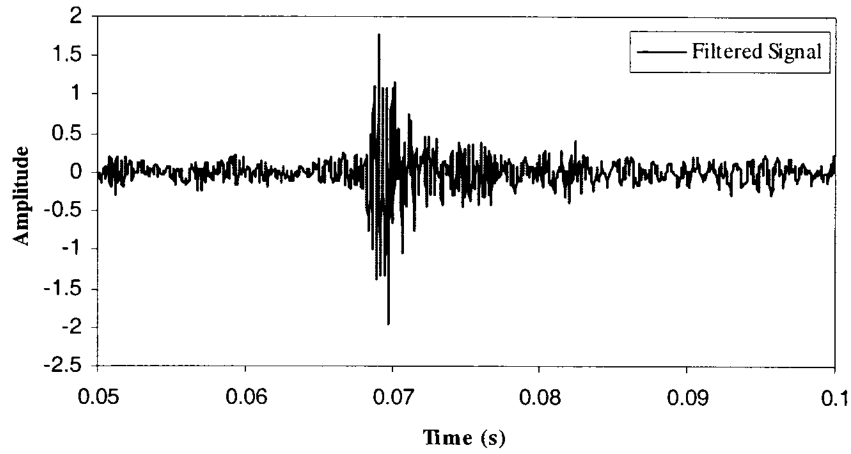
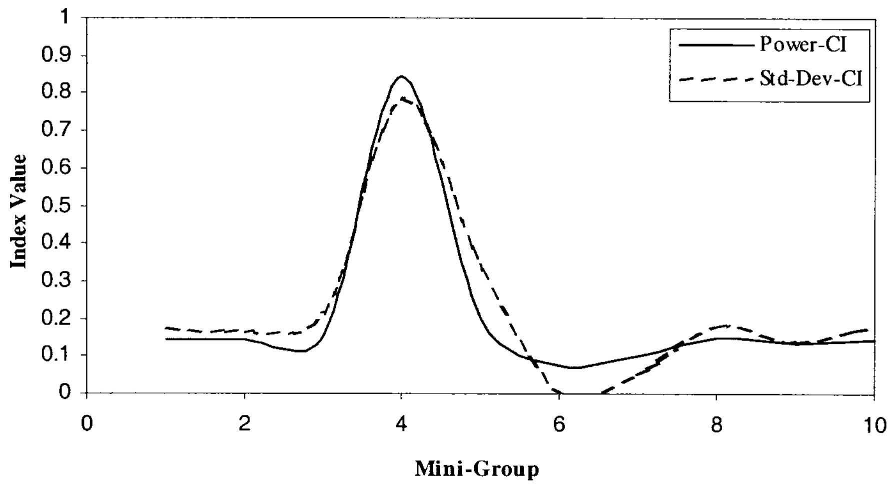



Figure B.61 Fan end bearing, fault detection – 0.007" fault diameter at outer race, 1 hp and 1772 rpm.



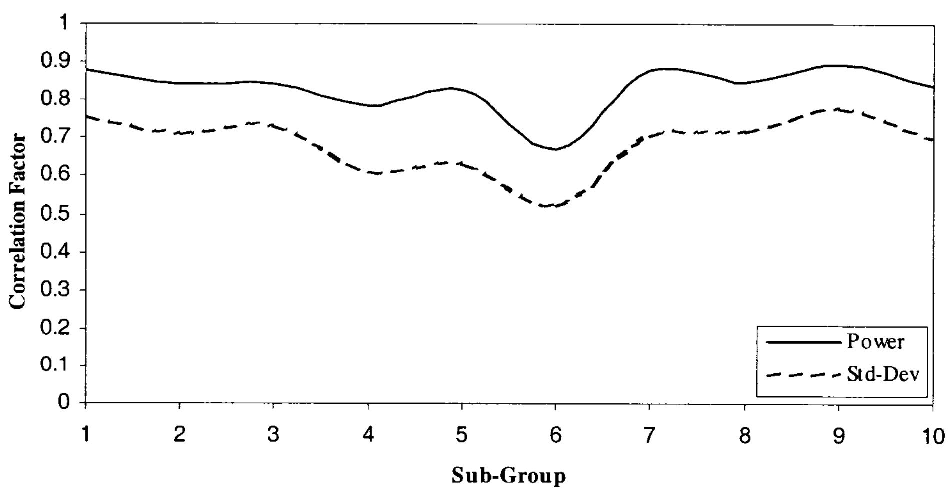




Figure B.62 Fan end bearing, fault detection – 0.007" fault diameter at outer, 2 hp and 1750 rpm.

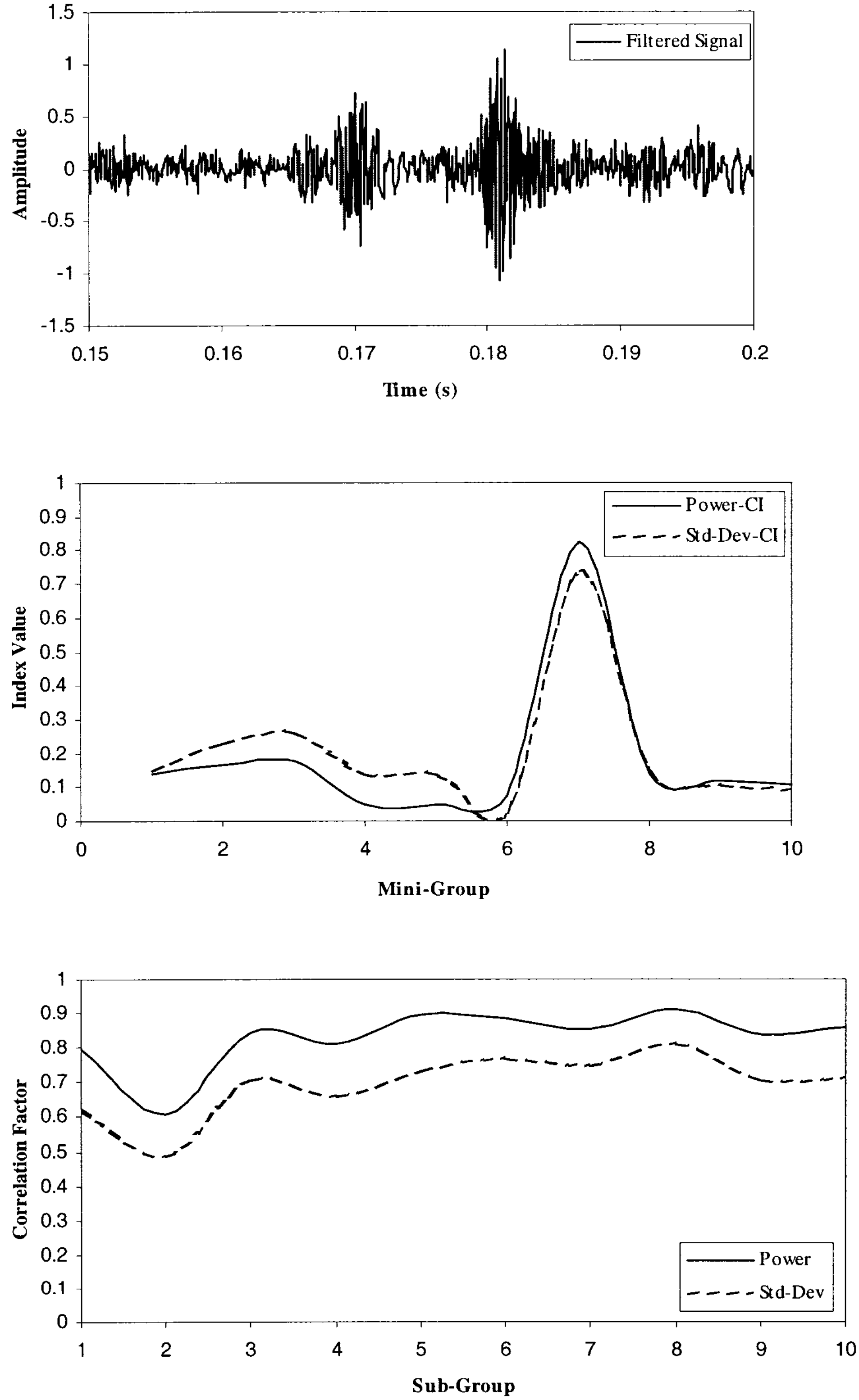
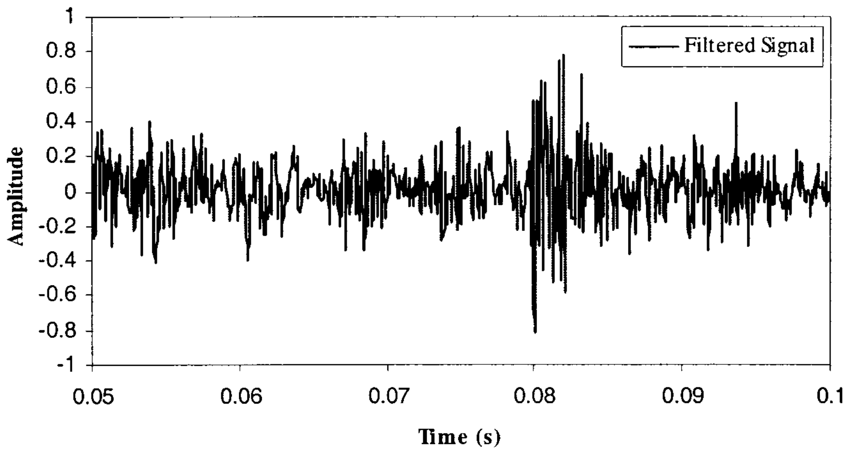
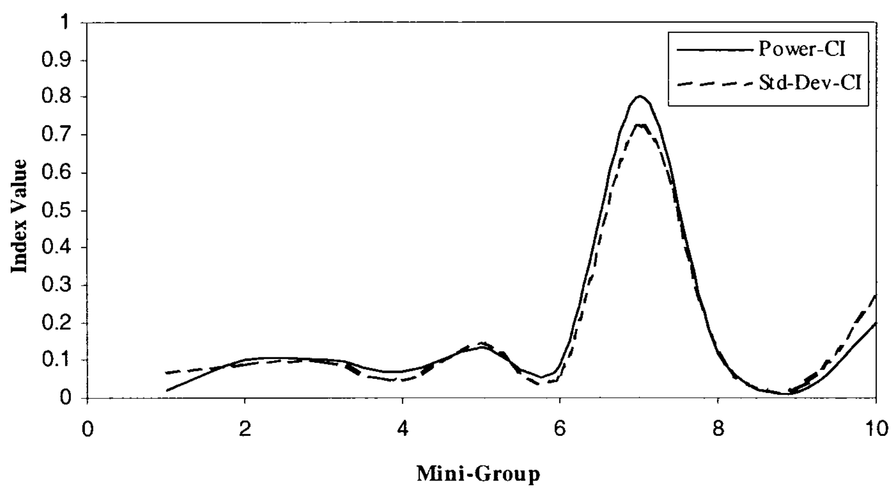



Figure B.63 Fan end bearing, fault detection – 0.007" fault diameter at outer race, 3 hp and 1730 rpm.



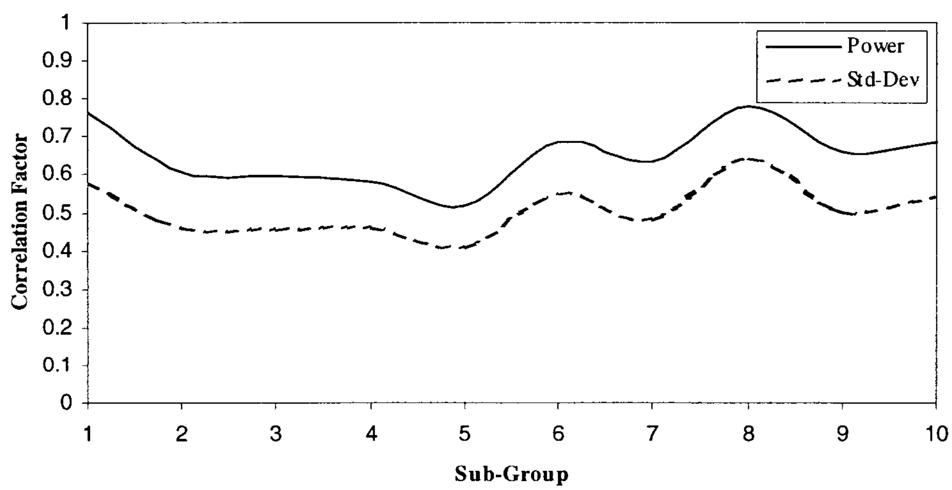




Figure B.64 Fan end bearing, fault detection – 0.014" fault diameter at inner race, 0 hp and 1797 rpm

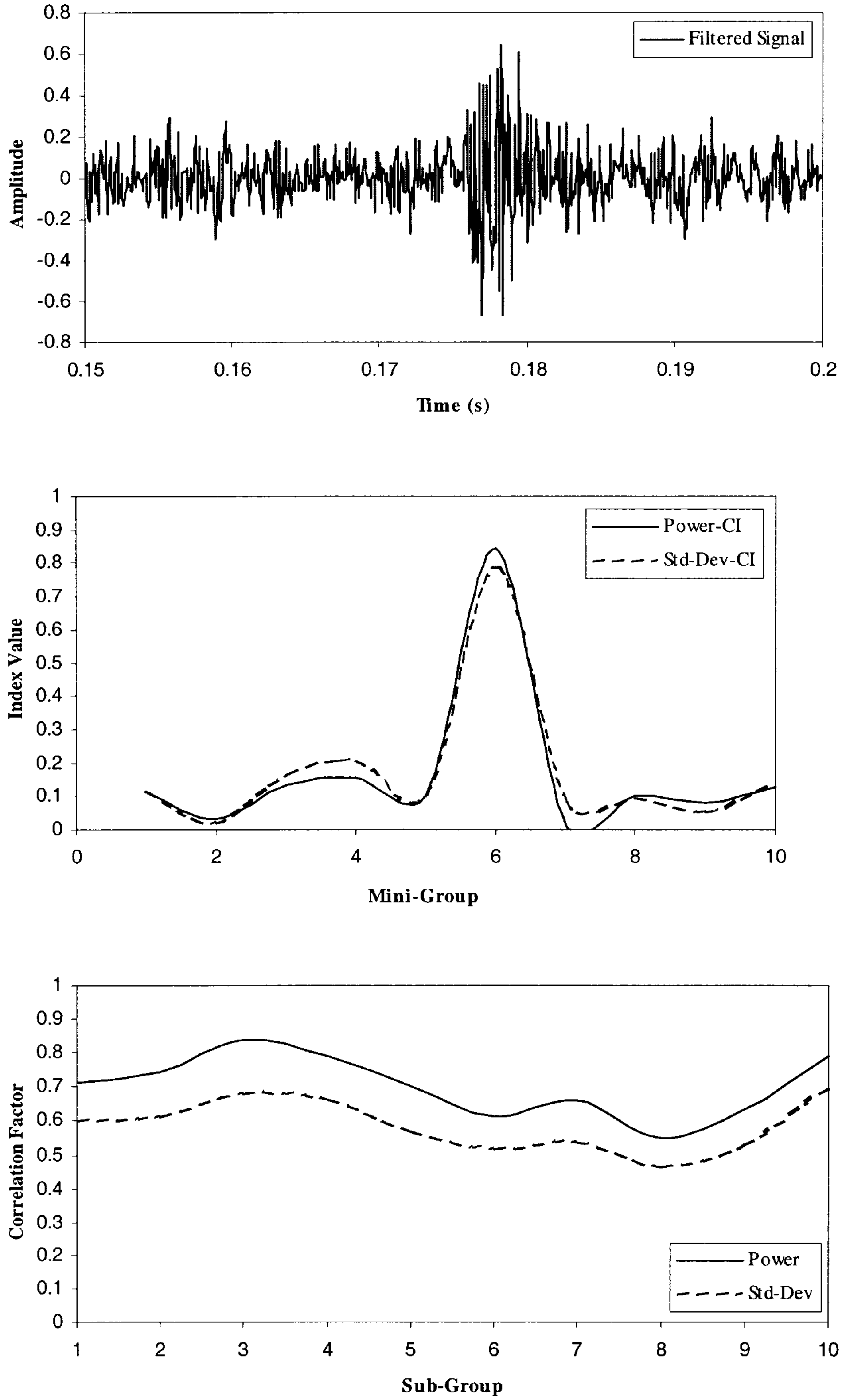
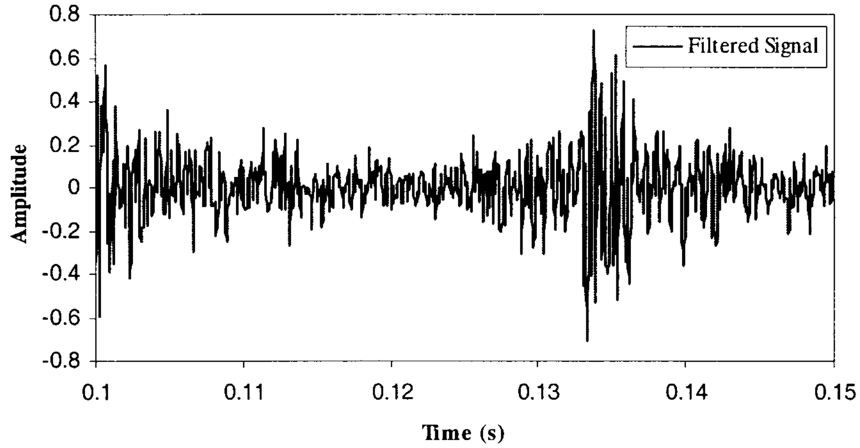
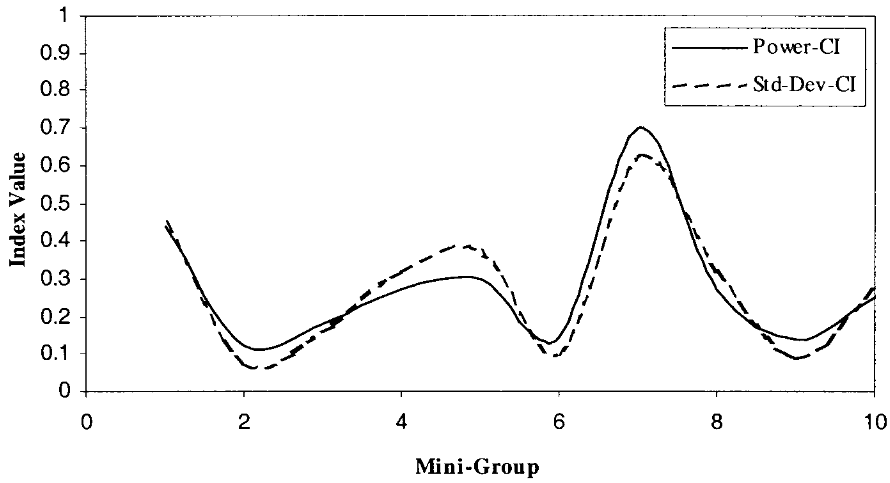



Figure B.65 Fan end bearing, fault detection – 0.014" fault diameter at inner race, 1 hp and 1772 rpm



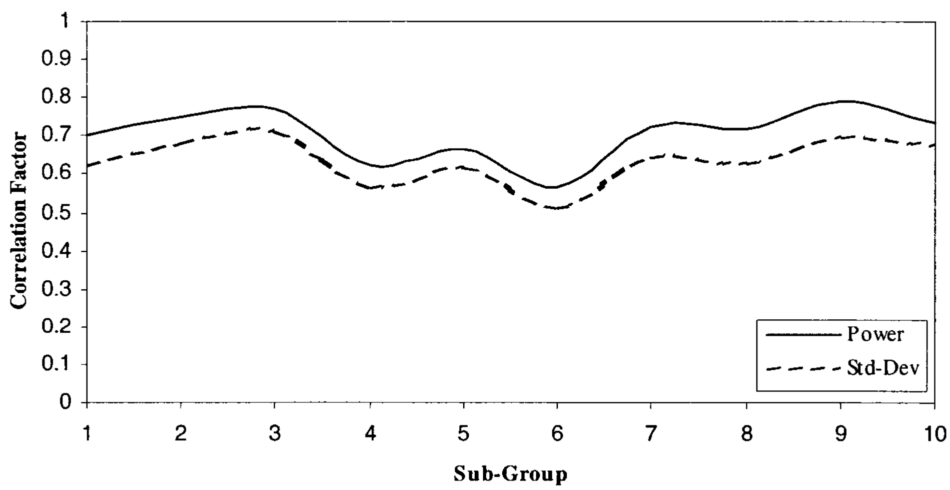




Figure B.66 Fan end bearing, fault detection – 0.014" fault diameter at inner race, 3 hp and 1730 rpm

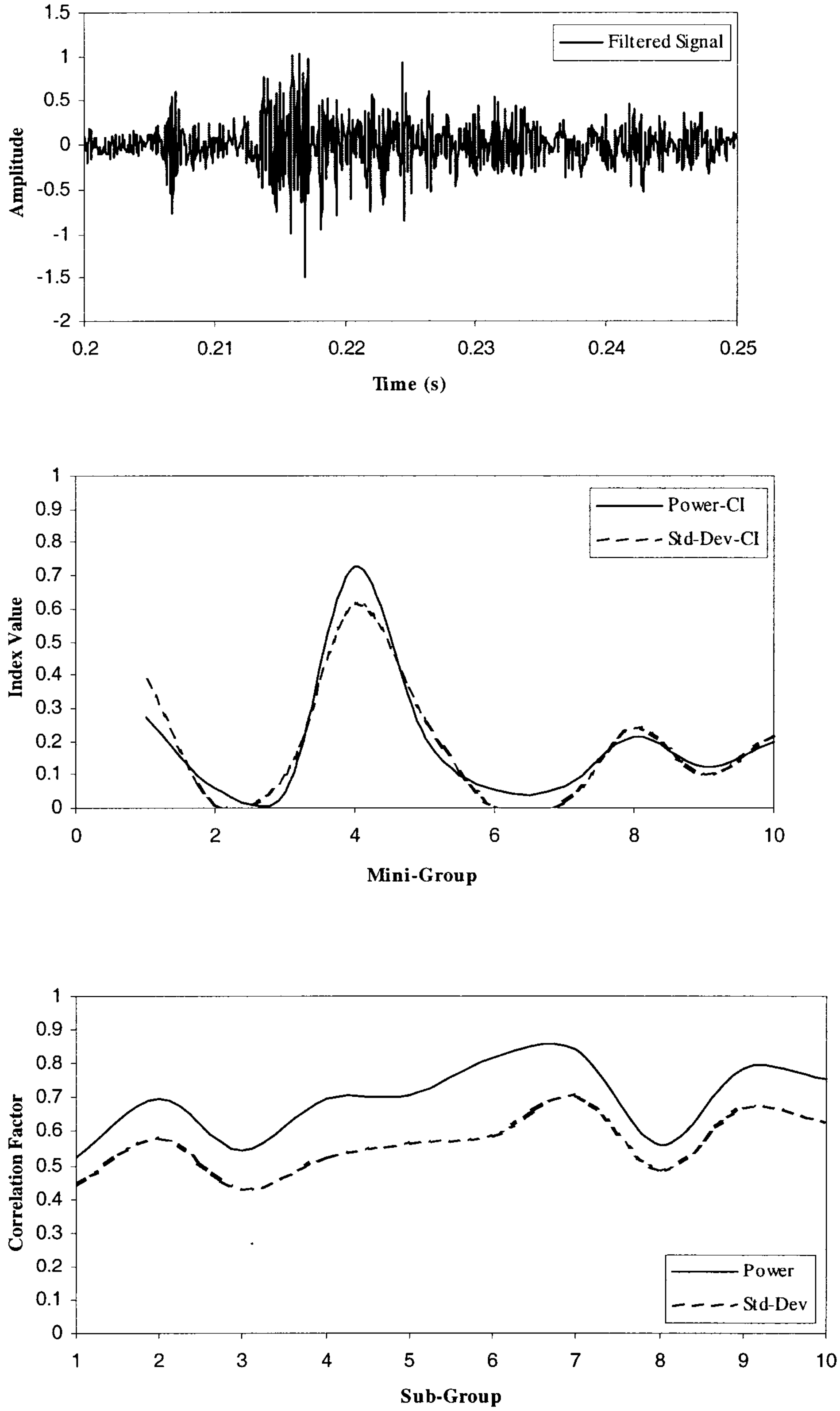



Figure B.67 Fan end bearing, fault detection – 0.014" fault diameter at ball, 0 hp and 1797 rpm.

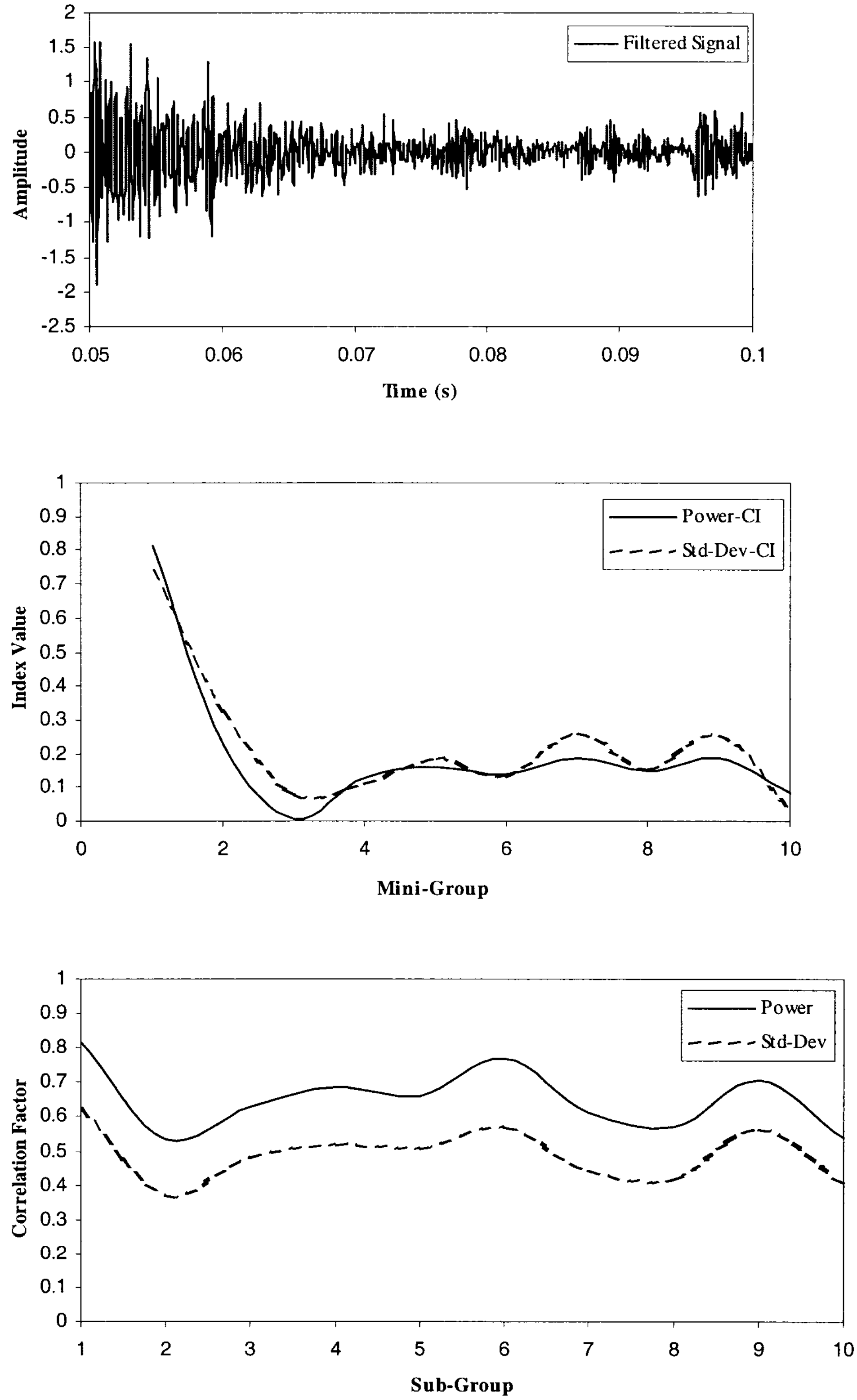



Figure B.68 Fan end bearing, fault detection – 0.014" fault diameter at ball, 2 hp and 1750 rpm.

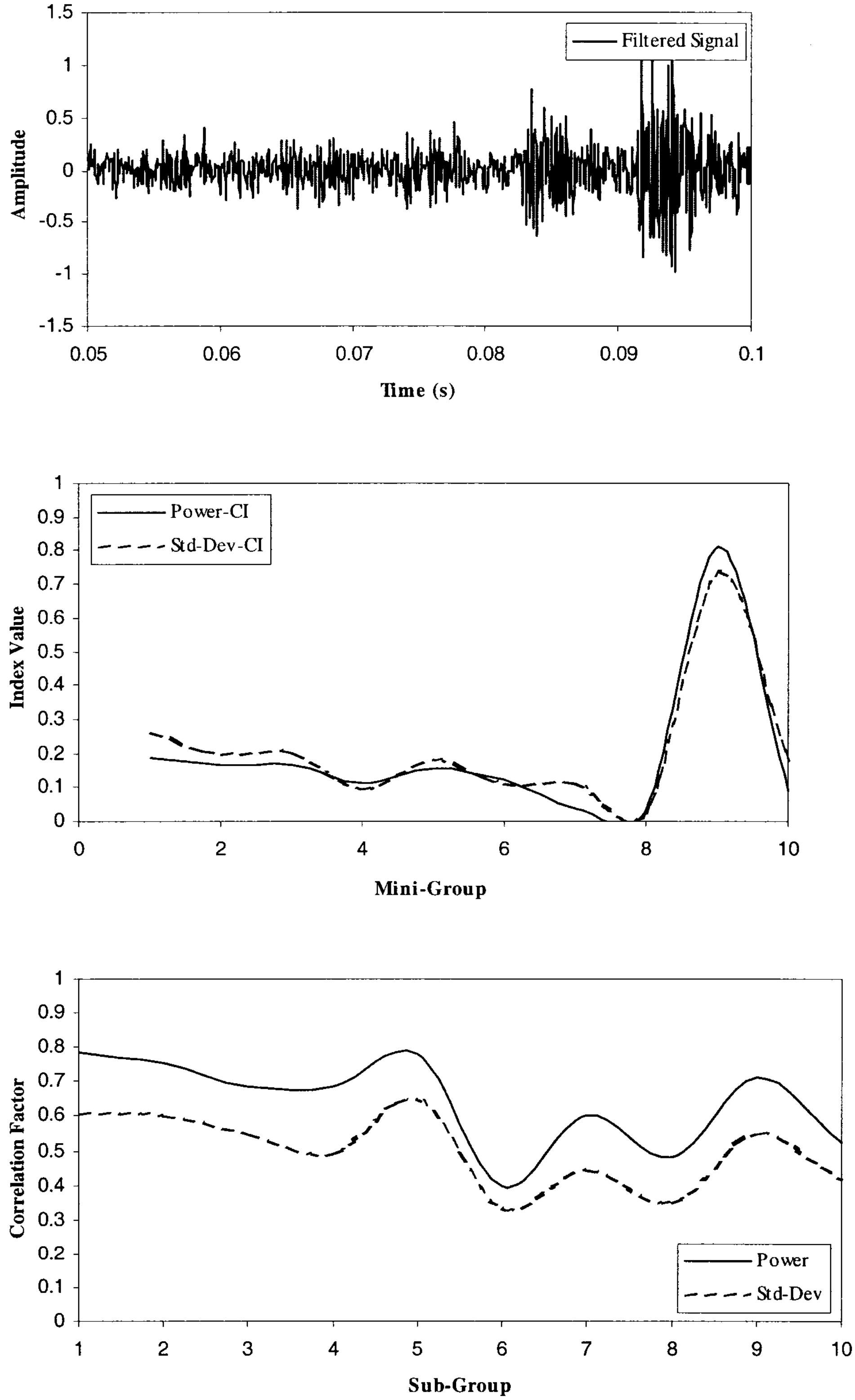
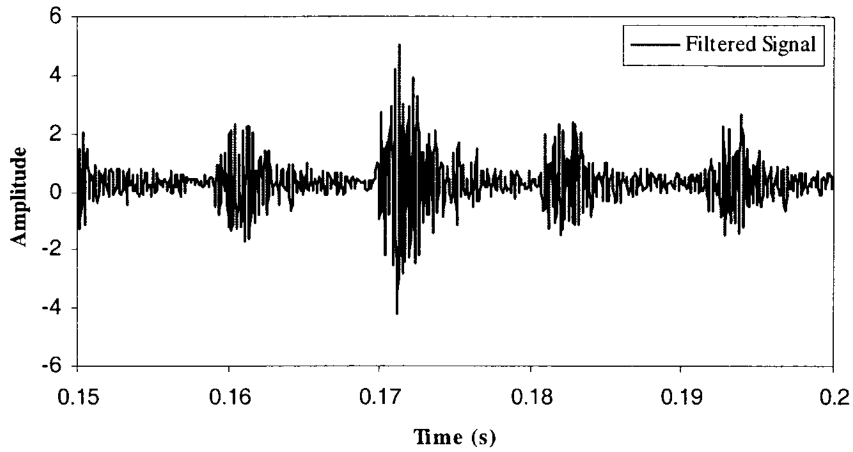
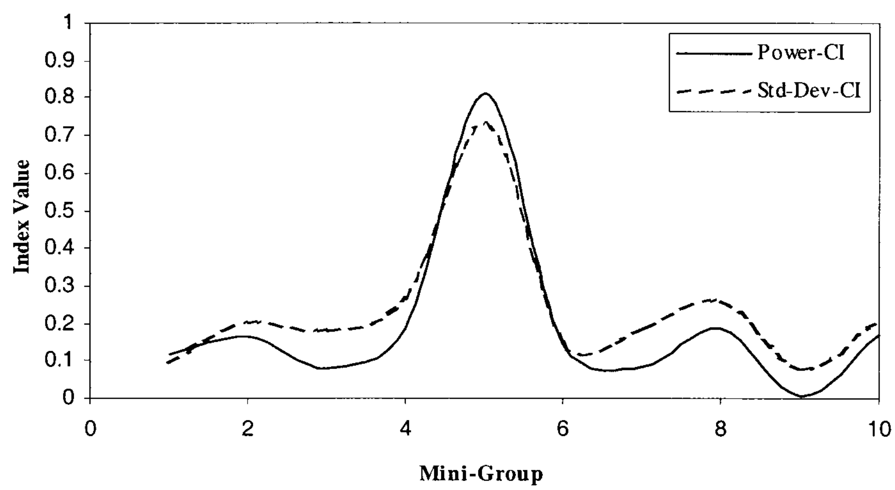



Figure B.69 Fan end bearing, fault detection – 0.014" fault diameter at ball, 3 hp and 1730 rpm.



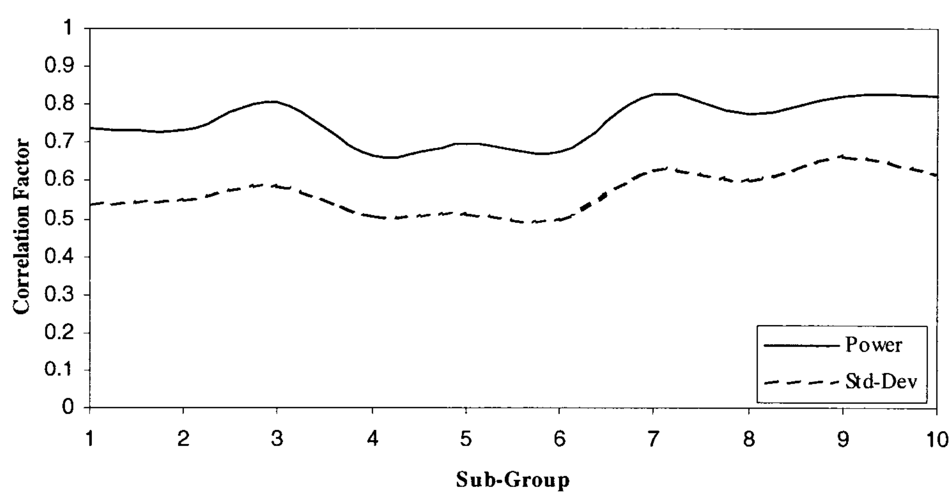
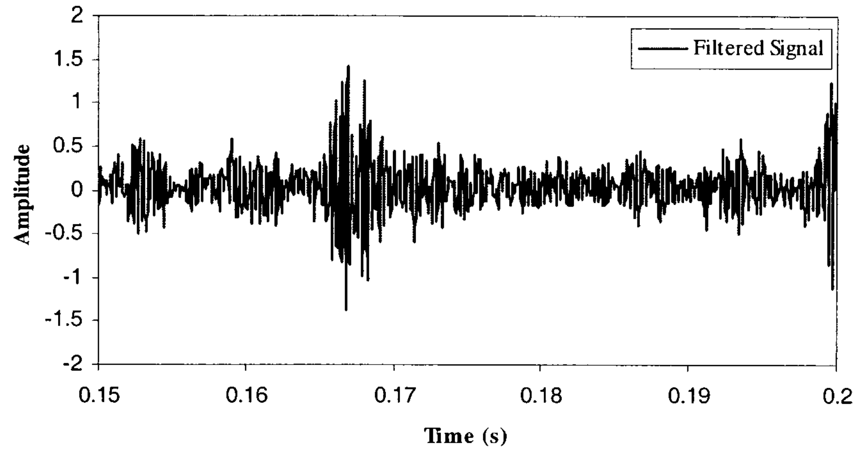
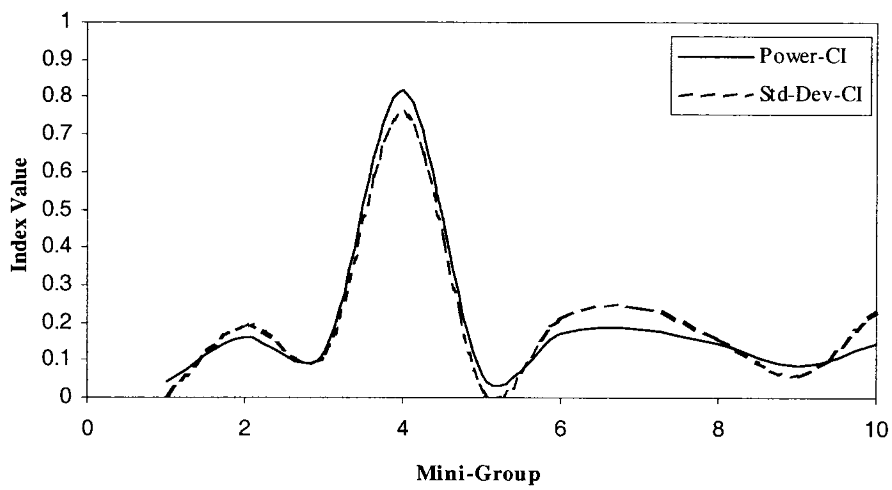




Figure B.70 Fan end bearing, fault detection – 0.014" fault diameter at outer race, 0 hp and 1797 rpm.



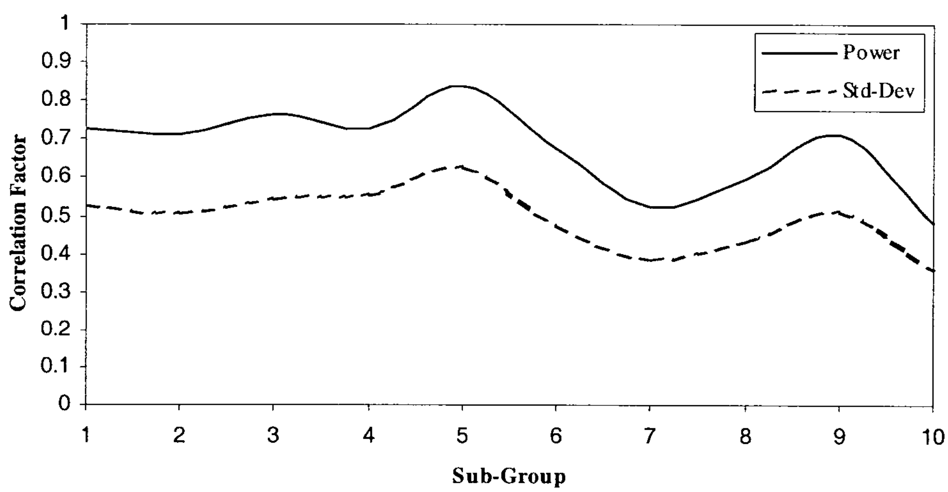




Figure B.71 Fan end bearing, fault detection – 0.021" fault diameter at inner race, 0 hp and 1797 rpm.

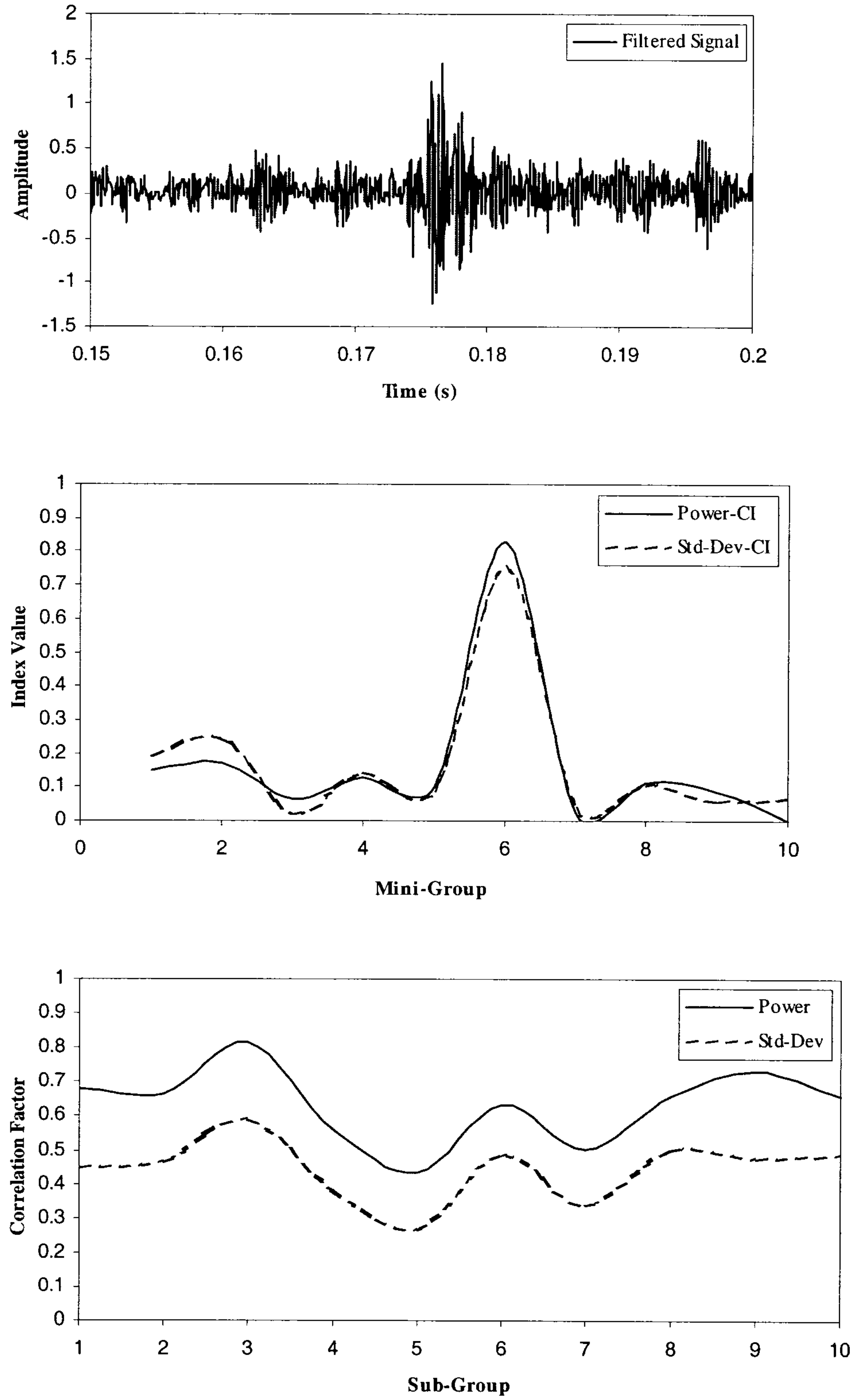



Figure B.72 Fan end bearing, fault detection – 0.021" fault diameter at inner race, 1 hp and 1772 rpm

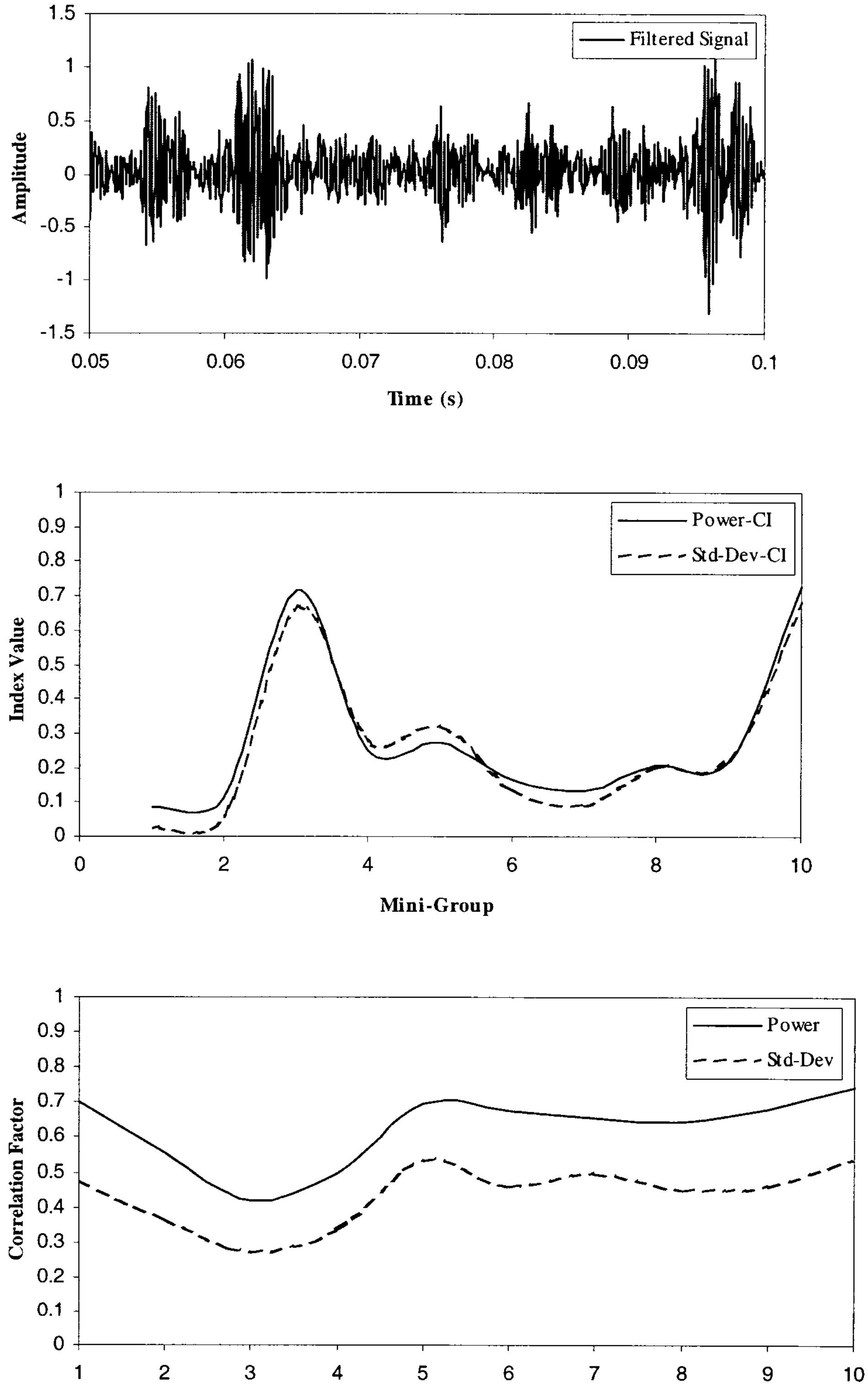



Figure B.73 Fan end bearing, fault detection – 0.021" fault diameter at inner race, 2 hp and 1750 rpm.

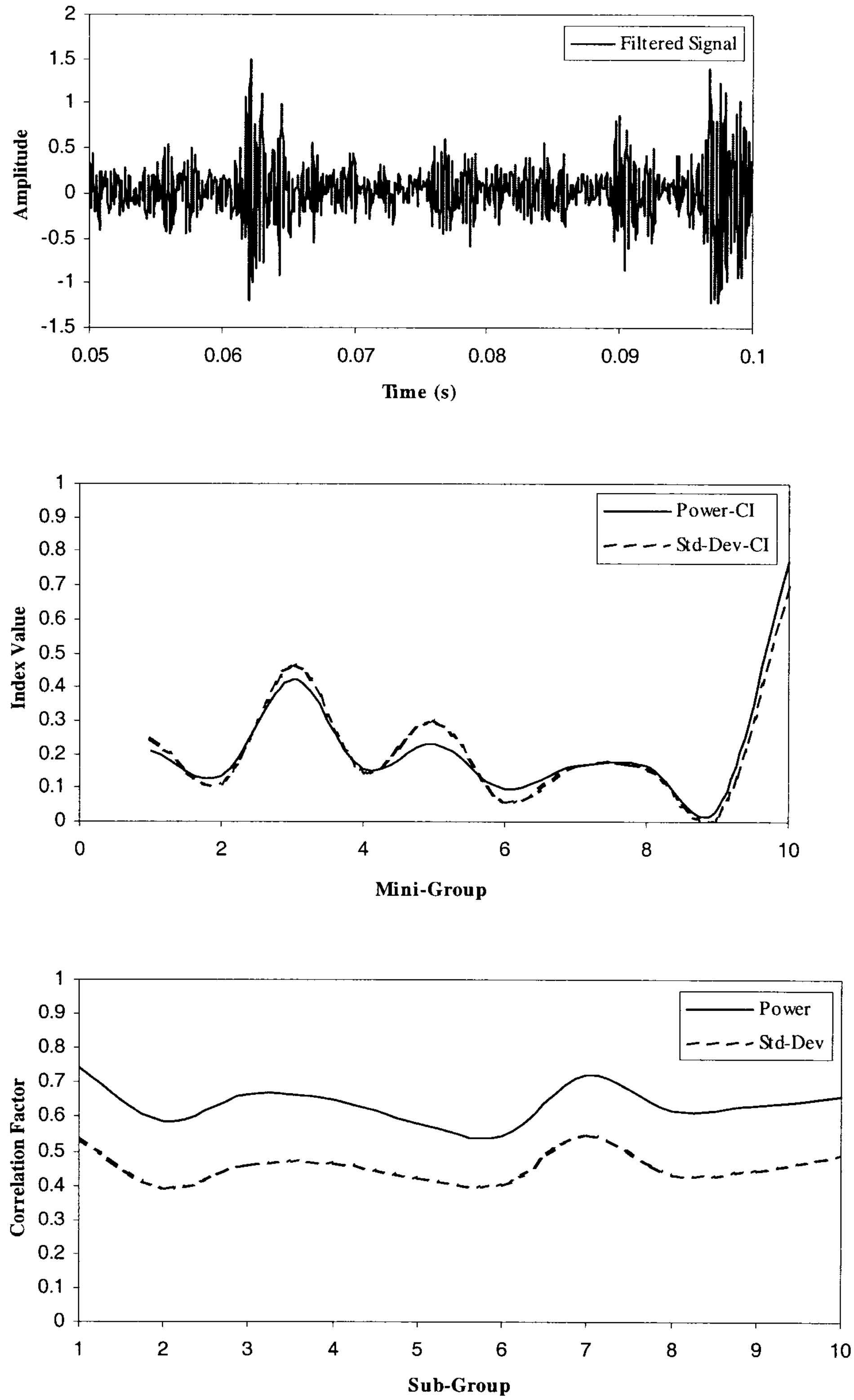
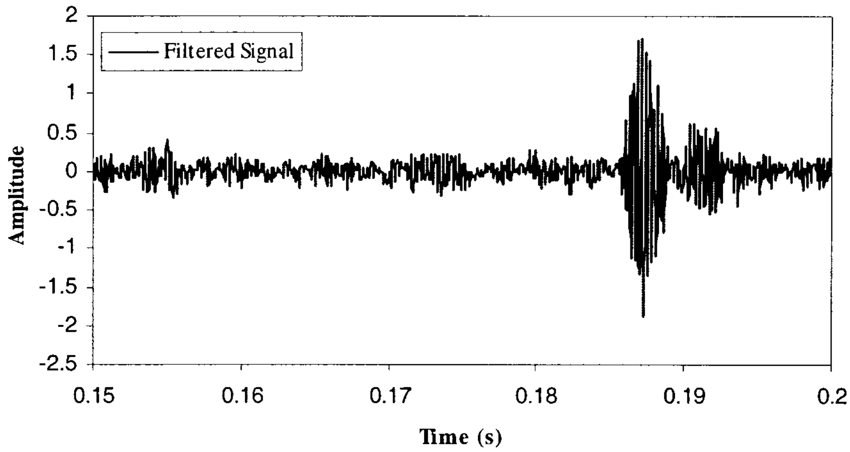
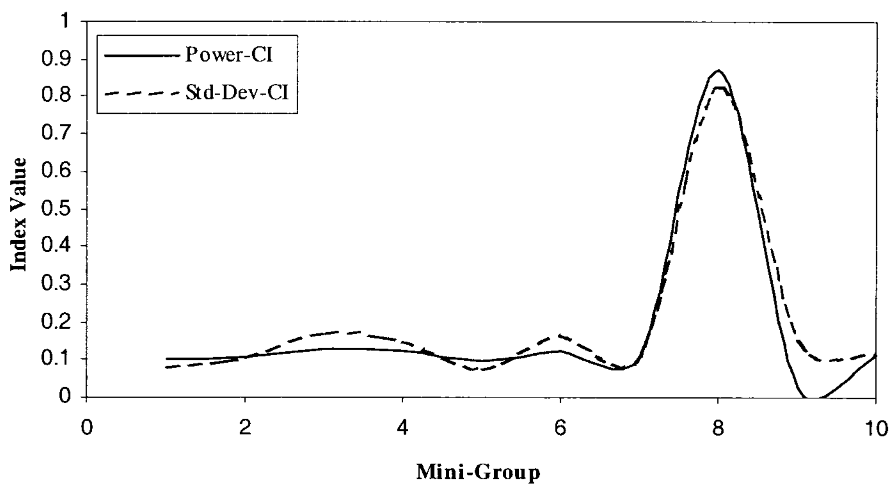



Figure B.74 Fan end bearing, fault detection – 0.021" fault diameter at inner race, 3 hp and 1730 rpm.



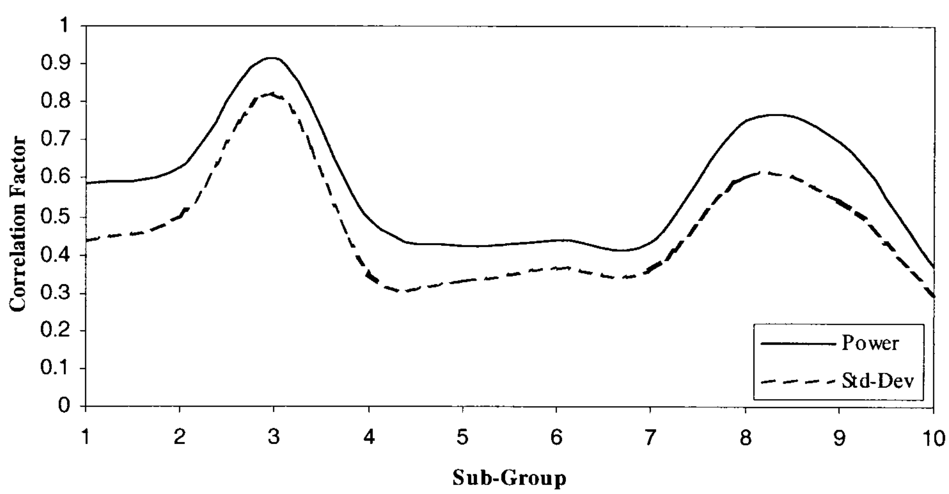




Figure B.75 Fan end bearing, fault detection – 0.021" fault diameter at ball, 0 hp and 1797 rpm.

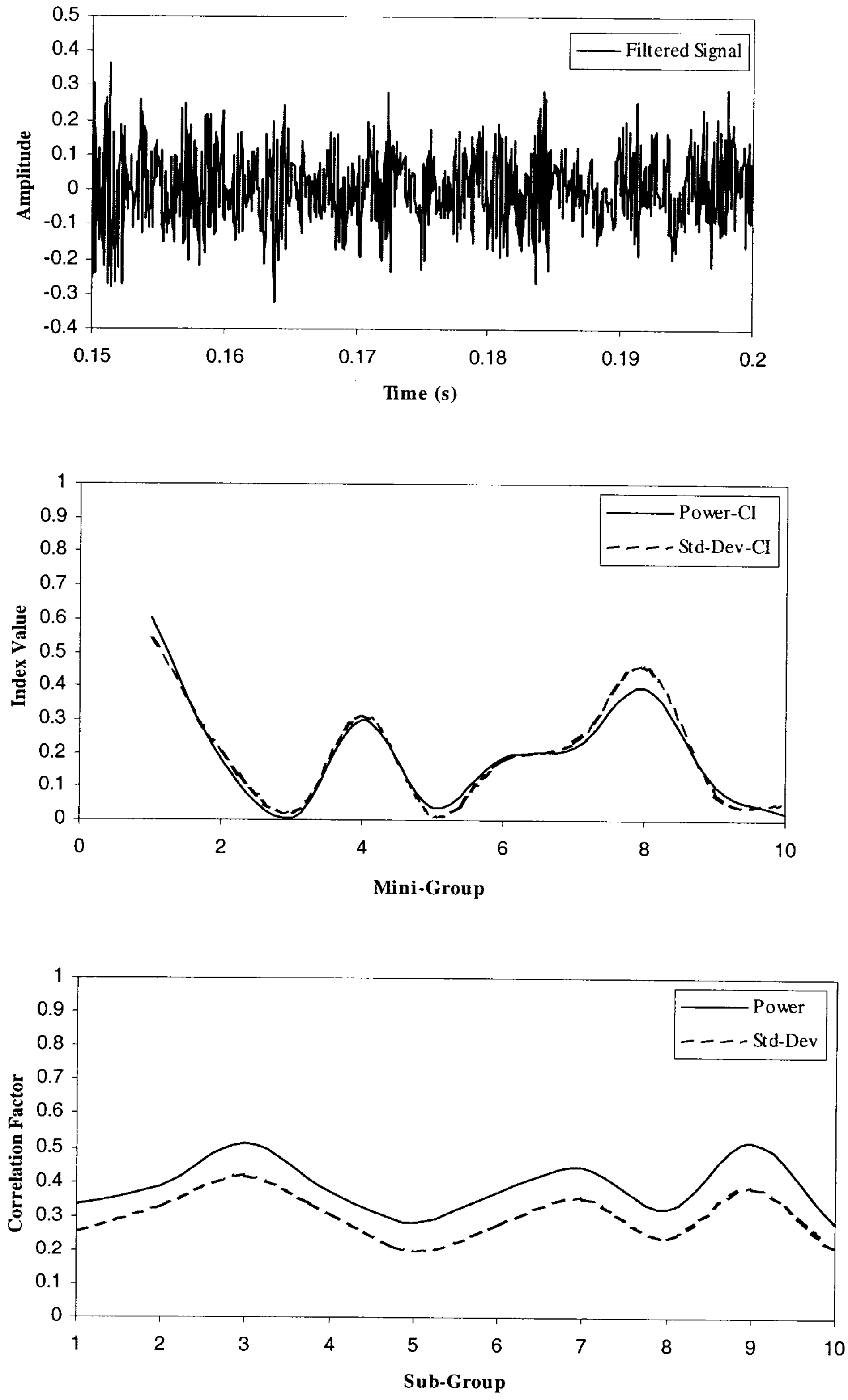
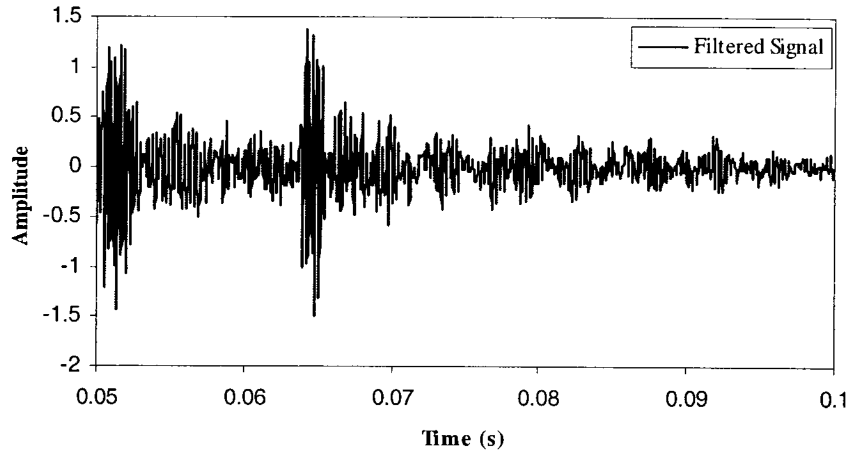
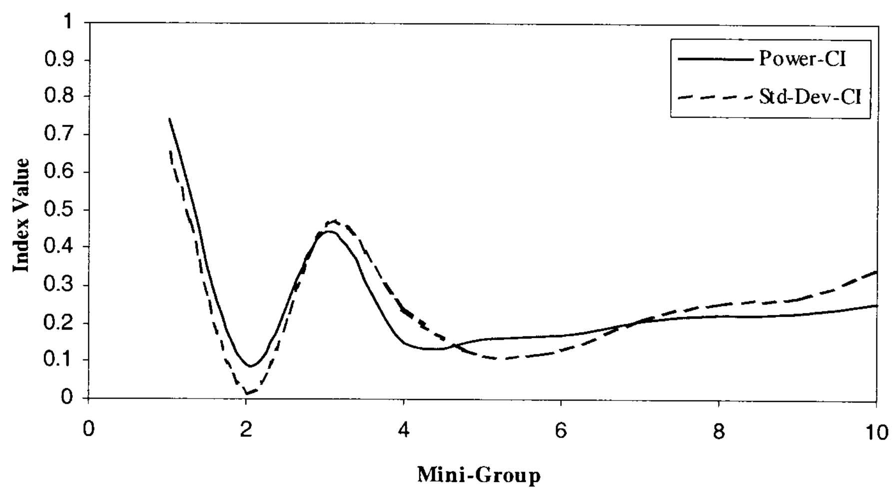



Figure B.76 Fan end bearing, fault detection – 0.021" fault diameter at ball, 1 hp and 1772 rpm.



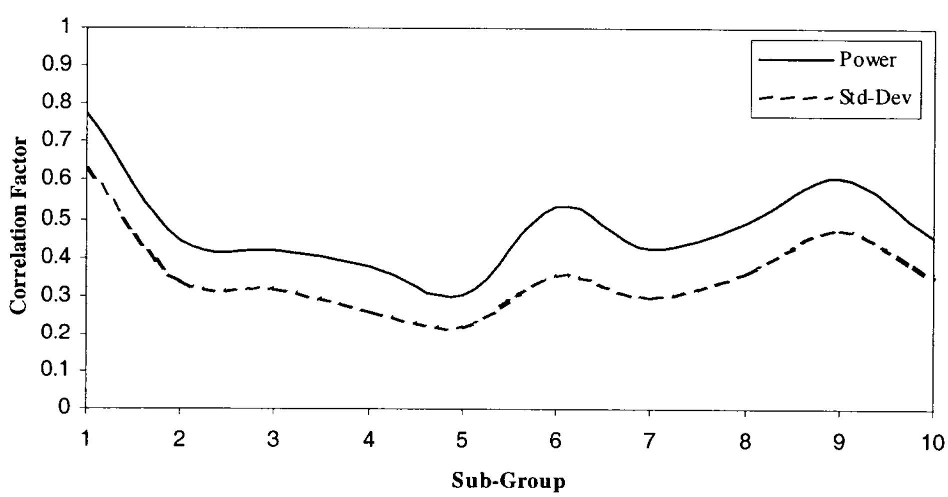




Figure B.77 Fan end bearing, fault detection – 0.021" fault diameter at ball, 2 hp and 1750 rpm.

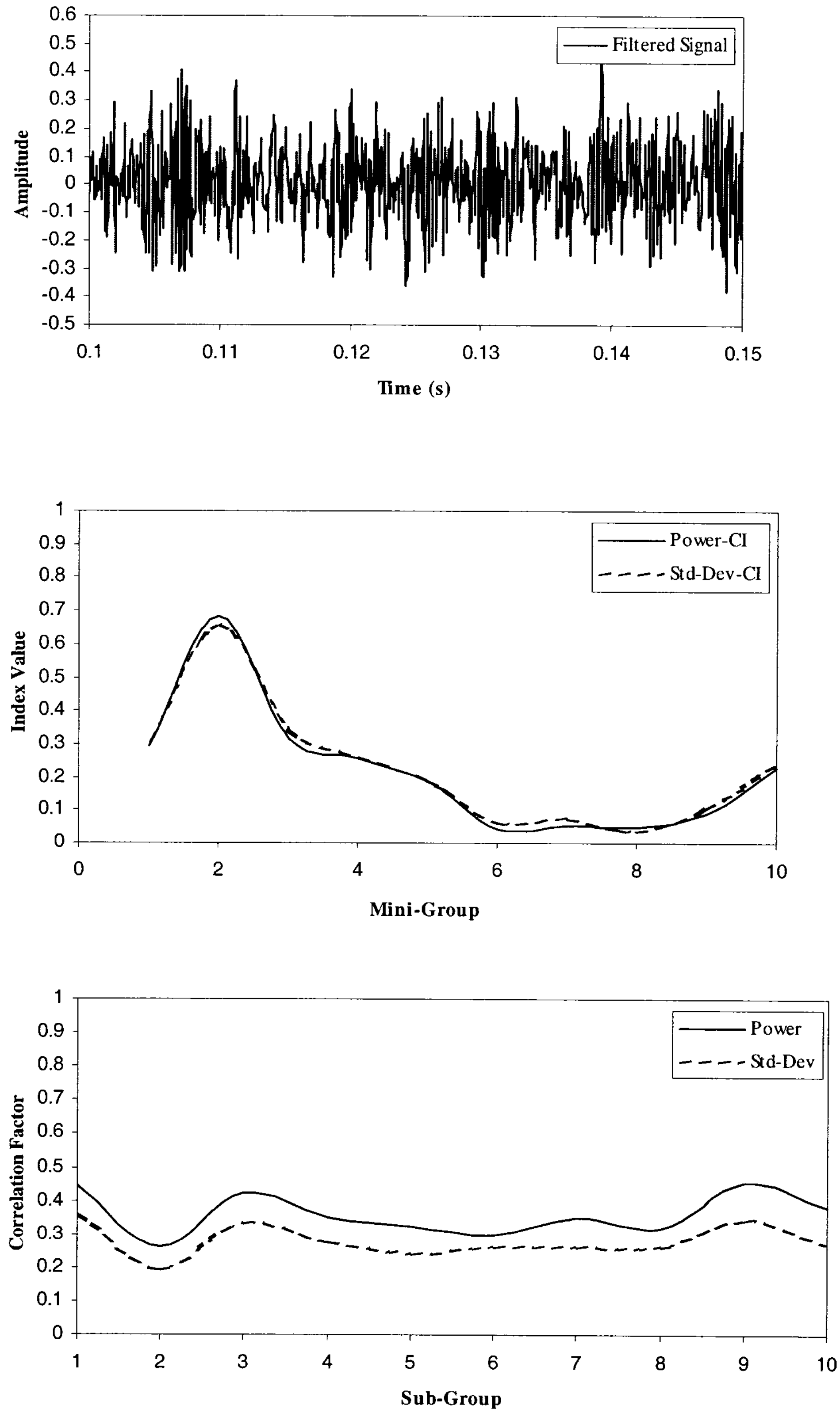



Figure B.78 Fan end bearing, fault detection – 0.021" fault diameter at ball, 3 hp and 1730 rpm.

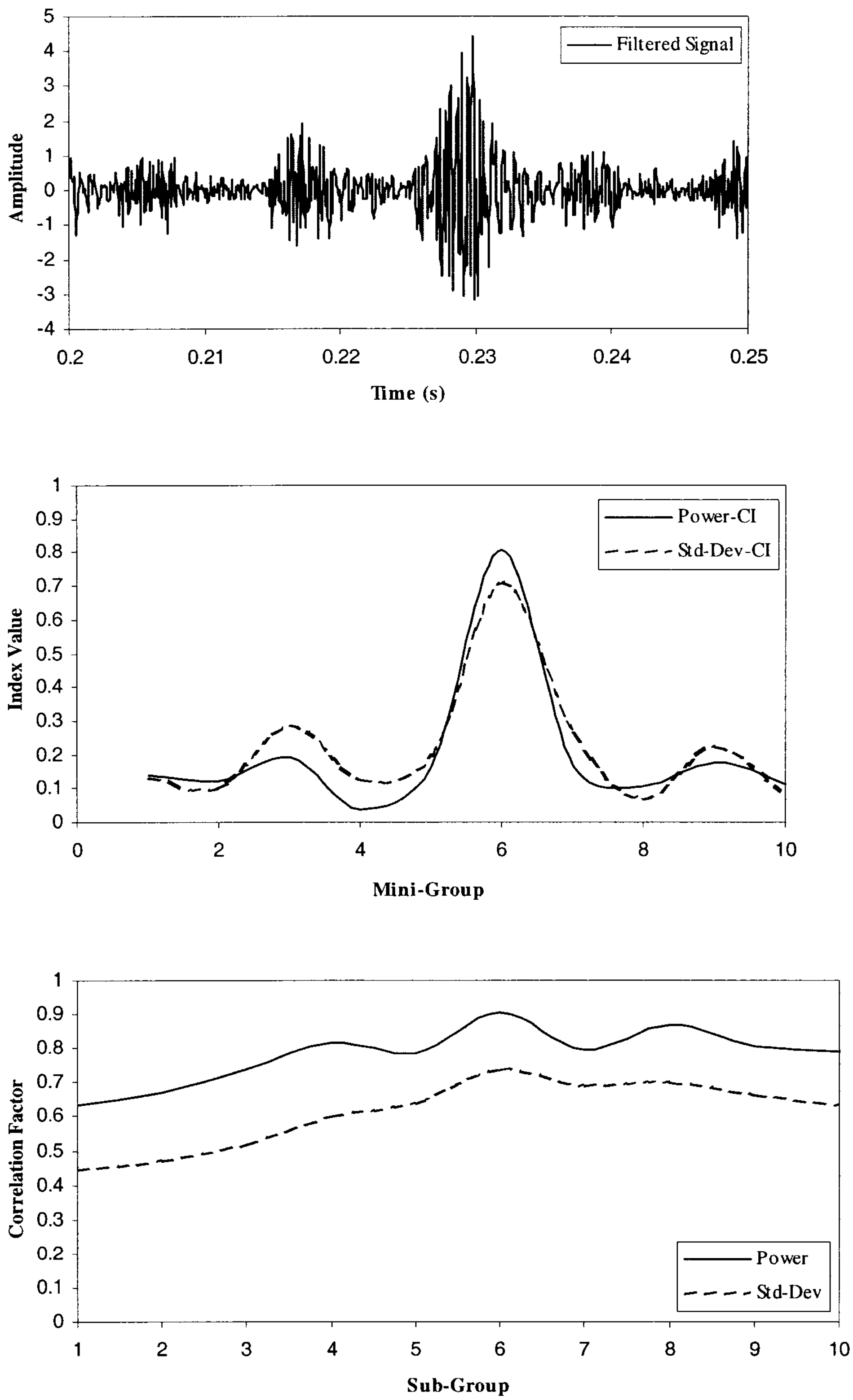



Figure B.79 Fan end bearing, fault detection – 0.021" fault diameter at outer race, 0 hp and 1797 rpm.

# Appendix C

## MATLAB Program *Fuzzy*

```
%-------------------------------------------------------------------
% File called Fuzzy.m
% This program uses the Sugeno method to calculate the fuzzy fused
% index FFI as described in Chapter 4
% ------------------------------------------------------------------
% Clear values from memory.
clear prob_PCI
clear prob_SCI
clear x
clear y
clear xx
clear yy
clear A
clear B
clear C
clear D
clear E
clear F
clear G
clear H
clear fuzzy
clear input
clear permut
clear permuta
clear pos
clear arm_out
clear outp
clear CI
clear dat
% First Read the results produced by the Monitoring.m file
% Use xlsread or database

fpci=max(power_CI); % fuzzy power CI = max power within sub-group
fsci=max(std_CI); % fuzzy std CI = max std dev  within sub-group
fpcf=1-corfp; % fuzzy power correlation factor = complement of power
correlation factor
fscf=1-corfs; % fuzzy std correlation factor
fuzzy(1)=fpci; % copy values into a matrix
fuzzy(2)=fsci;
fuzzy(3)=fpcf;
fuzzy(4)=fscf;
CI(dd,1)=fpci;
CI(dd,2)=fsci;
CI(dd,3)=corfp;
CI(dd,4)=corfs;
CI(dd,5)=time;
if (fpci>maxii & dd<5)
    pow(1:10)=power_CI(1:10);
    stdd(1:10)=std_CI(1:10);
    data(1:size-start+1)=d(start:size);
    t=start/fs:1/fs:size/fs-1/fs;
    maxii=fpci;
end;
A=0;
B=0;
C=0;
D=0;
```

```
E=0;
F=0;
G=0;
H=0;
AA=0;
BB=0;
CC=0;
DD=0;
EE=0;
FF=0;
GG=0;
HH=0;
for i=1:4
    input=fuzzy(i);
if input < 1/6
    if i==1
        A=3*input;
        B=1-A;
        AA=1;
        BB=0;
    else
        if i==2
            C=3*input;
            D=1-C;
            CC=1;
            DD=0;
        else
            if i==3
                E=3*input;
                F=1-E;
                EE=1;
                FF=0;
            else
                G=3*input;
                H=1-G;
                GG=1;
                HH=0;
            end;
        end;
    end;
end;
if (1/6<=input&input<1/3)
    if i==1
        A=1-3*input;
        B=1-A;
        AA=0;
        BB=1;
    else
        if i==2
            C=1-3*input;
            D=1-C;
            CC=0;
            DD=1;
        else
            if i==3
                E=1-3*input;
                F=1-E;
```

```
                EE=0;
                FF=1;
            else
                G=1-3*input;
                H=1-G;
                GG=0;
                HH=1;
            end;
        end;
    end;
end;
if (2/6<=input&input<3/6)
   if i==1
        A=3*input-1;
        B=1-A;
        AA=2;
        BB=1;
    else
        if i==2
            C=3*input-1;
            D=1-C;
            CC=2;
            DD=1;
        else
            if i==3
                E=3*input-1;
                F=1-E;
                EE=2;
                FF=1;
            else
                G=3*input-1;
                H=1-G;
                GG=2;
                HH=1;
            end;
        end;
    end;
end;
if (3/6<=input&input<4/6)
    if i==1
        A=2-3*input;
        B=1-A;
        AA=1;
        BB=2;
    else
        if i==2
            C=2-3*input;
            D=1-C;
            CC=1;
            DD=2;
        else
            if i==3
                E=2-3*input;
                F=1-E;
                EE=1;
                FF=2;
            else
```

```
                G=2-3*input;
                H=1-G;
                GG=1;
                HH=2;
            end;
        end;
    end;
end;
if (4/6<=input&input<5/6)
    if i==1
        A=3*input-2;
        B=1-A;
        AA=3;
        BB=2;
    else
        if i==2
            C=3*input-2;
            D=1-C;
            CC=3;
            DD=2;
        else
            if i==3
                E=3*input-2;
                F=1-E;
                EE=3;
                FF=2;
            else
                G=3*input-2;
                H=1-G;
                GG=3;
                HH=2;
            end;
        end;
    end;
end;
if (5/6<=input&input<6/6)
    if i==1
        A=3-3*input;
        B=1-A;
        AA=2;
        BB=3;
    else
        if i==2
            C=3-3*input;
            D=1-C;
            CC=2;
            DD=3;
        else
            if i==3
                E=3-3*input;
                F=1-E;
                EE=2;
                FF=3;
            else
                G=3-3*input;
                H=1-G;
                GG=2;
```

```
                HH=3;
            end;
        end;
    end;
end;
end;
 % Possible permutations
permut1(1)=A;% used to calculate the height
permut1(2)=C;
permut1(3)=E;
permut1(4)=G;
permuta1(1)=AA; % used to calculate the output weight
permuta1(2)=CC;
permuta1(3)=EE;
permuta1(4)=GG;
arm(1)= min(permut1); % minimum height for permutation
outp(1)=output(permuta1); % output is a function that finds the
location based on rules
arm_out(1)=arm(1)*outp(1); % height* location
permut2(1)=A;
permut2(2)=C;
permut2(3)=E;
permut2(4)=H;
permuta2(1)=AA;
permuta2(2)=CC;
permuta2(3)=EE;
permuta2(4)=HH;
arm(2)= min(permut2);
outp(2)=output(permuta2);
arm_out(2)=arm(2)*outp(2);
permut3(1)=A;
permut3(2)=C;
permut3(3)=F;
permut3(4)=G;
permuta3(1)=AA;
permuta3(2)=CC;
permuta3(3)=FF;
permuta3(4)=GG;
arm(3)= min(permut3);
outp(3)=output(permuta3);
arm_out(3)=arm(3)*outp(3);
permut4(1)=A;
permut4(2)=C;
permut4(3)=F;
permut4(4)=H;
permuta4(1)=AA;
permuta4(2)=CC;
permuta4(3)=FF;
permuta4(4)=HH;
arm(4)= min(permut4);
outp(4)=output(permuta4);
arm_out(4)=arm(4)*outp(4);
permut5(1)=A;
permut5(2)=D;
permut5(3)=E;
permut5(4)=G;
permuta5(1)=AA;
```

```
permuta5(2)=DD;
permuta5(3)=EE;
permuta5(4)=GG;
arm(5)= min(permut5);
outp(5)=output(permuta5);
arm_out(5)=arm(5)*outp(5);
permut6(1)=A;
permut6(2)=D;
permut6(3)=E;
permut6(4)=H;
permuta6(1)=AA;
permuta6(2)=DD;
permuta6(3)=EE;
permuta6(4)=HH;
arm(6)= min(permut6);
outp(6)=output(permuta6);
arm_out(6)=arm(6)*outp(6);
permut7(1)=A;
permut7(2)=D;
permut7(3)=F;
permut7(4)=G;
permuta7(1)=AA;
permuta7(2)=DD;
permuta7(3)=FF;
permuta7(4)=GG;
arm(7)= min(permut7);
outp(7)=output(permuta7);
arm_out(7)=arm(7)*outp(7);
permut8(1)=A;
permut8(2)=D;
permut8(3)=F;
permut8(4)=H;
permuta8(1)=AA;
permuta8(2)=DD;
permuta8(3)=FF;
permuta8(4)=HH;
arm(8)= min(permut8);
outp(8)=output(permuta8);
arm_out(8)=arm(8)*outp(8);
permut9(1)=B;
permut9(2)=C;
permut9(3)=E;
permut9(4)=G;
permuta9(1)=BB;
permuta9(2)=CC;
permuta9(3)=EE;
permuta9(4)=GG;
arm(9)= min(permut9);
outp(9)=output(permuta9);
arm_out(9)=arm(9)*outp(9);
permut10(1)=B;
permut10(2)=C;
permut10(3)=E;
permut10(4)=H;
permuta10(1)=BB;
permuta10(2)=CC;
permuta10(3)=EE;
```

```
permuta10(4)=HH;
arm(10)= min(permut10);
outp(10)=output(permuta10);
arm_out(10)=arm(10)*outp(10);
permut11(1)=B;
permut11(2)=C;
permut11(3)=F;
permut11(4)=G;
permuta11(1)=BB;
permuta11(2)=CC;
permuta11(3)=FF;
permuta11(4)=GG;
arm(11)= min(permut11);
outp(11)=output(permuta11);
arm_out(11)=arm(11)*outp(11);
permut12(1)=B;
permut12(2)=C;
permut12(3)=F;
permut12(4)=H;
permuta12(1)=BB;
permuta12(2)=CC;
permuta12(3)=FF;
permuta12(4)=HH;
arm(12)= min(permut12);
outp(12)=output(permuta12);
arm_out(12)=arm(12)*outp(12);
permut13(1)=B;
permut13(2)=D;
permut13(3)=E;
permut13(4)=G;
permuta13(1)=BB;
permuta13(2)=DD;
permuta13(3)=EE;
permuta13(4)=GG;
arm(13)= min(permut13);
outp(13)=output(permuta13);
arm_out(13)=arm(13)*outp(13);
permut14(1)=B;
permut14(2)=D;
permut14(3)=E;
permut14(4)=H;
permuta14(1)=BB;
permuta14(2)=DD;
permuta14(3)=EE;
permuta14(4)=HH;
arm(14)= min(permut14);
outp(14)=output(permuta14);
arm_out(14)=arm(14)*outp(14);
permut15(1)=B;
permut15(2)=D;
permut15(3)=F;
permut15(4)=G;
permuta15(1)=BB;
permuta15(2)=DD;
permuta15(3)=FF;
permuta15(4)=GG;
arm(15)= min(permut15);
```

```
outp(15)=output(permuta15);
arm_out(15)=arm(15)*outp(15);
permut16(1)=B;
permut16(2)=D;
permut16(3)=F;
permut16(4)=H;
permuta16(1)=BB;
permuta16(2)=DD;
permuta16(3)=FF;
permuta16(4)=HH;
arm(16)= min(permut16);
outp(16)=output(permuta16);
arm_out(16)=arm(16)*outp(16);
position=sum(arm_out)/sum(arm); % fuzzy fused index
pos(dd)=position
end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
```

```
% This function "output" provides the output spike based on
% triggered rules. This function is used in "Fuzzy" program.
%-----------------------------------------------------------------------

function [out]=output(in);
nn=length(in);
% define output values
ov0=0.0;
ov1=.5;
ov2=1;
% define index search
% first column for index search only. 2nd, 3rd, 4th and fifth are the
% inputs and sixth is the output
index=sum(in)+10;
rules(1,1)=0+10;
rules(1,2)=0;
rules(1,3)=0;
rules(1,4)=0;
rules(1,5)=0;
rules(1,6)=ov0;
rules(2,1)=1+10;
rules(2,2)=0;
rules(2,3)=0;
rules(2,4)=0;
rules(2,5)=1;
rules(2,6)=ov0;
rules(3,1)=2+10;
rules(3,2)=0;
rules(3,3)=0;
rules(3,4)=0;
rules(3,5)=2;
rules(3,6)=ov2;
rules(4,1)=3+10;
rules(4,2)=0;
rules(4,3)=0;
rules(4,4)=0;
rules(4,5)=3;
rules(4,6)=ov2;
rules(5,1)=1+10;
rules(5,2)=0;
rules(5,3)=0;
rules(5,4)=1;
rules(5,5)=0;
rules(5,6)=ov0;
rules(6,1)=2+10;
rules(6,2)=0;
rules(6,3)=0;
rules(6,4)=1;
rules(6,5)=1;
rules(6,6)=ov2;
rules(7,1)=13;
rules(7,2)=0;
rules(7,3)=0;
rules(7,4)=1;
rules(7,5)=2;
rules(7,6)=ov2;
rules(8,1)=14;
```

```
rules(8,2)=0;
rules(8,3)=0;
rules(8,4)=1;
rules(8,5)=3;
rules(8,6)=ov2;
rules(9,1)=12;
rules(9,2)=0;
rules(9,3)=0;
rules(9,4)=2;
rules(9,5)=0;
rules(9,6)=ov1;
rules(10,1)=13;
rules(10,2)=0;
rules(10,3)=0;
rules(10,4)=2;
rules(10,5)=1;
rules(10,6)=ov2;
rules(11,1)=14;
rules(11,2)=0;
rules(11,3)=0;
rules(11,4)=2;
rules(11,5)=2;
rules(11,6)=ov2;
rules(12,1)=15;
rules(12,2)=0;
rules(12,3)=0;
rules(12,4)=2;
rules(12,5)=3;
rules(12,6)=ov2;
rules(13,1)=13;
rules(13,2)=0;
rules(13,3)=0;
rules(13,4)=3;
rules(13,5)=0;
rules(13,6)=ov1;
rules(14,1)=14;
rules(14,2)=0;
rules(14,3)=0;
rules(14,4)=3;
rules(14,5)=1;
rules(14,6)=ov2;
rules(15,1)=15;
rules(15,2)=0;
rules(15,3)=0;
rules(15,4)=3;
rules(15,5)=2;
rules(15,6)=ov2;
rules(16,1)=16;
rules(16,2)=0;
rules(16,3)=0;
rules(16,4)=3;
rules(16,5)=3;
rules(16,6)=ov2;
rules(17,1)=11;
rules(17,2)=0;
rules(17,3)=1;
rules(17,4)=0;
```

```
rules(17,5)=0;
rules(17,6)=ov0;
rules(18,1)=12;
rules(18,2)=0;
rules(18,3)=1;
rules(18,4)=0;
rules(18,5)=1;
rules(18,6)=ov0;
rules(19,1)=13;
rules(19,2)=0;
rules(19,3)=1;
rules(19,4)=0;
rules(19,5)=2;
rules(19,6)=ov2;
rules(20,1)=14;
rules(20,2)=0;
rules(20,3)=1;
rules(20,4)=0;
rules(20,5)=3;
rules(20,6)=ov2;
rules(21,1)=12;
rules(21,2)=0;
rules(21,3)=1;
rules(21,4)=1;
rules(21,5)=0;
rules(21,6)=ov0;
rules(22,1)=13;
rules(22,2)=0;
rules(22,3)=1;
rules(22,4)=1;
rules(22,5)=1;
rules(22,6)=ov2;
rules(23,1)=14;
rules(23,2)=0;
rules(23,3)=1;
rules(23,4)=1;
rules(23,5)=2;
rules(23,6)=ov2;
rules(24,1)=15;
rules(24,2)=0;
rules(24,3)=1;
rules(24,4)=1;
rules(24,5)=3;
rules(24,6)=ov2;
rules(25,1)=13;
rules(25,2)=0;
rules(25,3)=1;
rules(25,4)=2;
rules(25,5)=0;
rules(25,6)=ov1;
rules(26,1)=14;
rules(26,2)=0;
rules(26,3)=1;
rules(26,4)=2;
rules(26,5)=1;
rules(26,6)=ov2;
rules(27,1)=15;
```

```
rules(27,2)=0;
rules(27,3)=1;
rules(27,4)=2;
rules(27,5)=2;
rules(27,6)=ov2;
rules(28,1)=16;
rules(28,2)=0;
rules(28,3)=1;
rules(28,4)=2;
rules(28,5)=3;
rules(28,6)=ov2;
rules(29,1)=14;
rules(29,2)=0;
rules(29,3)=1;
rules(29,4)=3;
rules(29,5)=0;
rules(29,6)=ov1;
rules(30,1)=15;
rules(30,2)=0;
rules(30,3)=1;
rules(30,4)=3;
rules(30,5)=1;
rules(30,6)=ov2;
rules(31,1)=16;
rules(31,2)=0;
rules(31,3)=1;
rules(31,4)=3;
rules(31,5)=2;
rules(31,6)=ov2;
rules(32,1)=17;
rules(32,2)=0;
rules(32,3)=1;
rules(32,4)=3;
rules(32,5)=3;
rules(32,6)=ov2;
rules(33,1)=12;
rules(33,2)=0;
rules(33,3)=2;
rules(33,4)=0;
rules(33,5)=0;
rules(33,6)=ov0;
rules(34,1)=13;
rules(34,2)=0;
rules(34,3)=2;
rules(34,4)=0;
rules(34,5)=1;
rules(34,6)=ov0;
rules(35,1)=14;
rules(35,2)=0;
rules(35,3)=2;
rules(35,4)=0;
rules(35,5)=2;
rules(35,6)=ov1;
rules(36,1)=15;
rules(36,2)=0;
rules(36,3)=2;
rules(36,4)=0;
```

```
rules(36,5)=3;
rules(36,6)=ov2;
rules(37,1)=13;
rules(37,2)=0;
rules(37,3)=2;
rules(37,4)=1;
rules(37,5)=0;
rules(37,6)=ov0;
rules(38,1)=14;
rules(38,2)=0;
rules(38,3)=2;
rules(38,4)=1;
rules(38,5)=1;
rules(38,6)=ov2;
rules(39,1)=15;
rules(39,2)=0;
rules(39,3)=2;
rules(39,4)=1;
rules(39,5)=2;
rules(39,6)=ov2;
rules(40,1)=16;
rules(40,2)=0;
rules(40,3)=2;
rules(40,4)=1;
rules(40,5)=3;
rules(40,6)=ov2;
rules(41,1)=14;
rules(41,2)=0;
rules(41,3)=2;
rules(41,4)=2;
rules(41,5)=0;
rules(41,6)=ov1;
rules(42,1)=15;
rules(42,2)=0;
rules(42,3)=2;
rules(42,4)=2;
rules(42,5)=1;
rules(42,6)=ov2;
rules(43,1)=16;
rules(43,2)=0;
rules(43,3)=2;
rules(43,4)=2;
rules(43,5)=2;
rules(43,6)=ov2;
rules(44,1)=17;
rules(44,2)=0;
rules(44,3)=2;
rules(44,4)=2;
rules(44,5)=3;
rules(44,6)=ov2;
rules(45,1)=15;
rules(45,2)=0;
rules(45,3)=2;
rules(45,4)=3;
rules(45,5)=0;
rules(45,6)=ov1;
rules(46,1)=16;
```

```
rules(46,2)=0;
rules(46,3)=2;
rules(46,4)=3;
rules(46,5)=1;
rules(46,6)=ov2;
rules(47,1)=17;
rules(47,2)=0;
rules(47,3)=2;
rules(47,4)=3;
rules(47,5)=2;
rules(47,6)=ov2;
rules(48,1)=18;
rules(48,2)=0;
rules(48,3)=2;
rules(48,4)=3;
rules(48,5)=3;
rules(48,6)=ov2;
rules(49,1)=13;
rules(49,2)=0;
rules(49,3)=3;
rules(49,4)=0;
rules(49,5)=0;
rules(49,6)=ov1;
rules(50,1)=14;
rules(50,2)=0;
rules(50,3)=3;
rules(50,4)=0;
rules(50,5)=1;
rules(50,6)=ov0;
rules(51,1)=15;
rules(51,2)=0;
rules(51,3)=3;
rules(51,4)=0;
rules(51,5)=2;
rules(51,6)=ov2;
rules(52,1)=16;
rules(52,2)=0;
rules(52,3)=3;
rules(52,4)=0;
rules(52,5)=3;
rules(52,6)=ov2;
rules(53,1)=14;
rules(53,2)=0;
rules(53,3)=3;
rules(53,4)=1;
rules(53,5)=0;
rules(53,6)=ov0;
rules(54,1)=15;
rules(54,2)=0;
rules(54,3)=3;
rules(54,4)=1;
rules(54,5)=1;
rules(54,6)=ov2;
rules(55,1)=16;
rules(55,2)=0;
rules(55,3)=3;
rules(55,4)=1;
```

```
rules(55,5)=2;
rules(55,6)=ov2;
rules(56,1)=17;
rules(56,2)=0;
rules(56,3)=3;
rules(56,4)=1;
rules(56,5)=3;
rules(56,6)=ov2;
rules(57,1)=15;
rules(57,2)=0;
rules(57,3)=3;
rules(57,4)=2;
rules(57,5)=0;
rules(57,6)=ov1;
rules(58,1)=16;
rules(58,2)=0;
rules(58,3)=3;
rules(58,4)=2;
rules(58,5)=1;
rules(58,6)=ov2;
rules(59,1)=17;
rules(59,2)=0;
rules(59,3)=3;
rules(59,4)=2;
rules(59,5)=2;
rules(59,6)=ov2;
rules(60,1)=18;
rules(60,2)=0;
rules(60,3)=3;
rules(60,4)=2;
rules(60,5)=3;
rules(60,6)=ov2;
rules(61,1)=16;
rules(61,2)=0;
rules(61,3)=3;
rules(61,4)=3;
rules(61,5)=0;
rules(61,6)=ov1;
rules(62,1)=17;
rules(62,2)=0;
rules(62,3)=3;
rules(62,4)=3;
rules(62,5)=1;
rules(62,6)=ov2;
rules(63,1)=18;
rules(63,2)=0;
rules(63,3)=3;
rules(63,4)=3;
rules(63,5)=2;
rules(63,6)=ov2;
rules(64,1)=19;
rules(64,2)=0;
rules(64,3)=3;
rules(64,4)=3;
rules(64,5)=3;
rules(64,6)=ov2;
rules(65,1)=11;
```

```
rules(65,2)=1;
rules(65,3)=0;
rules(65,4)=0;
rules(65,5)=0;
rules(65,6)=ov0;
rules(66,1)=12;
rules(66,2)=1;
rules(66,3)=0;
rules(66,4)=0;
rules(66,5)=1;
rules(66,6)=ov0;
rules(67,1)=13;
rules(67,2)=1;
rules(67,3)=0;
rules(67,4)=0;
rules(67,5)=2;
rules(67,6)=ov2;
rules(68,1)=14;
rules(68,2)=1;
rules(68,3)=0;
rules(68,4)=0;
rules(68,5)=3;
rules(68,6)=ov2;
rules(69,1)=12;
rules(69,2)=1;
rules(69,3)=0;
rules(69,4)=1;
rules(69,5)=0;
rules(69,6)=ov0;
rules(70,1)=13;
rules(70,2)=1;
rules(70,3)=0;
rules(70,4)=1;
rules(70,5)=1;
rules(70,6)=ov2;
rules(71,1)=14;
rules(71,2)=1;
rules(71,3)=0;
rules(71,4)=1;
rules(71,5)=2;
rules(71,6)=ov2;
rules(72,1)=15;
rules(72,2)=1;
rules(72,3)=0;
rules(72,4)=1;
rules(72,5)=3;
rules(72,6)=ov2;
rules(73,1)=13;
rules(73,2)=1;
rules(73,3)=0;
rules(73,4)=2;
rules(73,5)=0;
rules(73,6)=ov1;
rules(74,1)=14;
rules(74,2)=1;
rules(74,3)=0;
rules(74,4)=2;
```

```
rules(74,5)=1;
rules(74,6)=ov2;
rules(75,1)=15;
rules(75,2)=1;
rules(75,3)=0;
rules(75,4)=2;
rules(75,5)=2;
rules(75,6)=ov2;
rules(76,1)=16;
rules(76,2)=1;
rules(76,3)=0;
rules(76,4)=2;
rules(76,5)=3;
rules(76,6)=ov2;
rules(77,1)=14;
rules(77,2)=1;
rules(77,3)=0;
rules(77,4)=3;
rules(77,5)=0;
rules(77,6)=ov1;
rules(78,1)=15;
rules(78,2)=1;
rules(78,3)=0;
rules(78,4)=3;
rules(78,5)=1;
rules(78,6)=ov2;
rules(79,1)=16;
rules(79,2)=1;
rules(79,3)=0;
rules(79,4)=3;
rules(79,5)=2;
rules(79,6)=ov2;
rules(80,1)=17;
rules(80,2)=1;
rules(80,3)=0;
rules(80,4)=3;
rules(80,5)=3;
rules(80,6)=ov2;
rules(81,1)=12;
rules(81,2)=1;
rules(81,3)=1;
rules(81,4)=0;
rules(81,5)=0;
rules(81,6)=ov0;
rules(82,1)=13;
rules(82,2)=1;
rules(82,3)=1;
rules(82,4)=0;
rules(82,5)=1;
rules(82,6)=ov0;
rules(83,1)=14;
rules(83,2)=1;
rules(83,3)=1;
rules(83,4)=0;
rules(83,5)=2;
rules(83,6)=ov2;
rules(84,1)=15;
```

```
rules(84,2)=1;
rules(84,3)=1;
rules(84,4)=0;
rules(84,5)=3;
rules(84,6)=ov2;
rules(85,1)=13;
rules(85,2)=1;
rules(85,3)=1;
rules(85,4)=1;
rules(85,5)=0;
rules(85,6)=ov0;
rules(86,1)=14;
rules(86,2)=1;
rules(86,3)=1;
rules(86,4)=1;
rules(86,5)=1;
rules(86,6)=ov2;
rules(87,1)=15;
rules(87,2)=1;
rules(87,3)=1;
rules(87,4)=1;
rules(87,5)=2;
rules(87,6)=ov2;
rules(88,1)=16;
rules(88,2)=1;
rules(88,3)=1;
rules(88,4)=1;
rules(88,5)=3;
rules(88,6)=ov2;
rules(89,1)=14;
rules(89,2)=1;
rules(89,3)=1;
rules(89,4)=2;
rules(89,5)=0;
rules(89,6)=ov1;
rules(90,1)=15;
rules(90,2)=1;
rules(90,3)=1;
rules(90,4)=2;
rules(90,5)=1;
rules(90,6)=ov2;
rules(91,1)=16;
rules(91,2)=1;
rules(91,3)=1;
rules(91,4)=2;
rules(91,5)=2;
rules(91,6)=ov2;
rules(92,1)=17;
rules(92,2)=1;
rules(92,3)=1;
rules(92,4)=2;
rules(92,5)=3;
rules(92,6)=ov2;
rules(93,1)=15;
rules(93,2)=1;
rules(93,3)=1;
rules(93,4)=3;
```

```
rules(93,5)=0;
rules(93,6)=ov1;
rules(94,1)=16;
rules(94,2)=1;
rules(94,3)=1;
rules(94,4)=3;
rules(94,5)=1;
rules(94,6)=ov2;
rules(95,1)=17;
rules(95,2)=1;
rules(95,3)=1;
rules(95,4)=3;
rules(95,5)=2;
rules(95,6)=ov2;
rules(96,1)=18;
rules(96,2)=1;
rules(96,3)=1;
rules(96,4)=3;
rules(96,5)=3;
rules(96,6)=ov2;
rules(97,1)=13;
rules(97,2)=1;
rules(97,3)=2;
rules(97,4)=0;
rules(97,5)=0;
rules(97,6)=ov0;
rules(98,1)=14;
rules(98,2)=1;
rules(98,3)=2;
rules(98,4)=0;
rules(98,5)=1;
rules(98,6)=ov0;
rules(99,1)=15;
rules(99,2)=1;
rules(99,3)=2;
rules(99,4)=0;
rules(99,5)=2;
rules(99,6)=ov2;
rules(100,1)=16;
rules(100,2)=1;
rules(100,3)=2;
rules(100,4)=0;
rules(100,5)=3;
rules(100,6)=ov2;
rules(101,1)=14;
rules(101,2)=1;
rules(101,3)=2;
rules(101,4)=1;
rules(101,5)=0;
rules(101,6)=ov0;
rules(102,1)=15;
rules(102,2)=1;
rules(102,3)=2;
rules(102,4)=1;
rules(102,5)=1;
rules(102,6)=ov2;
rules(103,1)=16;
```

```
rules(103,2)=1;
rules(103,3)=2;
rules(103,4)=1;
rules(103,5)=2;
rules(103,6)=ov2;
rules(104,1)=17;
rules(104,2)=1;
rules(104,3)=2;
rules(104,4)=1;
rules(104,5)=3;
rules(104,6)=ov2;
rules(105,1)=15;
rules(105,2)=1;
rules(105,3)=2;
rules(105,4)=2;
rules(105,5)=0;
rules(105,6)=ov1;
rules(106,1)=16;
rules(106,2)=1;
rules(106,3)=2;
rules(106,4)=2;
rules(106,5)=1;
rules(106,6)=ov2;
rules(107,1)=17;
rules(107,2)=1;
rules(107,3)=2;
rules(107,4)=2;
rules(107,5)=2;
rules(107,6)=ov2;
rules(108,1)=18;
rules(108,2)=1;
rules(108,3)=2;
rules(108,4)=2;
rules(108,5)=3;
rules(108,6)=ov2;
rules(109,1)=16;
rules(109,2)=1;
rules(109,3)=2;
rules(109,4)=3;
rules(109,5)=0;
rules(109,6)=ov1;
rules(110,1)=17;
rules(110,2)=1;
rules(110,3)=2;
rules(110,4)=3;
rules(110,5)=1;
rules(110,6)=ov2;
rules(111,1)=18;
rules(111,2)=1;
rules(111,3)=2;
rules(111,4)=3;
rules(111,5)=2;
rules(111,6)=ov2;
rules(112,1)=19;
rules(112,2)=1;
rules(112,3)=2;
rules(112,4)=3;
```

```
rules(112,5)=3;
rules(112,6)=ov2;
rules(113,1)=14;
rules(113,2)=1;
rules(113,3)=3;
rules(113,4)=0;
rules(113,5)=0;
rules(113,6)=ov0;
rules(114,1)=15;
rules(114,2)=1;
rules(114,3)=3;
rules(114,4)=0;
rules(114,5)=1;
rules(114,6)=ov0;
rules(115,1)=16;
rules(115,2)=1;
rules(115,3)=3;
rules(115,4)=0;
rules(115,5)=2;
rules(115,6)=ov2;
rules(116,1)=17;
rules(116,2)=1;
rules(116,3)=3;
rules(116,4)=0;
rules(116,5)=3;
rules(116,6)=ov2;
rules(117,1)=15;
rules(117,2)=1;
rules(117,3)=3;
rules(117,4)=1;
rules(117,5)=0;
rules(117,6)=ov0;
rules(118,1)=16;
rules(118,2)=1;
rules(118,3)=3;
rules(118,4)=1;
rules(118,5)=1;
rules(118,6)=ov2;
rules(119,1)=17;
rules(119,2)=1;
rules(119,3)=3;
rules(119,4)=1;
rules(119,5)=2;
rules(119,6)=ov2;
rules(120,1)=18;
rules(120,2)=1;
rules(120,3)=3;
rules(120,4)=1;
rules(120,5)=3;
rules(120,6)=ov2;
rules(121,1)=16;
rules(121,2)=1;
rules(121,3)=3;
rules(121,4)=2;
rules(121,5)=0;
rules(121,6)=ov1;
rules(122,1)=17;
```

```
rules(122,2)=1;
rules(122,3)=3;
rules(122,4)=2;
rules(122,5)=1;
rules(122,6)=ov2;
rules(123,1)=18;
rules(123,2)=1;
rules(123,3)=3;
rules(123,4)=2;
rules(123,5)=2;
rules(123,6)=ov2;
rules(124,1)=19;
rules(124,2)=1;
rules(124,3)=3;
rules(124,4)=2;
rules(124,5)=3;
rules(124,6)=ov2;
rules(125,1)=17;
rules(125,2)=1;
rules(125,3)=3;
rules(125,4)=3;
rules(125,5)=0;
rules(125,6)=ov1;
rules(126,1)=18;
rules(126,2)=1;
rules(126,3)=3;
rules(126,4)=3;
rules(126,5)=1;
rules(126,6)=ov2;
rules(127,1)=19;
rules(127,2)=1;
rules(127,3)=3;
rules(127,4)=3;
rules(127,5)=2;
rules(127,6)=ov2;
rules(128,1)=20;
rules(128,2)=1;
rules(128,3)=3;
rules(128,4)=3;
rules(128,5)=3;
rules(128,6)=ov2;
rules(129,1)=12;
rules(129,2)=2;
rules(129,3)=0;
rules(129,4)=0;
rules(129,5)=0;
rules(129,6)=ov0;
rules(130,1)=13;
rules(130,2)=2;
rules(130,3)=0;
rules(130,4)=0;
rules(130,5)=1;
rules(130,6)=ov0;
rules(131,1)=14;
rules(131,2)=2;
rules(131,3)=0;
rules(131,4)=0;
```

```
rules(131,5)=2;
rules(131,6)=ov2;
rules(132,1)=15;
rules(132,2)=2;
rules(132,3)=0;
rules(132,4)=0;
rules(132,5)=3;
rules(132,6)=ov2;
rules(133,1)=13;
rules(133,2)=2;
rules(133,3)=0;
rules(133,4)=1;
rules(133,5)=0;
rules(133,6)=ov0;
rules(134,1)=14;
rules(134,2)=2;
rules(134,3)=0;
rules(134,4)=1;
rules(134,5)=1;
rules(134,6)=ov2;
rules(135,1)=15;
rules(135,2)=2;
rules(135,3)=0;
rules(135,4)=1;
rules(135,5)=2;
rules(135,6)=ov2;
rules(136,1)=16;
rules(136,2)=2;
rules(136,3)=0;
rules(136,4)=1;
rules(136,5)=3;
rules(136,6)=ov2;
rules(137,1)=14;
rules(137,2)=2;
rules(137,3)=0;
rules(137,4)=2;
rules(137,5)=0;
rules(137,6)=ov1;
rules(138,1)=15;
rules(138,2)=2;
rules(138,3)=0;
rules(138,4)=2;
rules(138,5)=1;
rules(138,6)=ov2;
rules(139,1)=16;
rules(139,2)=2;
rules(139,3)=0;
rules(139,4)=2;
rules(139,5)=2;
rules(139,6)=ov2;
rules(140,1)=17;
rules(140,2)=2;
rules(140,3)=0;
rules(140,4)=2;
rules(140,5)=3;
rules(140,6)=ov2;
rules(141,1)=15;
```

```
rules(141,2)=2;
rules(141,3)=0;
rules(141,4)=3;
rules(141,5)=0;
rules(141,6)=ov1;
rules(142,1)=16;
rules(142,2)=2;
rules(142,3)=0;
rules(142,4)=3;
rules(142,5)=1;
rules(142,6)=ov2;
rules(143,1)=17;
rules(143,2)=2;
rules(143,3)=0;
rules(143,4)=3;
rules(143,5)=2;
rules(143,6)=ov2;
rules(144,1)=18;
rules(144,2)=2;
rules(144,3)=0;
rules(144,4)=3;
rules(144,5)=3;
rules(144,6)=ov2;
rules(145,1)=13;
rules(145,2)=2;
rules(145,3)=1;
rules(145,4)=0;
rules(145,5)=0;
rules(145,6)=ov0;
rules(146,1)=14;
rules(146,2)=2;
rules(146,3)=1;
rules(146,4)=0;
rules(146,5)=1;
rules(146,6)=ov0;
rules(147,1)=15;
rules(147,2)=2;
rules(147,3)=1;
rules(147,4)=0;
rules(147,5)=2;
rules(147,6)=ov2;
rules(148,1)=16;
rules(148,2)=2;
rules(148,3)=1;
rules(148,4)=0;
rules(148,5)=3;
rules(148,6)=ov2;
rules(149,1)=14;
rules(149,2)=2;
rules(149,3)=1;
rules(149,4)=1;
rules(149,5)=0;
rules(149,6)=ov0;
rules(150,1)=15;
rules(150,2)=2;
rules(150,3)=1;
rules(150,4)=1;
```

```
rules(150,5)=1;
rules(150,6)=ov2;
rules(151,1)=16;
rules(151,2)=2;
rules(151,3)=1;
rules(151,4)=1;
rules(151,5)=2;
rules(151,6)=ov2;
rules(152,1)=17;
rules(152,2)=2;
rules(152,3)=1;
rules(152,4)=1;
rules(152,5)=3;
rules(152,6)=ov2;
rules(153,1)=15;
rules(153,2)=2;
rules(153,3)=1;
rules(153,4)=2;
rules(153,5)=0;
rules(153,6)=ov1;
rules(154,1)=16;
rules(154,2)=2;
rules(154,3)=1;
rules(154,4)=2;
rules(154,5)=1;
rules(154,6)=ov2;
rules(155,1)=17;
rules(155,2)=2;
rules(155,3)=1;
rules(155,4)=2;
rules(155,5)=2;
rules(155,6)=ov2;
rules(156,1)=18;
rules(156,2)=2;
rules(156,3)=1;
rules(156,4)=2;
rules(156,5)=3;
rules(156,6)=ov2;
rules(157,1)=16;
rules(157,2)=2;
rules(157,3)=1;
rules(157,4)=3;
rules(157,5)=0;
rules(157,6)=ov1;
rules(158,1)=17;
rules(158,2)=2;
rules(158,3)=1;
rules(158,4)=3;
rules(158,5)=1;
rules(158,6)=ov2;
rules(159,1)=18;
rules(159,2)=2;
rules(159,3)=1;
rules(159,4)=3;
rules(159,5)=2;
rules(159,6)=ov2;
rules(160,1)=19;
```

```
rules(160,2)=2;
rules(160,3)=1;
rules(160,4)=3;
rules(160,5)=3;
rules(160,6)=ov2;
rules(161,1)=14;
rules(161,2)=2;
rules(161,3)=2;
rules(161,4)=0;
rules(161,5)=0;
rules(161,6)=ov0;
rules(162,1)=15;
rules(162,2)=2;
rules(162,3)=2;
rules(162,4)=0;
rules(162,5)=1;
rules(162,6)=ov0;
rules(163,1)=16;
rules(163,2)=2;
rules(163,3)=2;
rules(163,4)=0;
rules(163,5)=2;
rules(163,6)=ov2;
rules(164,1)=17;
rules(164,2)=2;
rules(164,3)=2;
rules(164,4)=0;
rules(164,5)=3;
rules(164,6)=ov2;
rules(165,1)=15;
rules(165,2)=2;
rules(165,3)=2;
rules(165,4)=1;
rules(165,5)=0;
rules(165,6)=ov0;
rules(166,1)=16;
rules(166,2)=2;
rules(166,3)=2;
rules(166,4)=1;
rules(166,5)=1;
rules(166,6)=ov2;
rules(167,1)=17;
rules(167,2)=2;
rules(167,3)=2;
rules(167,4)=1;
rules(167,5)=2;
rules(167,6)=ov2;
rules(168,1)=18;
rules(168,2)=2;
rules(168,3)=2;
rules(168,4)=1;
rules(168,5)=3;
rules(168,6)=ov2;
rules(169,1)=16;
rules(169,2)=2;
rules(169,3)=2;
rules(169,4)=2;
```

```
rules(169,5)=0;
rules(169,6)=ov1;
rules(170,1)=17;
rules(170,2)=2;
rules(170,3)=2;
rules(170,4)=2;
rules(170,5)=1;
rules(170,6)=ov2;
rules(171,1)=18;
rules(171,2)=2;
rules(171,3)=2;
rules(171,4)=2;
rules(171,5)=2;
rules(171,6)=ov2;
rules(172,1)=19;
rules(172,2)=2;
rules(172,3)=2;
rules(172,4)=2;
rules(172,5)=3;
rules(172,6)=ov2;
rules(173,1)=17;
rules(173,2)=2;
rules(173,3)=2;
rules(173,4)=3;
rules(173,5)=0;
rules(173,6)=ov1;
rules(174,1)=18;
rules(174,2)=2;
rules(174,3)=2;
rules(174,4)=3;
rules(174,5)=1;
rules(174,6)=ov2;
rules(175,1)=19;
rules(175,2)=2;
rules(175,3)=2;
rules(175,4)=3;
rules(175,5)=2;
rules(175,6)=ov2;
rules(176,1)=20;
rules(176,2)=2;
rules(176,3)=2;
rules(176,4)=3;
rules(176,5)=3;
rules(176,6)=ov2;
rules(177,1)=15;
rules(177,2)=2;
rules(177,3)=3;
rules(177,4)=0;
rules(177,5)=0;
rules(177,6)=ov2;
rules(178,1)=16;
rules(178,2)=2;
rules(178,3)=3;
rules(178,4)=0;
rules(178,5)=1;
rules(178,6)=ov2;
rules(179,1)=17;
```

```
rules(179,2)=2;
rules(179,3)=3;
rules(179,4)=0;
rules(179,5)=2;
rules(179,6)=ov2;
rules(180,1)=18;
rules(180,2)=2;
rules(180,3)=3;
rules(180,4)=0;
rules(180,5)=3;
rules(180,6)=ov2;
rules(181,1)=16;
rules(181,2)=2;
rules(181,3)=3;
rules(181,4)=1;
rules(181,5)=0;
rules(181,6)=ov2;
rules(182,1)=17;
rules(182,2)=2;
rules(182,3)=3;
rules(182,4)=1;
rules(182,5)=1;
rules(182,6)=ov2;
rules(183,1)=18;
rules(183,2)=2;
rules(183,3)=3;
rules(183,4)=1;
rules(183,5)=2;
rules(183,6)=ov2;
rules(184,1)=19;
rules(184,2)=2;
rules(184,3)=3;
rules(184,4)=1;
rules(184,5)=3;
rules(184,6)=ov2;
rules(185,1)=17;
rules(185,2)=2;
rules(185,3)=3;
rules(185,4)=2;
rules(185,5)=0;
rules(185,6)=ov2;
rules(186,1)=18;
rules(186,2)=2;
rules(186,3)=3;
rules(186,4)=2;
rules(186,5)=1;
rules(186,6)=ov2;
rules(187,1)=19;
rules(187,2)=2;
rules(187,3)=3;
rules(187,4)=2;
rules(187,5)=2;
rules(187,6)=ov2;
rules(188,1)=20;
rules(188,2)=2;
rules(188,3)=3;
rules(188,4)=2;
```

```
rules(188,5)=3;
rules(188,6)=ov2;
rules(189,1)=18;
rules(189,2)=2;
rules(189,3)=3;
rules(189,4)=3;
rules(189,5)=0;
rules(189,6)=ov2;
rules(190,1)=19;
rules(190,2)=2;
rules(190,3)=3;
rules(190,4)=3;
rules(190,5)=1;
rules(190,6)=ov2;
rules(191,1)=20;
rules(191,2)=2;
rules(191,3)=3;
rules(191,4)=3;
rules(191,5)=2;
rules(191,6)=ov2;
rules(192,1)=21;
rules(192,2)=2;
rules(192,3)=3;
rules(192,4)=3;
rules(192,5)=3;
rules(192,6)=ov2;
rules(193,1)=13;
rules(193,2)=3;
rules(193,3)=0;
rules(193,4)=0;
rules(193,5)=0;
rules(193,6)=ov1;
rules(194,1)=14;
rules(194,2)=3;
rules(194,3)=0;
rules(194,4)=0;
rules(194,5)=1;
rules(194,6)=ov2;
rules(195,1)=15;
rules(195,2)=3;
rules(195,3)=0;
rules(195,4)=0;
rules(195,5)=2;
rules(195,6)=ov2;
rules(196,1)=16;
rules(196,2)=3;
rules(196,3)=0;
rules(196,4)=0;
rules(196,5)=3;
rules(196,6)=ov2;
rules(197,1)=14;
rules(197,2)=3;
rules(197,3)=0;
rules(197,4)=1;
rules(197,5)=0;
rules(197,6)=ov2;
rules(198,1)=15;
```

```
rules(198,2)=3;
rules(198,3)=0;
rules(198,4)=1;
rules(198,5)=1;
rules(198,6)=ov2;
rules(199,1)=16;
rules(199,2)=3;
rules(199,3)=0;
rules(199,4)=1;
rules(199,5)=2;
rules(199,6)=ov2;
rules(200,1)=17;
rules(200,2)=3;
rules(200,3)=0;
rules(200,4)=1;
rules(200,5)=3;
rules(200,6)=ov2;
rules(201,1)=15;
rules(201,2)=3;
rules(201,3)=0;
rules(201,4)=2;
rules(201,5)=0;
rules(201,6)=ov2;
rules(202,1)=16;
rules(202,2)=3;
rules(202,3)=0;
rules(202,4)=2;
rules(202,5)=1;
rules(202,6)=ov2;
rules(203,1)=17;
rules(203,2)=3;
rules(203,3)=0;
rules(203,4)=2;
rules(203,5)=2;
rules(203,6)=ov2;
rules(204,1)=18;
rules(204,2)=3;
rules(204,3)=0;
rules(204,4)=2;
rules(204,5)=3;
rules(204,6)=ov2;
rules(205,1)=16;
rules(205,2)=3;
rules(205,3)=0;
rules(205,4)=3;
rules(205,5)=0;
rules(205,6)=ov2;
rules(206,1)=17;
rules(206,2)=3;
rules(206,3)=0;
rules(206,4)=3;
rules(206,5)=1;
rules(206,6)=ov2;
rules(207,1)=18;
rules(207,2)=3;
rules(207,3)=0;
rules(207,4)=3;
```

```
rules(207,5)=2;
rules(207,6)=ov2;
rules(208,1)=19;
rules(208,2)=3;
rules(208,3)=0;
rules(208,4)=3;
rules(208,5)=3;
rules(208,6)=ov2;
rules(209,1)=14;
rules(209,2)=3;
rules(209,3)=1;
rules(209,4)=0;
rules(209,5)=0;
rules(209,6)=ov2;
rules(210,1)=15;
rules(210,2)=3;
rules(210,3)=1;
rules(210,4)=0;
rules(210,5)=1;
rules(210,6)=ov2;
rules(211,1)=16;
rules(211,2)=3;
rules(211,3)=1;
rules(211,4)=0;
rules(211,5)=2;
rules(211,6)=ov2;
rules(212,1)=17;
rules(212,2)=3;
rules(212,3)=1;
rules(212,4)=0;
rules(212,5)=3;
rules(212,6)=ov2;
rules(213,1)=15;
rules(213,2)=3;
rules(213,3)=1;
rules(213,4)=1;
rules(213,5)=0;
rules(213,6)=ov2;
rules(214,1)=16;
rules(214,2)=3;
rules(214,3)=1;
rules(214,4)=1;
rules(214,5)=1;
rules(214,6)=ov2;
rules(215,1)=17;
rules(215,2)=3;
rules(215,3)=1;
rules(215,4)=1;
rules(215,5)=2;
rules(215,6)=ov2;
rules(216,1)=18;
rules(216,2)=3;
rules(216,3)=1;
rules(216,4)=1;
rules(216,5)=3;
rules(216,6)=ov2;
rules(217,1)=16;
```

```
rules(217,2)=3;
rules(217,3)=1;
rules(217,4)=2;
rules(217,5)=0;
rules(217,6)=ov2;
rules(218,1)=17;
rules(218,2)=3;
rules(218,3)=1;
rules(218,4)=2;
rules(218,5)=1;
rules(218,6)=ov2;
rules(219,1)=18;
rules(219,2)=3;
rules(219,3)=1;
rules(219,4)=2;
rules(219,5)=2;
rules(219,6)=ov2;
rules(220,1)=19;
rules(220,2)=3;
rules(220,3)=1;
rules(220,4)=2;
rules(220,5)=3;
rules(220,6)=ov2;
rules(221,1)=17;
rules(221,2)=3;
rules(221,3)=1;
rules(221,4)=3;
rules(221,5)=0;
rules(221,6)=ov2;
rules(222,1)=18;
rules(222,2)=3;
rules(222,3)=1;
rules(222,4)=3;
rules(222,5)=1;
rules(222,6)=ov2;
rules(223,1)=19;
rules(223,2)=3;
rules(223,3)=1;
rules(223,4)=3;
rules(223,5)=2;
rules(223,6)=ov2;
rules(224,1)=20;
rules(224,2)=3;
rules(224,3)=1;
rules(224,4)=3;
rules(224,5)=3;
rules(224,6)=ov2;
rules(225,1)=15;
rules(225,2)=3;
rules(225,3)=2;
rules(225,4)=0;
rules(225,5)=0;
rules(225,6)=ov2;
rules(226,1)=16;
rules(226,2)=3;
rules(226,3)=2;
rules(226,4)=0;
```

```
rules(226,5)=1;
rules(226,6)=ov2;
rules(227,1)=17;
rules(227,2)=3;
rules(227,3)=2;
rules(227,4)=0;
rules(227,5)=2;
rules(227,6)=ov2;
rules(228,1)=18;
rules(228,2)=3;
rules(228,3)=2;
rules(228,4)=0;
rules(228,5)=3;
rules(228,6)=ov2;
rules(229,1)=16;
rules(229,2)=3;
rules(229,3)=2;
rules(229,4)=1;
rules(229,5)=0;
rules(229,6)=ov2;
rules(230,1)=17;
rules(230,2)=3;
rules(230,3)=2;
rules(230,4)=1;
rules(230,5)=1;
rules(230,6)=ov2;
rules(231,1)=18;
rules(231,2)=3;
rules(231,3)=2;
rules(231,4)=1;
rules(231,5)=2;
rules(231,6)=ov2;
rules(232,1)=19;
rules(232,2)=3;
rules(232,3)=2;
rules(232,4)=1;
rules(232,5)=3;
rules(232,6)=ov2;
rules(233,1)=17;
rules(233,2)=3;
rules(233,3)=2;
rules(233,4)=2;
rules(233,5)=0;
rules(233,6)=ov2;
rules(234,1)=18;
rules(234,2)=3;
rules(234,3)=2;
rules(234,4)=2;
rules(234,5)=1;
rules(234,6)=ov2;
rules(235,1)=19;
rules(235,2)=3;
rules(235,3)=2;
rules(235,4)=2;
rules(235,5)=2;
rules(235,6)=ov2;
rules(236,1)=20;
```

```
rules(236,2)=3;
rules(236,3)=2;
rules(236,4)=2;
rules(236,5)=3;
rules(236,6)=ov2;
rules(237,1)=18;
rules(237,2)=3;
rules(237,3)=2;
rules(237,4)=3;
rules(237,5)=0;
rules(237,6)=ov2;
rules(238,1)=19;
rules(238,2)=3;
rules(238,3)=2;
rules(238,4)=3;
rules(238,5)=1;
rules(238,6)=ov2;
rules(239,1)=20;
rules(239,2)=3;
rules(239,3)=2;
rules(239,4)=3;
rules(239,5)=2;
rules(239,6)=ov2;
rules(240,1)=21;
rules(240,2)=3;
rules(240,3)=2;
rules(240,4)=3;
rules(240,5)=3;
rules(240,6)=ov2;
rules(241,1)=16;
rules(241,2)=3;
rules(241,3)=3;
rules(241,4)=0;
rules(241,5)=0;
rules(241,6)=ov2;
rules(242,1)=17;
rules(242,2)=3;
rules(242,3)=3;
rules(242,4)=0;
rules(242,5)=1;
rules(242,6)=ov2;
rules(243,1)=18;
rules(243,2)=3;
rules(243,3)=3;
rules(243,4)=0;
rules(243,5)=2;
rules(243,6)=ov2;
rules(244,1)=19;
rules(244,2)=3;
rules(244,3)=3;
rules(244,4)=0;
rules(244,5)=3;
rules(244,6)=ov2;
rules(245,1)=17;
rules(245,2)=3;
rules(245,3)=3;
rules(245,4)=1;
```

```
rules(245,5)=0;
rules(245,6)=ov2;
rules(246,1)=18;
rules(246,2)=3;
rules(246,3)=3;
rules(246,4)=1;
rules(246,5)=1;
rules(246,6)=ov2;
rules(247,1)=19;
rules(247,2)=3;
rules(247,3)=3;
rules(247,4)=1;
rules(247,5)=2;
rules(247,6)=ov2;
rules(248,1)=20;
rules(248,2)=3;
rules(248,3)=3;
rules(248,4)=1;
rules(248,5)=3;
rules(248,6)=ov2;
rules(249,1)=18;
rules(249,2)=3;
rules(249,3)=3;
rules(249,4)=2;
rules(249,5)=0;
rules(249,6)=ov2;
rules(250,1)=19;
rules(250,2)=3;
rules(250,3)=3;
rules(250,4)=2;
rules(250,5)=1;
rules(250,6)=ov2;
rules(251,1)=20;
rules(251,2)=3;
rules(251,3)=3;
rules(251,4)=2;
rules(251,5)=2;
rules(251,6)=ov2;
rules(252,1)=21;
rules(252,2)=3;
rules(252,3)=3;
rules(252,4)=2;
rules(252,5)=3;
rules(252,6)=ov2;
rules(253,1)=19;
rules(253,2)=3;
rules(253,3)=3;
rules(253,4)=3;
rules(253,5)=0;
rules(253,6)=ov2;
rules(254,1)=20;
rules(254,2)=3;
rules(254,3)=3;
rules(254,4)=3;
rules(254,5)=1;
rules(254,6)=ov2;
rules(255,1)=21;
```

```
rules(255,2)=3;
rules(255,3)=3;
rules(255,4)=3;
rules(255,5)=2;
rules(255,6)=ov2;
rules(256,1)=22;
rules(256,2)=3;
rules(256,3)=3;
rules(256,4)=3;
rules(256,5)=3;
rules(256,6)=ov2;
for di=1:length(rules(:,1))
   if(all(rules(di,2:5) == in))
      out=rules(di,6);
      %di
   end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

# Appendix D

## Fuzzy Fused Index for Drive End and Fan End Bearings

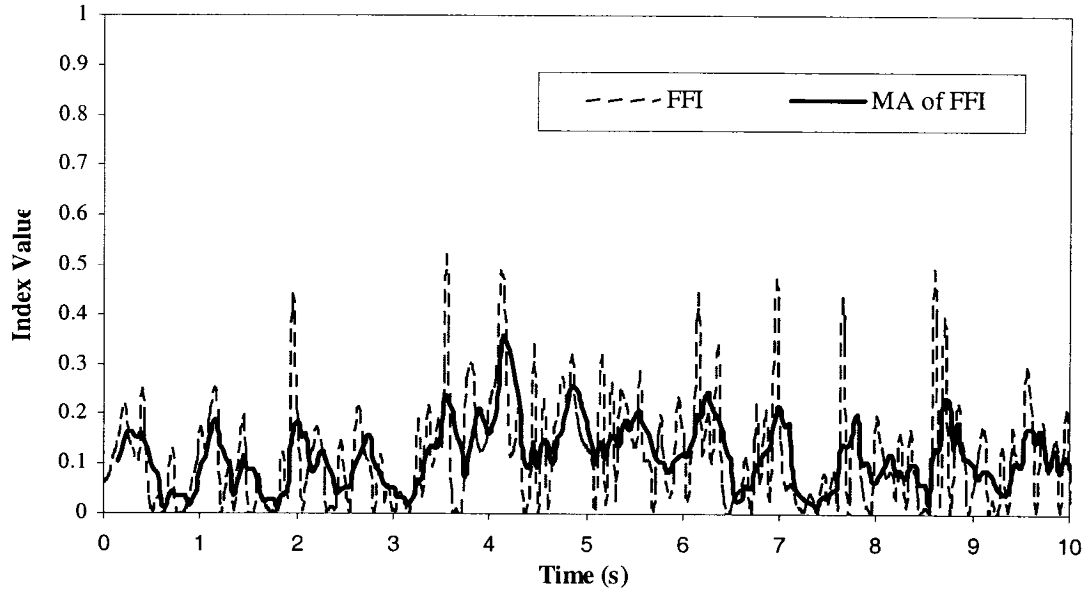



Figure D.1 Fuzzy fused index and its moving average shown for drive end bearing, normal condition at 1797 rpm and 0 hp motor load.

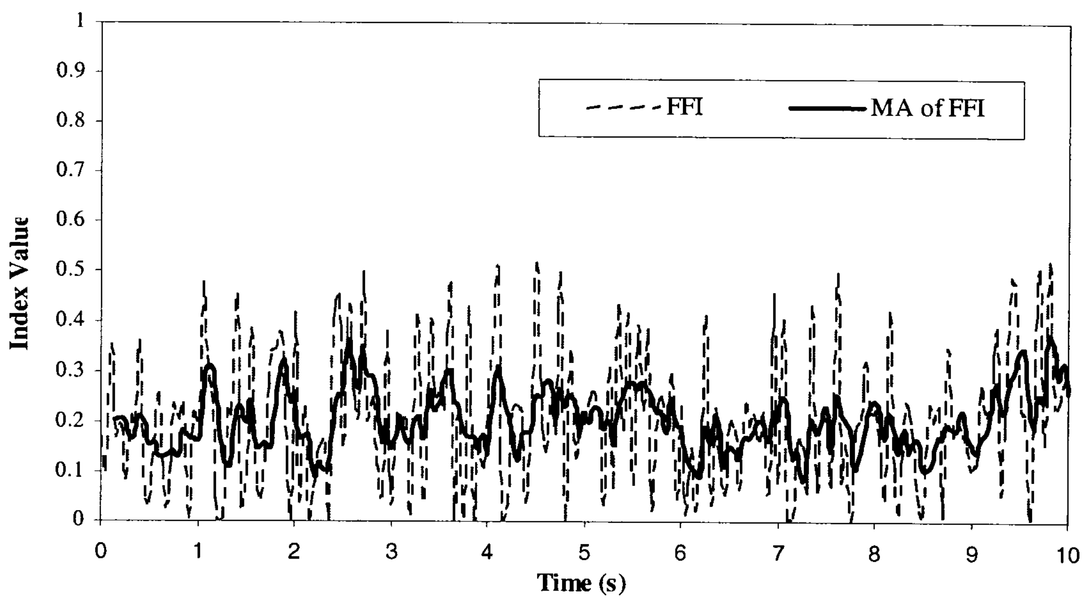



Figure D.2 Fuzzy fused index and its moving average shown for drive end bearing, normal condition at 1772 rpm and 1 hp motor load.

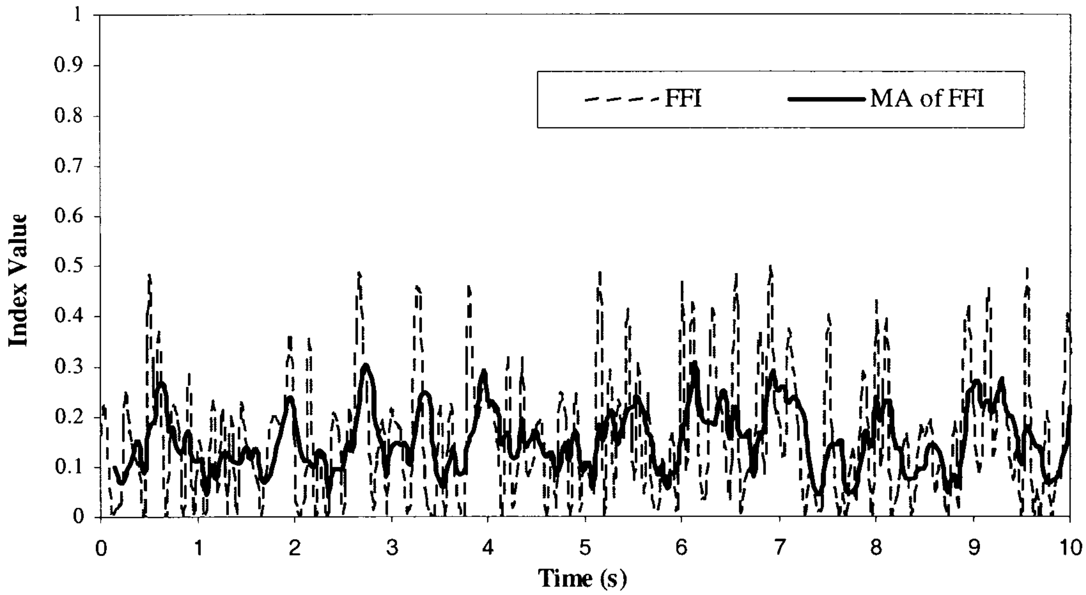



Figure D.3 Fuzzy fused index and its moving average shown for drive end bearing, normal condition at 1750 rpm and 2 hp motor load.

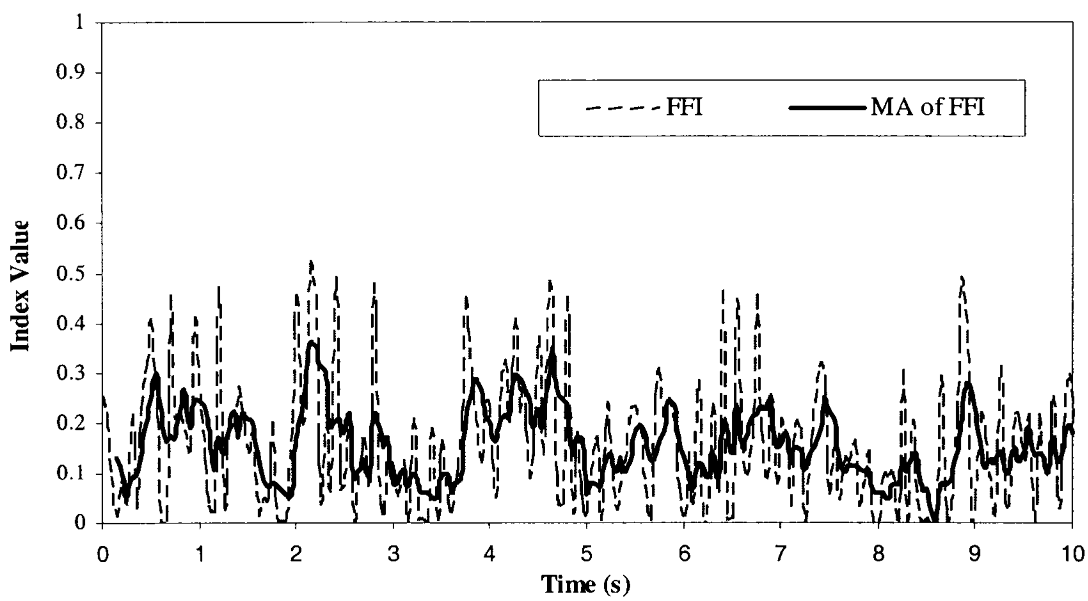



Figure D.4 Fuzzy fused index and its moving average shown for drive end bearing, normal condition at 1730 rpm and 3 hp motor load.

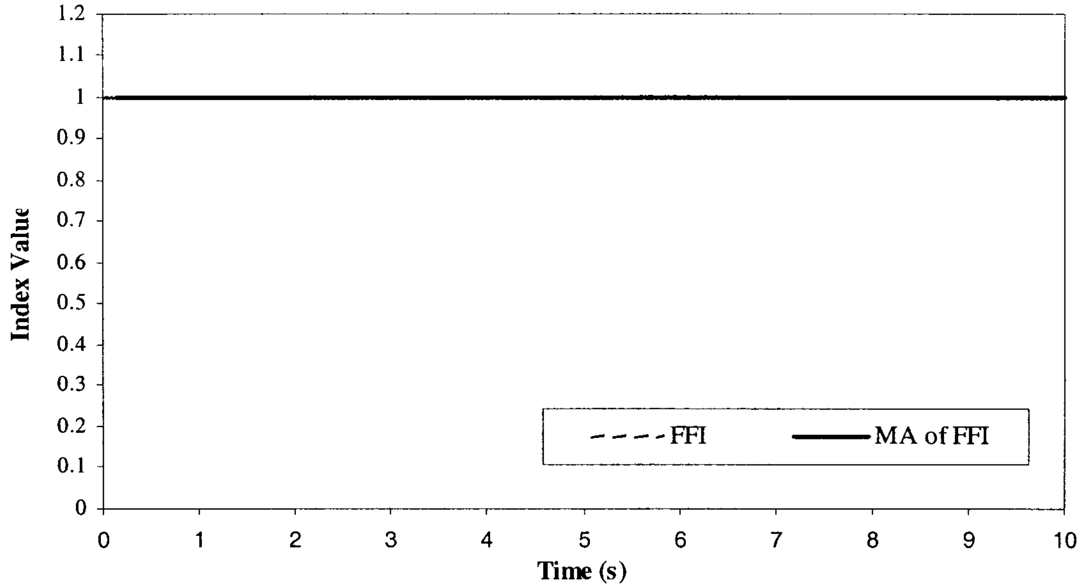



Figure D.5 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at inner race, 1797 rpm and 0 hp motor load.

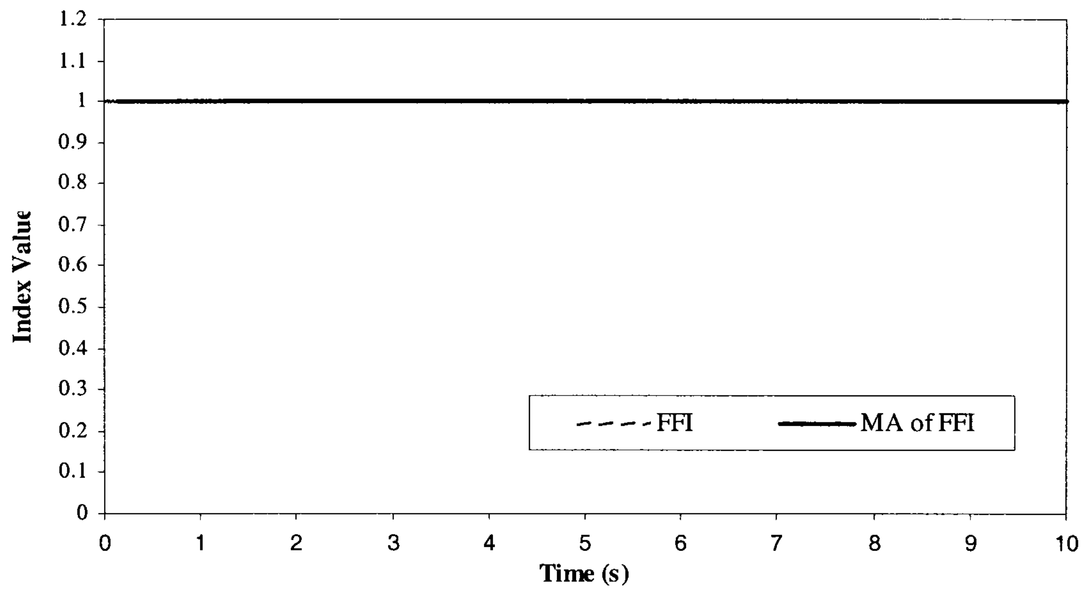



Figure D.6 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at inner race, 1772 rpm and 1 hp motor load.

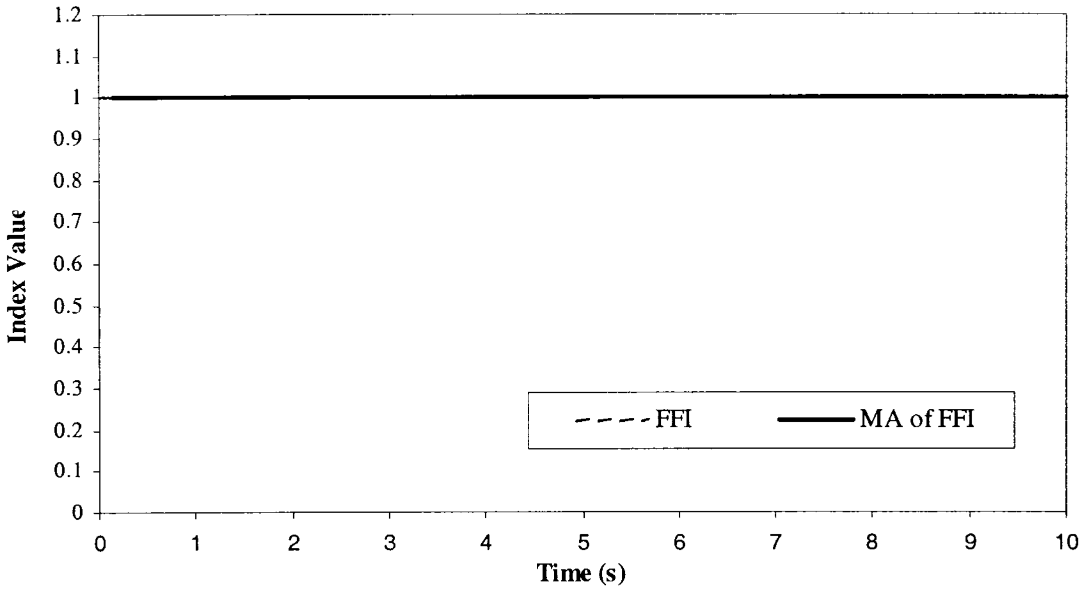



Figure D.7 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at inner race, 1750 rpm and 2 hp motor load.

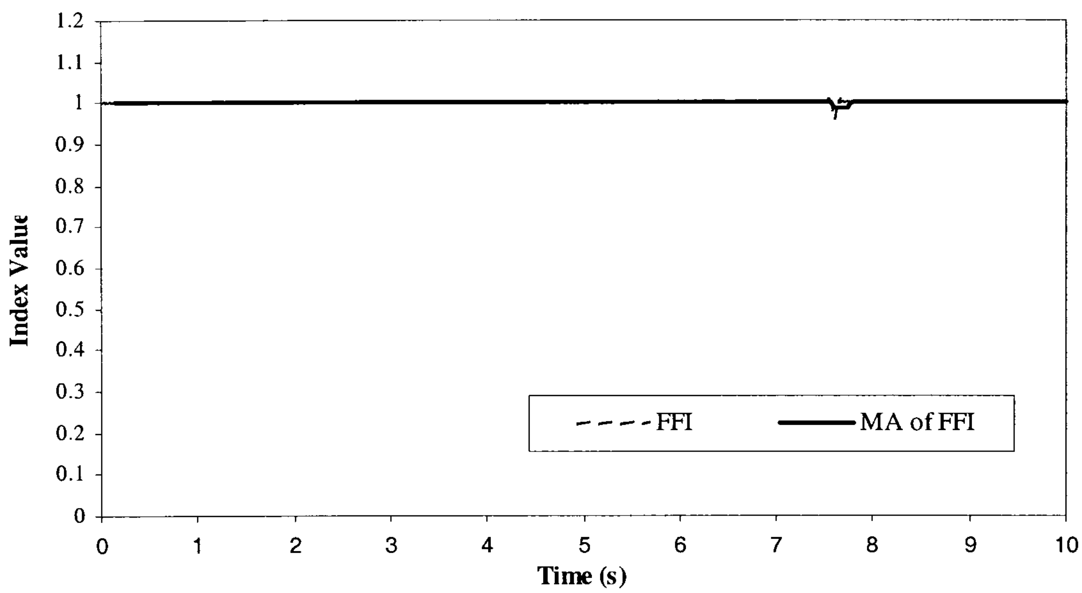



Figure D.8 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at inner race, 1730 rpm and 3 hp motor load.

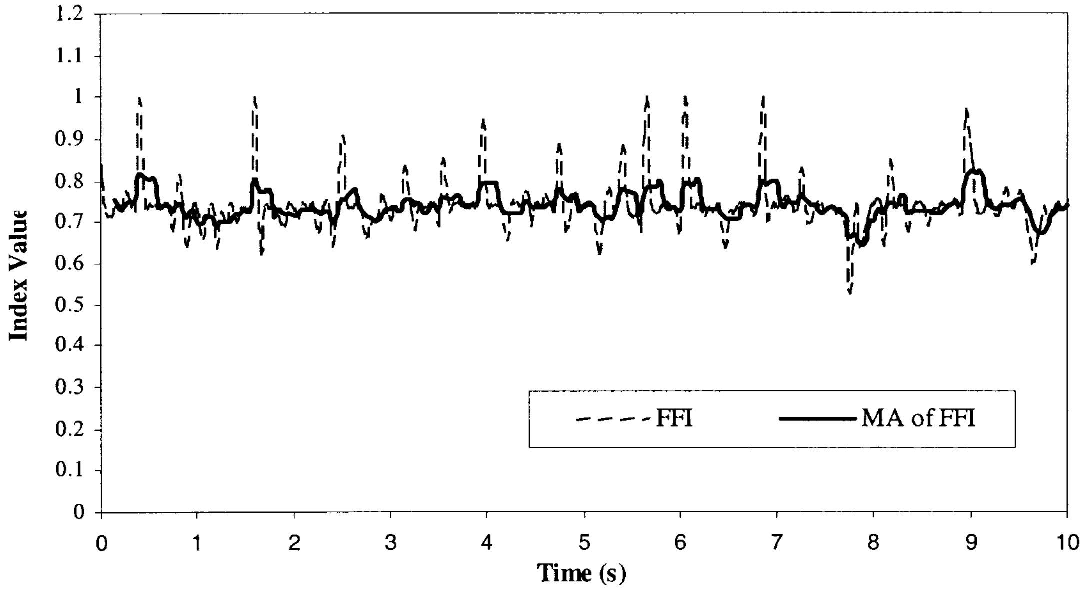



Figure D.9 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at inner race, 1797 rpm and 0 hp motor load.

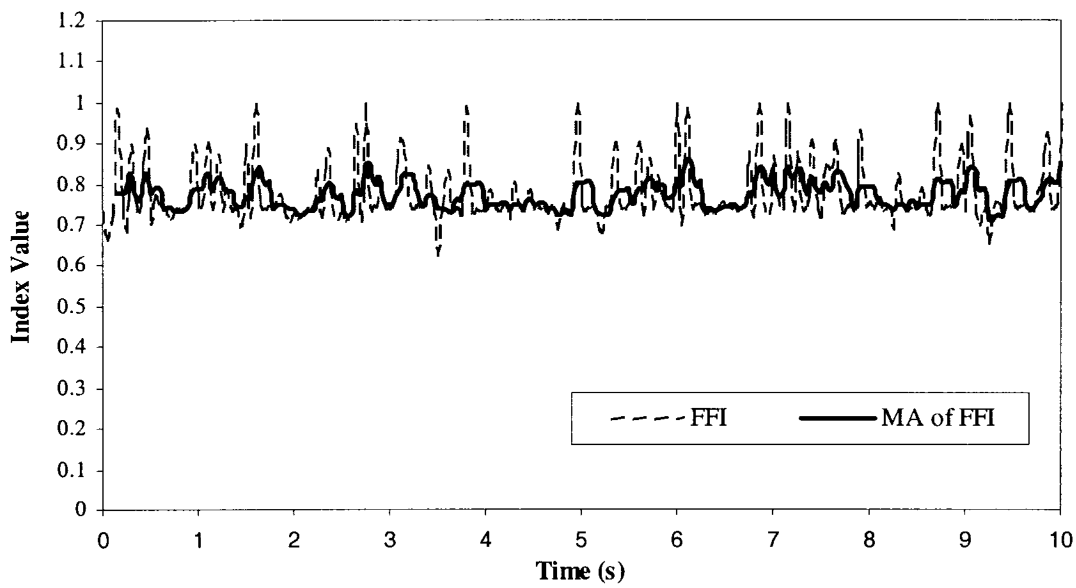



Figure D.10 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at inner race, 1772 rpm and 1 hp motor load.

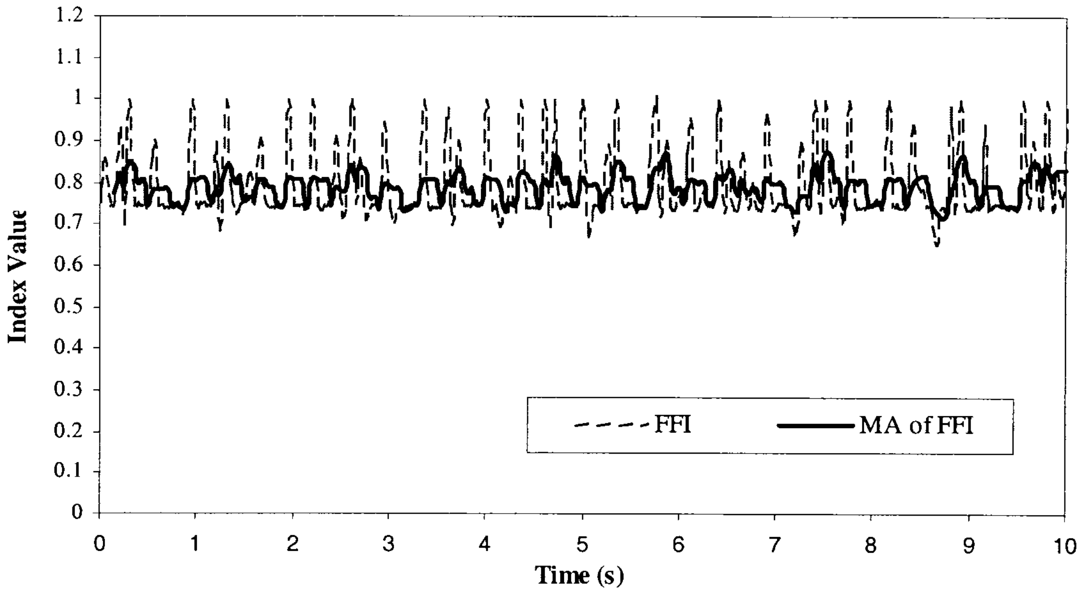



Figure D.11 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at inner race, 1750 rpm and 2 hp motor load.

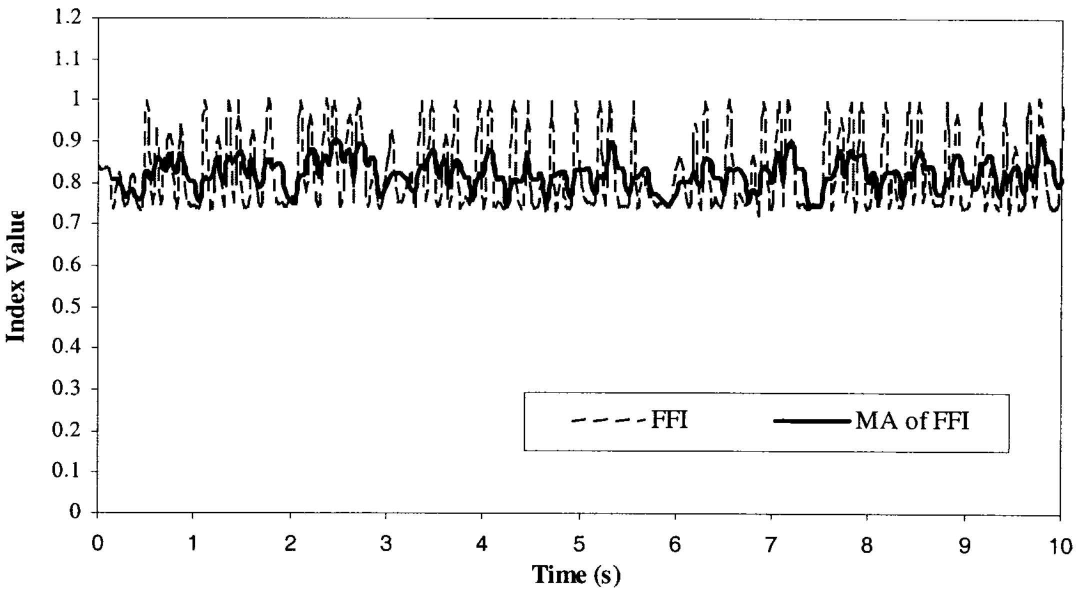



Figure D.12 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at inner race, 1730 rpm and 3 hp motor load.

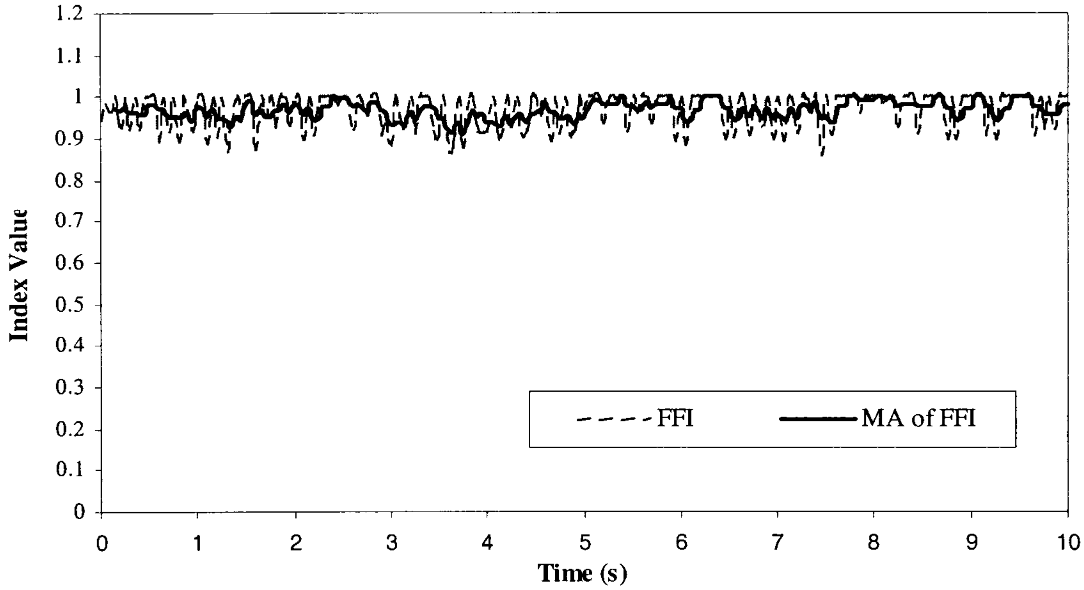



Figure D.13 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at inner race, 1797 rpm and 0 hp motor load.

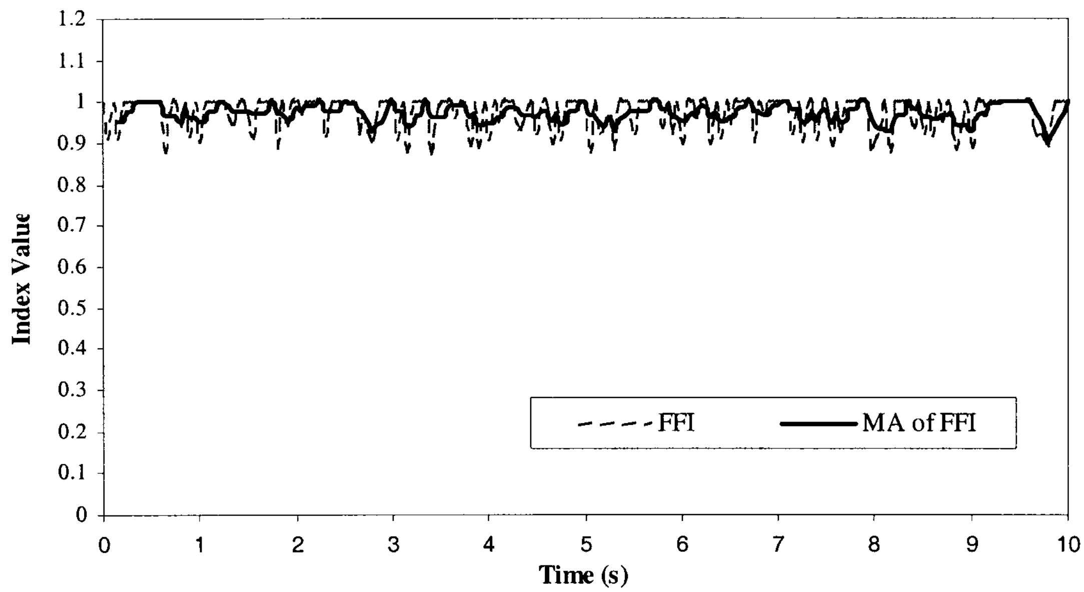



Figure D.14 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at inner race, 1772 rpm and 1 hp motor load.

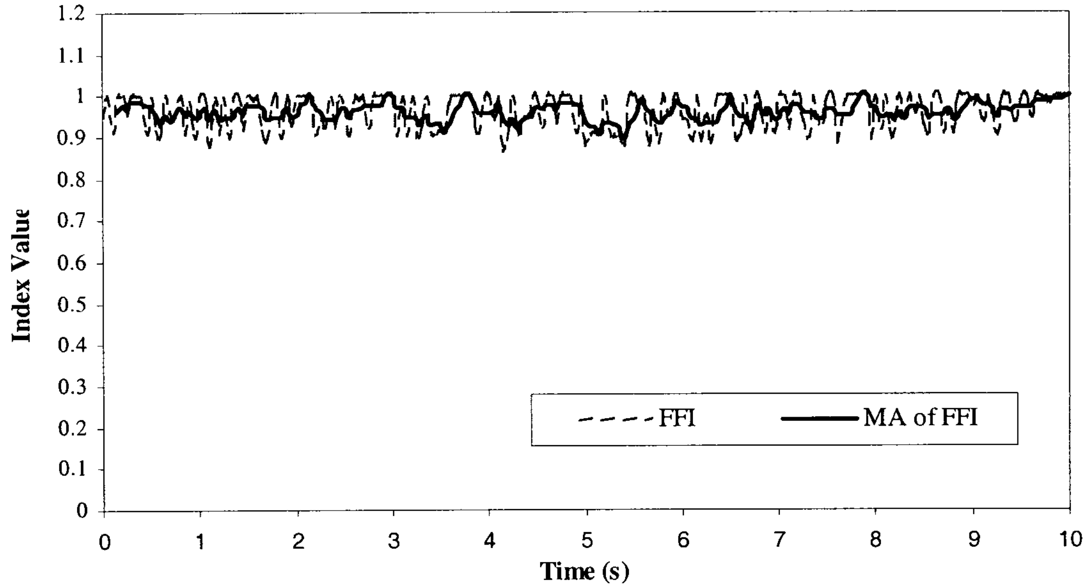



Figure D.15 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at inner race, 1750 rpm and 2 hp motor load.

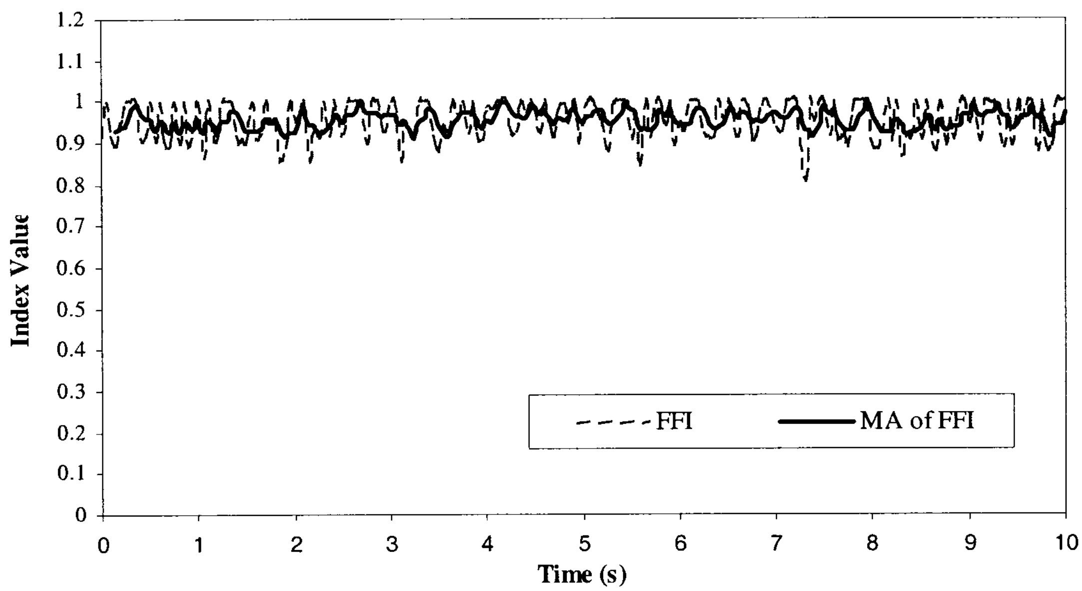



Figure D.16 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at inner race, 1730 rpm and 3 hp motor load.

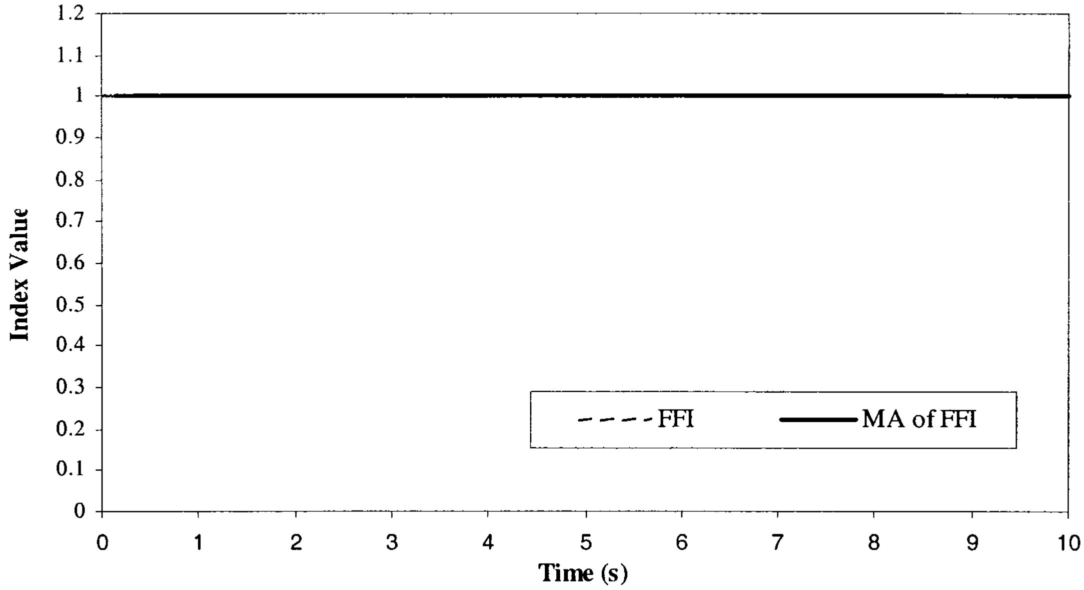



Figure D.17 Fuzzy fused index and its moving average shown for drive end bearing, 0.028" fault diameter at inner race, 1797 rpm and 0 hp motor load.

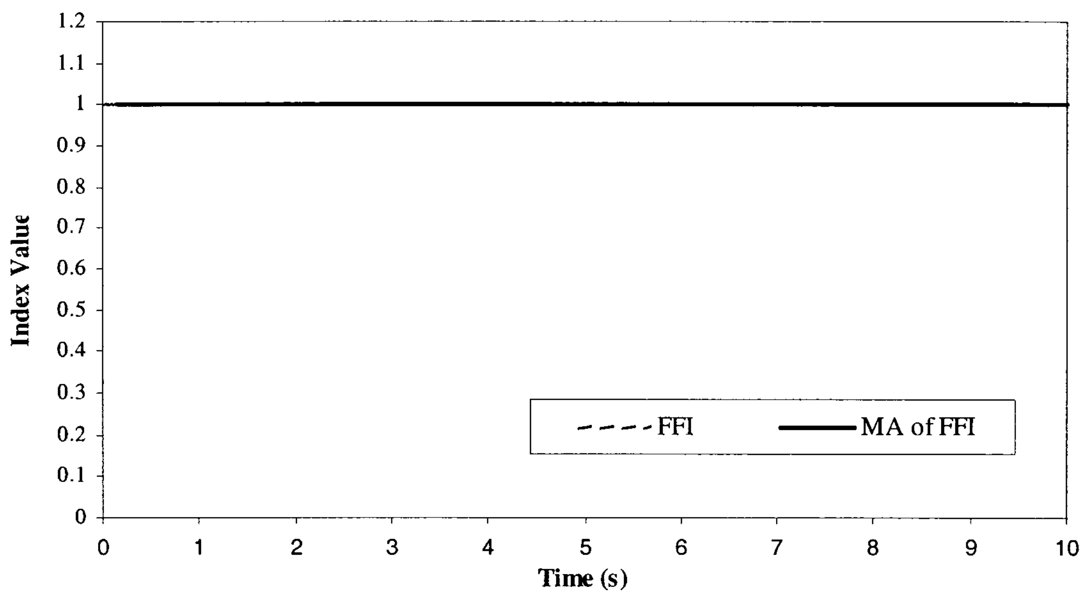



Figure D.18 Fuzzy fused index and its moving average shown for drive end bearing, 0.028" fault diameter at inner race, 1772 rpm and 1 hp motor load.

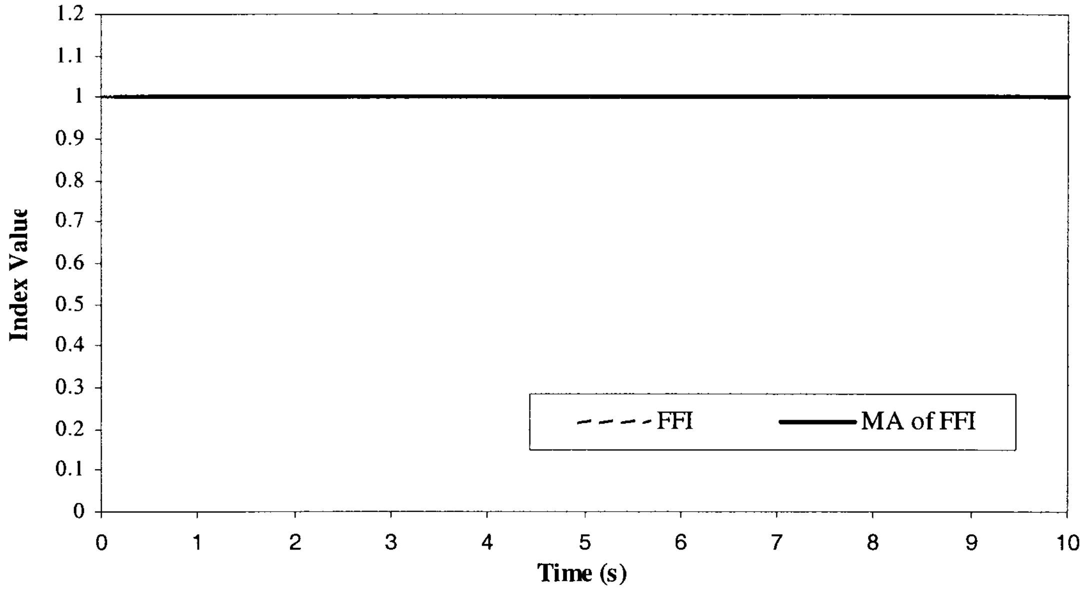



Figure D.19 Fuzzy fused index and its moving average shown for drive end bearing, 0.028" fault diameter at inner race, 1750 rpm and 2 hp motor load.

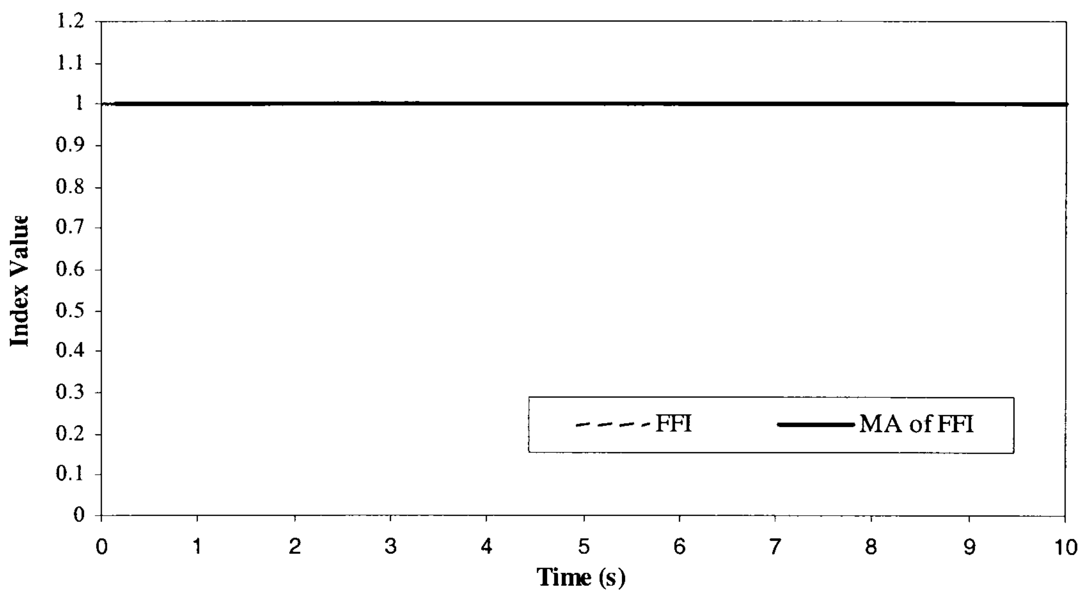



Figure D.20 Fuzzy fused index and its moving average shown for drive end bearing, 0.028" fault diameter at inner race, 1730 rpm and 3 hp motor load.

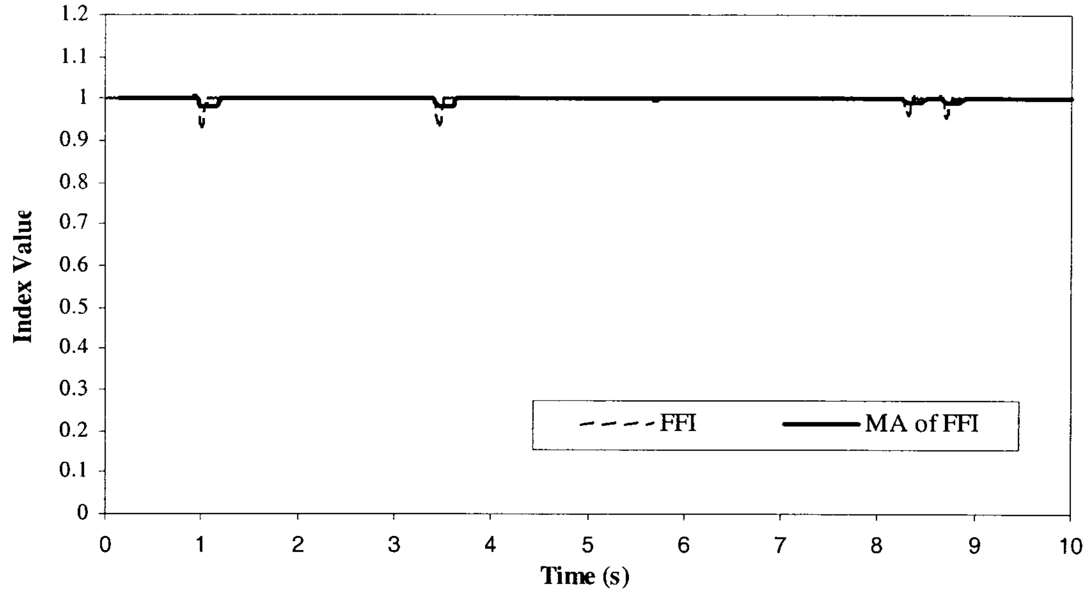



Figure D.21 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at ball, 1797 rpm and 0 hp motor load.

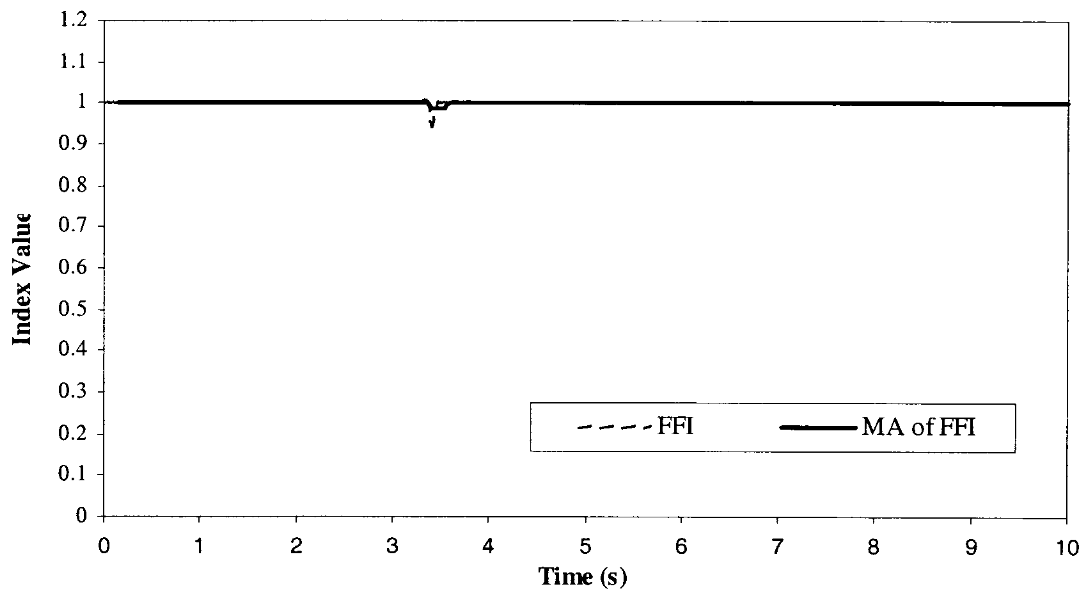



Figure D.22 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at ball, 1772 rpm and 1 hp motor load.

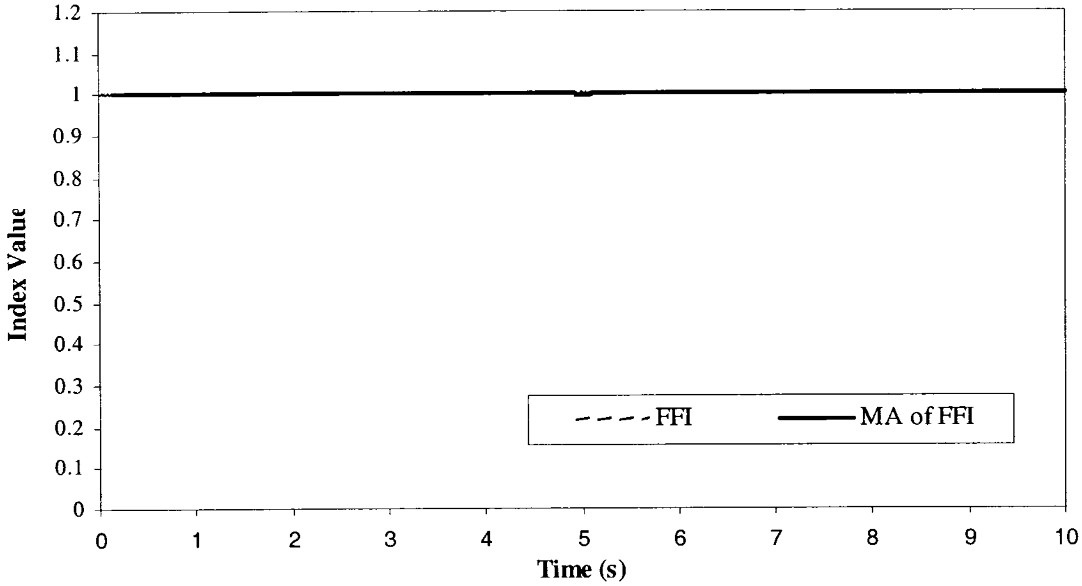



Figure D.23 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at ball, 1750 rpm and 2 hp motor load.

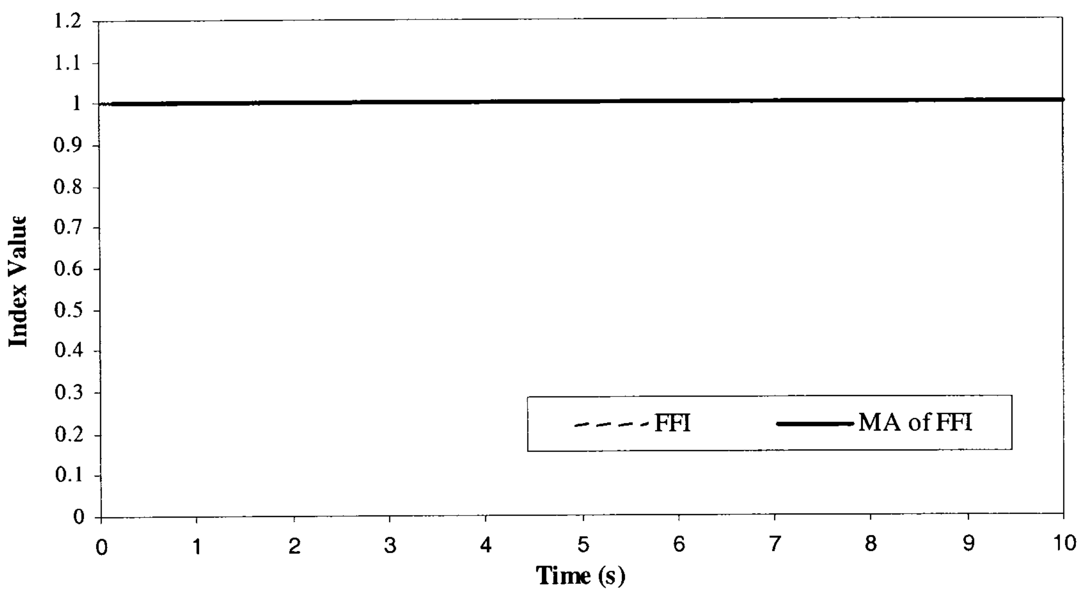



Figure D.24 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at ball, 1730 rpm and 3 hp motor load.

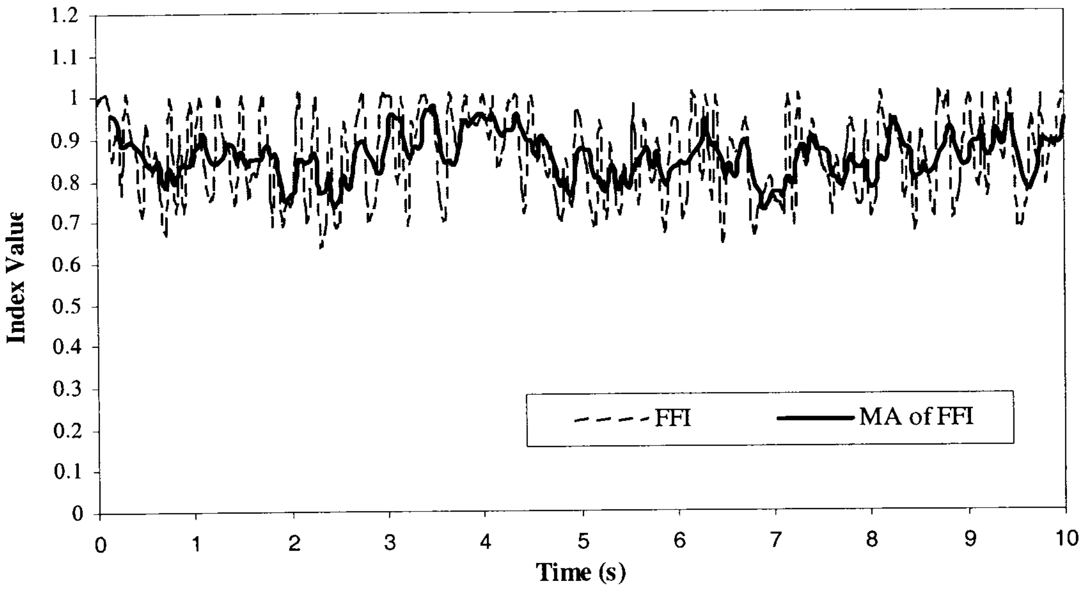



Figure D.25 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at ball, 1797 rpm and 0 hp motor load.

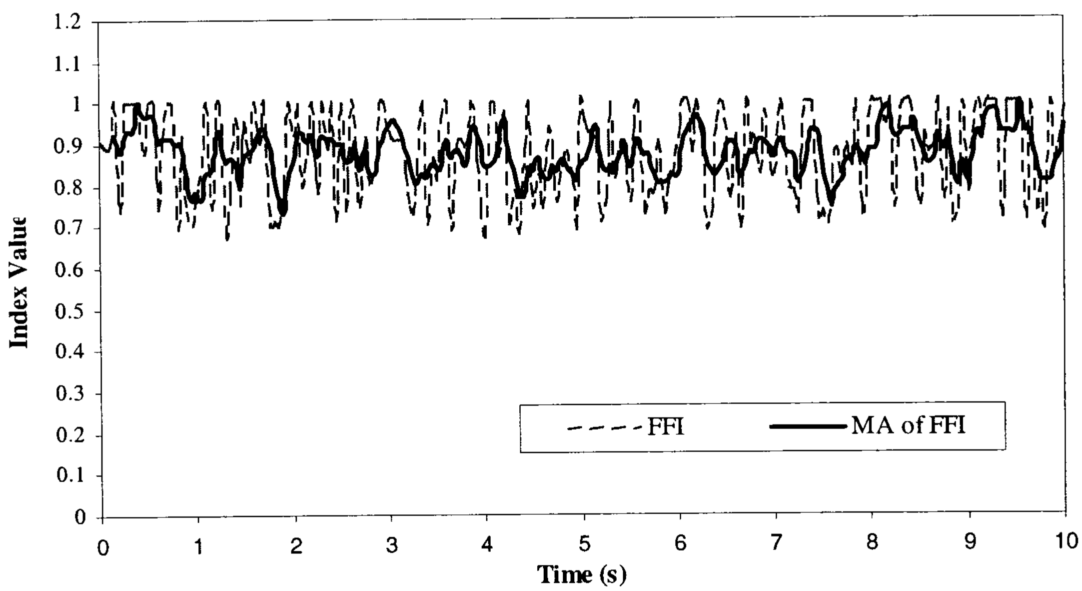



Figure D.26 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at ball, 1772 rpm and 1 hp motor load.

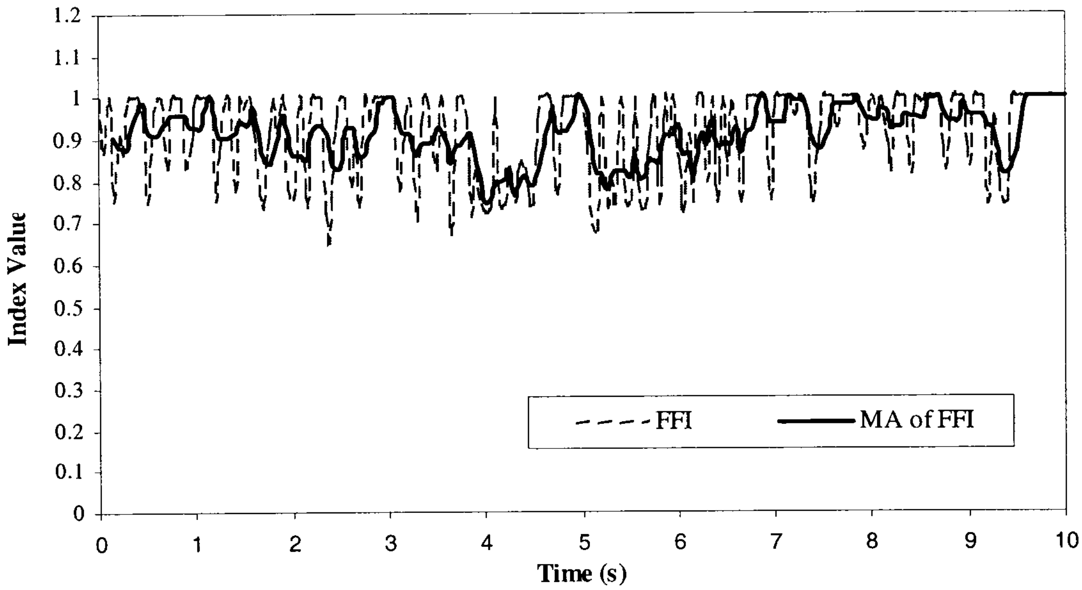



Figure D.27 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at ball, 1750 rpm and 2 hp motor load.

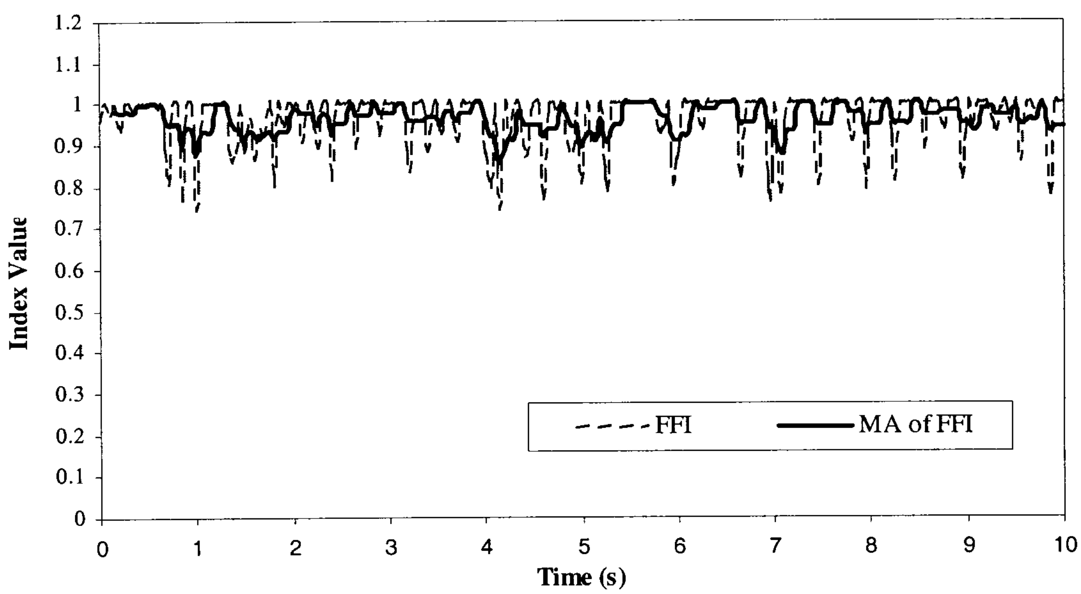



Figure D.28 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at ball, 1797 rpm and 0 hp motor load.

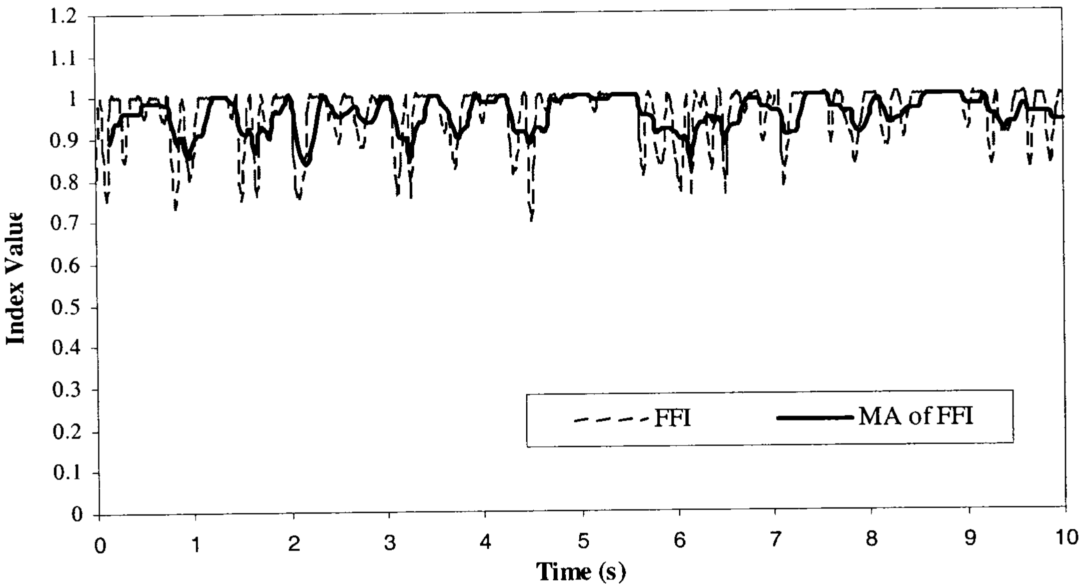



Figure D.29 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at ball, 1772 rpm and 1 hp motor load.

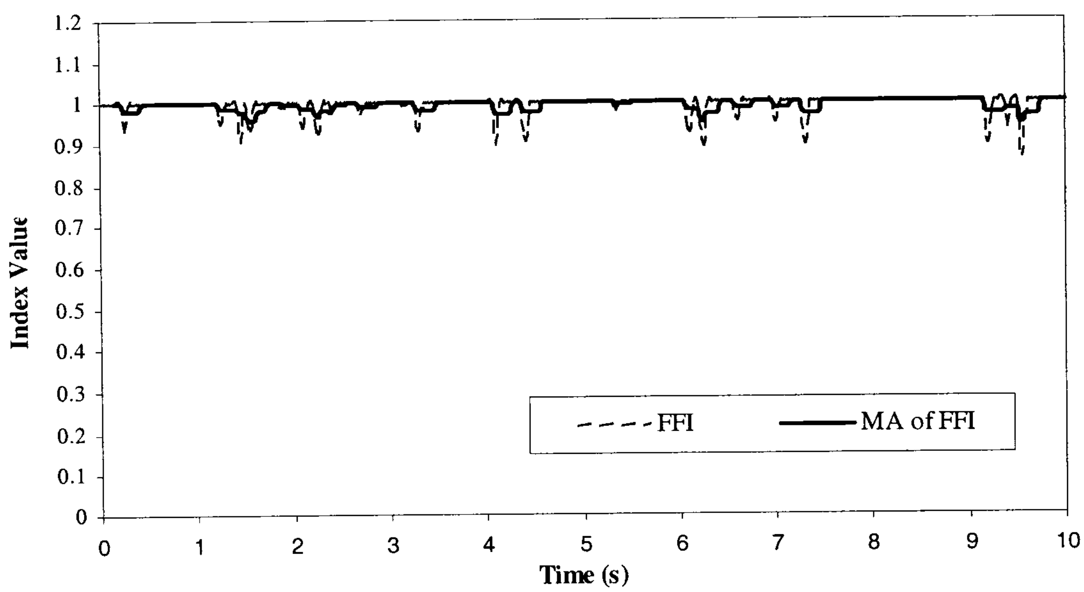



Figure D.30 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at ball, 1750 rpm and 2 hp motor load.

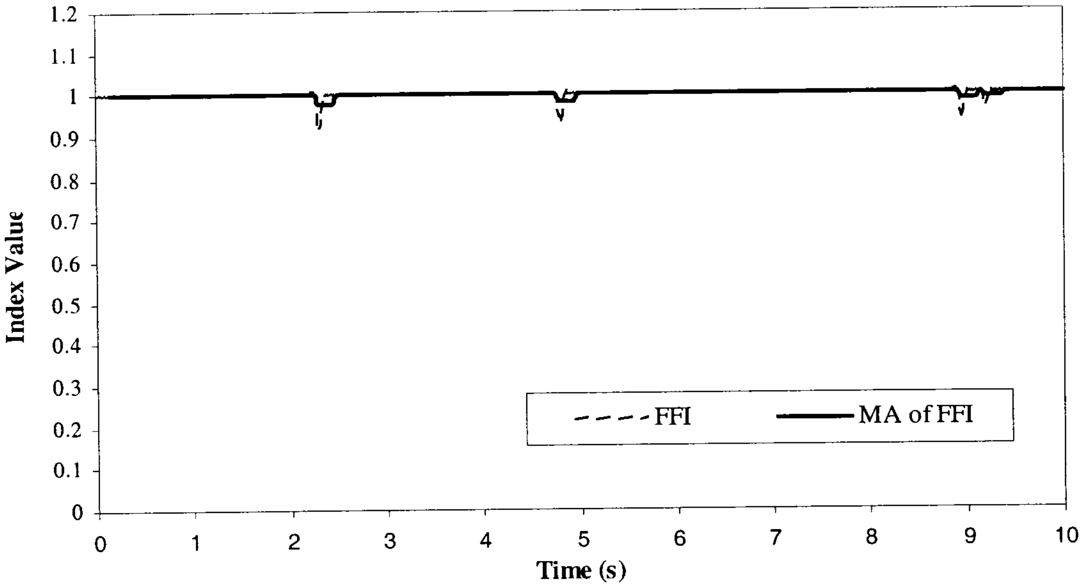



Figure D.31 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at ball, 1730 rpm and 3 hp motor load.

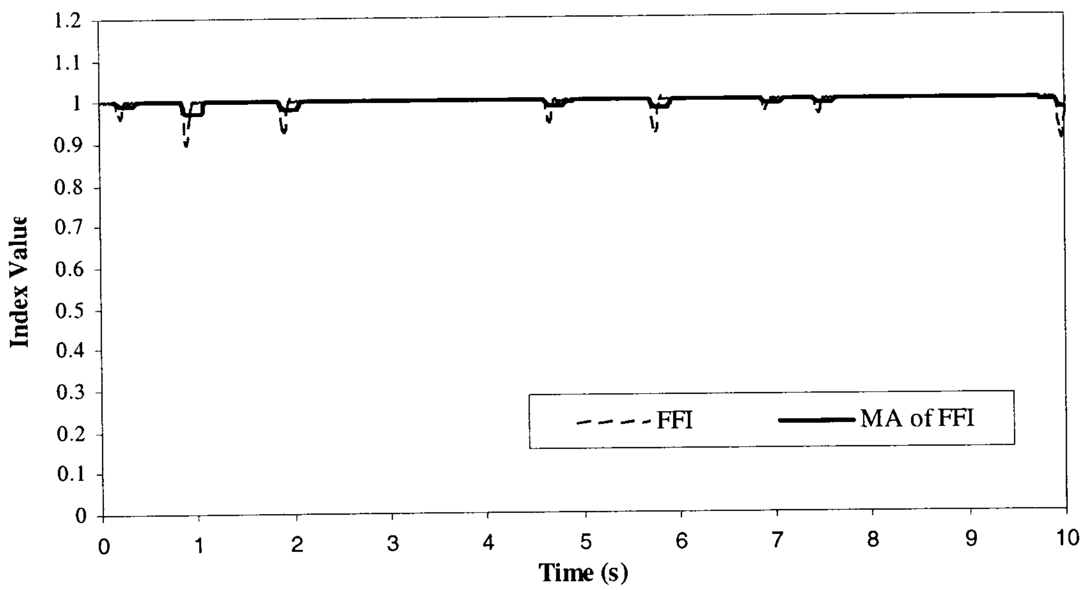



Figure D.32 Fuzzy fused index and its moving average shown for drive end bearing, 0.028" fault diameter at ball, 1797 rpm and 0 hp motor load.

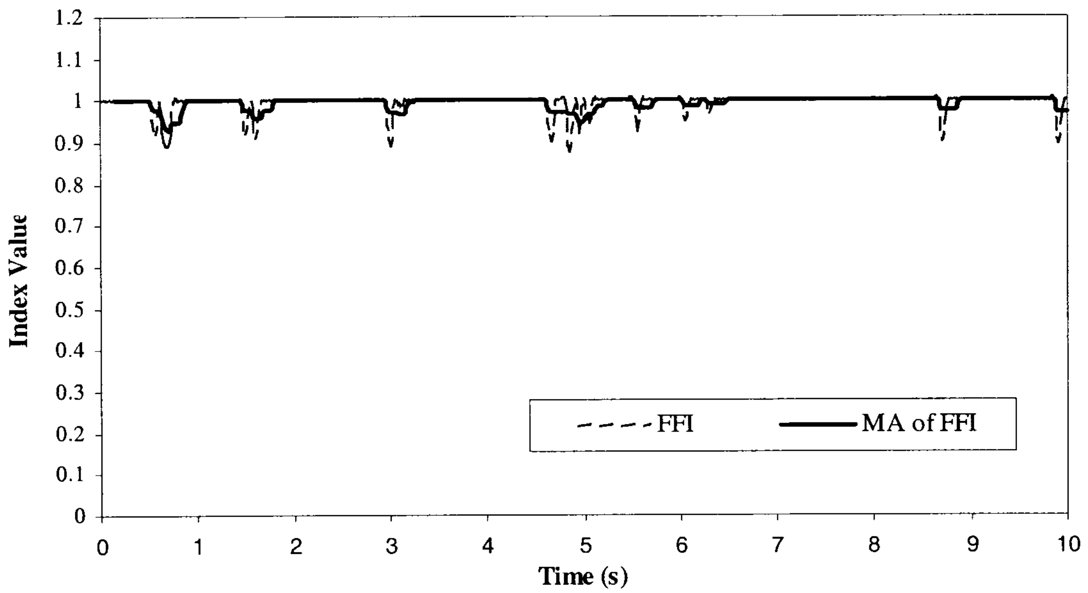



Figure D.33 Fuzzy fused index and its moving average shown for drive end bearing, 0.028" fault diameter at ball, 1772 rpm and 1 hp motor load.

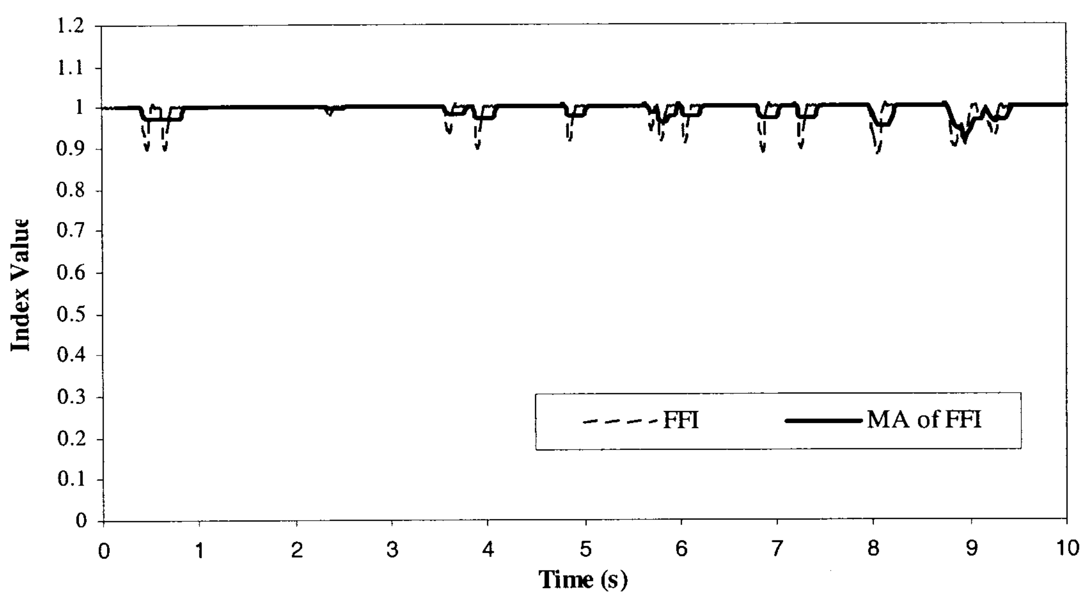



Figure D.34 Fuzzy fused index and its moving average shown for drive end bearing, 0.028" fault diameter at ball, 1750 rpm and 2 hp motor load.

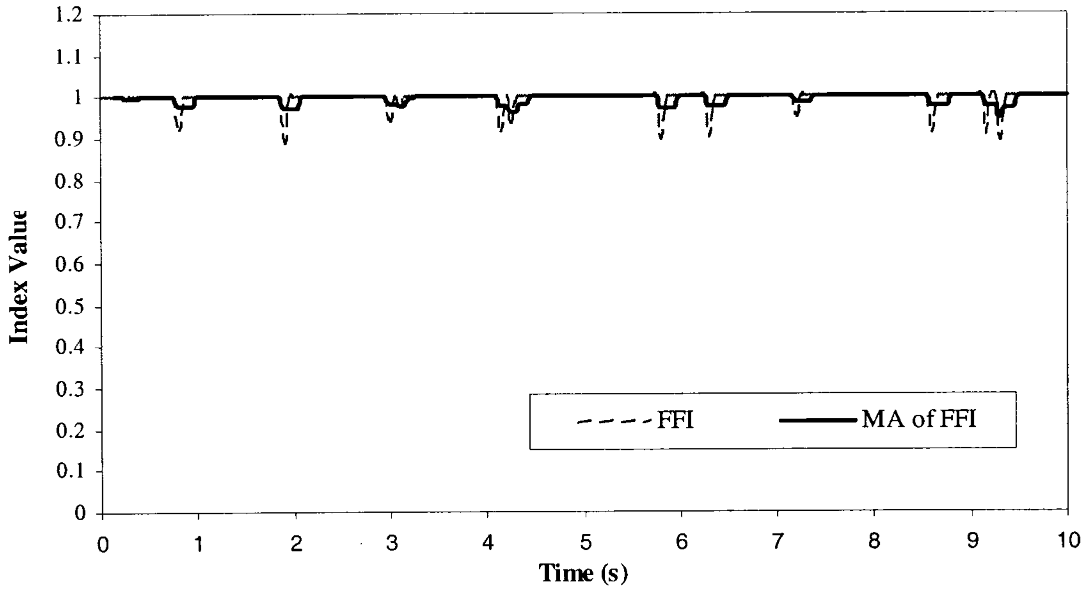



Figure D.35 Fuzzy fused index and its moving average shown for drive end bearing, 0.028" fault diameter at ball, 1730 rpm and 3 hp motor load.

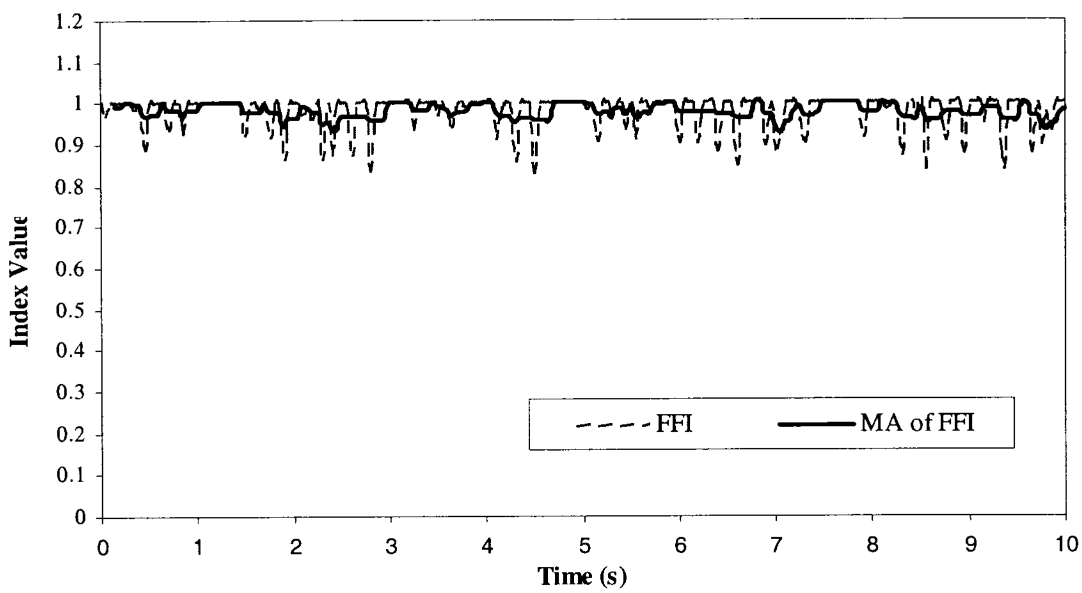



Figure D.36 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at outer race, 1797 rpm and 0 hp motor load.

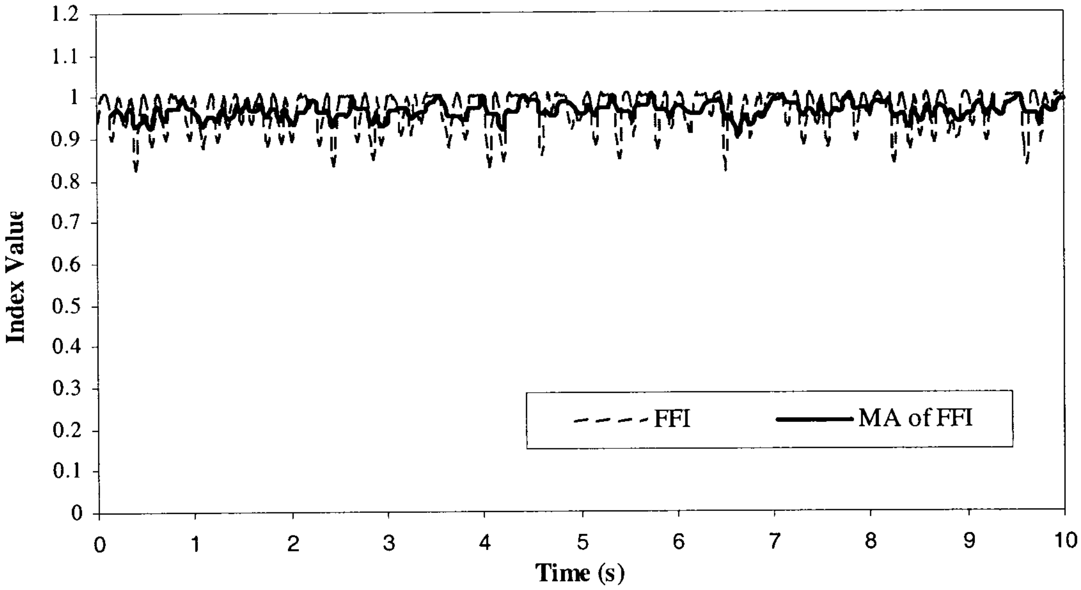



Figure D.37 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at outer race, 1772 rpm and 1 hp motor load.

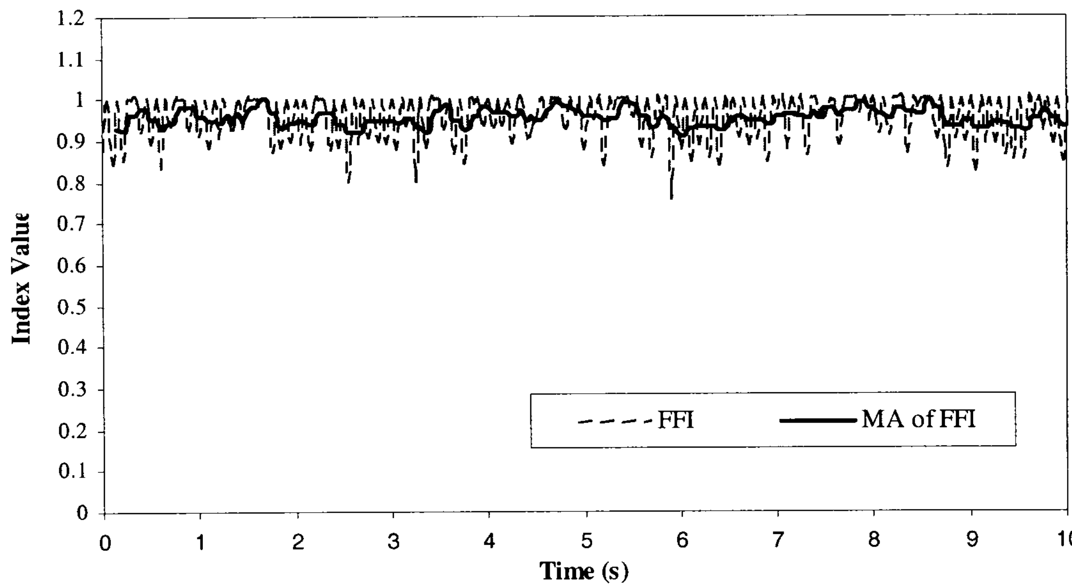



Figure D.38 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at outer race, 1750 rpm and 2 hp motor load.

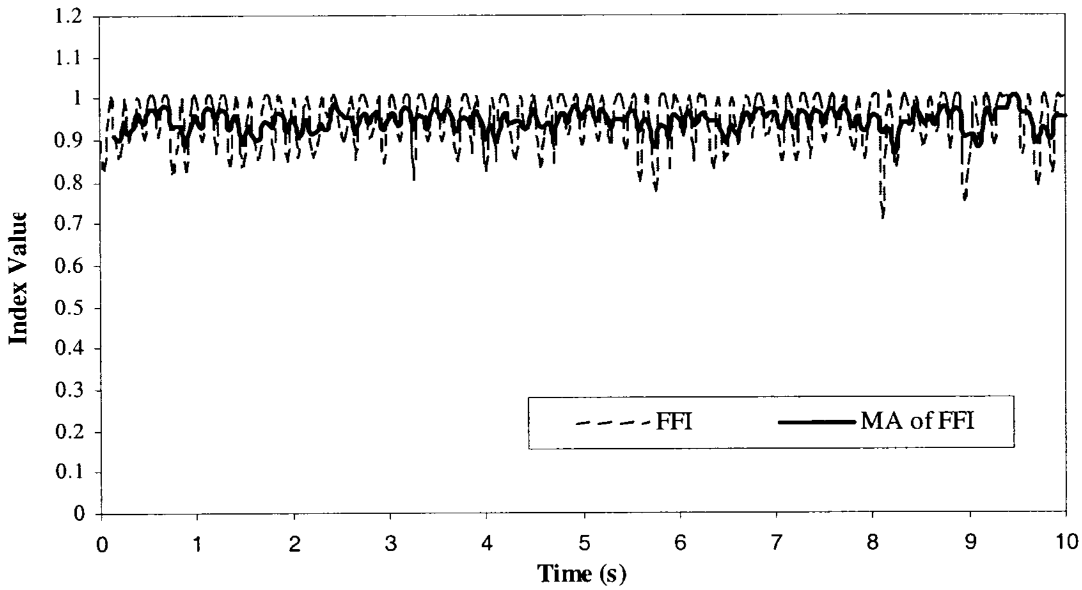



Figure D.39 Fuzzy fused index and its moving average shown for drive end bearing, 0.007" fault diameter at outer race, 1730 rpm and 3 hp motor load.

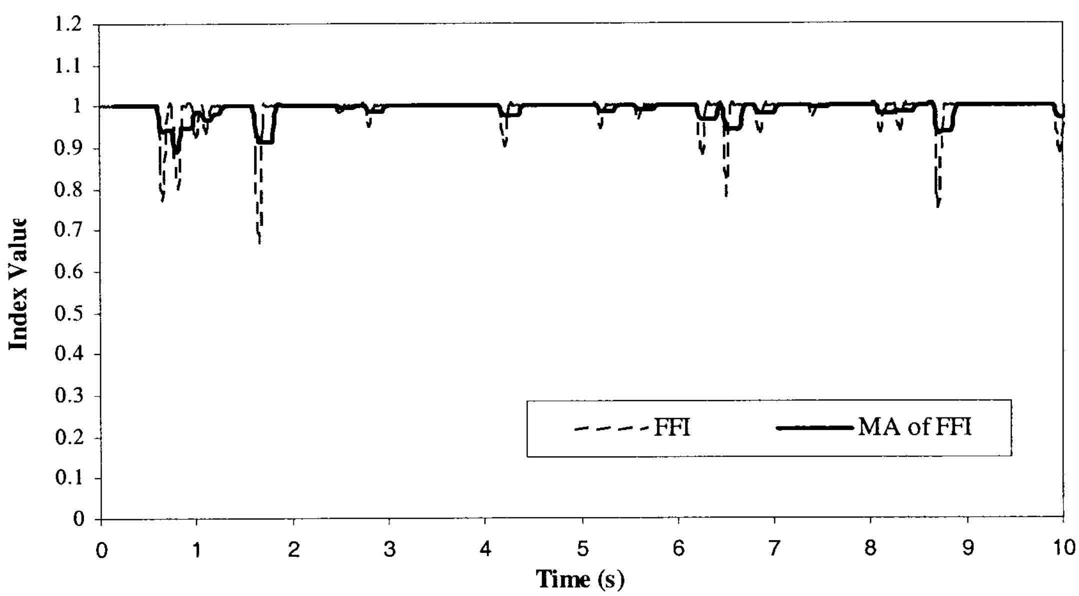



Figure D.40 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at outer race, 1797 rpm and 0 hp motor load.

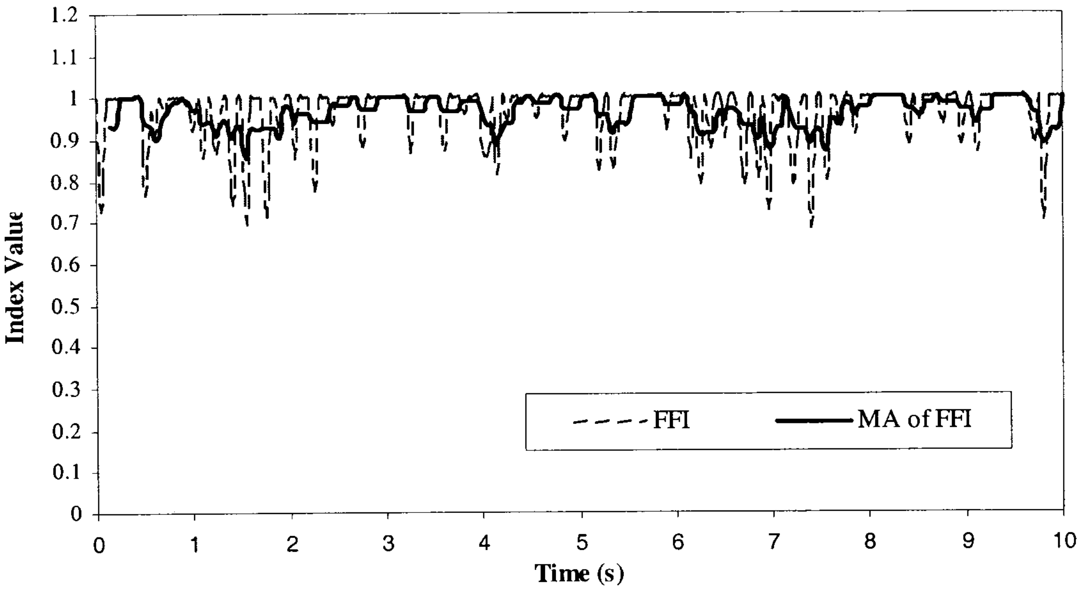



Figure D.41 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at outer race, 1772 rpm and 1 hp motor load.

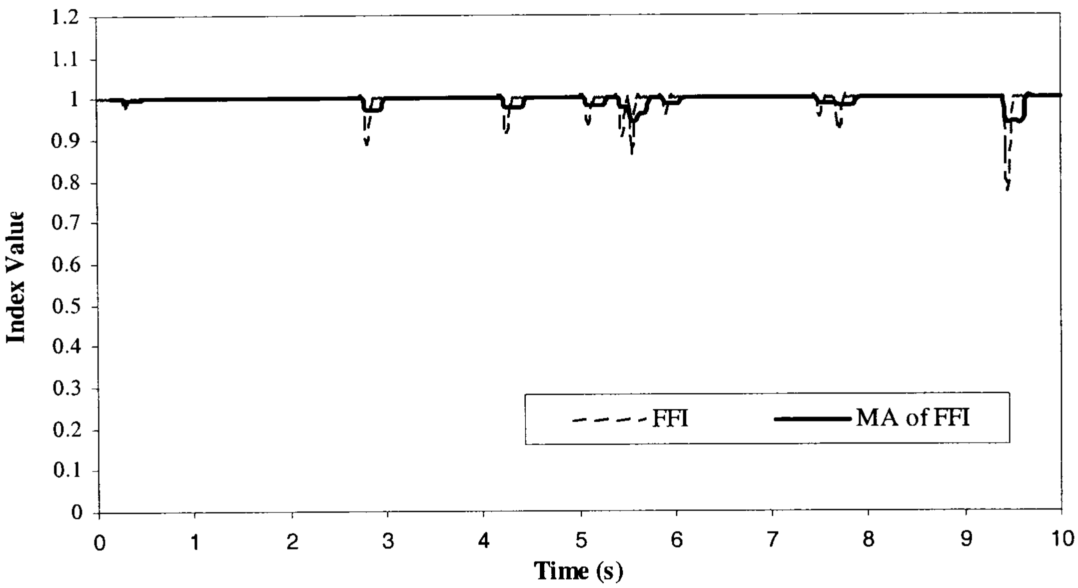



Figure D.42 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at outer race, 1750 rpm and 2 hp motor load.

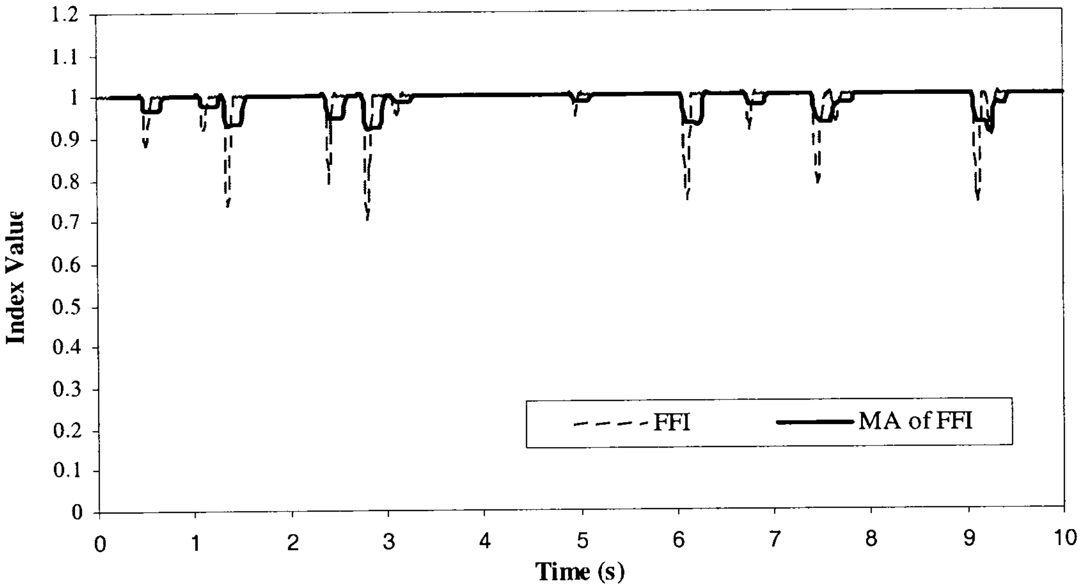



Figure D.43 Fuzzy fused index and its moving average shown for drive end bearing, 0.014" fault diameter at outer race, 1730 rpm and 3 hp motor load.

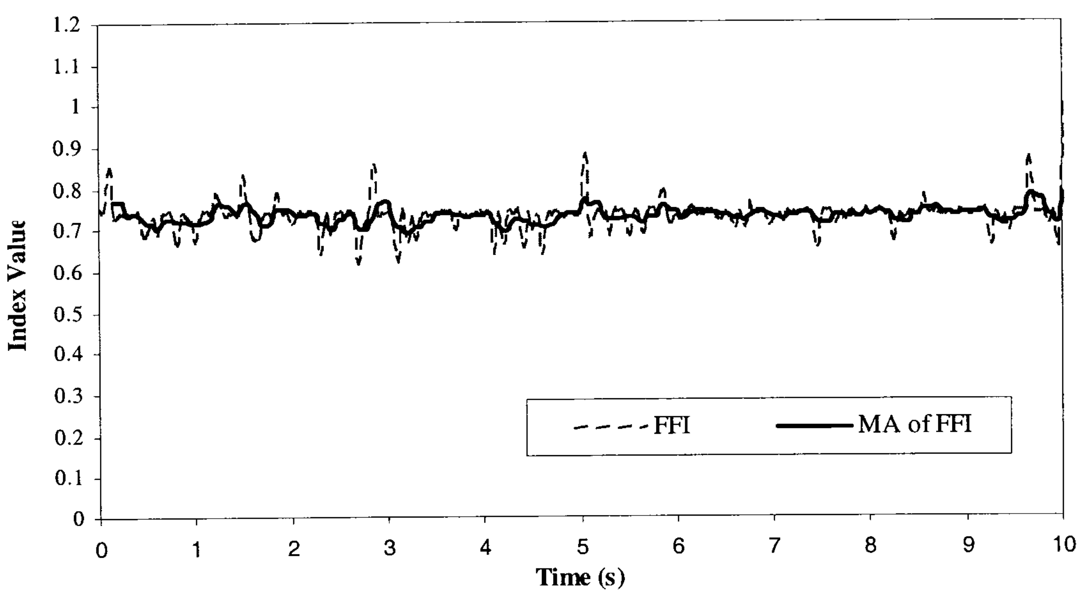



Figure D.44 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at outer race, 1797 rpm and 0 hp motor load.

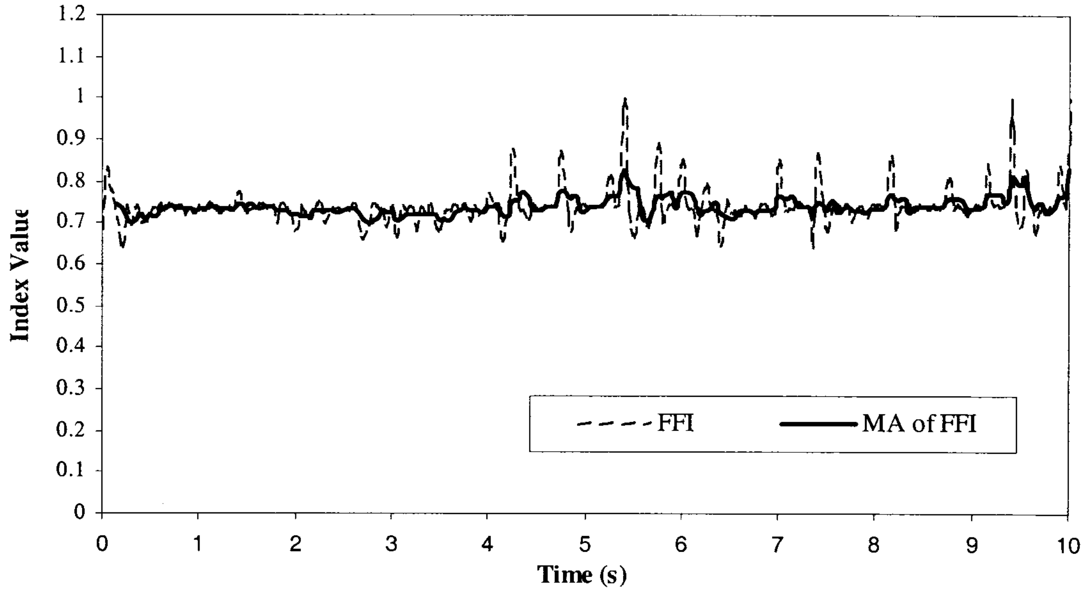



Figure D.45 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at outer race, 1772 rpm and 1 hp motor load.

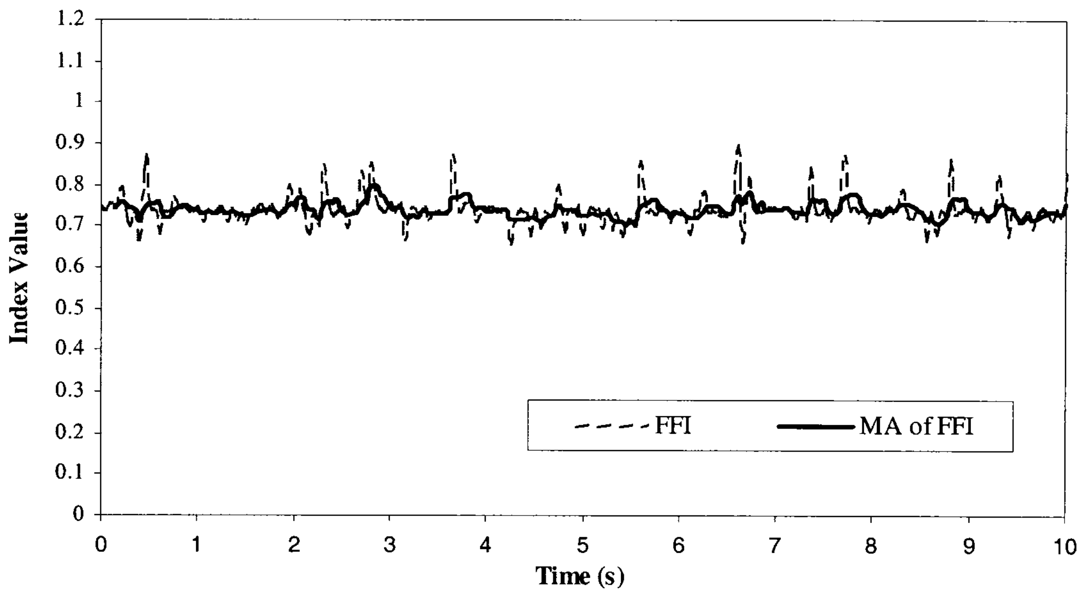



Figure D.46 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at outer race, 1750 rpm and 2 hp motor load.

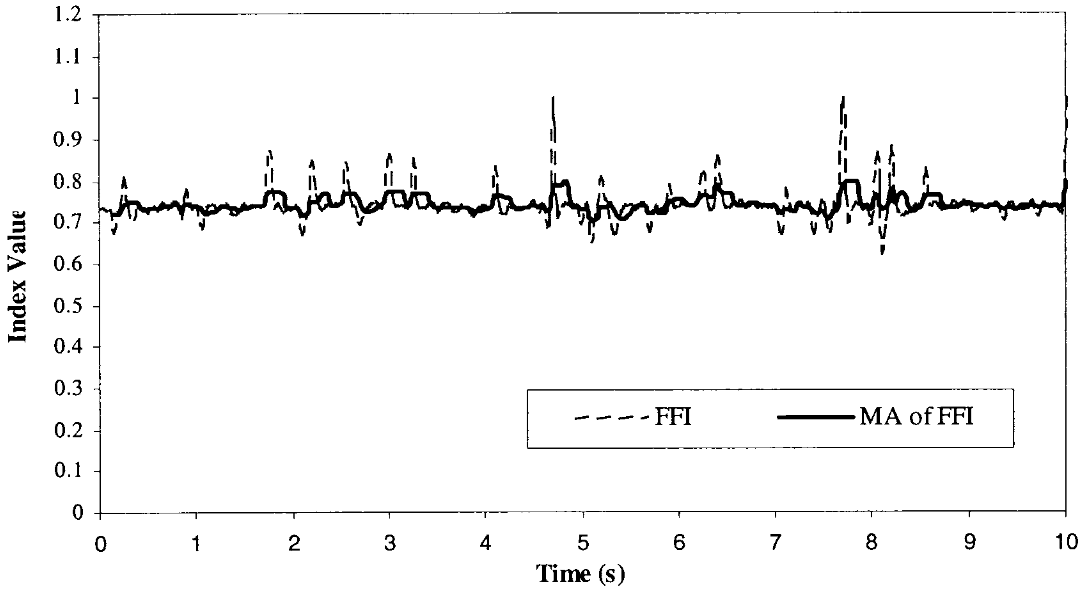



Figure D.47 Fuzzy fused index and its moving average shown for drive end bearing, 0.021" fault diameter at outer race, 1730 rpm and 3 hp motor load.

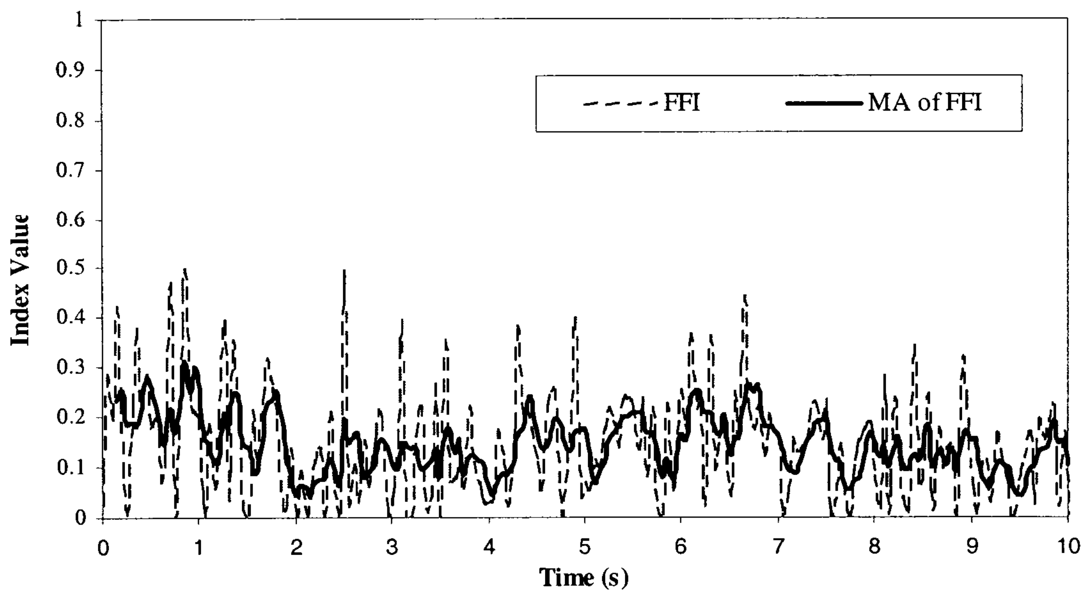



Figure D.48 Fuzzy fused index and its moving average shown for fan end bearing, normal condition at 1797 rpm and 0 hp motor load.

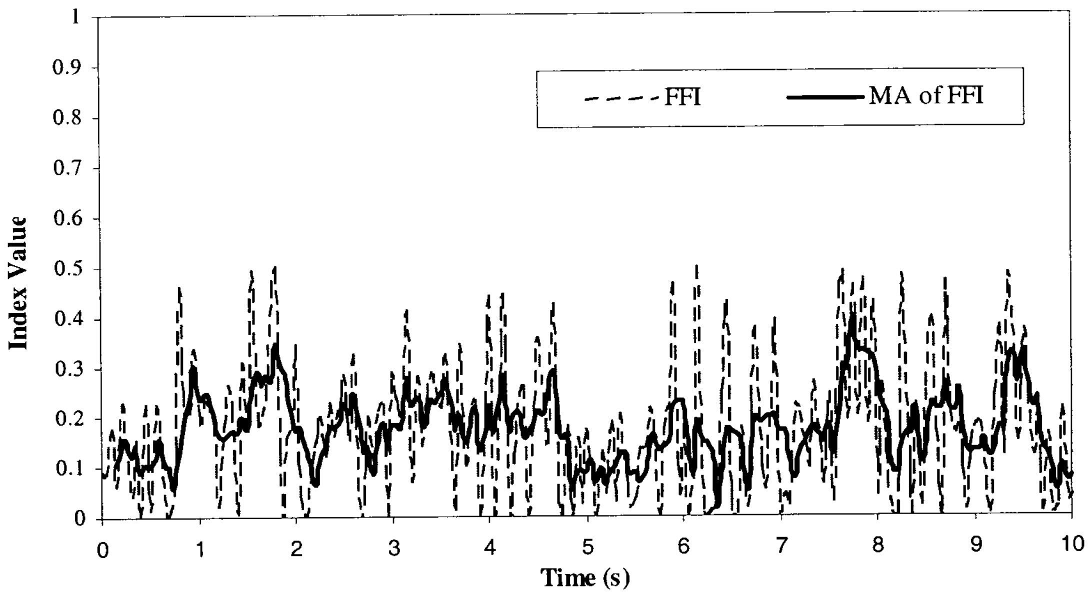



Figure D.49 Fuzzy fused index and its moving average shown for fan end bearing, normal condition at 1772 rpm and 1 hp motor load.

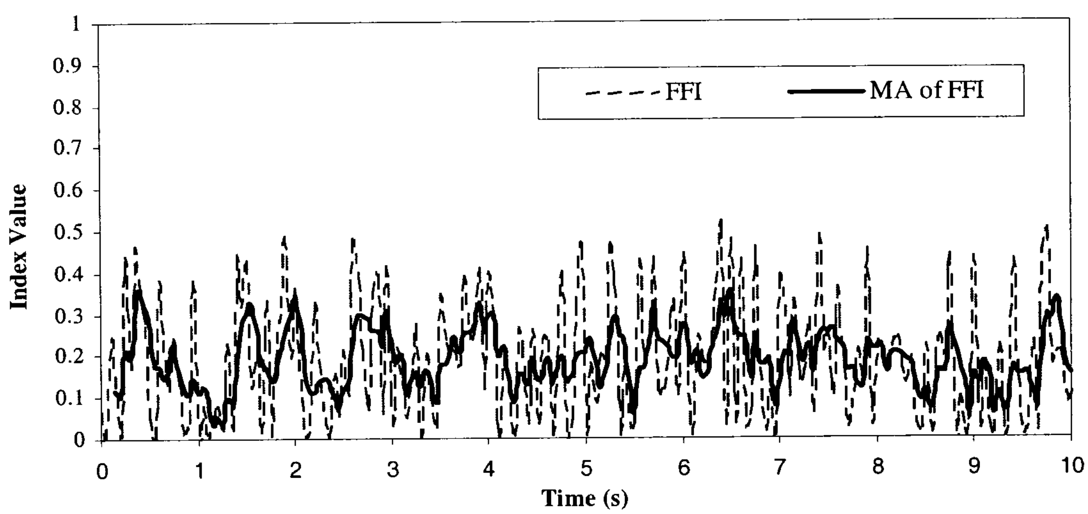



Figure D.50 Fuzzy fused index and its moving average shown for fan end bearing, normal condition at 1750 rpm and 2 hp motor load.

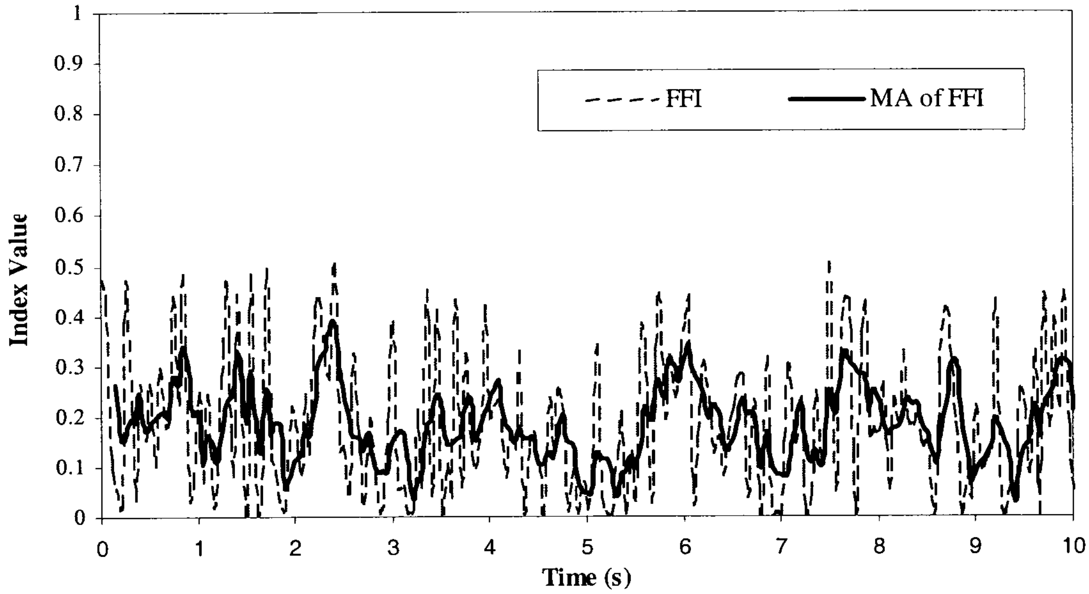



Figure D.51 Fuzzy fused index and its moving average shown for fan end bearing, normal condition at 1730 rpm and 3 hp motor load.

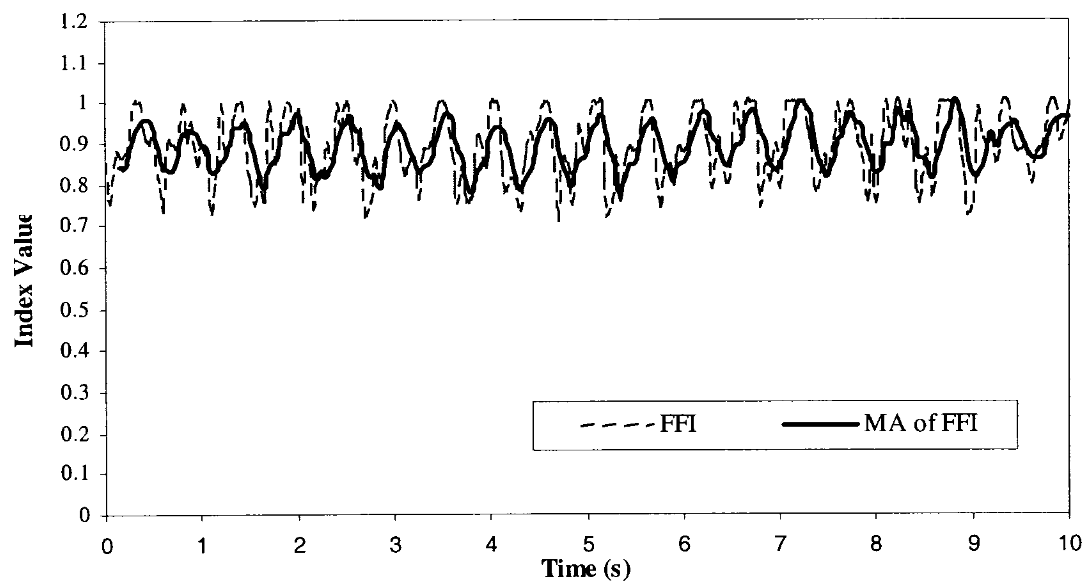



Figure D.52 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at inner race, 1797 rpm and 0 hp motor load.

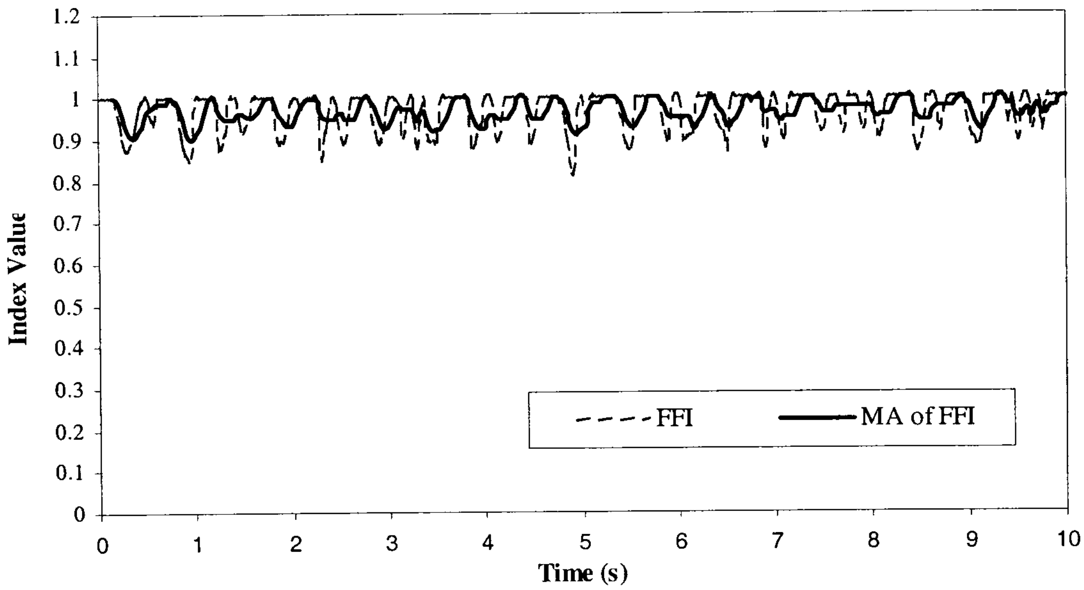



Figure D.53 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at inner race, 1772 rpm and 1 hp motor load.

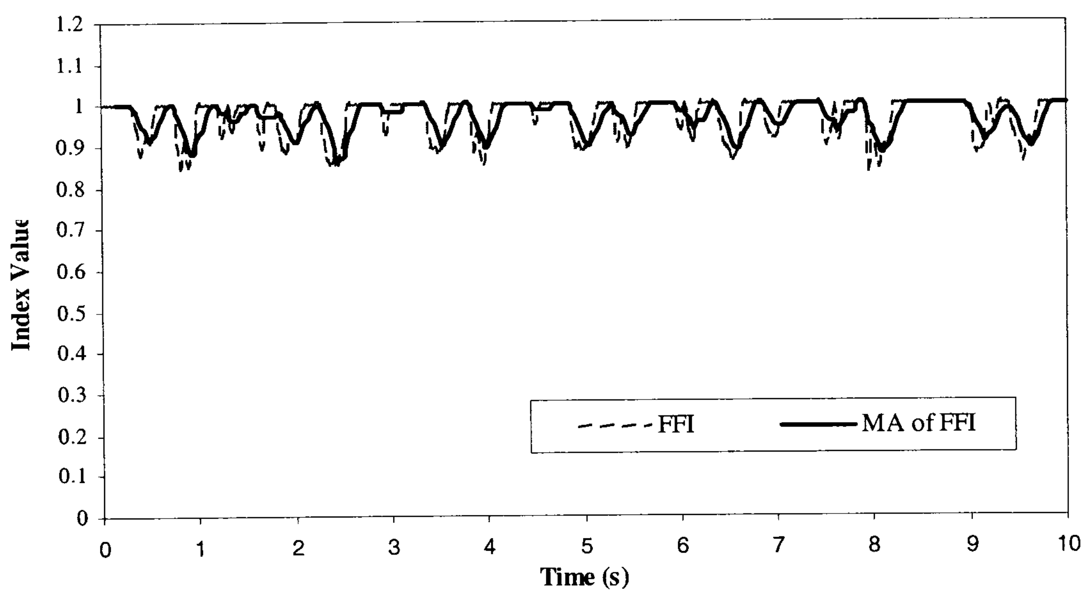



Figure D.54 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at inner race, 1750 rpm and 2 hp motor load.

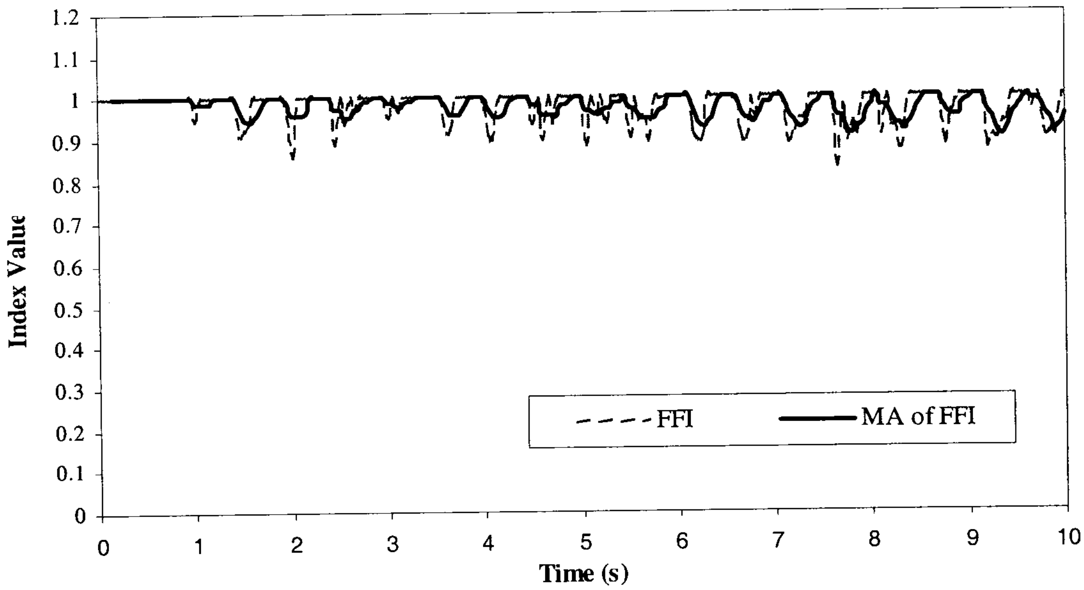



Figure D.55 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at inner race, 1730 rpm and 3 hp motor load.

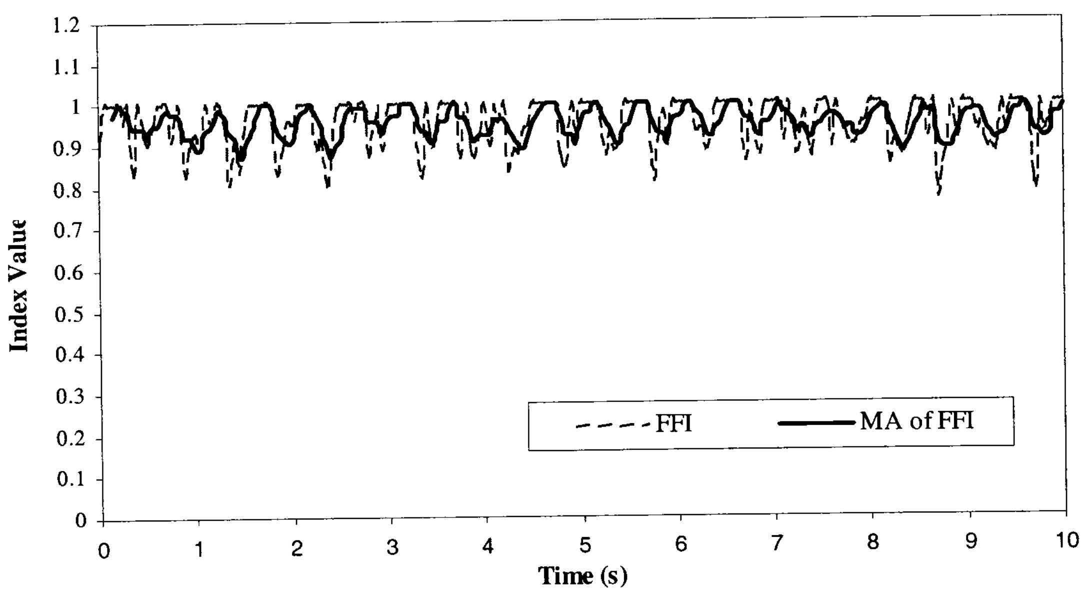



Figure D.56 Fuzzy fused index and its moving average shown for fan end bearing, 0.014" fault diameter at inner race, 1797 rpm and 0 hp motor load.

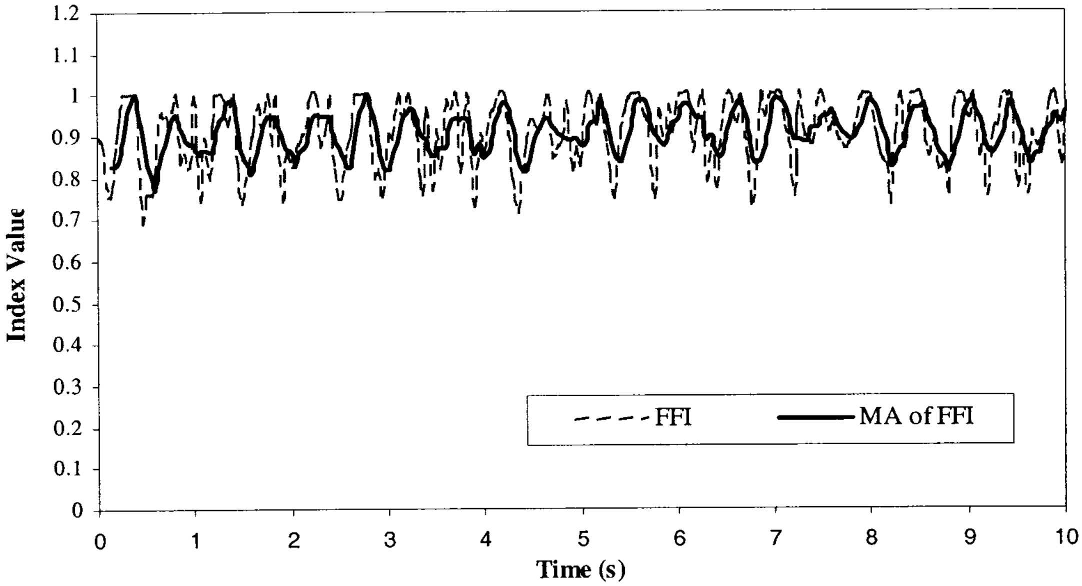



Figure D.57 Fuzzy fused index and its moving average shown for fan end bearing, 0.014" fault diameter at inner race, 1772 rpm and 1 hp motor load.

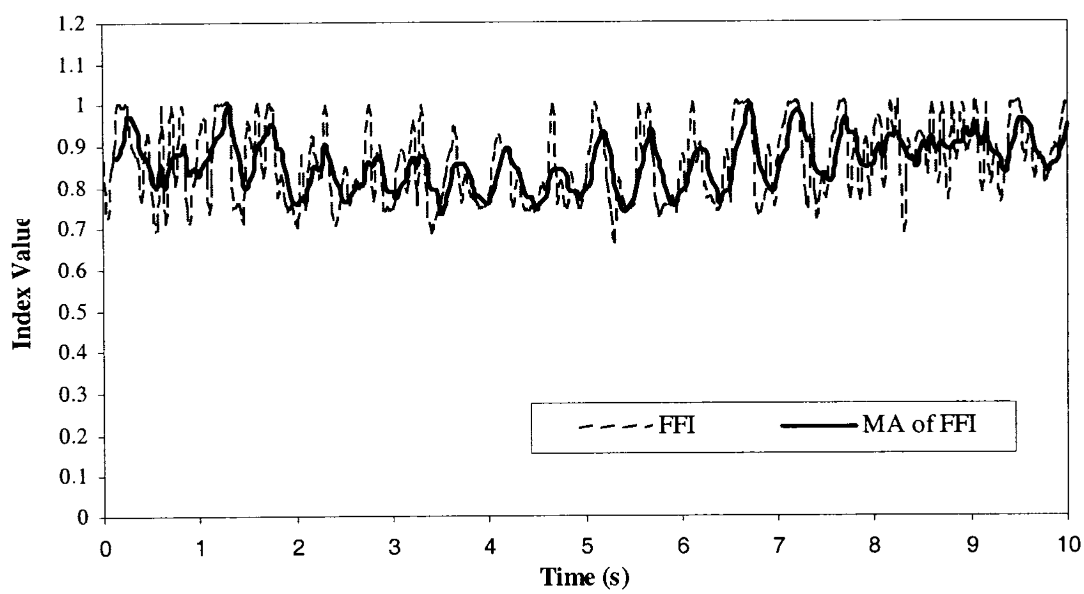



Figure D.58 Fuzzy fused index and its moving average shown for fan end bearing, 0.014" fault diameter at inner race, 1730 rpm and 3 hp motor load.

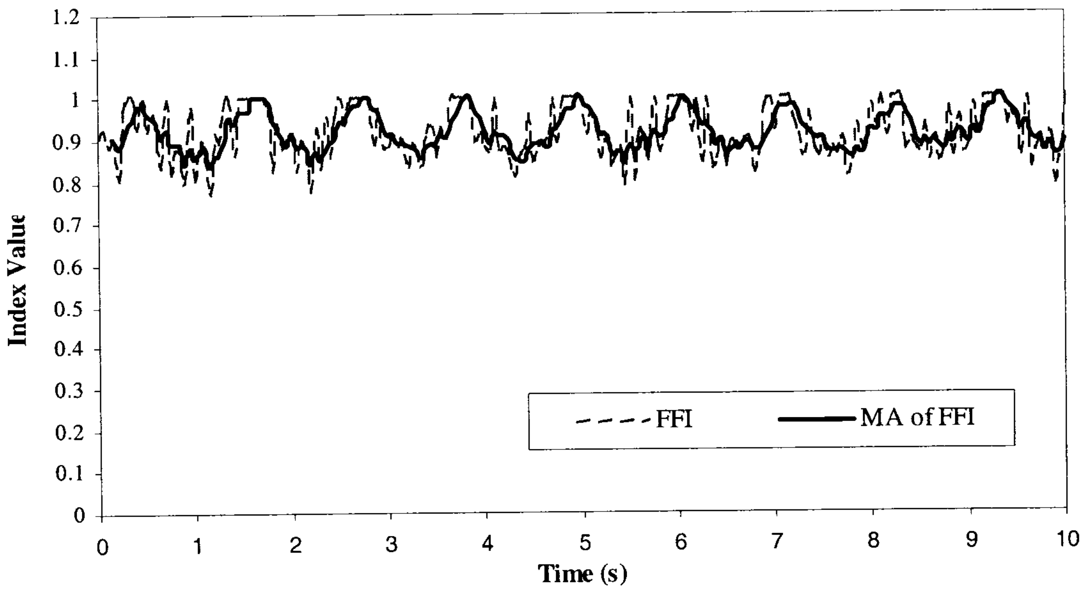



Figure D.59 Fuzzy fused index and its moving average shown for fan end bearing, 0.021" fault diameter at inner race, 1797 rpm and 0 hp motor load.

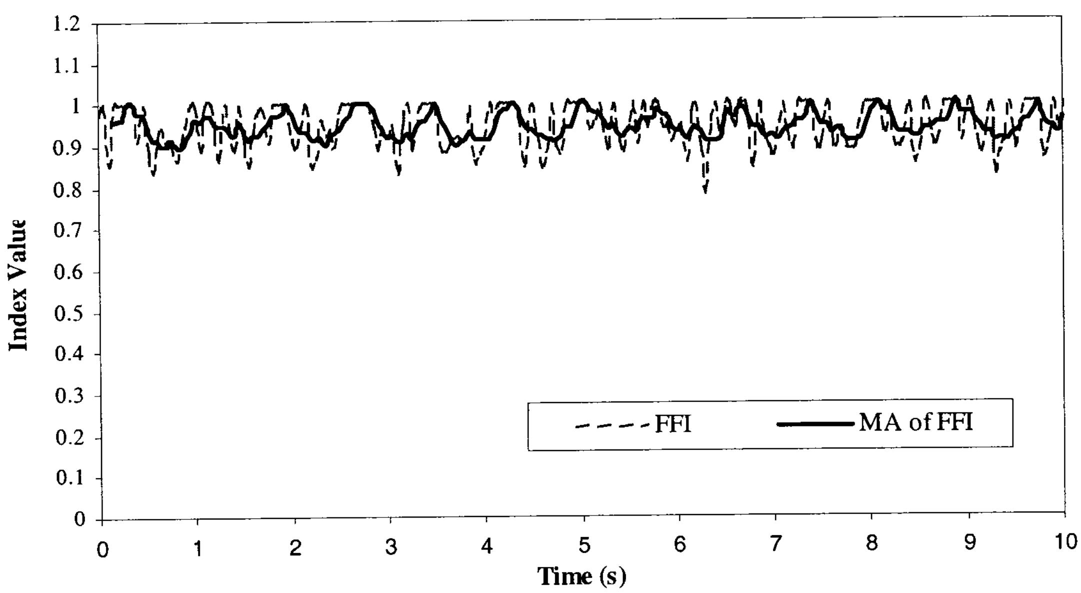



Figure D.60 Fuzzy fused index and its moving average shown for fan end bearing, 0.021" fault diameter at inner race, 1772 rpm and 1 hp motor load.

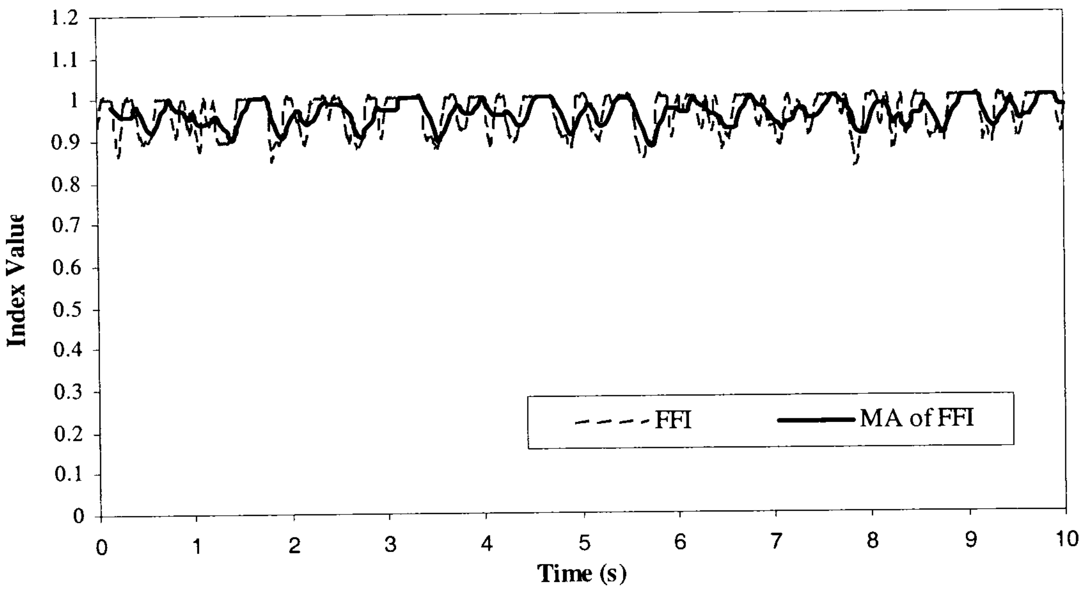



Figure D.61 Fuzzy fused index and its moving average shown for fan end bearing, 0.021" fault diameter at inner race, 1750 rpm and 2 hp motor load.

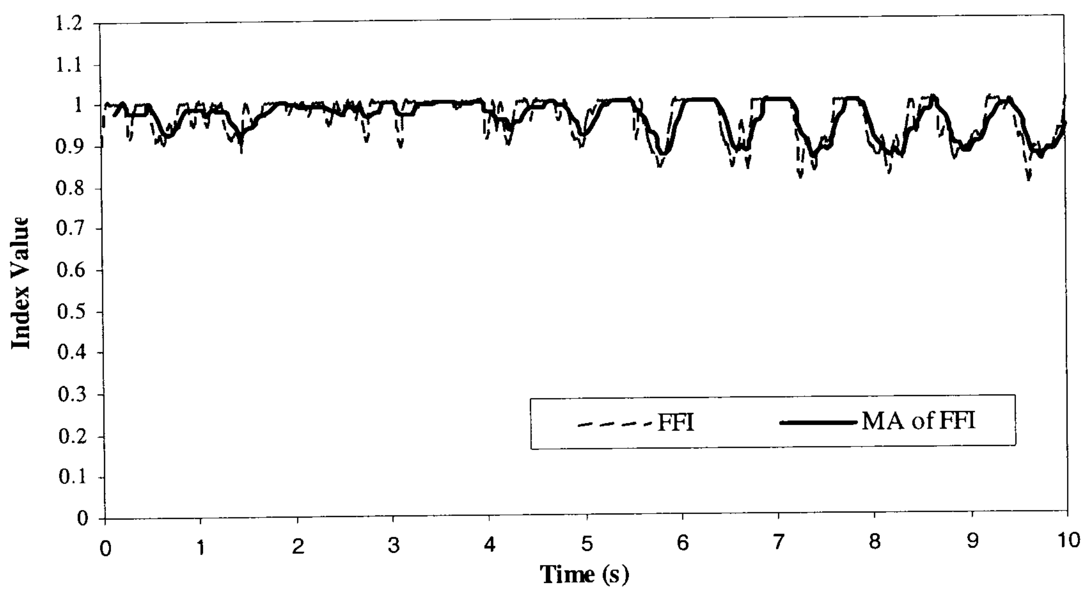



Figure D.62 Fuzzy fused index and its moving average shown for fan end bearing, 0.021" fault diameter at inner race, 1730 rpm and 3 hp motor load.

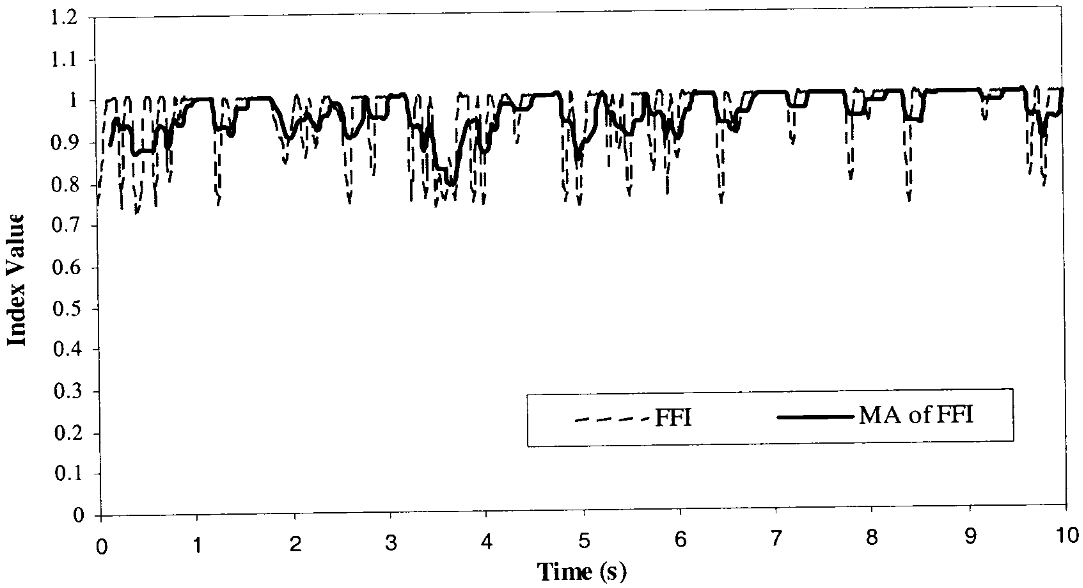



Figure D.63 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at ball, 1797 rpm and 0 hp motor load.

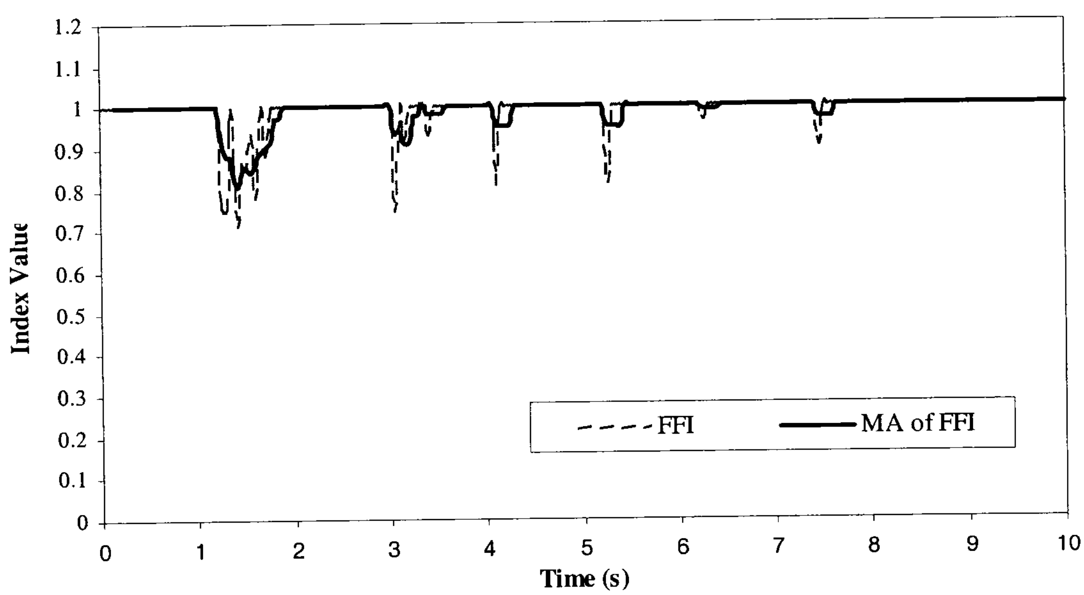



Figure D.64 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at ball, 1772 rpm and 1 hp motor load.

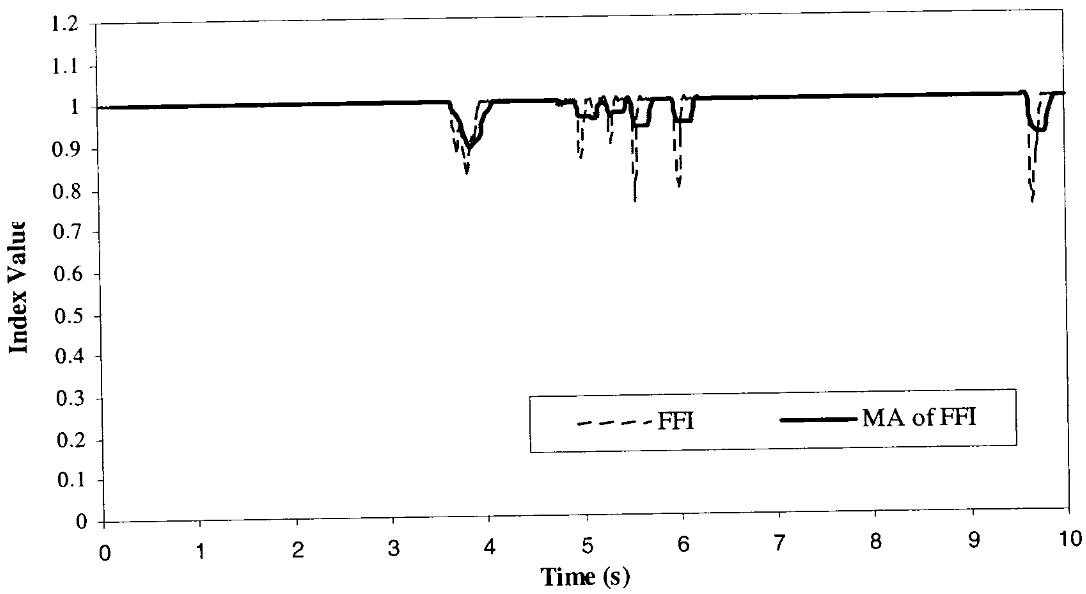



Figure D.65 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at ball, 1750 rpm and 2 hp motor load.

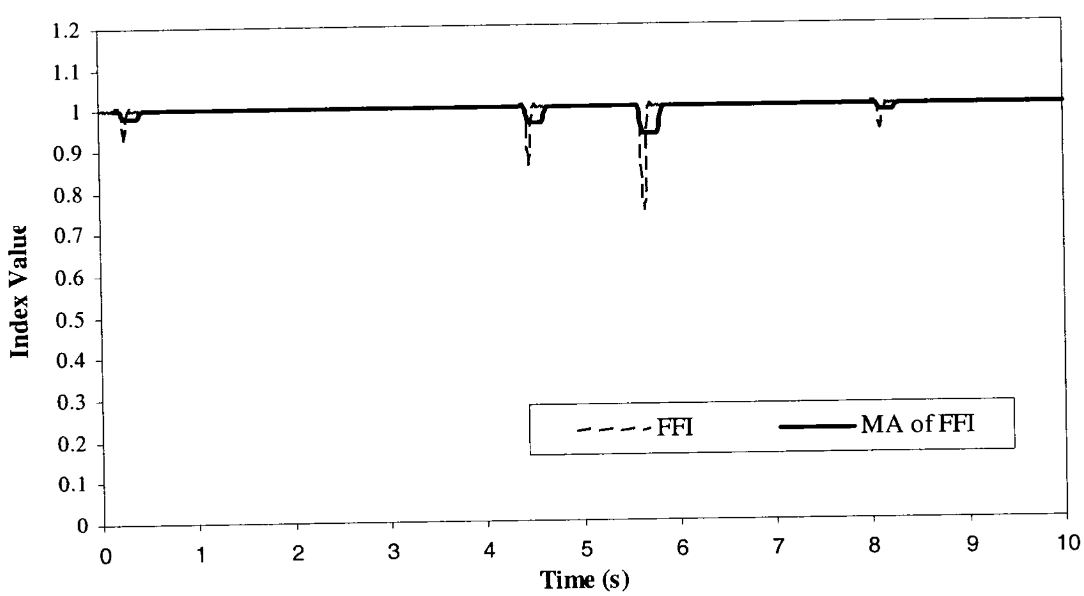



Figure D.66 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at ball, 1730 rpm and 3 hp motor load.

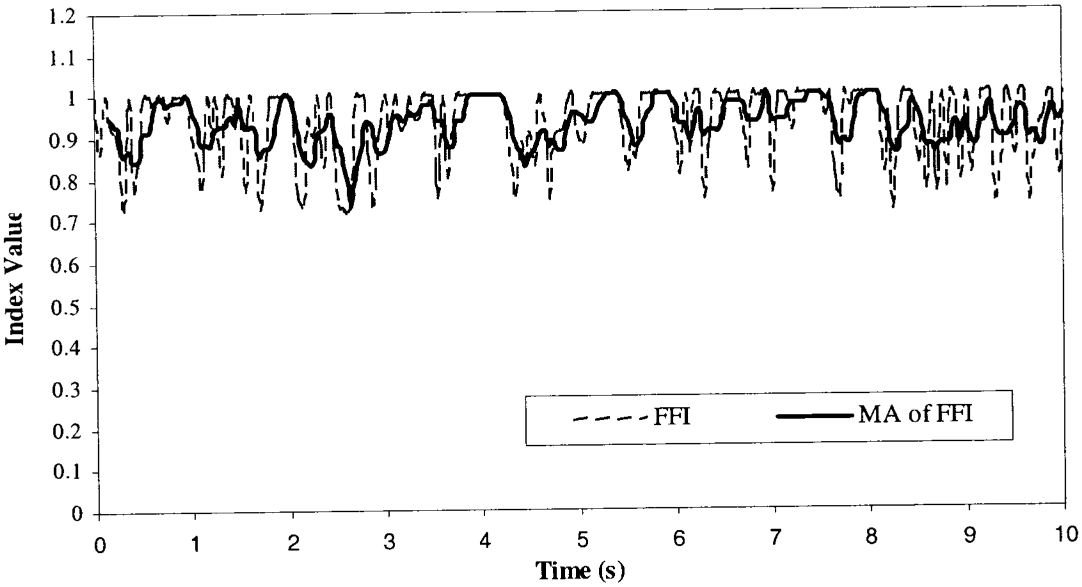



Figure D.67 Fuzzy fused index and its moving average shown for fan end bearing, 0.014" fault diameter at ball, 1797 rpm and 0 hp motor load.

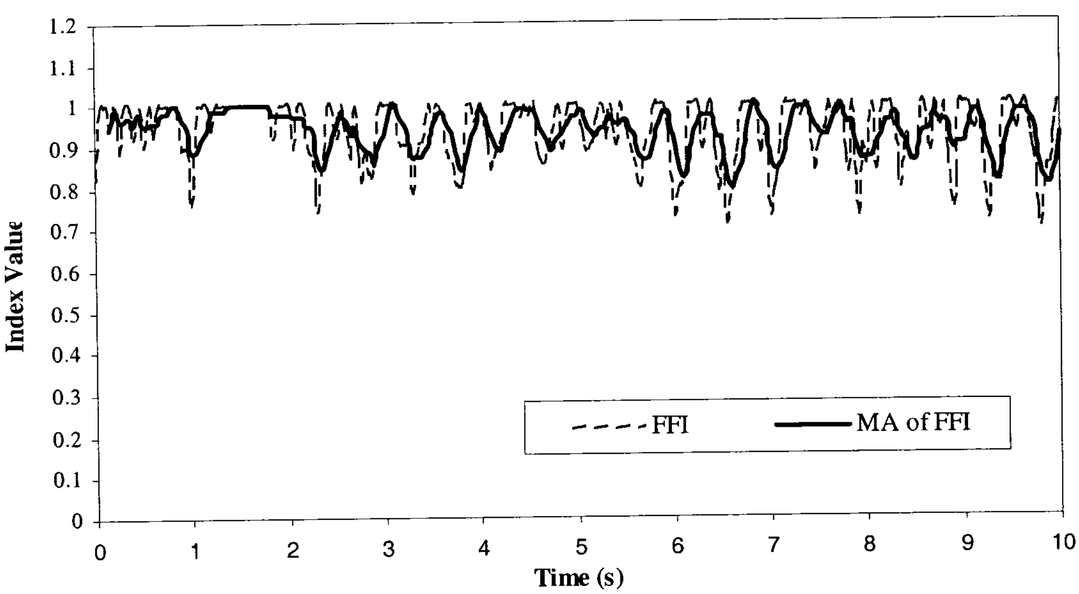



Figure D.68 Fuzzy fused index and its moving average shown for fan end bearing, 0.014" fault diameter at ball, 1750 rpm and 2 hp motor load.

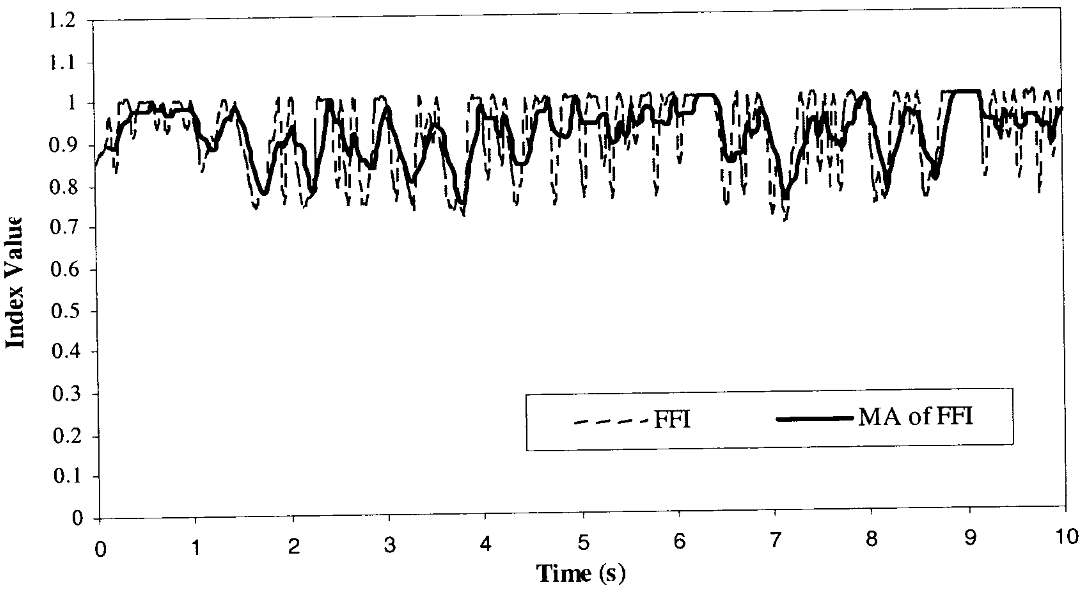



Figure D.69 Fuzzy fused index and its moving average shown for fan end bearing, 0.014" fault diameter at ball, 1730 rpm and 3 hp motor load.

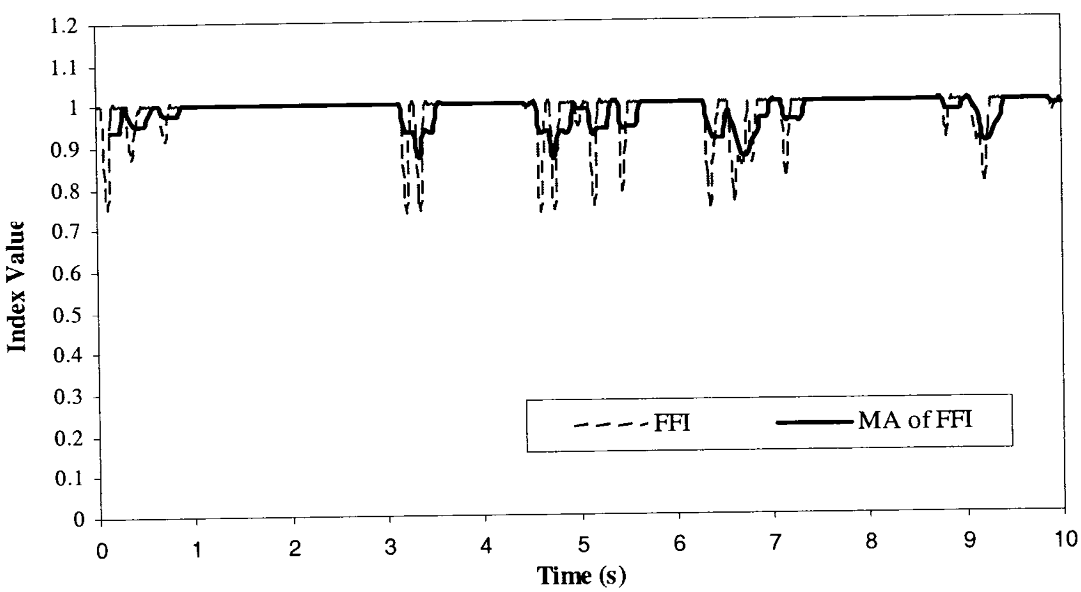



Figure D.70 Fuzzy fused index and its moving average shown for fan end bearing, 0.021" fault diameter at ball, 1797 rpm and 0 hp motor load.

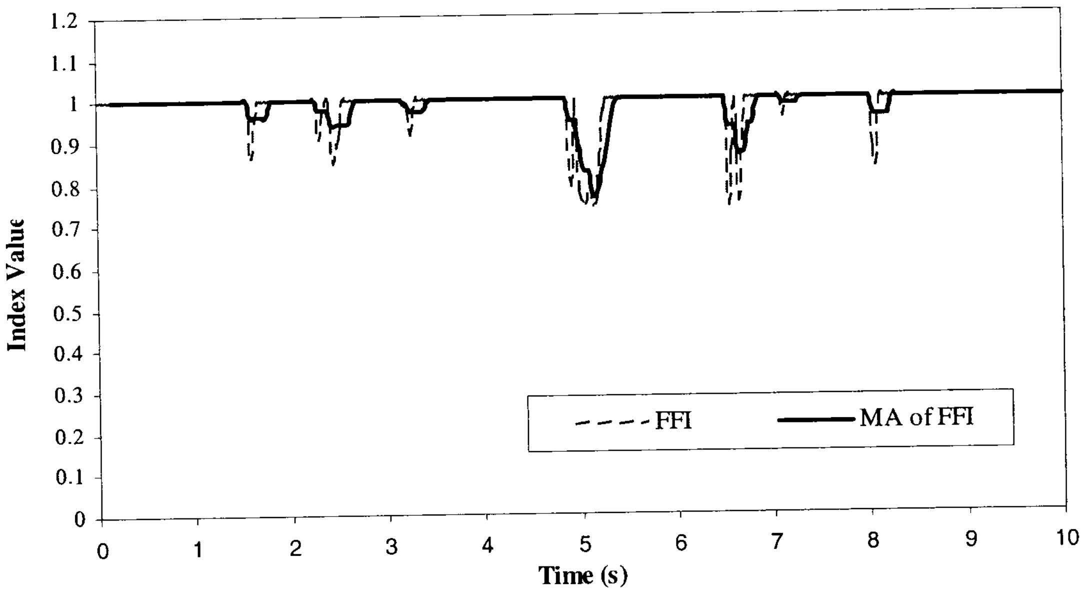



Figure D.71 Fuzzy fused index and its moving average shown for fan end bearing, 0.021" fault diameter at ball, 1772 rpm and 1 hp motor load.

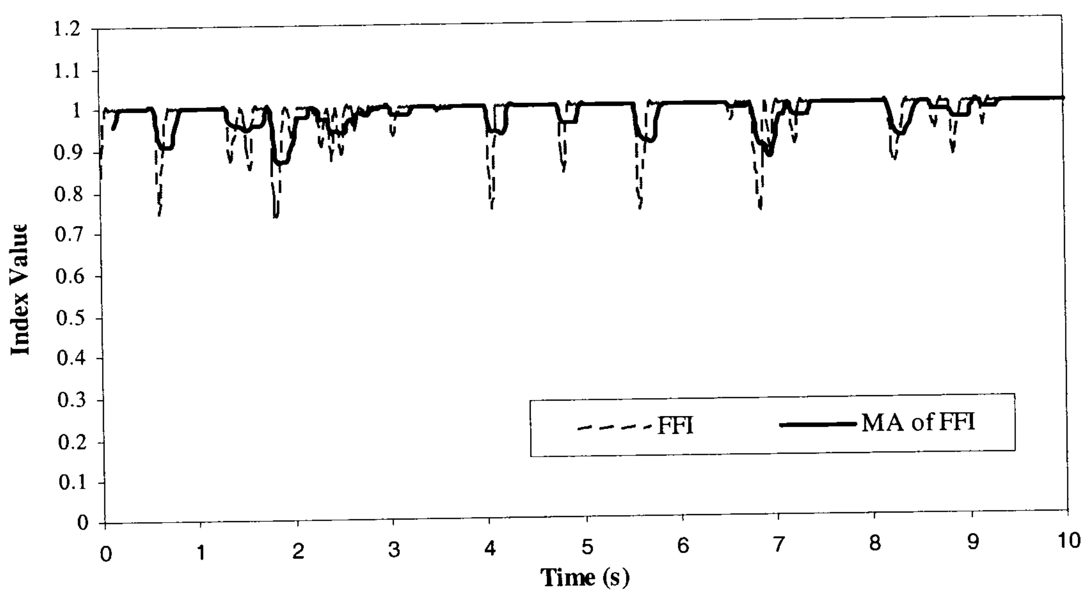



Figure D.72 Fuzzy fused index and its moving average shown for fan end bearing, 0.021" fault diameter at ball, 1750 rpm and 2 hp motor load.

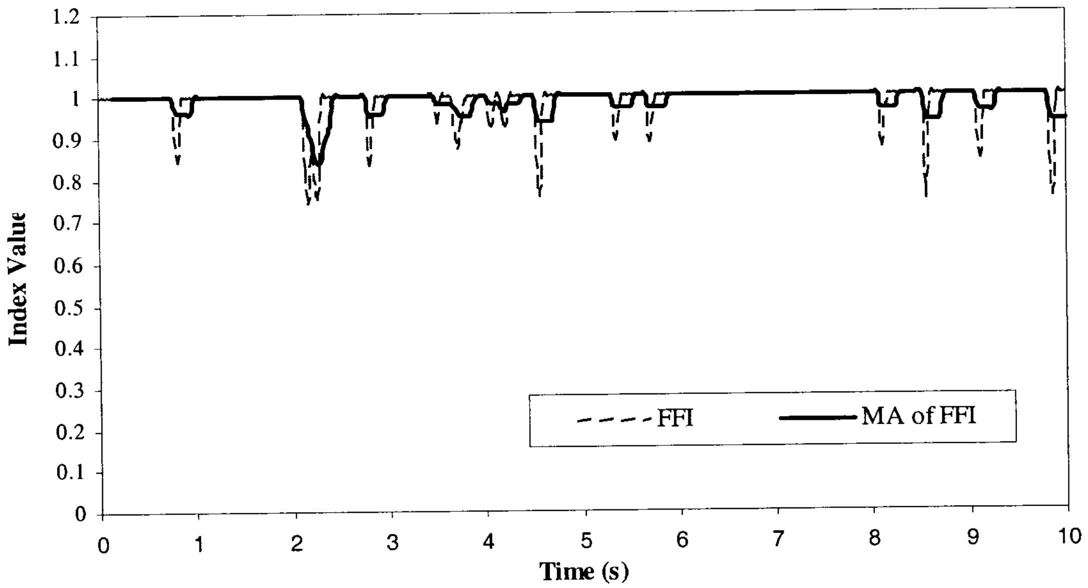



Figure D.73 Fuzzy fused index and its moving average shown for fan end bearing, 0.021" fault diameter at ball, 1730 rpm and 3 hp motor load.

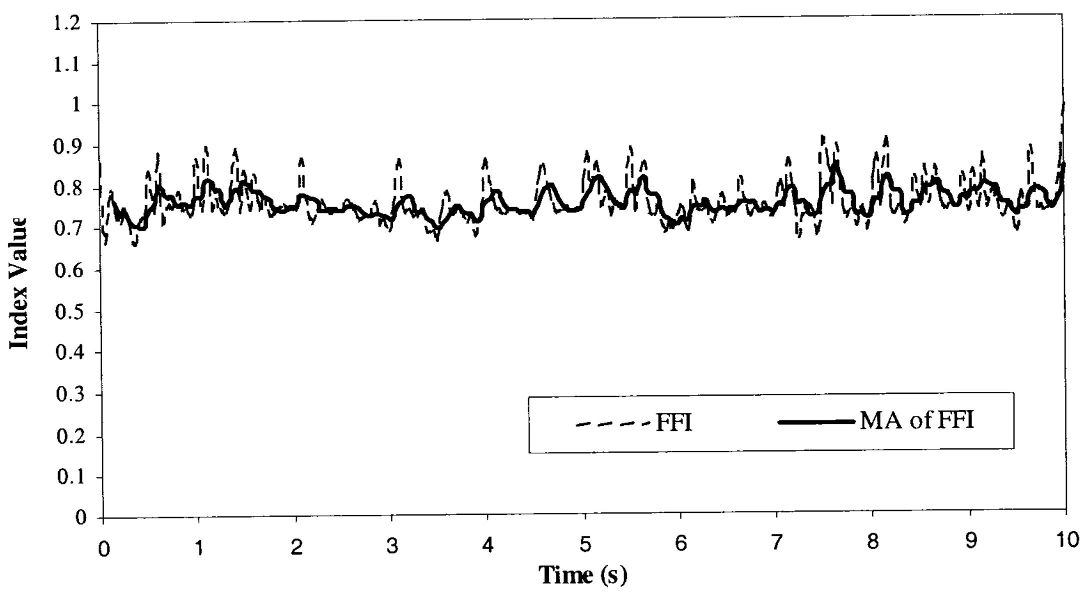



Figure D.74 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at outer race, 1797 rpm and 0 hp motor load.

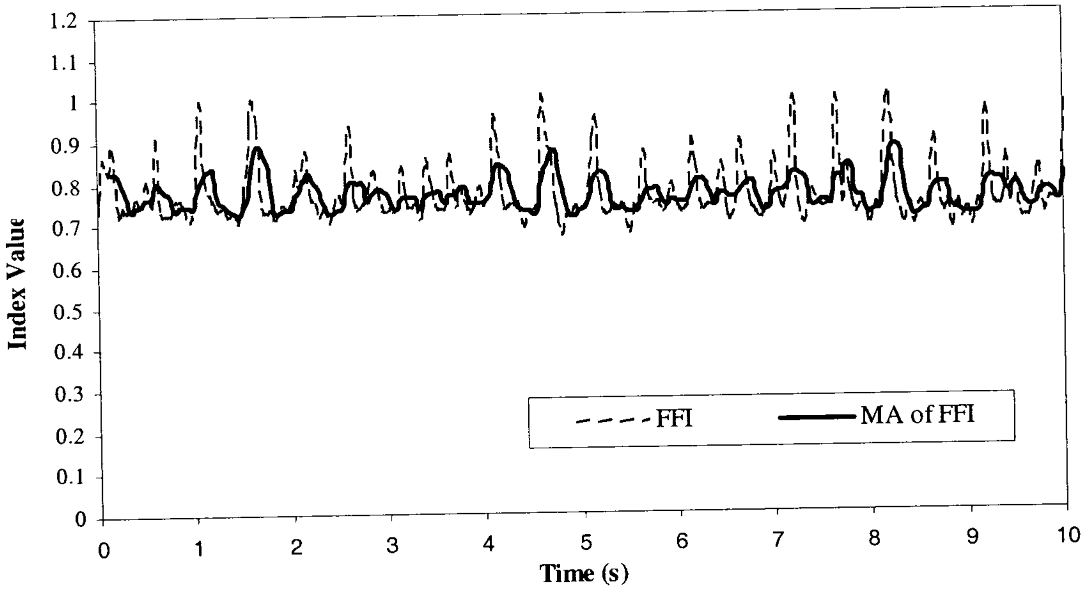



Figure D.75 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at outer race, 1772 rpm and 1 hp motor load.

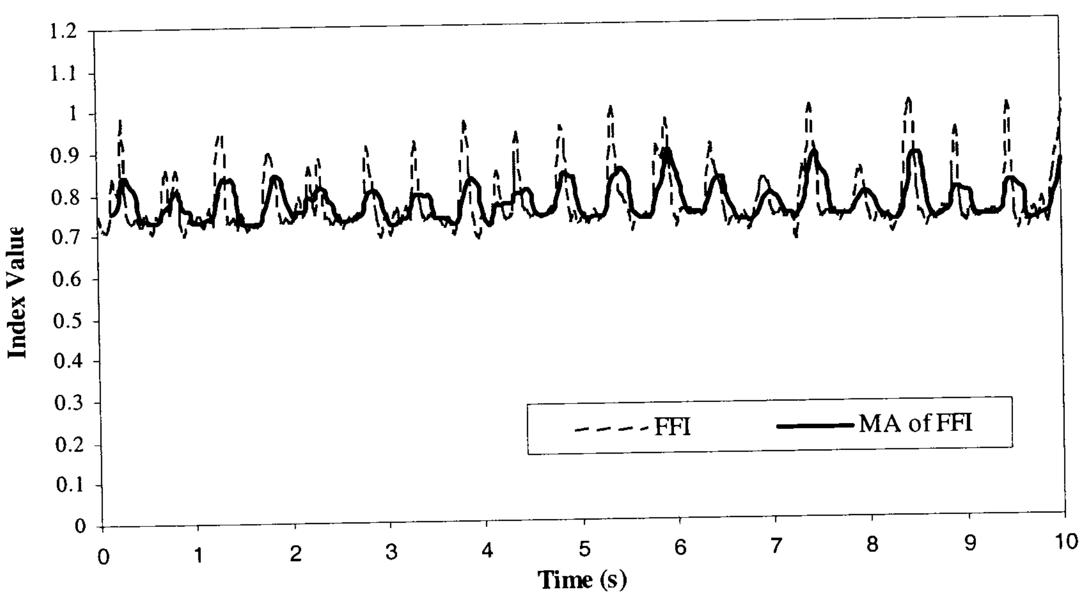



Figure D.76 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at outer race, 1750 rpm and 2 hp motor load.

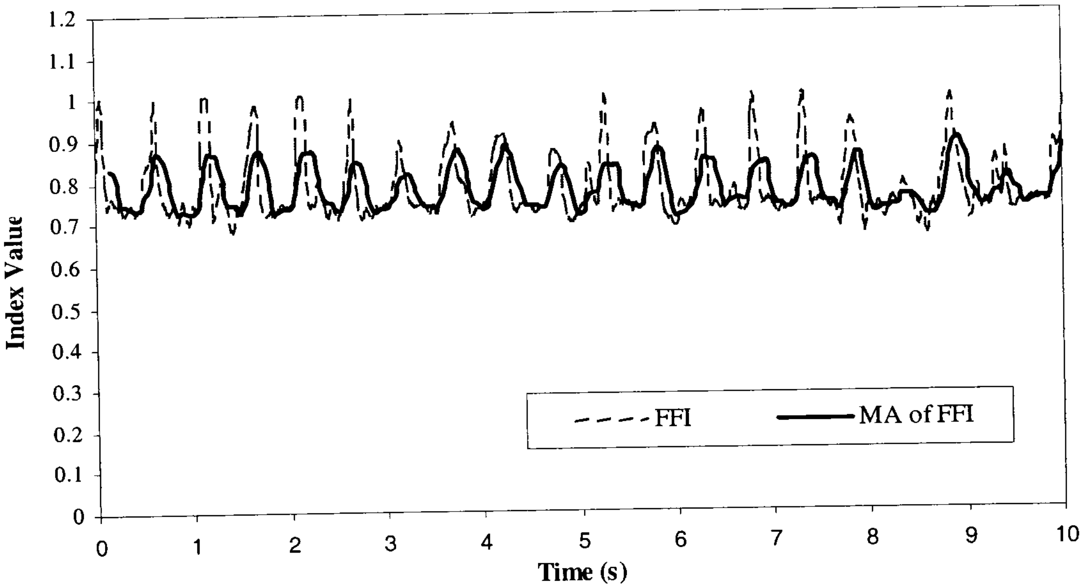



Figure D.77 Fuzzy fused index and its moving average shown for fan end bearing, 0.007" fault diameter at outer race, 1730 rpm and 3 hp motor load.

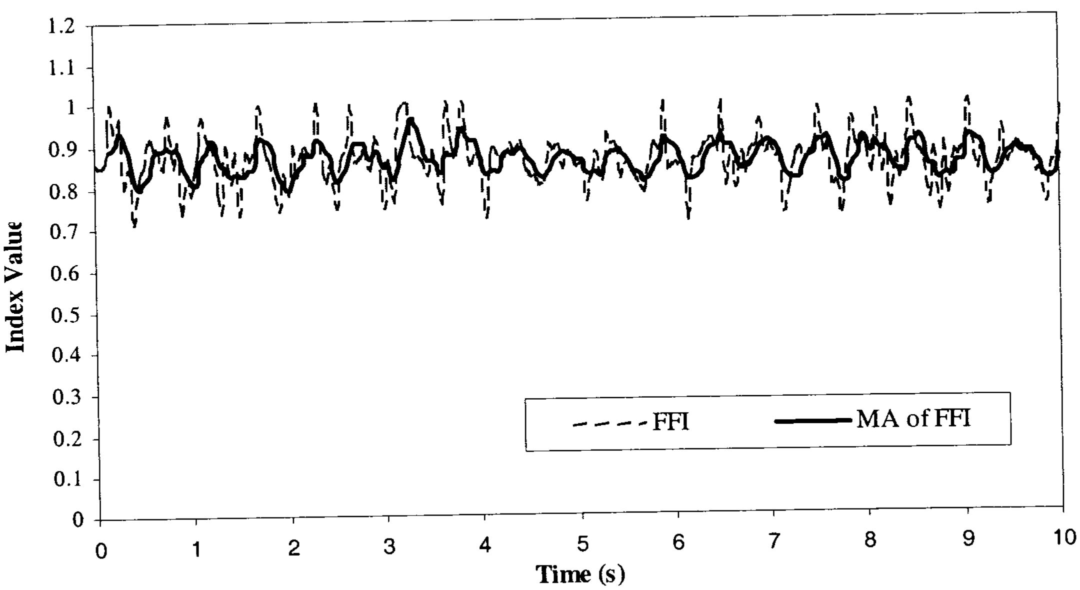



Figure D.78 Fuzzy fused index and its moving average shown for fan end bearing, 0.014" fault diameter at outer race, 1797 rpm and 0 hp motor load.

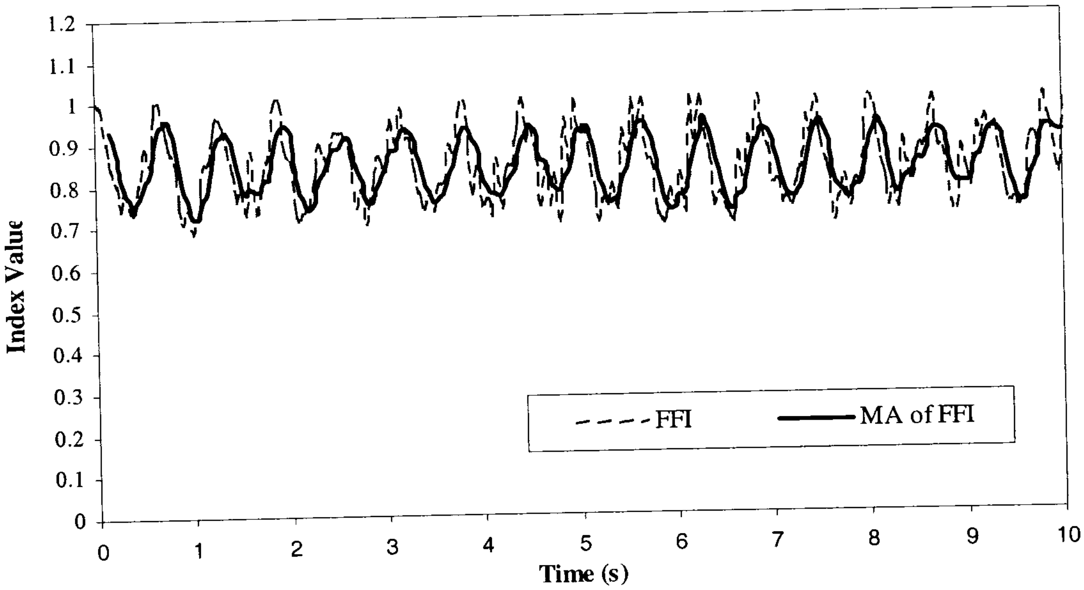



Figure D.79 Fuzzy fused index and its moving average shown for fan end bearing, 0.021" fault diameter at outer race, 1797 rpm and 0 hp motor load.

# Appendix E

## MATLAB Programs *Training* and *Diagnosis*

```
%-----------------------------------------------------------------------
% File name Training.m
% To train the Hidden Markov Model
% ----------------------------------------------------------------------
%% This is a template program that needs to be run through few loops to
%% cover all training stages. Some interactions are required during
%% training.
%% Starting with reference codebook, then Viterbi and Baum Welch
%% methods.
%% Template should be modified for different applications (tool and
%% vibration). The example shown below is related to the bearing
%% problem.
%-----------------------------------------------------------------------
% Clear values from memory.
clear average_scale
clear d
clear b1
clear b2
clear b3;
clear b4;
clear b5;
clear scale
clear norm_scale
clear x1
clear x2
clear x3
clear x4
clear f1
clear f2
clear f3
clear f4
clear f5
clear x5
clear length(x1)
clear length (x2)
clear length(x3)
clear length(x4)
clear length(x5)
clear dis
clear mindis
clear ref
clear prob_model
clear new_code
clear ac
clear scales
clear norm_scales
clear mesh
clear specgram
clear sum_norm_scale
clear count_symbol
clear bb
clear new_bb
clear q
clear a
clear aa
clear delta
clear listdelta
```

```
clear maximum
clear state
clear ddis
clear sum_ddis
clear new_aa
clear Index
clear max
%clear store % cacel effect after each state
% Select the bearing location
Bearing='DE'
signal =3; % to determine reference vectors and store them
nb=0; %
index=220;%index of file for each symptom
% Load file and read data
load('a237.mat');
d=X237_DE_time;
% Trained data for Drive End (DE) bearing
if (Bearing=='DE'&signal==1)% signal=1 i.e. stage 1 of the training
 % Codebook defines reference symbols
v=17;
% Define number of States
N=5;
% Choose initial reference vectors based on collected data.
% Reference vector 1
code(1,1)=1;
code(1,2)=.347;
code(1,3)=.140;
code(1,4)=.001;
code(1,5)=.003;
% Reference vetor 2
code(2,1)=.167;
code(2,2)=.362;
code(2,3)=.97;
code(2,4)=1;
code(2,5)=.039;
% Reference vector 3
code(3,1)=.124;
code(3,2)=.265;
code(3,3)=.476;
code(3,4)=1;
code(3,5)=.022;
% Reference vector 4
code(4,1)=.04;
code(4,2)=.028;
code(4,3)=.43;
code(4,4)=1;
code(4,5)=.009;
% Reference vector 5
code(5,1)=.092;
code(5,2)=.045;
code(5,3)=.924;
code(5,4)=1;
code(5,5)=.04;
% Reference vector 6
code(6,1)=.069;
code(6,2)=.022;
code(6,3)=.59;
```

```
code(6,4)=1;
code(6,5)=.025;
% Reference vector 7
code(7,1)=.097;
code(7,2)=.206;
code(7,3)=.58;
code(7,4)=1;
code(7,5)=.017;
% Reference vector 8
code(8,1)=.310
code(8,2)=.16;
code(8,3)=.72;
code(8,4)=1;
code(8,5)=.015;
% Reference vector 9
code(9,1)=.221;
code(9,2)=.06;
code(9,3)=.15;
code(9,4)=1;
code(9,5)=.075;
% Reference vector 10
code(10,1)=.08;
code(10,2)=.024;
code(10,3)=1;
code(10,4)=.45;
code(10,5)=.002;
% Reference vector 11
code(11,1)=.20;
code(11,2)=.22;
code(11,3)=.36;
code(11,4)=1;
code(11,5)=.028;
% Reference vector 12
code(12,1)=.1;
code(12,2)=.038;
code(12,3)=.39;
code(12,4)=1;
code(12,5)=.0342;
% Reference vector 13
code(13,1)=.14;
code(13,2)=.042;
code(13,3)=.83;
code(13,4)=1;
code(13,5)=.01;
% Reference vector 14
code(14,1)=.38;
code(14,2)=.385;
code(14,3)=1;
code(14,4)=.15;
code(14,5)=.002;
% Reference vector 15
code(15,1)=.815;
code(15,2)=.499;
code(15,3)=1;
code(15,4)=.172;
code(15,5)=.004;
% Reference vector 16
```

```
code(16,1)=1;
code(16,2)=.477;
code(16,3)=.808;
code(16,4)=.155;
code(16,5)=.007;
% Reference vector 17
code(17,1)=.001;
code(17,2)=.008;
code(17,3)=1;
code(17,4)=.36;
code(17,5)=.006;
end;
% Define the trained data for Fan End (FE) bearing.
if (Bearing=='FE'&signal==1)
% Codebook defines reference symbols
v=17;
% Define number of States
N=4;
%% Choose initial reference vectors based on collected data.
% Reference vector 1
code(1,1)=1;
code(1,2)=.347;
code(1,3)=.140;
code(1,4)=.001;
code(1,5)=.003;
% Reference vector 2
code(2,1)=.185;
code(2,2)=.341;
code(2,3)=.121;
code(2,4)=1;
code(2,5)=.051;
% Reference vector 3
code(3,1)=.362;
code(3,2)=.759;
code(3,3)=.138;
code(3,4)=1;
code(3,5)=.077;
% Reference vector 4
code(4,1)=.261;
code(4,2)=.038;
code(4,3)=.263;
code(4,4)=1;
code(4,5)=.164;
% Reference vector 5
code(5,1)=.754;
code(5,2)=.144;
code(5,3)=.263;
code(5,4)=1;
code(5,5)=.166;
% Reference vector 6
code(6,1)=.491;
code(6,2)=.133;
code(6,3)=.195;
code(6,4)=1;
code(6,5)=.094;
% Reference vector 7
code(7,1)=1;
```

```
code(7,2)=.284;
code(7,3)=.169;
code(7,4)=.775;
code(7,5)=.128;
% Reference vector 8
code(8,1)=.063;
code(8,2)=.060;
code(8,3)=.104;
code(8,4)=1;
code(8,5)=.019;
% Reference vector 9
code(9,1)=.406;
code(9,2)=.309;
code(9,3)=.414;
code(9,4)=1;
code(9,5)=.458;
% Reference vector 10
code(10,1)=.179;
code(10,2)=.044;
code(10,3)=.271;
code(10,4)=1;
code(10,5)=.374;
% Reference vector 11
code(11,1)=.048;
code(11,2)=.038;
code(11,3)=.745;
code(11,4)=1;
code(11,5)=.193;
% Reference vector 12
code(12,1)=.081;
code(12,2)=.054;
code(12,3)=.413;
code(12,4)=1;
code(12,5)=.259;
% Reference vector 13
code(13,1)=.009
code(13,2)=.011;
code(13,3)=.211;
code(13,4)=1;
code(13,5)=.021;
% Reference vector 14
code(14,1)=.045;
code(14,2)=.020;
code(14,3)=.154;
code(14,4)=1;
code(14,5)=.052;
% Reference vector 15
code(15,1)=.181;
code(15,2)=.037;
code(15,3)=.111;
code(15,4)=1;
code(15,5)=.112;
% Reference vector 16
code(16,1)=.193;
code(16,2)=.102;
code(16,3)=.103;
code(16,4)=1;
```

```
code(16,5)=.071;
% Reference vector 17
code(17,1)=.081;
code(17,2)=.282;
code(17,3)=.313;
code(17,4)=1;
code(17,5)=.057;
end;
 if signal ==2|3 % stage 2 of the training
     if Bearing=='DE'
  % Codebook defines reference symbols
v=17;
% Define number of States
N=5;
% These reference vectors are prepared as per stage 1 of the training
% Reference vector 1
code(1,1)=1;
code(1,2)=.347;
code(1,3)=.140;
code(1,4)=.001;
code(1,5)=.003;
% Reference vector 2
code(2,1)=.163;
code(2,2)=.359;
code(2,3)=.918;
code(2,4)=1;
code(2,5)=.033;
% Reference vector 3
code(3,1)=.124;
code(3,2)=.265;
code(3,3)=.476;
code(3,4)=1;
code(3,5)=.022;
% Reference vector 4
code(4,1)=.07;
code(4,2)=.038;
code(4,3)=.39;
code(4,4)=1;
code(4,5)=.009;
% Reference vector 5
code(5,1)=.092;
code(5,2)=.045;
code(5,3)=.924;
code(5,4)=1;
code(5,5)=.04;
% Reference vector 6
code(6,1)=.076;
code(6,2)=.04;
code(6,3)=.59;
code(6,4)=1;
code(6,5)=.025;
% Reference vector 7
code(7,1)=.097;
code(7,2)=.206;
code(7,3)=.58;
code(7,4)=1;
code(7,5)=.017;
```

```
% Reference vector 8
code(8,1)=.340
code(8,2)=.16;
code(8,3)=.75;
code(8,4)=1;
code(8,5)=.015;
% Reference vector 9
code(9,1)=.221;
code(9,2)=.045;
code(9,3)=.181;
code(9,4)=1;
code(9,5)=.055;
% Reference vector 10
code(10,1)=.064;
code(10,2)=.022;
code(10,3)=1;
code(10,4)=.399;
code(10,5)=.002;
% Reference vector 11
code(11,1)=.228;
code(11,2)=.246;
code(11,3)=.377;
code(11,4)=1;
code(11,5)=.026;
% Reference vector 12
code(12,1)=.07;
code(12,2)=.038;
code(12,3)=.39;
code(12,4)=1;
code(12,5)=.0342;
% Reference vector 13
code(13,1)=.14;
code(13,2)=.042;
code(13,3)=.83;
code(13,4)=1;
code(13,5)=.01;
% Reference vector 14
code(14,1)=.416;
code(14,2)=.385;
code(14,3)=1;
code(14,4)=.171;
code(14,5)=.005;
% Reference vector 15
code(15,1)=.815;
code(15,2)=.499;
code(15,3)=1;
code(15,4)=.172;
code(15,5)=.004;
% Reference vector 16
code(16,1)=1;
code(16,2)=.477;
code(16,3)=.808;
code(16,4)=.155;
code(16,5)=.007;
% Reference vector 17
code(17,1)=.001;
code(17,2)=.008;
```

```
code(17,3)=1;
code(17,4)=.447;
code(17,5)=.006;
end;
if Bearing=='FE'
% Codebook defines reference symbols
% These reference vectors are prepared as per stage 1 of the training
v=17;
% Define number of States
N=4;
% Reference vector 1
code(1,1)=1;
code(1,2)=.347;
code(1,3)=.140;
code(1,4)=.001;
code(1,5)=.003;
% Reference vector 2
code(2,1)=.185;
code(2,2)=.341;
code(2,3)=.121;
code(2,4)=1;
code(2,5)=.051;
% Reference vector 3
code(3,1)=.362;
code(3,2)=.759;
code(3,3)=.138;
code(3,4)=1;
code(3,5)=.077;
% Reference vector 4
code(4,1)=.261;
code(4,2)=.038;
code(4,3)=.263;
code(4,4)=1;
code(4,5)=.164;
% Reference vector 5
code(5,1)=.754;
code(5,2)=.144;
code(5,3)=.263;
code(5,4)=1;
code(5,5)=.166;
% Reference vector 6
code(6,1)=.491;
code(6,2)=.133;
code(6,3)=.195;
code(6,4)=1;
code(6,5)=.094;
% Reference vector 7
code(7,1)=1;
code(7,2)=.284;
code(7,3)=.169;
code(7,4)=.775;
code(7,5)=.128;
% Reference vector 8
code(8,1)=.073;
code(8,2)=.052;
code(8,3)=.104;
code(8,4)=1;
```

```
code(8,5)=.019;
% Reference vector 9
code(9,1)=.406;
code(9,2)=.309;
code(9,3)=.414;
code(9,4)=1;
code(9,5)=.458;
% Reference vector 10
code(10,1)=.179;
code(10,2)=.044;
code(10,3)=.271;
code(10,4)=1;
code(10,5)=.374;
% Reference vector 11
code(11,1)=.048;
code(11,2)=.038;
code(11,3)=.745;
code(11,4)=1;
code(11,5)=.193;
% Reference vector 12
code(12,1)=.081;
code(12,2)=.054;
code(12,3)=.413;
code(12,4)=1;
code(12,5)=.259;
% Reference vector 13
code(13,1)=.009
code(13,2)=.011;
code(13,3)=.211;
code(13,4)=1;
code(13,5)=.021;
% Reference vector 14
code(14,1)=.045;
code(14,2)=.022;
code(14,3)=.154;
code(14,4)=1;
code(14,5)=.052;
% Reference vector 15
code(15,1)=.181;
code(15,2)=.037;
code(15,3)=.111;
code(15,4)=1;
code(15,5)=.112;
% Reference vector 16
code(16,1)=.193;
code(16,2)=.102;
code(16,3)=.103;
code(16,4)=1;
code(16,5)=.071;
% Reference vector 17
code(17,1)=.081;
code(17,2)=.282;
code(17,3)=.313;
code(17,4)=1;
code(17,5)=.057;
     end;
 end;
```

```
% fs is the sampling frequency and fss is the Nyquisit frequency
fs=12000;
fss=fs/2;
% Q number of filters in bankfilter
Q=5;
w1=100;
w2=1100;
for i=1:Q
%%%if i==Q
%%% w1=6500;
%%% w2=7100;
%%%end;
% Filtering the noise using a high pass Butterworth filter where wl is
the cut-off frequency
w=[w1 w2];
c=fir1(3000,w/fss,'bandpass',hanning(3001));
% Filter the noise using a bank of banspass filters
% 'b' is the final filtered vector
if i==1;
    b1=filter(c,1,d);
end;
if i==2;
    b2=filter(c,1,d);
end;
if i==3;
    b3=filter(c,1,d);
end;
if i==4;
    b4=filter(c,1,d);
end;
if i==5
    b5=filter(c,1,d);
end;

w1=w1+1000;
w2=w2+1000;
end;
% Specifying the number of samples in a mini group.
space=3000;

for nn=1:1
% Specifying the starting and the end points for each sub-group

k1=nn+1;
k2=nn;
size=k1*space;
start=k2*space;
% Nbs is the number of observations
Nbs=20;
% Initialize the number of times each scale is maximum frequency band)
nb_max_scale1=0;
nb_max_scale2=0;
nb_max_scale3=0;
nb_max_scale4=0;
nb_max_scale5=0;
linda_ref=1; %% to be used in Baum Welch method
for iiii=1:Nbs
```

```
% Calculate power % time or frequency domain will give the same result.
[x1,f1]=psd(b1(start+(iiii-1)*space:start+iiii*space),space,fs);
[x2,f2]=psd(b2(start+(iiii-1)*space:start+iiii*space),space,fs);
[x3,f3]=psd(b3(start+(iiii-1)*space:start+iiii*space),space,fs);
[x4,f4]=psd(b4(start+(iiii-1)*space:start+iiii*space),space,fs);
[x5,f5]=psd(b5(start+(iiii-1)*space:start+iiii*space),space,fs);

% calculate the average power of observation in every scale (filtering
% range)
scale1=(sum((x1))/length(x1));
scale2=(sum((x2))/length(x2));
scale3=(sum((x3))/length(x3));
scale4=(sum((x4))/length(x4));
scale5=(sum((x5))/length(x5));
% Put values into matrix
scale(iiii,1)=scale1;
scale(iiii,2)=scale2;
scale(iiii,3)=scale3;
scale(iiii,4)=scale4;
scale(iiii,5)=scale5;
scales(1)=scale1;
scales(2)=scale2;
scales(3)=scale3;
scales(4)=scale4;
scales(5)=scale5;
sum_scale=sum(scales);
% Compute percentage of each scale with respect to sum
scales(1)=scale1/sum_scale;
scales(2)=scale2/sum_scale;
scales(3)=scale3/sum_scale;
scales(4)=scale4/sum_scale;
scales(5)=scale5/sum_scale;
max_scale=max(scales);
if scale1<0
    max_scale=min(scales);
end;
if max_scale==scale1
    nb_max_scale1=nb_max_scale1+1;
end
if max_scale==scale2
    nb_max_scale2=nb_max_scale2+1;
end;
if max_scale==scale3
    nb_max_scale3=nb_max_scale3+1;
end;
if max_scale==scale4
    nb_max_scale4=nb_max_scale4+1;
end;
if max_scale==scale5
    nb_max_scale5=nb_max_scale5+1;
end;
% Normalize w.r.t to maximum scale already calculated as a percentage
norm_scale1=scales(1)/max_scale;
norm_scale2=scales(2)/max_scale;
norm_scale3=scales(3)/max_scale;
norm_scale4=scales(4)/max_scale;
norm_scale5=scales(5)/max_scale;
```

```
%Put values into matrix
norm_scale(iiii,1)=norm_scale1;
norm_scale(iiii,2)=norm_scale2;
norm_scale(iiii,3)=norm_scale3;
norm_scale(iiii,4)=norm_scale4;
norm_scale(iiii,5)=norm_scale5;
norm_scales(1)=norm_scale1;
norm_scales(2)=norm_scale2;
norm_scales(3)=norm_scale3;
norm_scales(4)=norm_scale4;
norm_scales(5)=norm_scale5;
end;
for i=1:Nbs
    i;
    for j=1:Q
        scale(i,j);
        norm_scale(i,j);
        x_axis(j)=j;
    end;
end;
% Calculate the distance between observed symbol and reference symbols
to
% find the closest vector.
for i=1:Nbs
    minidis=1000000;
        for ii=1:v
        sum_ddis(ii)=0;
        dis(i,ii)=0;
        sum_norm_scale(i)=0;
        for iii=1:Q
   dis(i,ii)=dis(i,ii)+abs(norm_scale(i,iii )-code(ii,iii));
   ddis(i,ii)=(abs(norm_scale(i,iii )-code(ii,iii)))/code(ii,iii);
   sum_norm_scale(i)=sum_norm_scale(i)+norm_scale(i,iii);
   sum_ddis(ii)=sum_ddis(ii)+ddis(i,ii);
        end;
  if ((dis(i,ii))<minidis);
        minidis=(dis(i,ii));
        mindis(i)=minidis;% mindis is the minimum distance between the
observation symbol and the codebook
        ref(i)=ii; % ref contains the codebook number
    end;
    end;
  %% Define an acceptable convergence percentage. In this case 15% was
taken
   if signal==1 % only for first stage of training
    if ((sum_ddis(ref(i)))/(Q))>.15;
      sum_ddis(ref(i))/Q
  'codebook updated'
  % If convergence is not satisfied, then calculate the new code based
on the
  % new centroid.
      for iii=1:Q;
      code(ref(i),iii)=(code(ref(i),iii)+norm_scale(i,iii))/2;
      end;
  else
      % sum_ddis(ref(i))/Q
      'codebook convergence'
```

```
  end;
   end;
end;
% Store observations for each file of every symptom ... make sure all
% data are collected before starting analysis
for ddd=1:Nbs
store(index+ddd)=ref(ddd); % nb is the file index of each file
end;
end; % close "nn" loop
init=0;
if signal==3
% Restore store matrix in ref matrix
for i=1:length(store)
    ref(i)=store(i);
end;
Nbs=length(store);
for i=1:v
count_symbol(i)=0;
end;
for i=1:N
    q(i)=0;
end;
if Bearing=='DE'
% Take average occurrence of observations to speed up convergence
if init==0
aa(1,1)=.75;
aa(1,2)=.1;
aa(1,3)=.05;
aa(1,4)=.05;
aa(1,5)=.05;
aa(2,1)=.01;
aa(2,2)=.70;
aa(2,3)=.20;
aa(2,4)=.05;
aa(2,5)=.05;
aa(3,1)=0.01;
aa(3,2)=0.1;
aa(3,3)=.75;
aa(3,4)=.1;
aa(3,5)=.04;
aa(4,1)=.01;
aa(4,2)=.10;
aa(4,3)=.10;
aa(4,4)=.70;
aa(4,5)=.09;
aa(5,1)=.01;
aa(5,2)=.01;
aa(5,3)=.01;
aa(5,4)=.01;
aa(5,5)=.96;
bb(1,1)=.92;   % probability to be in state 1 with symbol 1
bb(1,2)=.005 ; % Probability to be in state 1 with symbol 2
bb(1,3)=.005 ; % Probability to be in state 1 with symbol 3
bb(1,4)=.005; % Probability to be in state 1 with symbol 4
bb(1,5)=.005 ;
bb(1,6)=.005  ;
bb(1,7)=.005 ;
```

```
bb(1,8)=.005  ;
bb(1,9)=.005  ;
bb(1,10)=.005 ;
bb(1,11)=.005 ;
bb(1,12)=.005 ;
bb(1,13)=.005  ;
bb(1,14)=.005 ;
bb(1,15)=.005  ;
bb(1,16)=.005  ;
bb(1,17)=.005  ;
bb(2,1)=.01 ;
bb(2,2)=.25 ;
bb(2,3)=.1;
bb(2,4)=.47;
bb(2,5)=.01;
bb(2,6)=.01;
bb(2,7)=.01;
bb(2,8)=.01;
bb(2,9)=.01;
bb(2,10)=.01;
bb(2,11)=.01;
bb(2,12)=.01;
bb(2,13)=.01;
bb(2,14)=.01;
bb(2,15)=.01;
bb(2,16)=.01;
bb(2,17)=.01;
bb(3,1)=.01;
bb(3,2)=.027;
bb(3,3)=.01;
bb(3,4)=.01;
bb(3,5)=.11;
bb(3,6)=.182;
bb(3,7)=0.076;
bb(3,8)=.145;
bb(3,9)=.289;
bb(3,10)=.01;
bb(3,11)=.01;
bb(3,12)=.013;
bb(3,13)=.068;
bb(3,14)=.01;
bb(3,15)=.01;
bb(3,16)=.01;
bb(3,17)=.01;
bb(4,1)=.01;
bb(4,2)=.01;
bb(4,3)=.01;
bb(4,4)=.155;
bb(4,5)=.013;
bb(4,6)=.178;
bb(4,7)=.01;
bb(4,8)=.01;
bb(4,9)=.01;
bb(4,10)=.178;
bb(4,11)=.178;
bb(4,12)=.178;
bb(4,13)=.178;
```

```
bb(4,14)=.01;
bb(4,15)=.01;
bb(4,16)=.01;
bb(4,17)=.01;
bb(5,1)=.01;
bb(5,2)=.01;
bb(5,3)=.01;
bb(5,4)=.01;
bb(5,5)=.01;
bb(5,6)=.01;
bb(5,7)=.01;
bb(5,8)=.01;
bb(5,9)=.01;
bb(5,10)=.01;
bb(5,11)=.01;
bb(5,12)=.01;
bb(5,13)=.01;
bb(5,14)=.25;
bb(5,15)=.125;
bb(5,16)=.125;
bb(5,17)=.37;
end;
for i=1:Nbs;
     if ref(i)==1
         count_symbol(1)=count_symbol(1)+1;
      end;
     if ref(i)==2
         count_symbol(2)=count_symbol(2)+1;
        q(2)=q(2)+1;
     end;
     if ref(i)==3
         count_symbol(3)=count_symbol(3)+1;
     end;
     if ref(i)==4
         count_symbol(4)=count_symbol(4)+1;
     end;
     if ref(i)==5
         count_symbol(5)=count_symbol(5)+1;
     end;
     if ref(i)==6
         count_symbol(6)=count_symbol(6)+1;
     end;
     if ref(i)==7
         count_symbol(7)=count_symbol(7)+1;
      end;
     if ref(i)==8
         count_symbol(8)=count_symbol(8)+1;
     end;
     if ref(i)==9
         count_symbol(9)=count_symbol(9)+1;
     end;
     if ref(i)==10
         count_symbol(10)=count_symbol(10)+1;
      end;
     if ref(i)==11
         count_symbol(11)=count_symbol(11)+1;
     end;
```

```
    if ref(i)==12
        count_symbol(12)=count_symbol(12)+1;
     end;
    if ref(i)==13
        count_symbol(13)=count_symbol(13)+1;
     end;
    if ref(i)==14
        count_symbol(14)=count_symbol(14)+1;
     end;
    if ref(i)==15
        count_symbol(15)=count_symbol(15)+1;
     end;
    if ref(i)==16
        count_symbol(16)=count_symbol(16)+1;
     end;
    if ref(i)==17
        count_symbol(17)=count_symbol(17)+1;
    end;
end;
end;
% Reevaluate the probability of observing symbol v in state j using the
% learning data
if Bearing=='FE'
    % Take average occurrence of observations to speed up convergence
    if init==0
aa(1,1)=.8;
aa(1,2)=.1;
aa(1,3)=.05;
aa(1,4)=.05;
aa(2,1)=0.05;
aa(2,2)=.80;
aa(2,3)=.1;
aa(2,4)=.05;
aa(3,1)=0.01;
aa(3,2)=0.01;
aa(3,3)=.94;
aa(3,4)=.04;
aa(4,1)=.01;
aa(4,2)=.08 ;
aa(4,3)=.01;
aa(4,4)=.9;
bb(1,1)=.84; % probability to be in state 1 with symbol 1
bb(1,2)=.01; % Probability to be in state 1 with symbol 2
bb(1,3)=.01; % Probability to be in state 1 with symbol 3
bb(1,4)=.01; % Probability to be in state 1 with symbol 4
bb(1,5)=.01;
bb(1,6)=.01;
bb(1,7)=.01;
bb(1,8)=.01;
bb(1,9)=.01;
bb(1,10)=.01;
bb(1,11)=.01;
bb(1,12)=.01;
bb(1,13)=.01;
bb(1,14)=.01;
bb(1,15)=.01;
bb(1,16)=.01;
```

```
bb(1,17)=.01;
bb(2,1)=.01;
bb(2,2)=.2;
bb(2,3)=.1;
bb(2,4)=.11;
bb(2,5)=.05;
bb(2,6)=.03;
bb(2,7)=.15;
bb(2,8)=.21;
bb(2,9)=.01;
bb(2,10)=.01;
bb(2,11)=.01;
bb(2,12)=.01;
bb(2,13)=.01;
bb(2,14)=.04;
bb(2,15)=.01;
bb(2,16)=.03;
bb(2,17)=.01;
bb(3,1)=.01;
bb(3,2)=.01;
bb(3,3)=.01;
bb(3,4)=.01;
bb(3,5)=.01;
bb(3,6)=.01;
bb(3,7)=.01;
bb(3,8)=.01;
bb(3,9)=.32;
bb(3,10)=.13;
bb(3,11)=.11;
bb(3,12)=.17;
bb(3,13)=.14;
bb(3,14)=.02;
bb(3,15)=.01;
bb(3,16)=.01;
bb(3,17)=.01;
bb(4,1)=.01;
bb(4,2)=.01;
bb(4,3)=.01;
bb(4,4)=.02;
bb(4,5)=.01;
bb(4,6)=.01;
bb(4,7)=.01;
bb(4,8)=.05;
bb(4,9)=.01;
bb(4,10)=.01;
bb(4,11)=.01;
bb(4,12)=.01;
bb(4,13)=.01;
bb(4,14)=.41;
bb(4,15)=.2;
bb(4,16)=.1;
bb(4,17)=.11;
     end;
 for i=1:Nbs;
     if ref(i)==1
         count_symbol(1)=count_symbol(1)+1;
      end;
```

```
    if ref(i)==2
        count_symbol(2)=count_symbol(2)+1;
     end;
    if ref(i)==3
        count_symbol(3)=count_symbol(3)+1;
     end;
    if ref(i)==4
        count_symbol(4)=count_symbol(4)+1;
     end;
    if ref(i)==5
        count_symbol(5)=count_symbol(5)+1;
     end;
    if ref(i)==6
        count_symbol(6)=count_symbol(6)+1;
     end;
    if ref(i)==7
        count_symbol(7)=count_symbol(7)+1;
     end;
    if ref(i)==8
        count_symbol(8)=count_symbol(8)+1;
       end;
    if ref(i)==9
        count_symbol(9)=count_symbol(9)+1;
       end;
    if ref(i)==10
        count_symbol(10)=count_symbol(10)+1;
     end;
    if ref(i)==11
        count_symbol(11)=count_symbol(11)+1;
     end;
    if ref(i)==12
        count_symbol(12)=count_symbol(12)+1;
     end;
    if ref(i)==13
        count_symbol(13)=count_symbol(13)+1;
     end;
    if ref(i)==14
        count_symbol(14)=count_symbol(14)+1;
     end;
    if ref(i)==15
        count_symbol(15)=count_symbol(15)+1;
     end;
    if ref(i)==16
        count_symbol(16)=count_symbol(16)+1;
     end;
    if ref(i)==17
        count_symbol(17)=count_symbol(17)+1;
     end;
end;
end;
maxii=0;
for ppp=1:v
    if count_symbol(ppp)>maxii
        maxii=count_symbol(ppp);
        max_sym=ppp
    end;
end;
```

```
initial_prob=0;
pi1=1;
repeat=1;
while repeat < 11
%% Use Viterbi Algorithm to find the sates sequence
% Initialization
for i=1:N
    delta(1,i)=pi1*bb(i,ref(1));
end;
linda_ref=1; % Initialize the probability of the reference model
%%%for jj=1:Nbs
% Recursion
for t=2:Nbs
    for j=1:N
        for i=1:N
            listdelta(i)=delta(t-1,i)*aa(i,j);
       end;
 delta(t,j)=max(listdelta)*bb(j,ref(t));
    end;
end;
for t=1:Nbs
    maxi=0;
    for j=1:N
        if delta(t,j)>maxi;
        maxi=delta(t,j);
        ref_state=j;
        end;
     end;
    state(t)=ref_state;
end;
%% Find the probable state
for i=1:Nbs;
    if state(i)==1
       q(1)=q(1)+1;
    if i==1
    ini_state=1;
    end;
    end;
    if state(i)==2
       q(2)=q(2)+1;
    if i==1
    ini_state=2;
    end;
    end;
    if state(i)==3
      q(3)=q(3)+1;
    if i==1
    ini_state=3;
    end;
    end;
    if state(i)==4
      q(4)=q(4)+1;
    if i==1
    ini_state=4;
    end;
    end;
    if ref(i)==5
```

```
        q(5)=q(5)+1;
     if i==1
     ini_state=5;
     end;
     end;
end;
%max_count=max(count_symbol);
max_state=max(q);
for i=1:N;
    if max_state==q(i);
        if i==5
            qq=5;
        else
            if i==4
                qq=4;
            else
                if i==3
                    qq=3;
                else
                    if i==2
                        qq=2;
                    else
                        qq=1;
                    end;
                end;

            end;
        end;
    end;
end;
% Update values of b matrix (Welch Method)(probability of observing
% symbol v such that state is j
'qq'
qq
for ii=1:v
    if(count_symbol(ii)>0)
    bb(qq,ii)=(bb(qq,ii)+count_symbol(ii)/Nbs)/2; % define new b matrix
    end;
%if bb(qq,ii)==0
%    bb(qq,ii)=0.01;
%end;
end;
% Ensure sum of bb is equal to 1
if sum(bb(qq,1:v))==1
        'ok sum bb'
    else
        bb(qq,max_sym)=bb(qq,max_sym)-abs(1-sum(bb(qq,1:v)));
    end;
% Initialize the number of occurrences of states
countstate1=0;
countstate2=0;
countstate3=0;
countstate4=0;
countstate5=0;
% Set to zero
for i=1:N
    for j=1:N
```

```
        ac(i,j)=0;
    end;
end;
% Assume initail state was 1 (normal)
%aa(1,ini_state)=(aa(1,ini_state)+1/Nbs)/2;
for i=2:Nbs
    if state(i)==1
         countstate1=countstate1+1;
         if state(i)==state(i-1)
             ac(1,1)=ac(1,1)+1;
             aa(1,1)=(aa(1,1)+ac(1,1)/(Nbs-1))/2; % define new A matrix
         else
             ac(state(i-1),1)=ac(state(i-1),1)+1;
             aa(state(i-1),1)=(aa(state(i-1),1)+ac(state(i-1),1)/(Nbs-
1))/2;
         end;
    end;
    if state(i)==2
         countstate2=countstate2+1;
         if state(i)==state(i-1)
             ac(2,2)=ac(2,2)+1;
             aa(2,2)=(aa(2,2)+ac(2,2)/(Nbs-1))/2;
         else
             ac(state(i-1),2)=ac(state(i-1),2)+1;
             aa(state(i-1),2)=(aa(state(i-1),2)+ac(state(i-1),2)/(Nbs-
1))/2;
         end;
    end;
    if state(i)==3
         countstate3=countstate3+1;
         if state(i)==state(i-1)
             ac(3,3)=ac(3,3)+1;
             aa(3,3)=(aa(3,3)+ac(3,3)/(Nbs-1))/2;
         else
             ac(state(i-1),3)=ac(state(i-1),3)+1;
             aa(state(i-1),3)=(aa(state(i-1),3)+ac(state(i-1),3)/(Nbs-
1))/2;
         end;
    end;
    if state(i)==4
         countstate4=countstate4+1;
         if state(i)==state(i-1)
             ac(4,4)=ac(4,4)+1;
             aa(4,4)=(aa(4,4)+ac(4,4)/(Nbs-1))/2;
         else
             ac(state(i-1),4)=ac(state(i-1),4)+1;
             aa(state(i-1),4)=(aa(state(i-1),4)+ac(state(i-1),4)/(Nbs-
1))/2;
         end;
    end;
    if state(i)==5
         countstate5=countstate5+1;
         if state(i)==state(i-1)
             ac(5,5)=ac(5,5)+1;
             aa(5,5)=(aa(5,5)+ac(5,5)/(Nbs-1))/2;
         else
             ac(state(i-1),5)=ac(state(i-1),5)+1;
```

```
            aa(state(i-1),5)=(aa(state(i-1),5)+ac(state(i-1),5)/(Nbs-
1))/2;
         end;
    end;
end;
% Ensure sum of probabilities are equal to 1
for sss=1:N
    if sum(aa(sss,1:5))==1
        'ok sum aa'
    else
        if sum(aa(sss,1:5))>1
        aa(sss,sss)=aa(sss,sss)-abs(1-sum(aa(sss,1:5)));
        else
          aa(sss,sss)=aa(sss,sss)+abs(1-sum(aa(sss,1:5)));
        end;
    end;
end;
if Bearing == 'FE'
    aa(qq,5)=0;
end;
init=1; % change initialization condition
pi(1)=1;
pi(2)=0;
pi(3)=0;
pi(4)=0;
pi(5)=0;
% Initialization
for j=1:N
   alpha(1,j)=pi(j)*bb(j,ref(1));
end;

for t=1:Nbs-1
    for j=1:N;
        sum_alpha=0;
        for i=1:N
        sum_alpha=sum_alpha +alpha(t,i)*aa(i,j);
        end;
        alpha(t+1,j)=sum_alpha*bb(j,ref(t+1));
    end;
end;
prob_model(1)=alpha(Nbs,1);
prob_model(2)=alpha(Nbs,2);
prob_model(3)=alpha(Nbs,3);
prob_model(4)=alpha(Nbs,4);
prob_model(5)=alpha(Nbs,5);
prob_mod=sum(prob_model);
if initial_prob>0
    if ((log10(prob_mod)-
log10(initial_prob))/(log10(initial_prob)))<0.1
        'convergence'
        repeat=11;
        %xlswrite()  save matrices
    else
        initial_prob=prob_mod;
        'reestimate matrices'
    end;
end;
```

```
if initial_prob==0
    initial_prob=prob_mod;
end;

repeat=repeat+1;
end; % close loop on "repeat"
end; % close loop on "signal =3".
```

```
%------------------------------------------------------------------------
% File called Diagnosis.m
% Compute the probability of HMM
% -----------------------------------------------------------------------
% This is a template that can be used for other applications.
clear average_scale
clear d
clear b1
clear b2
clear b3;
clear b4;
clear b5;
clear scale
clear norm_scale
clear x1
clear x2
clear x3
clear x4
clear f1
clear f2
clear f3
clear f4
clear f5
clear x5
clear length(x1)
clear length (x2)
clear length(x3)
clear length(x4)
clear length(x5)
clear dis
clear mindis
clear ref
clear prob_model
clear new_code
clear scales
clear norm_scales
clear sum_norm_scale
clear count_symbol
clear bb
clear new_bb
clear q
clear a
clear aa
clear delta
clear listdelta
clear maximum
clear state
clear ddis
clear sum_ddis
clear new_aa
clear Index
% Give the bearing location
Bearing='FE'
% Load file and read data
load('af271.mat');
d=X271_FE_time;
% Trained data for Drive End (DE) bearing
```

```
if Bearing=='DE'
% Codebook defines reference symbols (based on training)
v=17;
% Define number of States
N=5;
% Reference code 1
code(1,1)=1;
code(1,2)=.347;
code(1,3)=.140;
code(1,4)=.001;
code(1,5)=.003;
% Reference code 2
code(2,1)=.163;
code(2,2)=.359;
code(2,3)=.918;
code(2,4)=1;
code(2,5)=.033;
% Reference code 3
code(3,1)=.124;
code(3,2)=.265;
code(3,3)=.476;
code(3,4)=1;
code(3,5)=.022;
% Reference code 4
code(4,1)=.07;
code(4,2)=.038;
code(4,3)=.39;
code(4,4)=1;
code(4,5)=.009;
% Reference code 5
code(5,1)=.092;
code(5,2)=.045;
code(5,3)=.924;
code(5,4)=1;
code(5,5)=.04;
% Reference code 6
code(6,1)=.076;
code(6,2)=.04;
code(6,3)=.59;
code(6,4)=1;
code(6,5)=.025;
% Reference code 7
code(7,1)=.097;
code(7,2)=.206;
code(7,3)=.58;
code(7,4)=1;
code(7,5)=.017;
% Reference code 8
code(8,1)=.340;
code(8,2)=.16;
code(8,3)=.75;
code(8,4)=1;
code(8,5)=.015;
% Reference code 9
code(9,1)=.221;
code(9,2)=.045;
code(9,3)=.181;
```

```
code(9,4)=1;
code(9,5)=.055;
% Reference code 10
code(10,1)=.064;
code(10,2)=.022;
code(10,3)=1;
code(10,4)=.399;
code(10,5)=.002;
% Reference code 11
code(11,1)=.228;
code(11,2)=.246;
code(11,3)=.377;
code(11,4)=1;
code(11,5)=.026;
% Reference code 12
code(12,1)=.07;
code(12,2)=.038;
code(12,3)=.39;
code(12,4)=1;
code(12,5)=.0342;
% Reference code 13
code(13,1)=.14;
code(13,2)=.042;
code(13,3)=.83;
code(13,4)=1;
code(13,5)=.01;
% Reference code 14
code(14,1)=.416;
code(14,2)=.385;
code(14,3)=1;
code(14,4)=.171;
code(14,5)=.005;
% Reference code 15
code(15,1)=.815;
code(15,2)=.499;
code(15,3)=1;
code(15,4)=.172;
code(15,5)=.004;
% Reference code 16
code(16,1)=1;
code(16,2)=.477;
code(16,3)=.808;
code(16,4)=.155;
code(16,5)=.007;
% Reference code 17
code(17,1)=.001;
code(17,2)=.008;
code(17,3)=1;
code(17,4)=.447;
code(17,5)=.006;
% Define the transition matrix aij--Initialization
aa(1,1)=.75;
aa(1,2)=.1;
aa(1,3)=.05;
aa(1,4)=.05;
aa(1,5)=.05;
aa(2,1)=.009;
```

```
aa(2,2)=.731;
aa(2,3)=.18;
aa(2,4)=.04;
aa(2,5)=.04;
aa(3,1)=0.01;
aa(3,2)=0.1;
aa(3,3)=.75;
aa(3,4)=.1;
aa(3,5)=.04;
aa(4,1)=.001;
aa(4,2)=.141;
aa(4,3)=.125;
aa(4,4)=.63;
aa(4,5)=.1;
aa(5,1)=.01;
aa(5,2)=.01;
aa(5,3)=.01;
aa(5,4)=.01;
aa(5,5)=.96;
end;
% Define pii (probability of being at state i at time 0)
pi1=1;
% Probability of observation symbol v such in state i
bb(1,1)=.92;   % probability to be in state 1 with symbol 1
bb(1,2)=.005 ; % Probability to be in state 1 with symbol 2
bb(1,3)=.005 ; % Probability to be in state 1 with symbol 3
bb(1,4)=.005; % Probability to be in state 1 with symbol 4
bb(1,5)=.005 ;
bb(1,6)=.005  ;
bb(1,7)=.005 ;
bb(1,8)=.005  ;
bb(1,9)=.005  ;
bb(1,10)=.005 ;
bb(1,11)=.005 ;
bb(1,12)=.005 ;
bb(1,13)=.005  ;
bb(1,14)=.005 ;
bb(1,15)=.005  ;
bb(1,16)=.005  ;
bb(1,17)=.005  ;
bb(2,1)=.01 ;
bb(2,2)=.25 ;
bb(2,3)=.083;
bb(2,4)=.483;
bb(2,5)=.01;
bb(2,6)=.054;
bb(2,7)=.01;
bb(2,8)=.01;
bb(2,9)=.01;
bb(2,10)=.01;
bb(2,11)=.01;
bb(2,12)=.01;
bb(2,13)=.01;
bb(2,14)=.01;
bb(2,15)=.01;
bb(2,16)=.01;
bb(2,17)=.01;
```

```
bb(3,1)=.01;
bb(3,2)=.027;
bb(3,3)=.01;
bb(3,4)=.01;
bb(3,5)=.11;
bb(3,6)=.182;
bb(3,7)=0.076;
bb(3,8)=.145;
bb(3,9)=.289;
bb(3,10)=.01;
bb(3,11)=.01;
bb(3,12)=.013;
bb(3,13)=.068;
bb(3,14)=.01;
bb(3,15)=.01;
bb(3,16)=.01;
bb(3,17)=.01;
bb(4,1)=.01;
bb(4,2)=.01;
bb(4,3)=.01;
bb(4,4)=.155;
bb(4,5)=.013;
bb(4,6)=.108;
bb(4,7)=.01;
bb(4,8)=.01;
bb(4,9)=.01;
bb(4,10)=.201;
bb(4,11)=.135;
bb(4,12)=.138;
bb(4,13)=.15;
bb(4,14)=.01;
bb(4,15)=.01;
bb(4,16)=.01;
bb(4,17)=.01;
bb(5,1)=.01;
bb(5,2)=.01;
bb(5,3)=.01;
bb(5,4)=.01;
bb(5,5)=.01;
bb(5,6)=.01;
bb(5,7)=.01;
bb(5,8)=.01;
bb(5,9)=.01;
bb(5,10)=.01;
bb(5,11)=.01;
bb(5,12)=.01;
bb(5,13)=.01;
bb(5,14)=.25;
bb(5,15)=.125;
bb(5,16)=.125;
bb(5,17)=.37;
% Define the trained data for Fan End (FE) bearing.
if Bearing=='FE'
% Codebook defines reference symbols
v=17;
% Define number of States
N=4;
```

```
% Reference vectors based on training
% Reference code 1
code(1,1)=1;
code(1,2)=.347;
code(1,3)=.140;
code(1,4)=.001;
code(1,5)=.003;
% Reference code 2
code(2,1)=.185;
code(2,2)=.341;
code(2,3)=.121;
code(2,4)=1;
code(2,5)=.051;
% Reference code 3
code(3,1)=.362;
code(3,2)=.759;
code(3,3)=.138;
code(3,4)=1;
code(3,5)=.077;
% Reference code 4
code(4,1)=.261;
code(4,2)=.038;
code(4,3)=.263;
code(4,4)=1;
code(4,5)=.164;
% Reference code 5
code(5,1)=.754;
code(5,2)=.144;
code(5,3)=.263;
code(5,4)=1;
code(5,5)=.166;
% Reference code 6
code(6,1)=.491;
code(6,2)=.133;
code(6,3)=.195;
code(6,4)=1;
code(6,5)=.094;
% Reference code 7
code(7,1)=1;
code(7,2)=.284;
code(7,3)=.169;
code(7,4)=.775;
code(7,5)=.128;
% Reference code 8
code(8,1)=.073;
code(8,2)=.052;
code(8,3)=.104;
code(8,4)=1;
code(8,5)=.019;
% Reference code 9
code(9,1)=.406;
code(9,2)=.309;
code(9,3)=.414;
code(9,4)=1;
code(9,5)=.458;
% Reference code 10
code(10,1)=.179;
```

```
code(10,2)=.044;
code(10,3)=.271;
code(10,4)=1;
code(10,5)=.374;
% Reference code 11
code(11,1)=.048;
code(11,2)=.038;
code(11,3)=.745;
code(11,4)=1;
code(11,5)=.193;
% Reference code 12
code(12,1)=.081;
code(12,2)=.054;
code(12,3)=.413;
code(12,4)=1;
code(12,5)=.259;
% Reference code 13
code(13,1)=.009;
code(13,2)=.011;
code(13,3)=.211;
code(13,4)=1;
code(13,5)=.021;
% Reference code 14
code(14,1)=.045;
code(14,2)=.022;
code(14,3)=.154;
code(14,4)=1;
code(14,5)=.052;
% Reference code 15
code(15,1)=.181;
code(15,2)=.037;
code(15,3)=.111;
code(15,4)=1;
code(15,5)=.112;
% Reference code 16
code(16,1)=.193;
code(16,2)=.102;
code(16,3)=.103;
code(16,4)=1;
code(16,5)=.071;
% Reference code 17
code(17,1)=.081;
code(17,2)=.282;
code(17,3)=.313;
code(17,4)=1;
code(17,5)=.057;
% Define the transition matrix aij--Initialization
aa(1,1)=.8;
aa(1,2)=.1;
aa(1,3)=.05;
aa(1,4)=.05;
aa(2,1)=0.01;
aa(2,2)=.81;
aa(2,3)=.1;
aa(2,4)=.08;
aa(3,1)=0.01;
aa(3,2)=0.01;
```

```
aa(3,3)=.94;
aa(3,4)=.04;
aa(4,1)=.01;
aa(4,2)=.08 ;
aa(4,3)=.01;
aa(4,4)=.9;
% Define pii (probability of being at state i at time 0)
pi1=1;
% Probability of observing symbol v while being in state i
bb(1,1)=.84; % probability to be in state 1 and observing symbol 1
bb(1,2)=.01; % Probability to be in state 1 and observing symbol 2
bb(1,3)=.01; % Probability to be in state 1 and observing symbol 3
bb(1,4)=.01; % Probability to be in state 1 and observing symbol 4
bb(1,5)=.01;
bb(1,6)=.01;
bb(1,7)=.01;
bb(1,8)=.01;
bb(1,9)=.01;
bb(1,10)=.01;
bb(1,11)=.01;
bb(1,12)=.01;
bb(1,13)=.01;
bb(1,14)=.01;
bb(1,15)=.01;
bb(1,16)=.01;
bb(1,17)=.01;
bb(2,1)=.01;
bb(2,2)=.2;
bb(2,3)=.1;
bb(2,4)=.11;
bb(2,5)=.05;
bb(2,6)=.03;
bb(2,7)=.15;
bb(2,8)=.21;
bb(2,9)=.01;
bb(2,10)=.01;
bb(2,11)=.01;
bb(2,12)=.01;
bb(2,13)=.01;
bb(2,14)=.04;
bb(2,15)=.01;
bb(2,16)=.03;
bb(2,17)=.01;
bb(3,1)=.01;
bb(3,2)=.01;
bb(3,3)=.01;
bb(3,4)=.01;
bb(3,5)=.01;
bb(3,6)=.01;
bb(3,7)=.01;
bb(3,8)=.01;
bb(3,9)=.32;
bb(3,10)=.13;
bb(3,11)=.11;
bb(3,12)=.17;
bb(3,13)=.14;
bb(3,14)=.02;
```

```
bb(3,15)=.01;
bb(3,16)=.01;
bb(3,17)=.01;
bb(4,1)=.01;
bb(4,2)=.01;
bb(4,3)=.01;
bb(4,4)=.02;
bb(4,5)=.01;
bb(4,6)=.01;
bb(4,7)=.01;
bb(4,8)=.05;
bb(4,9)=.01;
bb(4,10)=.01;
bb(4,11)=.01;
bb(4,12)=.01;
bb(4,13)=.01;
bb(4,14)=.41;
bb(4,15)=.2;
bb(4,16)=.1;
bb(4,17)=.11;
end;
% fs is the sampling frequency and fss is the Nyquisit frequency
fs=12000;
fss=fs/2;
% Q number of filters in bankfilter
Q=5;
w1=100;
w2=1100;
for i=1:Q
%%%if i==Q
%%%  w1=6500;
%%% w2=7100;
%%%end;
% Filtering the noise using a high pass Butterworth filter where w1 is
the cut-off frequency
w=[w1 w2];
c=fir1(3000,w/fss,'bandpass',hanning(3001));
% Filter the noise using a bnk of banspass filters
% 'b' is the final filtered vector
if i==1;
    b1=filter(c,1,d);
end;
if i==2;
    b2=filter(c,1,d);
 end;
if i==3;
    b3=filter(c,1,d);
 end;
if i==4;
    b4=filter(c,1,d);

end;
if i==5
    b5=filter(c,1,d);
 end;
w1=w1+1000;
w2=w2+1000;
```

```
end;
% Specifying the number of samples in a mini group.
space=3000;
% Specify sequence length
conseq=4;
% Specify number of sequences
nb_conseq=9;
for nn=1:nb_conseq
% Specifying the starting and the end points for each sub-group
k1=nn*conseq;
k2=(nn-1)*conseq+1;
size=k1*space;
start=k2*space;
% Nbs is the number of observations
Nbs=k1-k2+1;
% Initialize the number of times each scale is maximum frequency band)
nb_max_scale1=0;
nb_max_scale2=0;
nb_max_scale3=0;
nb_max_scale4=0;
nb_max_scale5=0;
linda_ref=1; %% to be used in Baum Welch method
for i=1:Nbs
% Calculate power % time or frequency domain will give the same result.
[x1,f1]=psd(b1(start+(i-1)*space:start+i*space),space,fs);
[x2,f2]=psd(b2(start+(i-1)*space:start+i*space),space,fs);
[x3,f3]=psd(b3(start+(i-1)*space:start+i*space),space,fs);
[x4,f4]=psd(b4(start+(i-1)*space:start+i*space),space,fs);
[x5,f5]=psd(b5(start+(i-1)*space:start+i*space),space,fs);
% calculate the average power of observation in every scale (filtering
% range)
scale1=(sum((x1))/length(x1));
scale2=(sum((x2))/length(x2));
scale3=(sum((x3))/length(x3));
scale4=(sum((x4))/length(x4));
scale5=(sum((x5))/length(x5));
% Put values into matrix
scale(i,1)=scale1;
scale(i,2)=scale2;
scale(i,3)=scale3;
scale(i,4)=scale4;
scale(i,5)=scale5;
scales(1)=scale1;
scales(2)=scale2;
scales(3)=scale3;
scales(4)=scale4;
scales(5)=scale5;
sum_scale=sum(scales);
% Compute percentage of each scale with respect to sum
scales(1)=scale1/sum_scale;
scales(2)=scale2/sum_scale;
scales(3)=scale3/sum_scale;
scales(4)=scale4/sum_scale;
scales(5)=scale5/sum_scale;
max_scale=max(scales);
if scale1<0
    max_scale=min(scales);
```

```
end;
if max_scale==scale1
    nb_max_scale1=nb_max_scale1+1;
end
if max_scale==scale2
    nb_max_scale2=nb_max_scale2+1;
end;
if max_scale==scale3
    nb_max_scale3=nb_max_scale3+1;
end;
if max_scale==scale4
    nb_max_scale4=nb_max_scale4+1;
end;
if max_scale==scale5
    nb_max_scale5=nb_max_scale5+1;
end;
% Normalize w.r.t to maximum scale already calculated as a percentage
norm_scale1=scales(1)/max_scale;
norm_scale2=scales(2)/max_scale;
norm_scale3=scales(3)/max_scale;
norm_scale4=scales(4)/max_scale;
norm_scale5=scales(5)/max_scale;
%Put values into matrix
norm_scale(i,1)=norm_scale1;
norm_scale(i,2)=norm_scale2;
norm_scale(i,3)=norm_scale3;
norm_scale(i,4)=norm_scale4;
norm_scale(i,5)=norm_scale5;
norm_scales(1)=norm_scale1;
norm_scales(2)=norm_scale2;
norm_scales(3)=norm_scale3;
norm_scales(4)=norm_scale4;
norm_scales(5)=norm_scale5;
%subplot(Nbs,1,i),plot(norm_scales)
norm_scales;
end;
for i=1:Nbs
    i;
    for j=1:Q
        scale(i,j);
        norm_scale(i,j);
        x_axis(j)=j;
    end;
end;
% Calculate the distance between observed symbol and reference symbols
% to find the closest vector.
for i=1:Nbs
    minidis=1000000;
        for ii=1:v
        sum_ddis(ii)=0;
        dis(i,ii)=0;
        sum_norm_scale(i)=0;
        for iii=1:Q
   dis(i,ii)=dis(i,ii)+abs(norm_scale(i,iii )-code(ii,iii));
   ddis(i,ii)=(abs(norm_scale(i,iii )-code(ii,iii)))/code(ii,iii);
   sum_norm_scale(i)=sum_norm_scale(i)+norm_scale(i,iii);
   sum_ddis(ii)=sum_ddis(ii)+ddis(i,ii);
```

```
        end;
    if ((dis(i,ii))<minidis);
        minidis=(dis(i,ii));
        mindis(i)=minidis;% mindis is the minimum distance between the
% observation symbol and the codebook
        ref(i)=ii; % ref contains the codebook number
    end;
    end;
  end;
pi(1)=1;
pi(2)=0;
pi(3)=0;
pi(4)=0;
pi(5)=0;
% Initialization
for j=1:N
   alpha(1,j)=pi(j)*bb(j,ref(1));
end;
for t=1:conseq-1
    for j=1:N;
        sum_alpha=0;
        for i=1:N
        sum_alpha=sum_alpha +alpha(t,i)*aa(i,j);
        end;
        alpha(t+1,j)=sum_alpha*bb(j,ref(t+1));
    end;
end;
prob_model(nn,1)=alpha(conseq,1);% probability of state 1
prob_model(nn,2)=alpha(conseq,2);
prob_model(nn,3)=alpha(conseq,3);
prob_model(nn,4)=alpha(conseq,4);
%prob_model(nn,5)=alpha(conseq,5);
end;
%success=xlswrite()
```