Development and Performance Evaluation of Prognostic Approaches for Technical Systems

zu Erlangung des akademischen Grades eines

DOKTORS DER INGENIEURWISSENSCHAFTEN (Dr.-Ing.)

der Fakultät für Maschinenbau

der Universität Paderborn

genehmigte

DISSERTATION

von

M.Sc. James Kuria Kimotho

aus Nairobi, Kenia

Tages des Kolloquiums: 10. November 2016
Referent: Prof. Dr.-Ing. habil. Walter Sextro
Korreferant: Prof. Dr.-Ing. Detmar Zimmer

Foreword

This work is a summary of four years of my research at the Chair of Mechatronics and Dynamics, University of Paderborn.

First and foremost, I would like to thank Prof. Dr.-Ing. habil Walter Sextro for showing confidence in me and giving me a chance to join his research group and also for being the first examiner of this thesis. His comments and constructive criticism guided me towards completion of my research and this dissertation. I also thank Prof. Dr.-Ing. Detmar Zimmer for being the second examiner and Prof. Dr.-Ing Iris Gräßler and Dr.-Ing Tobias Hemsel for being part of my examination committee.

I am very grateful to all my colleagues at the Chair of Mechatronics and Dynamics for providing me with a very friendly working environment and assisting me to continuously improve my German language. I acknowledge the great support accorded to me by the members of the design, optimization and reliability of mechatronic systems research group. I would like to particularly thank Tobias Meyer for our fruitful technical discussions, interesting suggestions that always provided a different dimension to my research and also for your friendship that made my stay in Germany interesting. I am very grateful to Dr.-Ing. Christoph Sondermann-Woelke for our initial extensive discussions on condition monitoring and prognostics in mechatronic systems. My gratitude also extends to Amelie Bender for your valuable comments on my dissertation and for sharing your technical experiences. I am grateful to Martin Liekenbroecker for the assistance in the laboratory. Special thanks also go to Dr.-Ing. Tobias Hemsel for your invaluable technical advice, encouragement and support throughout my research. I also thank Christian Lessmeier for providing experimental data and for our collaboration.

I acknowledge the support of the German Academic Exchange Service (DAAD) and the Ministry of Higher Education Science and Technology (MoHEST), Kenya for providing me with an opportunity to pursue my doctorate degree in Germany.

I am highly indebted to Priscillah Njeri and Fadhili Kimotho for your patience. You are the source of joy in my life.

Paderborn, November 2016

James Kuria Kimotho

Kurzfassung

Angesichts des steigenden Bedarfs zur Erhöhung der Zuverlässigkeit, Verfügbarkeit und Sicherheit technischer Systeme, wird eine große Anzahl von proaktiven Wartungsstrategien vorgeschlagen. Von größtem Interesse ist die Entwicklung von Strategien basierend auf Diagnose- und Prognosemethoden, bei denen die Wartung auf Basis des aktuellen und des vorhergesagten zukünftigen Zustands eines technischen Systems geplant wird. Darüber hinaus können Prognoseinformationen verwendet werden, um die Zuverlässigkeit von intelligenten mechatronischen Systemen zu steuern, um einen erfolgreichen Betrieb zu gewährleisten. Daher werden Methoden für eine zuverlässige und präzise Schätzung des aktuellen und zukünftigen Zustands des Systems benötigt. Mit dem Fortschritt in der Sensortechnik sind die Mehrheit der heutigen technischen Systeme mit einem Netzwerk von Sensoren für Zustands- oder Leistungsüberwachung ausgestattet. Dies führt zu einer gesteigerten Anwendung von maschinellen Lernverfahren in der Zustandsüberwachung und -prognose. In Abhängigkeit von den verfügbaren Sensordaten, können verschiedene Ansätze von maschinellen Lernverfahren für die Verwendung der Daten, angewendet werden. Allerdings fehlt ein Leitfaden zur Auswahl eines geeigneten Ansatzes für ein gegebenes System oder es ist noch nicht umfangreich untersucht worden. Daher stellt diese Arbeit einen Leitfaden für die Auswahl geeigneter Ansätze und maschineller Lernverfahren für ein gegebenes System, abhängig von den verfügbaren Zustandsüberwachungsdaten, bereit. Es werden fünf ausgewählte Prognoseansätze in Abhängigkeit von den verfügbaren Zustandsüberwachungsdaten und der Anwendung von maschinellen Lernverfahren innerhalb dieser Ansätze dargestellt. Da nicht alle extrahierten Merkmale relevante Informationen für die Diagnose und Prognose enthalten, werden Methoden für eine Merkmalauswahl vorgestellt. Die Ansätze werden anhand von Zustandsüberwachungsdaten von tatsächlichen Systemen evaluiert und validiert. Diese Informationen können als Anhaltspunkt für die Auswahl einer geeigneten Methode für eine bestimmte Systeme, in Abhängigkeit von den verfügbaren Zustandsüberwachungsdaten, dienen.

Abstract

As the need to improve reliability, availability and safety of technical systems increases, a large number of proactive maintenance strategies have been proposed. Of greater interest is the development of prognostic and health management strategies where maintenance is scheduled based on the current and the predicted future health state of a technical system. In addition, prognostic information can be used to control the reliability of intelligent mechatronic systems to ensure their mission objective is achieved. Therefore methodologies for estimating these current and future health states reliably and accurately are imperative. With the advancement in sensor technology, majority of the present day technical systems are installed with a network of sensors for condition or performance monitoring. This has led to the increased application of machine learning algorithms in condition monitoring. Depending on the sensor data available, different approaches for utilizing the data with machine learning algorithms can be applied. However, a guide for selecting the appropriate approach for a given system is either lacking or has not been explored extensively. Therefore, this work aims at providing a guide for selecting suitable approaches and machine learning algorithms for a given system depending on the available sensor data. Five approaches for prognostics and an ensemble of the best performing approaches are presented. Since the performance of machine learning algorithms is highly dependent on the input features, methods for feature extraction and selection are also presented. The approaches are evaluated and validated with run-to-failure condition monitoring data of actual systems. This information could serve as a guide for selecting the appropriate method for a given kind of system depending on the available condition monitoring data.

List of Publications

Parts of this thesis have been published in the following articles.

Peer-Reviewed Journal Articles

Kimotho, J. K.; Sextro, W.: Optimal Parameter Tuning for Multiclass Support Vector Machines in Machinery Health State Estimation. In: *Proceedings of Applied Mathematics and Mechanics,* Volume 14, pp 815-816, 2014

Kimotho, J. K.; Sondermann-Woelke, C.; Meyer, T.; Sextro, W.: Application of Event Based Decision Tree and Ensemble of Data Driven Methods for Maintenance Action Recommendation. In: *International Journal of Prognostics and Health Management,* Volume 4, 2013

Kimotho, J. K.; Sondermann-Woelke, C.; Meyer, T.; Sextro, W.: Machinery Prognostic Method Based on Multi-Class Support Vector Machines and Hybrid Differential Evolution–Particle Swarm Optimization. In: *Chemical Engineering Transactions,* Volume 33, pp 619-624, 2013

Meyer, T.; Sondermann-Woelke, C.; Kimotho, J. K.; Sextro, W.: Controlling the Remaining Useful Lifetime using Self-Optimization. *Chemical Engineering Transactions,* Volume 33, pp 625-630, 2013

Peer-Reviewed Articles in Conference Proceedings

Lessmeier, C.; Kimotho, J. K.; Zimmer, D.; Sextro, W.: Condition Monitoring of Bearing Damage in Electromechanical Drive Systems by Using Motor Current Signals of Electric Motors: A Benchmark Data Set for Data-Driven Classification. In: *Proceedings of the European Conference of the Prognostics and Health Management Society,* 2016

Kimotho, J. K.; Sextro, W.: Comparison and ensemble of temperature-based and vibration-based methods for machinery prognosis. In: *Proceedings of the Annual Conference of the Prognostics and Health Management Society,* 2015

Meyer, T.; Kimotho, J. K.; Sextro, W.: Anforderungen an Condition-Monitoring-Verfahren zur Nutzung im zuverlässigkeitsgeregelten Betrieb adaptiver Systeme. *27. Tagung Technische Zuverlässigkeit (TTZ 2015) - Entwicklung und Betrieb zuverlässiger Produkte,* 2015

Kimotho, J. K.; Sextro, W.: An approach for feature extraction and selection from non-trending data for machinery prognosis. In: *Proceedings of the Second European Conference of the Prognostics and Health Management Society,* Volume 5, 2014

Kimotho, J.K.; Meyer, T.; Sextro, W.: PEM fuel cell prognostics using particle filter with model parameter adaptation. In: *2014 IEEE International Conference on Prognostics and Health Management,* 2014

Non Peer-Reviewed Contributions in International Workshops

Kimotho, J.K.; Hemsel, T.; Sextro, W.: Application of Self-sensing in Prognostics and Health Management of Piezoelectric Transducers. *The 12th International Workshop on Piezoelectric Materials and Applications in Actuators (IWPMA 2015), Vilnius, Lithuania,* 2015

Kimotho, J.K.; Hemsel, T.; Sextro, W.: Application of Self-sensing in Monitoring Degradation of Piezoelectric Transducers. *The 11th Workshop on Direct and Inverse Problems in Piezoelectricity, Paderborn, Germany,* 2015

Table of Contents

Nomenclature

α	State model parameter 1
β	State model parameter 2
γ	Kernel parameter for kernel-based methods
σ	Standard deviation
A_m	Mean performance evaluation score
b	Machine learning algorithm bias
c	Class/cluster
ct	Centroid of features in a cluster
C	Regularization parameter
e_r	Relative error
d_j	Feature separation distance for feature selection
$f(\cdot)$	General function
FN	False negative (Number of predictions outside allowable negative error)
FP	False positive (Number of predictions outside allowable positive error)
HI	Health index
HI_a	Actual health index
HI_p	Predicted health index
HS	Health state
L_i	Percentage remaining useful life at health state i
n	Number of samples
N_c	Number of health states
N_T	Number of training data sets
o_k	System observation/measurement
p_j	Feature performance evaluation value for regression
P_i	Health state probability
q_k	System state
Q	Number of features
r	Signal residue
RUL	Remaining useful lifetime
RUL_a	Actual remaining useful lifetime

RUL_p	Predicted remaining useful lifetime
$s(t)$	Signal/sensor data
sf	Self-healing factor
t	Time
t_c	Current time
t_{EoL}	Time at end of life
T	Temperature
ΔT	Temperature change
$u(t)$	System input
$\mathbf{w}$	Support vector
w_i	Method weight in ensemble
$x(t)$	Extracted feature
$\mathbf{x}$	Feature matrix
$y(t)$	System output/target
y_p	Predicted output

Abbreviations and Acronyms

ANFIS	Adaptive Neural Fuzzy Inference System
ANN	Artificial Neural Networks
AR	Autoregressive
BPFI	Ball Pass Frequency of Inner Ring
BPFO	Ball Pass Frequency of Outer Ring
CI	Confidence Interval
CMB	Condition Based Maintenance
DE	Differential Evolution
DFT	Discrete Fourier Transform
DWT	Discrete Wavelet Transform
EMD	Empirical Mode Decomposition
ELM	Extreme Learning Machine
EoL	End of Life
FFT	Fast Fourier Transform
FMEA	Failure Mode Effect Analyis
FTF	Fundamental Train Trequency
HMM	Hidden Markov Model
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization
MA	Moving average
MAPE	Mean Absolute Percentage Error
MCSA	Motor Current Signature Analysis
MSE	Mean Square Error
PF	Particle Filter
PHM	Prognostic and Health Management
PSD	Power Spectral Density
PSO	Particle Swarm Optimization
RBF	Radial Basis Function
RF	Random Forests
RMS	Root Mean Square

SOM	Self-Organizing Maps Neural Networks
SVM	Support Vector Machines
TTF	Time To Failure

1 Introduction

Traditionally, preventive maintenance has been the most common maintenance strategy [1, 2]. In this strategy, maintenance tasks are carried out after elapsed time or operation that is based on statistical or historical failure times of a fleet of systems. In some cases, periodic manual inspection is carried out where the machine in question is periodically shut down and an inspection on the health of the components is carried out [3]. During this routine inspection, the machine often lies idle and there may be no faults detected. This machine downtime adds significantly to its cost of operation. Although this strategy is designed to prevent failure and consequently reduce operating costs, it ends up increasing the overall cost of maintenance by reducing availability as well as replacing components which have some remaining useful lifetime. In the recent years, condition based maintenance (CBM) is gradually becoming the most preferred maintenance strategy. The strategy involves the use of condition monitoring data to estimate the current health state of the system and when the system is likely to fail. This information is then used in maintenance planning.

Figure 1.1 shows the evolution of maintenance strategies visavis benefits and complexity [4]. In corrective maintenance, components are repaired or replaced when they fail. This reduces the availability, safety and reliability of the system. In preventive maintenance, components are replaced before failure and in some cases, the components may contain some amount of useful lifetime. This leads to high maintenance costs. With condition based maintenance, components are replaced depending on their current and future health state, leading to maximum utilization of the components and consequently reducing the maintenance costs. In addition, this strategy increases the availability, safety and reliability.

The overall objective of condition based maintenance is to eliminate unscheduled maintenance and increases the system reliability, availability, safety, maximize component useful lifetime while preventing failure and consequently secondary damage to other components and to assist in planning operations and inventory [5]. New technical systems, referred to as self-optimizing mechatronic systems with the ability to adaptively control reliability have been developed [6, 7]. These systems are able to react to changed operating conditions or faults within the system through behavior adaptation based on multi-objective optimization and thus ensuring that the system continues to operate until the next scheduled maintenance.

Prognostics and health management (PHM) refers to the integration of diagnostics, prognostics and a decision module into condition monitoring [8]. Diagnostics involves fault detection, isolation to locate the fault and identification of the type of fault. In most cases, expert knowledge on the type of faults that can occur is required. This information can be obtained through failure mode and effect analysis (FMEA). Prognostics involves assessment of the future health of a component and is defined as “estimation of time to failure and risk for one or more existing and future failure modes”, according

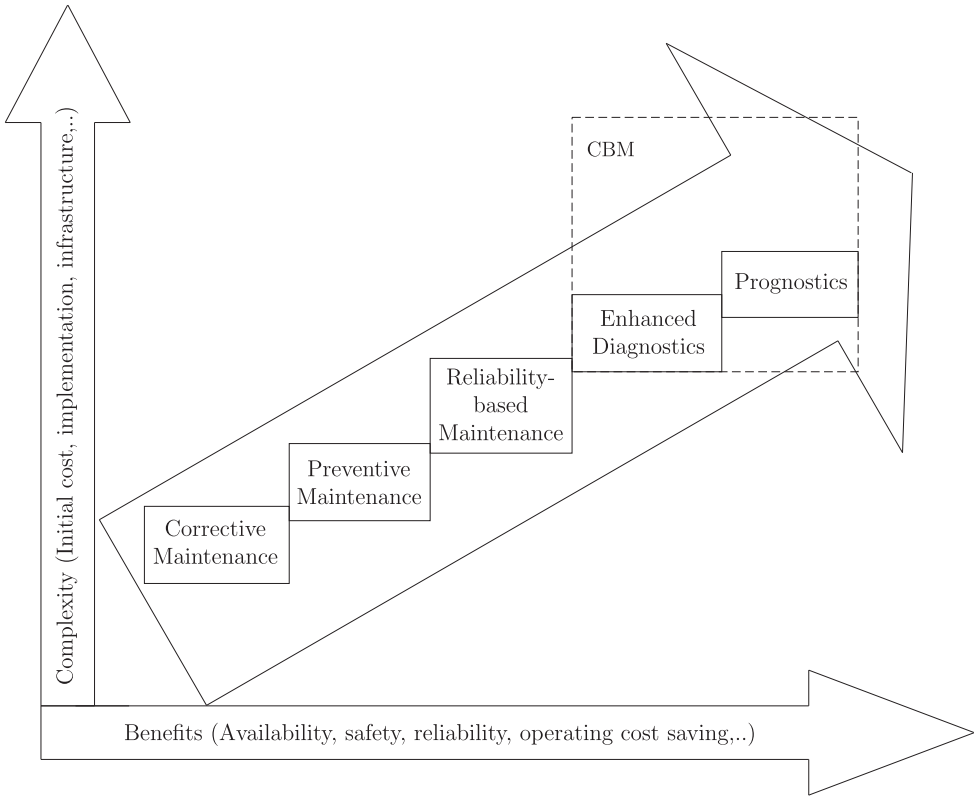

Figure 1.1: Evolution of maintenance strategies [4]

to International Organization for Standardization (ISO 13381-1) [9, 10]. Its objective is to predict the remaining useful lifetime (RUL) of a component before failure occurs given the current machine condition and past operation profile [11]. Figure 1.2 shows the key elements of a prognostics and health management system. In this approach, raw condition monitoring data is acquired from the system through a network of sensors and preprocessed. Relevant features that give the maximum information on the health of a system are extracted. These features are used in diagnostics to identify developing faults. A decision (maintenance or adaptive control) can be made at this stage. Alternatively, this information can be combined with the extracted features for prognostics to estimate when the system is likely to fail. The results of prognostics are then used as input to the decision module. In the framework of this work, the decision module has not been addressed and therefore, methods for estimating when the system will fail, will be referred to as prognostic methods.

In recent years, a huge number of prognostic methods have been developed. The methods can be broadly classified into three categories: 1) reliability based, which rely on failure times of similar units, 2) model based, which rely on mathematical models based on physics of failure and 3) data-driven methods, which rely on raw sensory data obtained from a system during operation [12]. Reliability based methods are the simplest to employ since they do not require condition monitoring data. However, their accuracy is relatively low especially for systems with varying lifetimes due to manufacturing tolerances and varying operating conditions. Model-based methods though found to be very accurate, are system or component specific and are not easily adaptable to different systems. In addition, due to the complexity of modern day systems, the system models are very

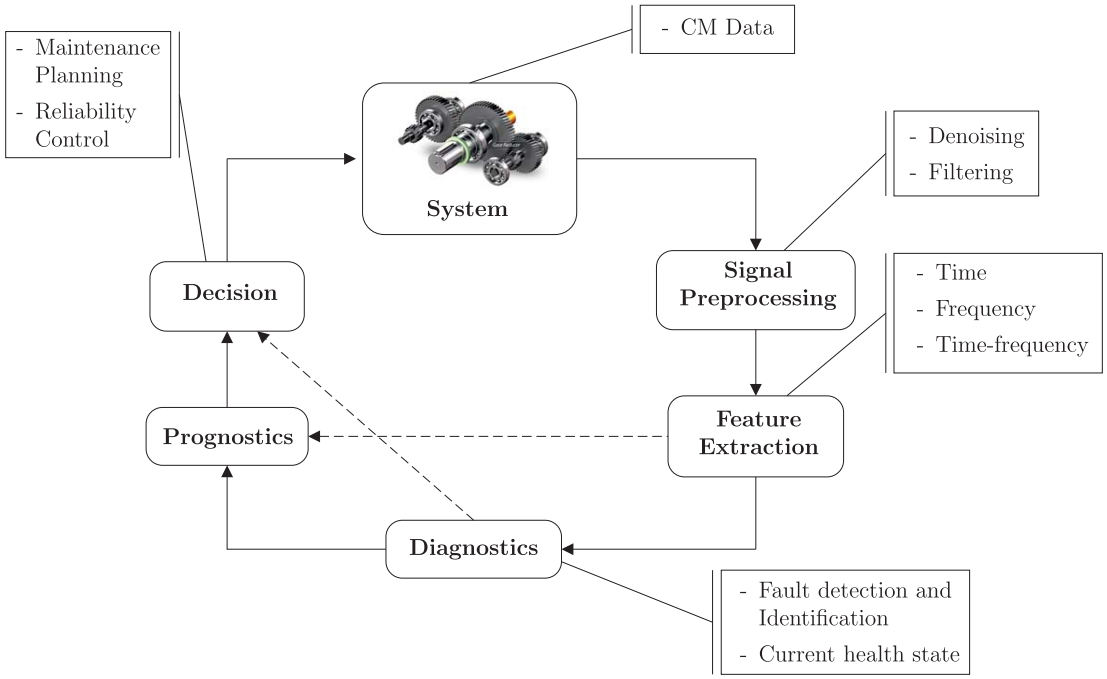

Figure 1.2: Elements of a prognostics and health management system [4]

complex and computationally intensive. Data-driven methods have received considerable efforts since they can be adapted to different systems. Data-driven methods utilize machine learning algorithms to learn the degradation behavior from condition monitoring data of technical systems. The trained algorithm is then used with condition monitoring data of similar system for real-time diagnosis and prognosis.

A significant number of machine learning algorithms have been developed and evaluated on their suitability for diagnostics and prognostics of technical systems. Different prognostic approaches describing how degradation indices can be defined from the condition monitoring data for use with machine learning algorithms have also been developed. Figure 1.3 shows the general workflow when applying machine learning algorithms for diagnostics and prognostics.

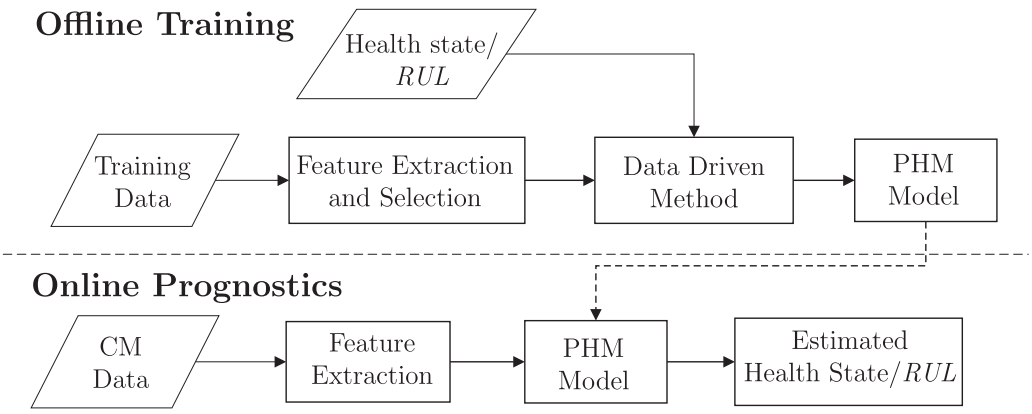

Figure 1.3: General workflow of data-driven PHM methods

The methodology involves two main stages: training and testing. In the training stage, a machine learning algorithm learns the underlying model that maps the input (features from condition monitoring data) to a target. Typical targets for technical systems include:

1. Health state - a value or descriptive name describing the current condition of a system depending on damage/fault severity. Health state is usually used with systems that undergo discrete states before failure, for instance normal-faulty-failed.
2. Health index - an index describing the damage severity of a system. It can be physically measurable from the system, for instance wear or defined virtually using a predetermined function. It is mainly used for tracking continuous damage.
3. Remaining useful lifetime (RUL) - the useful lifetime left on a system at a particular time of operation. Units of time such as days, hours or cycles are used depending on application. It is suited for applications where a large number of run-to-failure data of similar systems is available.
4. Fault type and location - a value or descriptive name describing different fault modes of a system. Mainly used in diagnostics.

In the testing/online prognostics stage, the trained model is used to map the input features of a similar system to an output from which an inference on the state of the system is made. The performance of most prognostic methods will depend on how the input features and the targets are defined.

A key component missing from published prognostic research is a systematic performance evaluation of different prognostic approaches as well as different machine learning algorithms on different types of systems. Such information would be useful in selection of a prognostic approach and corresponding machine learning algorithm and consequently reducing the time and overall cost required for the design and development of a PHM system for a specified technical system. This work attempts to fill that gap by developing a database of prognostic approaches evaluated on real world technical systems to serve as a way of selecting the prognostic approach and corresponding algorithm(s) depending on the type of technical system being monitored and the condition monitoring data available.

1.1 Problem Statement

Machinery health management has been extensively investigated and intelligent diagnostic and prognostic methods developed to monitor and manage machinery health. However, there are still some aspects that need to be explored for the design and development of efficient and reliable PHM systems.

The performance of proposed prognostic methods is subject to the technical system in use. Different methods perform differently on different technical systems depending on the type of condition monitoring data available. A database of prognostic methods evaluated on data from different technical systems is necessary. This information will provide a quick reference for the design and development of PHM systems for a given technical system depending on the condition monitoring data available, consequently reducing the time and cost time that would be incurred in evaluating different methods.

The performance of these algorithms also depends on suitability of the features extracted from the raw data [13]. For classification of health states, good features should demonstrate separability between different health states. For systems undergoing continuous damage where regression methods for approximating continuous functions (either a health index or remaining useful lifetime) are used, then the features should have the ability to capture the degradation trend, preferably monotonic change. In addition, feature selection helps in avoiding overfitting in some machine learning algorithms. Overfitting occurs when the trained prognostic model is excessively complex and overreacts to minor fluctuations in the training data leading to poor predictive performance when used with unseen testing data. A method of feature selection depending on application is therefore necessary.

Many of the machine learning algorithms available require tuning of various parameters for each type of system. A systematic method of parameter tuning using optimization algorithms as well as identifying the parameter search range in order to improve accuracy and reduce the computation time for training and testing of algorithms is necessary.

1.2 Objectives

The main objective of this study is to develop a database of machinery prognostic methods and evaluate their performance based on condition monitoring data of different kinds of technical systems.

The specific objectives of this research are:

1. To study and identify the degradation indices from condition monitoring data of various technical systems as a prerequisite for defining a prognostic approach.
2. To develop methods for feature selection and a method for selecting the optimum number of health states for unlabeled data.
3. To develop a database of prognostic methods for various technical systems based on the condition monitoring data and degradation information available.
4. To evaluate the performance of various prognostic approaches on their ability to estimate the current health state of a system and to predict the remaining useful lifetime so as to serve as a guide for selecting the most suitable method for a given application.

1.3 Thesis Outline

This thesis consists of five chapters. In this introductory chapter, the concept of prognostics and health management (PHM) as a state of the modern day maintenance strategy is introduced. Presently a lot of research on PHM methods is being conducted. However, a number of issues remain to be addressed in order to obtain efficient and robust prognostic methods. For this reason, in the following chapter 2, a detailed literature review on diagnostics, prognostics and the state of the art prognostic approaches that exist so far is presented. Diagnostic methods based on physics-based models and data driven methods are described in detail, including typical application areas. Prognostics methods which are the main focus of this research are also discussed in detail. These include reliability-based, model-based and data driven based methods. Gaps that exist in the current research are highlighted. In order to form a basis for evaluating and selecting prognostic methods, performance metrics for prognostics are discussed in this chapter.

Chapter 3 begins with a description of commonly used condition monitoring data for different technical systems. Data processing to convert the raw sensor data into useful information and also reduce the dimensionality of a data set is discussed. This includes data preprocessing, feature extraction and filtering methods. Since not all extracted features contain relevant information for diagnostics and prognostics, methods for selecting the features with most information on the health of a system are presented in this chapter.

In chapter 4, five selected prognostics approaches depending on the type of condition monitoring data available and the application of machine learning algorithms within three of these approaches are presented. The other two methods include health index propagation method and a simple model-based approach that utilizes temperature measurements for systems with wear. The performance of the methods is evaluated with condition monitoring data of three real technical systems using the prognostic metrics described in chapter 2. Further, a weighted ensemble approach that combines predictions from the best performing methods is presented and evaluated. This performance evaluation serves as a guide for selecting the prognostic approach for a given type of application and condition monitoring data available.

In chapter 5, prognostics of piezoelectric transducers using the concept of self-sensing in a laboratory experiment is introduced. The identification of health indices and the use of performance monitoring data for prognostics is discussed. The application of some selected prognostic methods from chapter four is discussed. The methods are evaluated with prognostic metrics and the suitable method for this application is identified.

Lastly, chapter 6 provides a summary the work presented and conclusions. An overview of prognostic method selection for a specified application and depending on the condition monitoring data available is presented. This chapter also includes an overview of proposed future work which does not fall under the scope of this research.

2 Literature Review

With today's competitive industrial environment, there is need to maximize machinery availability, safety and reliability in addition to reducing maintenance costs. Condition monitoring of technical systems is playing a significant role in technical systems as it provides information on operation and performance characteristics as well as the health state of a system [14]. Advances in information technology has led to the development of low cost microprocessors and multi-functional data acquisition systems with direct sensor connectivity. Advances in sensor technology, automated controls as well as data telemetry have led to innovative and low cost sensors for monitoring operation characteristics such as electric current, pressure, flow, temperature as well as for condition monitoring such as tribology, thermography, vibration, acoustics and force [14]. Technologies in remote condition monitoring have made it possible to monitor systems in locations that are not easily accessible. A state of the art real-time monitoring system referred to as Supervisory Control and Data Acquisition (SCADA) has been extensively employed in real-time performance and remote condition monitoring of complex industrial systems [14]. The sensor data acquired is mainly used for performance monitoring and for controlling various processes in a plant and therefore further analysis of the data for diagnostics and prognostics is necessary. Advancement in these technologies has led to condition based maintenance gaining popularity as the present day maintenance strategy. It aims at avoiding inopportune maintenance spending in addition to maximizing component and system life [9].

Prognostics and health management (PHM) involves integrating diagnostics, prognostics and a decision module into condition monitoring. A fault is first detected, located and identified. The evolution of the fault with continued operation is then tracked and prognostics is conducted at predetermined intervals. Presentation of the output of prognostic can vary depending on the application. Some typical outputs are as follows [15]:

- Time to failure (TTF) of a or system.
- Remaining useful lifetime (RUL).
- Probability that failure will occur before the next scheduled inspection, replacement or overhaul.

Extensive research on diagnostics and prognostics has been carried out and the following sections highlight the milestones and limitations identified in the available literature relevant to this research.

2.1 Diagnostic Methods

Diagnosis is an assessment of the current health state of a system based on observed features from condition monitoring data. It involves identifying anomalous behavior in

a system, locating the component with anomalous behavior and identifying the type of anomaly in that component. In some instances, diagnosis can imply the isolation of a faulty component, a failure mode, or a failure condition [8]. Fault diagnosis methods can be boldly categorized into model-based and data-driven.

2.1.1 Model Based Fault Diagnosis

Model based fault diagnosis involves the use of mathematical models of the physical laws governing the behavior of a system [16, 17]. Models of possible faults and how they influence the state of the system are developed and based on deviation of at least one characteristic property of a variable from an acceptable behavior, faults can be detected and identified [16].

Figure 2.1 shows a basic structure of a model-based diagnostic system. The system model G is described by differential equations with input $u(t)$, system parameters θ. The system produces a measurable output for some given disturbances $d(t)$ and faults $f(t)$. The output $y(t)$ and nominal output $\hat{y}(t)$ are used to generate residues $r(t)$ given by

$$\hat{y}(t) = G\big[u(\cdot), \theta\big], \tag{2.1}$$

$$r(t) = y(t) - \hat{y}(t). \tag{2.2}$$

The residue is compared to nominal behaviour of the system for diagnostic purposes. This approach is closely linked to analytical redundancy used for fault detection and isolation [18]. Analytical redundancy involves the use of analytical relationships to describe the interconnection between various system components and fault detection and isolation is done by comparing a system's available measurements to information from the analytical relationship [19].

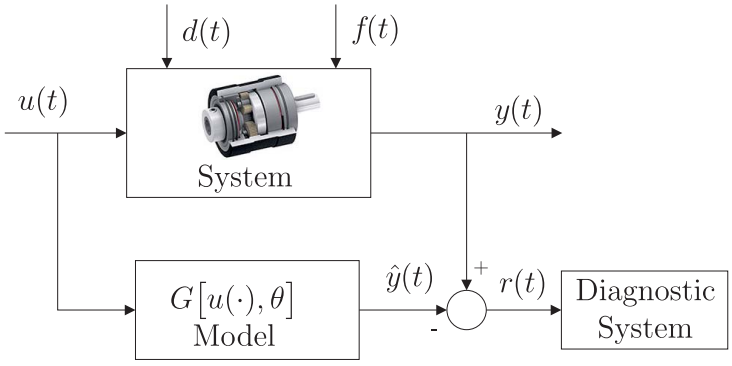

Figure 2.1: General structure of model-based fault diagnosis [16]

Model based diagnostic approach has been widely used in the aircraft industry and a number of studies have been reported. Of greater interest are actuators used in various mechatronic application and from which a number of failures have been reported [20, 21]. Isermann [16] proposed a process model based fault diagnosis method of a cabin pressure outflow valve actuator of a passenger aircraft. With the combination of parameter

estimation and parity equations, four measurements were used to diagnose faults with the help of fuzzy-logic inferencing. Byington et al. [22] proposed a model based approach to PHM that uses physical modeling and advanced parametric identification techniques along with fault detection and failure prediction algorithms for prognostics for each competitive failure mode within the system. The approach was validated and demonstrated on an electromechanical actuator (EMA) for flight control [22]. Varga and Ossmann [20] developed a fault detection and diagnosis method for flight actuator faults such as jamming, runaway, oscillatory failure or loss of efficiency. The method is based on a linear parameter varying (LPV) model of the actuator which relies on a fault detection filter to generate residuals for fault diagnosis. Daigle et al. [23] developed a medium fidelity lumped-parameter dynamical model of spacecraft propellant for loading cryogenic. A model based diagnostic approach was employed where deviations in measured values from model predicted values are compared for fault detection and isolation. Particle filters were used for fault identification through parameter estimation.

Application in robotics have also been reported in [24–27]. Caccavali et al. [24] developed a fault diagnosis approach for actuator faults of robot manipulators. This approach employs a diagnostic observer that computes the estimation error of the system state from which a residual vector is obtained. A fault is said to occur when the norm of the residual vector exceeds a predetermined threshold [24]. Fourlas et al. [25] presented a model based actuator fault diagnosis for a four wheel skid steering robot that use structural analysis based technique to generate residuals. A kinematic model of the robot used to create the structural model of the system was developed and parity equation derived from the system model were used to generate residuals for diagnosis [25]. Hsiao and Weng [26] proposed a hierarchical multiple-model scheme for robotic actuator faults that incorporates a small number of models which are mixtures of the dynamic and kinematic equations of a robot manipulator. All possible faults are contained in the models and a fault is said to occur when the probability of the kinematic fault model exceeds a predetermined threshold [26]. Wahrburg et al. [27] presented an interpretation of cartesian contact force estimation schemes in the context of fault diagnosis of robotic applications involving physical contact between the robot and the environment where collisions are likely to occur (e.g. human interaction or robotic assembly). The study provided a perspective of bridging the gap between robotics and fault diagnosis.

Rotating machinery are some of the most critical systems in industry and prone to failure due to the large number of moving parts. Model based diagnostics methods have also found application in rotating machinery [28–31]. Xiangyang and Wanqiang [28] developed a physical model of a rolling element bearing to simulate the normal state of the bearing. With the use of one-class support vector machines, the authors were successful in detecting 90% of bearing faults. Do and Chong [29] proposed an approach for fault diagnosis of induction motors using vibration signals. The proposed approach consists of two steps: fault detection and fault diagnosis. In the fault detection step, relevant features are extracted from the vibration signals using scaled invariant feature transform (SIFT) to generate fault symptoms while in the fault diagnosis step, the fault symptoms are used for pattern classification to identify the type of fault present. Wang and Wong [30] presented a model based technique for detection and diagnosis of gear faults. The proposed technique establishes an autoregressive (AR) model of an averaged signal obtained by time synchronous averaging under healthy state of the gearbox. The health condition of a gearbox is then diagnosed using the residual signal between the AR filtered signal and

the original healthy state signal. The feasibility of the method was demonstrated using in-flight vibration data of a helicopter gearbox. The method was found to successfully detect cracks in gears. Park et al. [31] proposed a lumped-parameter model to simulate the dynamic behavior of planetary gear trains. The approach uses the gear transmission error to derive various health indices that can be used for fault diagnosis. The results showed that the approach can be used to detect cracks in gears. Treetrong [32] proposed a method for condition monitoring of induction motor through parameter estimation using Genetic algorithm. The proposed method estimates key motor parameters such as stator and rotor resistance, stator and rotor reactance and magnetizing reactance from two measurements: stator phase current and rotor speed. Errors between estimated parameters and nominal parameters are used to diagnose stator and rotor faults [32].

Model based fault diagnosis has been found very accurate in detecting faults but requires mathematical models of the target system to be available. In addition, the approach may be computationally unfeasible for online diagnosis.

2.1.2 Data-Driven Fault Diagnosis

Data-driven methods process raw sensory data into useful information that indicates the state of health of a system. The methods involve advanced signal processing, feature extraction and a comparative analysis with known health conditions of a system. Traditionally, advanced signal processing coupled with expert knowledge of the system is used to identify possible faults. Graphical tools such as power spectrum graph, cepstrum graph, spectrogram, etc, are employed [11, 33]. In rotating machinery, fault characteristic frequencies such as bearing frequency characteristics, gear mesh frequencies, shaft frequencies and their harmonics, etc, have been extensively used in fault diagnosis [34, 35]. Figure 2.2 shows the frequency spectrum of a healthy and faulty bearings. In Figure 2.2(a) only the shaft frequency and its harmonics are visible. In faulty bearings, the fault characteristic frequencies are visible. A bearing with a fault on the outer ring will have the ball pass frequency of outer ring (BPFO) present while that with an inner ring fault will have the ball pass frequency of inner ring (BPFI) present, as shown in Figure 2.2 respectively.

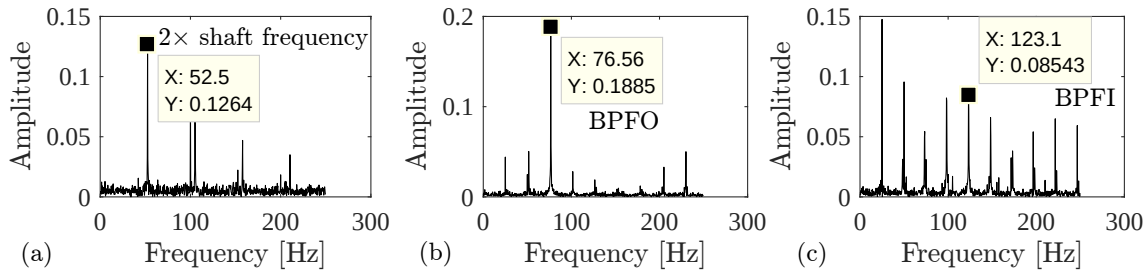

Figure 2.2: Frequency spectrum of ball bearing for (a) healthy bearing, (b) bearing with outer race fault (presence of BPFO) and (c) bearing with inner ring fault (presence of BPFI)

Various studies have been conducted on signal processing techniques to enhance signals in either time-, frequency- or time-frequency domain for accurate fault diagnosis. Zhang et al. [34] proposed a resonant demodulation method with adaptive frequency selection implemented in LabVIEW. The method employs wavelet particle decomposition to decompose and reconstruct a vibration signal and consequently extract fault information from the noisy signal [34]. The method was evaluated with vibration data of a rolling element bearing with inner ring fault. Yi et al. [35] proposed an approach for multi-fault diagnosis of railway axle bearings based on ensemble empirical mode decomposition (EEMD) and Hilbert marginal spectrum. EEMD decomposes a signal into intrisinc mode functions (IMFs) which are then subjected to Hilbert transformation to obtain the instantaneous frequencies from which fault characteristic frequencies are identified [35]. The approach was found to be effective in identifying single or multiple faults in bearings. Pan et al. [36] proposed a method for fault diagnosis in rolling element bearings through virtual instrument technology and Hilbert transform. The method involves detecting the envelope of the raw signal from which the frequency spectrum is obtained. Fault characteristic frequencies are then identified for diagnostic purposes. Sawalhi et al. [37] proposed an algorithm for enhancement of fault detection and diagnosis in rolling element bearings using combined spectral kurtosis (SK) with autoregressive (AR) based linear prediction filtering and minimum entropy deconvolution (MED). The algorithm separates impulses originating from a faulty bearing using an AR-based linear prediction filter. The impulses are then enhanced with MED by deconvolving the transmission path after which the signal is decomposed using complex Morlet wavelets and the SK is calculated to select the best filter for the envelope analysis [37]. The method can be further extended to prognosis by tracking the SK values with evolution of the fault.

Apart from rolling element bearings, the approach has also found applications in gearboxes, electric motors and combustion engines. Guoji et al. [38] proposed the uses of bispectrum analysis for gearbox fault diagnosis. The method exploits the presence of peaks in the harmonic mesh frequencies of gears due to local defects. Thomson and Gilmore [39] presented an analysis on the use of electric motor current signature analysis (MCSA) for fault diagnosis of induction motors. MCSA has been successfully applied in diagnosis of induction motor faults such as bearing faults, broken rotor bars, airgap eccentricity, shorted turns and bent shaft [39–41]. Lee and Kim [42] developed a two-stage adaptive line enhancer to enhance impulsive signals of a faulty combustion engine embedded in background noise. The method was used as a signal preprocessor prior to time-frequency analysis for fault diagnosis.

So far the methods discussed are system specific and suited for single-point faults which give rise to characteristic fault frequencies. In presence of distributed and multiple faults, the methods may not be suitable for diagnosis. In addition, the methods require expert knowledge of the system and may not be readily adaptable to different kinds of technical systems.

In the recent years, pattern recognition has been widely employed in fault diagnosis and overcomes some of the limitations highlighted in diagnostic methods based on signal processing. This approach employs algorithms to automatically identify patterns within condition monitoring data or extracted features. These patterns are further used to classify the data into different categories [43]. In fault diagnosis, the approach is based on the

fact that different kinds of faults in a system create unique patterns in the data/features, which can be used to detect their presence. These patterns can be used together with failure mode and effect analysis (FMEA) to classify different types of faults. Figure 2.3 shows clustering of features extracted from healthy and faulty bearings.

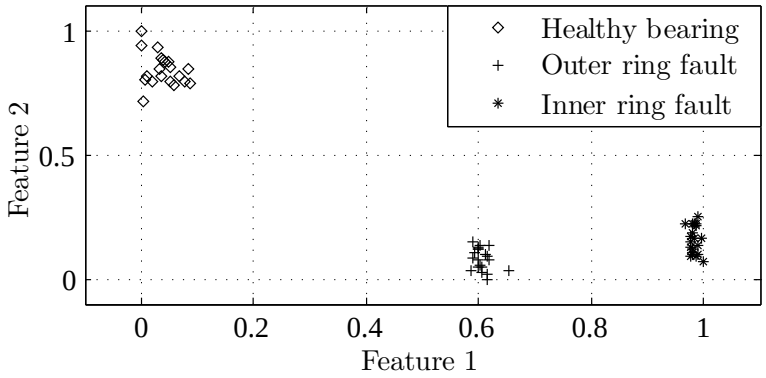

Figure 2.3: Clustering of features from different health states of a ball bearing

Data-driven methods based on pattern recognition are based on either unsupervised or supervised machine learning.

2.1.2.1 Fault Diagnosis through Unsupervised Pattern Recognition

Unsupervised pattern recognition, also referred to as clustering, identifies the underlying patterns within unknown multivariate data (mainly consisting of a number of features extracted from condition monitoring data). This approach is especially beneficial when dealing with data without apriori information [44]. This approach can be used to detect deviations from normal operation without prior knowledge of faults within the system by partitioning the data into clusters with similar regularities. Algorithms such as k-means, fuzzy c-means and self-organizing maps (SOM) neural networks can be employed. Clustering by k-means algorithm is done by partitioning observations in an $n \times p$ matrix into k clusters by iteratively finding k cluster centers. The distance between each point and the centroid of each cluster is computed and the point is assigned to the closest centroid [45]. Figure 2.4 shows the centroid of different health states of the ball bearing in Figure 2.3 computed using k-means clustering algorithm, assuming that the health states are unknown.

A number of studies have been conducted on application of unsupervised machine learning for fault diagnosis in technical systems. Liu et al. [46] proposed a two-step machinery fault diagnosis using fuzzy c-means and fuzzy integral theory. The fuzzy c-means is employed to group condition monitoring data into similar clusters in the first step while fuzzy measure and fuzzy integral theory are applied to make a final decision on the diagnosis process [46]. The approach was evaluated with vibration data from rolling element bearing and found to detect inner race, outer race and ball faults with an average of 95% accuracy. Datta et al. [44] proposed a hybrid model for unsupervised clustering

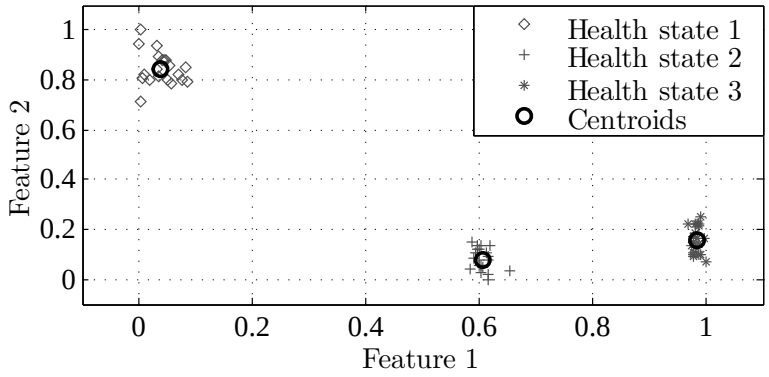

Figure 2.4: Cluster centroids of ball bearing health states computed using k-means algorithm

method which was applied in fault diagnosis of industrial robots. The approach combines SOM with quality threshold which is an algorithm initially developed for clustering of genes based on micro array data and has been found to have a higher cluster quality. Yu et al. [47] proposed a cluster based feature extraction method for machine fault diagnosis. The approach extracts features from coefficients of discrete wavelet transform and employs similar means to cluster data. The method was evaluated on ball bearing data. Widodo et al. [48] applied SOM in machine fault diagnosis using infrared images acquired through thermography camera. Results showed that SOM can diagnose faults such as shaft misalignment, bearing faults and unbalance in rotating machinery [48].

One disadvantage of most clustering algorithms is that they require an estimation of the possible number of clusters of the data in consideration [44]. However, methods such as multidimensional scaling (MDS) can be used to determine the number of clusters in a data set [44]. MDS projects the data into a higher dimensional space so as to maximize intra-cluster distance in the data [44]. This way, separation between clusters is revealed.

2.1.2.2 Fault Diagnosis through Supervised Pattern Recognition

In supervised pattern recognition, a machine learning algorithm is trained to map input features to a target. The training process involves determining and adjusting weights, biases and parameters which constitute a model such that the error between the actual target and the predicted target for the same input is minimized as shown in Figure 2.5 [49]. Once the model is obtained, it can be used to predict the output of a new set of input from a similar system. In fault diagnosis of technical systems, the input consists of either condition monitoring data or features extracted from the condition monitoring data while the output consists of the health state of the system. The health state may be determined through FMEA, expert knowledge from maintenance engineers [4], seeded fault testing [50], etc.

The machine learning algorithms are trained offline resulting in a model that can be used with input features obtained in real-time from sensory data to monitor the health of a system. A large number of studies on application of supervised machine learning algorithms for diagnosis have been conducted. Table 2.1 lists some of the most commonly used state of the art algorithms in machinery fault diagnosis.

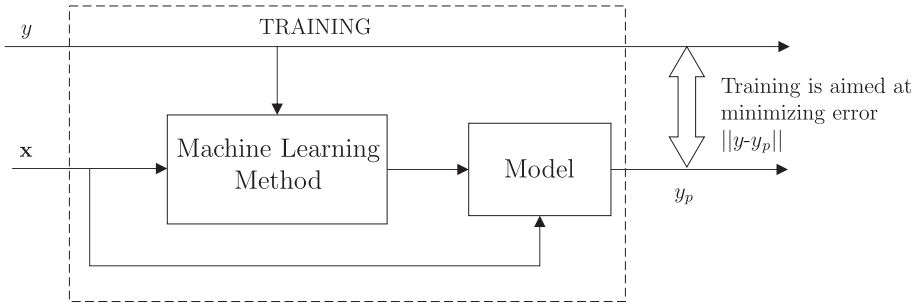

Figure 2.5: General approach to supervised machine learning, where $\mathbf{x}$ is the input features, y is the target and y_p is the predicted output for the same input

Table 2.1: List of most commonly used state of the art supervised machine learning algorithms for fault diagnosis

Algorithm	**Application**
Artificial neural networks (ANN)	Rolling element bearings [51–56], reciprocating compressor [57], HVAC systems [58], combustion engines [59, 60], gearbox [61, 62], induction machines [63, 64], turbine generators [65], robot manipulators [66]
Support vector machines (SVM)	Rolling element bearings [67–70], gearboxes [62, 71–74], shaft-rotor systems [75], centrifugal pump [76], induction motors [77]
Classification and regression trees (CART)	Rolling element bearings [78–80], induction machines [81, 82]
Neuro-Fuzzy	Rolling element bearings [78, 83–85], Combustion engines [86], pneumatic valve [87], induction machines [81]
Hidden Markov Models (HMMs)	Rolling element bearings [12, 88–90], shaft-rotor system [91], gearbox [92], induction machines [93]
k-Nearest Neignbors (kNN)	Rolling element bearings [94–97], induction machines [98], gearboxes [99–102]

As seen in Table 2.1, artificial neural network is the most widely used algorithm in machine diagnosis followed by support vector machine. An artificial neural network (ANN) is a computational model inspired by the structure of the human brain. It consists of neurons that are interconnected to form a network [103]. Figure 2.6 shows the general structure of a multi-layer feed-forward ANN commonly used in fault diagnosis.

ANN operates as a 'black box' model that does not require detailed knowledge about the system being modeled but rather, it learns the underlying relationship between the input and output of a system by adjusting its weights and biases. The main difference in the reviewed literature lies in the selection of the type of ANN and in tuning ANN parameters in order to improve generalization and consequently the classification accuracy. Generalization refers to the ability to accurately predict an output from unseen input data. Unal et al. [55] proposed the use of genetic algorithm to tune the number of hidden layers and neurons in each hidden layer for the classification of rolling element

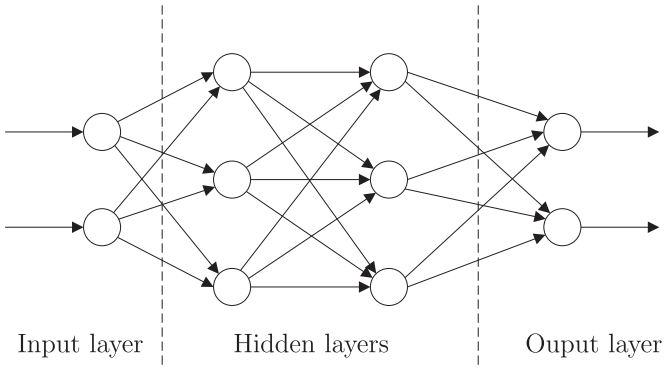

Figure 2.6: General structure of multi-layer feed-forward neural network [103]

bearing faults. Chen et al. [54] proposed the combined use of dependent feature vector and probability neural network (PNN) for fault diagnosis of rolling element bearings. The dependent feature vector method identifies the main feature that describes the fault characteristics and its dependent features. These features are used as inputs to the PNN. There is no clear guideline on selection of the type and structure of ANN for a given system and as such the application of ANN is not straightforward. In addition, ANN requires a large number of training samples for better performance.

Support vector machine has been identified as a power pattern recognition tool which is capable of learning the underlying relationship between input and outputs of a pattern recognition problem when provided with few training samples. SVM solves the classification problem by transforming the input features into a higher dimensional feature space in which a hyperplane that separates data between classes is defined [104]. However, to improve generalization, SVM requires tuning of its parameters for each application and various authors have developed methods for automatically tuning these parameters. Zhang et al. [67] proposed the use of ant colony algorithm for feature selection and SVM parameter tuning for fault diagnosis of a shaft-rotor system. Hang et al. [105] investigate the application of fuzzy support vector machine (FSVM) for fault diagnosis of a wind turbine. The study employed kernel fuzzy c-means clustering algorithm and particle swarm optimization (PSO) to optimize the kernel parameters of FSVM. Results showed that optimized FSVM has higher classification accuracies than standard SVM and ANN.

Although fault diagnosis is currently a mature technology, a guide for selection of algorithms to apply for a given technical system is lacking. The selection guide can be developed by evaluating the performance of the state of the art algorithms on different technical systems and creating a database of these algorithms as well their performance.

2.2 Prognostic Methods

Prognostics involves estimating the current health state of a technical system/component and predicting when the system/ component is likely to fail. Prognostics is concerned with estimating the remaining useful lifetime (RUL) of a system/component and is more complicated than diagnostics since it aims at predicting an event that is yet to occur [8]. The predictions can either be done as an event where the end of life is predicted or as a degradation prediction where the future trajectory until the end of life (EoL) is

predicted. There are various uncertainties, like future operating conditions of the system, that make the process complicated [4]. Due to benefits, such as increasing availability, safety, reliability and maximizing usable life, that would occur if the remaining useful lifetime could be estimated within acceptable limits, a lot of research is being conducted on accurate and robust prognostic methods. The prognostic methods detailed in the available literature depend on the type of data and system information available and can be broadly classified into three categories: 1) reliability based, 2) model based and 3) data-driven based [12].

2.2.1 Reliability Based Prognostic Methods

Reliability based prognostic methods stem from traditional reliability analysis that utilizes distribution of historical failure times of similar systems [106]. This approach is suitable for low risk systems that do not have sensor network for health monitoring and have relatively shorter lifetimes [107]. The prognostic methods require historical failure times as input [107]. A probability distribution is fit to the lifetimes of a population of similar units to give an estimate of the time of failure (tof) distribution. The remaining useful lifetime at a given probability of failure can then be estimated as follows

$$RUL = tof - t_c, \tag{2.3}$$

where t_c is the current operating time. Weibull distribution is the most common statistical model, especially in prognostics of electronic components [106, 107]. Other commonly used lifetime distributions include exponential, lognormal and normal distributions [107]. The Weibull model can be applied in a variety of forms such as 1-parameter, 2-parameter, 3-parameter or mixed parameter. The failure rate λ defined by a 2-parameter Weibull model is given by

$$\lambda(t) = \frac{\beta}{\eta}\left(\frac{t}{\eta}\right)^{\beta-1}, \tag{2.4}$$

where β is the shape parameter and η is the scale parameter that defines where the bulk of the distribution lies [106]. Several methods have been developed to estimate the distribution parameters to fit a lifetime distribution to a particular population of units. These include probability plotting, rank regression and maximum likelyhood estimation (MLE).

Several reliability based prognostic methods have been proposed. Yang et al. [108] investigated the use of Weibull distribution to predict the lifetime of a scrapper conveyor. Chen and Zheng [109] extended the use of reliability based prognostics to incorporate degradation data in order to improve lifetime predictions. Heng [110] proposed the combined use of survival probabilities, degradation-based failure probabilities and artificial neural networks to incorporate equipment with censored lifetime in prognostics. Fan and Hsu [111] presented an analysis of a series system under accelerated lifetime tests (ALT) with masked data (where components causing failure are not observed) while assuming that the components have independent exponential lifetime distributions. These methods have been found to be unreliable since they do not take into account the operating conditions of a system [106, 107]. In addition, the methods are not capable of detecting and tracking the incipient faults.

2.2.2 Model Based Prognostic Methods

Model based prognostic methods utilize a detailed mathematical model of a system, usually derived from first principles, to track its degradation. More often than not, the models are capable of capturing the nominal behavior of the system as well as the faulty behavior and how the faults evolve with time [112]. Therefore a model based approach requires an in-depth understanding of the underlying physic laws in order to produce an accurate and reliable system model that is able to diagnose faults and track their evolution [23]. Figure 2.7 shows the architecture of a model based prognostic approach.

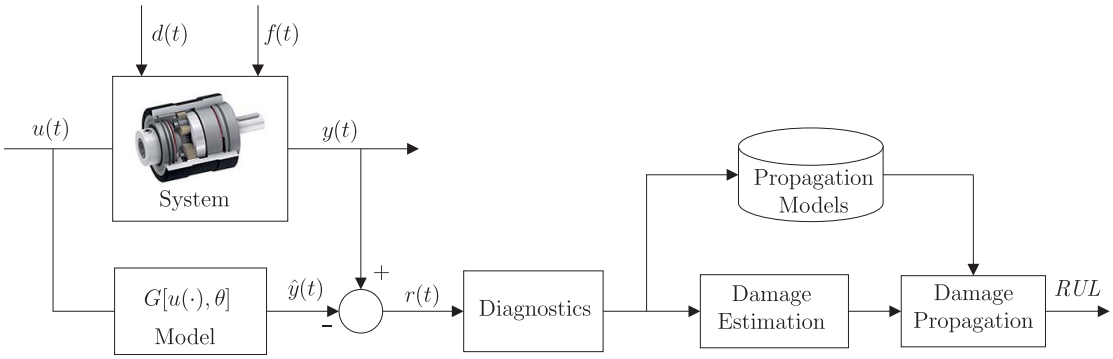

Figure 2.7: A general model based prognostic architecture

For some given set of inputs $u(t)$, disturbances $d(t)$ and faults $f(t)$, the system produces some measurable output $y(t)$ which is used together with the output of the model to produce a residue $r(t)$. The residue is used in defining a health index for diagnosis and prognosis. A suitable damage propagation model is selected from a database of models and used to propagate the health index forward in time to a predetermined threshold. The remaining useful lifetime is then estimated from the distribution of a number of model parameters as shown in Figure 2.8. Typical approach is to take the mean or a specified percentile of the distribution.

Algorithms such as extended Kalman filters, reduced order unknown input observers, interacting multiple models, particle filters, are used to track and propagate the damage index [113]. Particle filters have become popular in prognostics and health management where they are used to track degradation through state parameter estimation [112]. Particle filters are non-linear state observers that use Monte Carlo simulation to approximate the posterior state probability distribution as a set of discrete weighted samples called particles [114].

Several model based prognostic methods have been proposed. Byington et al. [115] proposed a model based approach to prognostics and health management that applies physical modeling and advanced parametric identification techniques together with fault detection and failure prediction algorithms to predict the time of failure of critical components in a system [115]. The feasibility of the approach was evaluated on an electromechanical actuator for flight control. Daigle and Goebel [112] developed a general model based prognostic framework of a pneumatic valve from a Space Shuttle cryogenic refueling system

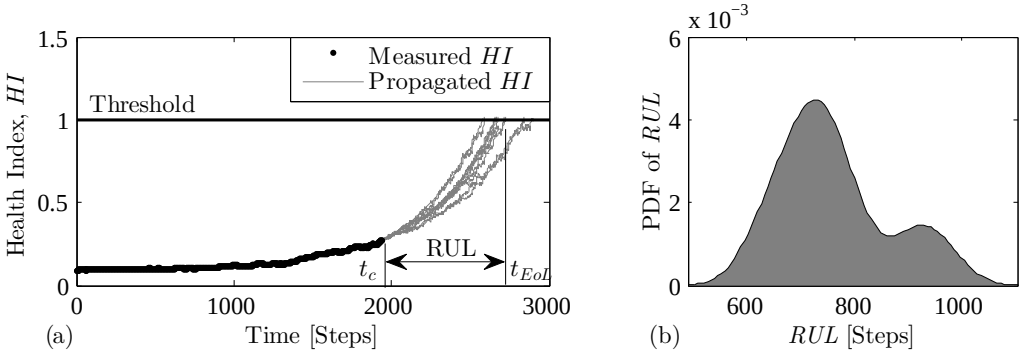

Figure 2.8: Propagating a health index to a threshold (a) trajectories of *HI* and (b) distribution of *RUL*

using particle filters. A nominal model as well as fault models that capture how model parameters change as a result of damage were developed. Kulkarni [116] developed a physics based degradation model schemes for electrolytic capacitors. A Bayesian tracking framework using Kalman filter and unscented Kalman filter was used to track degradation and for *RUL* estimation [116]. The method was evaluated with both accelerated and nominal degradation and found to be accurate and robust. Sankavaram et al. [113] discussed a systematic approach to prognostics based on a hybrid model based, data-driven and knowledge based integrated diagnosis and prognosis framework for automotive suspension, battery systems and on-board electronic systems.

Model based prognostic approaches have also been applied to rotating machinery. Patrick-Aldaco [117] developed a generic model based fault diagnosis and failure prognosis method applicable to large scale dynamical mechanical systems. The method constitutes model based vibration characterization of complex rotary systems, fault identification from vibration signals through reverse engineering and damage progression model based on particle filters for prognosis [117]. A finite element model of a planetary carrier of a helicopter gearbox was developed to evaluate the effect of crack geometry on crack growth parameters which were used in the model for crack progression. The approach also incorporates uncertainties in future loading conditions and noisy measurements. Bolander et al. [118] proposed a physics-based approach for predicting the remaining useful life of engine systems. The method employs finite element analysis (FEA) to model roller/spall impact in rolling element bearings, with spall severity as the damage index. Particle filters with Bayesian update were employed to track and propagate the damage index to a predetermined threshold. Luo et al. [119] employed singular perturbation methods of control theory coupled with state estimation techniques to estimate the remaining useful life of a system with multiple operational modes. The approach was evaluated with an automotive suspension system using lumped-parameter model excited through a rough road of different profiles to create the different operational modes [119].

The main advantage of model based prognostics is the ability to incorporate the physical understanding of the system which makes them very accurate. It is also easier to incorporate varying operating and loading conditions and their influence on the degradation process. However, the modern day technical systems are very complex requiring complex dynamic models which might be difficult to develop especially in situations where the manufacturer is not willing to supply the design details of the system [113]. In addition,

simulations might be too complex and computationally unfeasible for online prognostics.

2.2.3 Data-Driven Prognostic Methods

Data-driven prognostic methods make use of condition monitoring data to extract useful information that represents the degradation behavior of a system [8]. The methods use historical condition monitoring data of similar systems to learn the degradation behavior which is stored as a model in form of weights, biases and parameters, often done offline. The resulting model is then used with condition monitoring data of a running system to estimate the current health state and to predict when the system is likely to fail. This information is used in scheduling maintenance or for adaptive control of the reliability of the system [7]. The main feature of data-driven methods is the use of machine learning algorithms to learn patterns and trends in the data. These require input (often run-to-failure condition monitoring data) and a target (health index or remaining useful lifetime) and learn the underlying correlation between the input and target. In complex systems that display several failure modes, each failure mode may present a different degradation trend and as such, a diagnostic module is integrated to determine the type of failure mode present. A prognostic model for each failure mode is trained and stored in a database. During testing or online prognosis, the current health state and failure mode are identified and the appropriate model is retrieved. Figure 2.9 shows the general architecture of a prognostics method with integrated diagnostics. For systems that have several failure modes, each failure mode may have its own unique degradation trend. The diagnostic output in Figure 2.9 is used to identify the current failure mode so as to select the appropriate prognostic model for estimating the remaining useful lifetime.

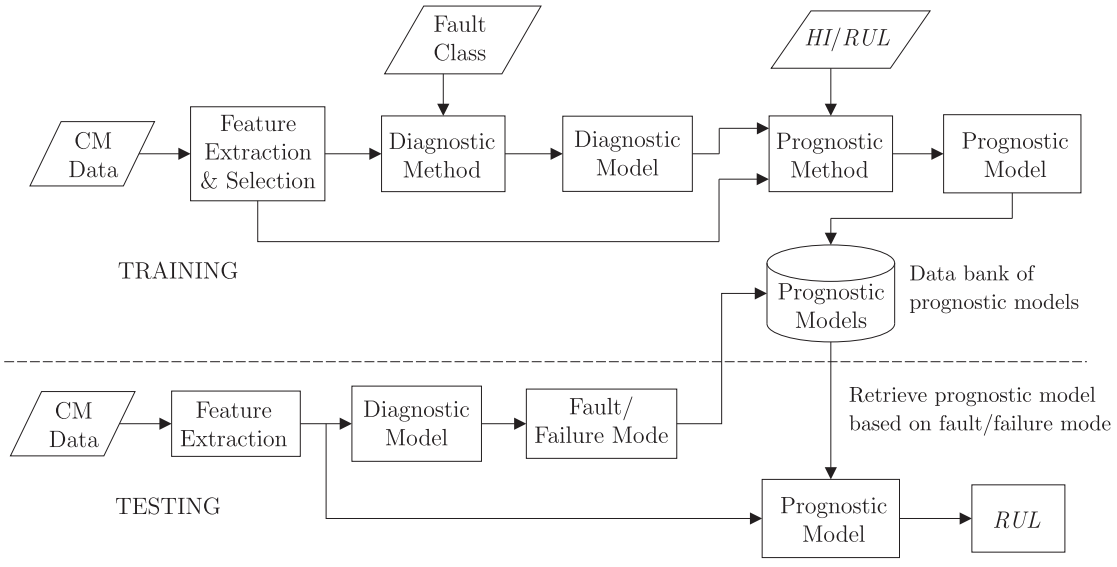

Figure 2.9: General architecture of a prognostic method with integrated diagnostics

Depending on the type of condition monitoring data and degradation information available, several data-driven methodologies have been proposed. In cases where the run-to-

failure condition monitoring data of a number of similar systems is available, machine learning algorithms can be trained to map the condition monitoring data or features extracted from the data to the remaining useful lifetime or probability of failure. This approach is suitable for systems run at constant operating conditions that have very small variance in lifetimes. Sutrisno et al. [120] evaluated the application of least square support vector machines (LS-SVR) in estimating the remaining useful lifetime of ball bearings. The input consisted of time-, frequency- and time-frequency features reduced through principal components analysis (PCA) while the target was the remaining useful lifetime. Tian et al. [121] presented a method of estimating the remaining useful lifetime of technical systems utilizing both failure and suspension histories of the systems using artificial neural networks (ANN). The method was evaluated with run-to-failure and suspended vibration data from rolling element bearings. Extracted features were used as input and percentage *RUL* was used as the target in training and evaluating the ANN. Caesarendra et al. [122] presented a combination of probability approach and support vector machines to predict failure degradation of rolling element bearings. Kurtosis of vibration data was used as the input while the target was failure rate computed through Cox proportional hazard model and reliability theory.

For systems that undergo discrete, usually unobservable degradation stages before failure, the prognostic methods can be trained to identify the current health state. Using historical percentage *RUL* at each health state or normalized *RUL*, the *RUL* from the current health state can be estimated. Figure 2.10 shows typical degradation stages of a ball bearing, where a fault develops and evolves with continued operation until failure. As the probability of one health state decreases, the probability of the next health state increases.

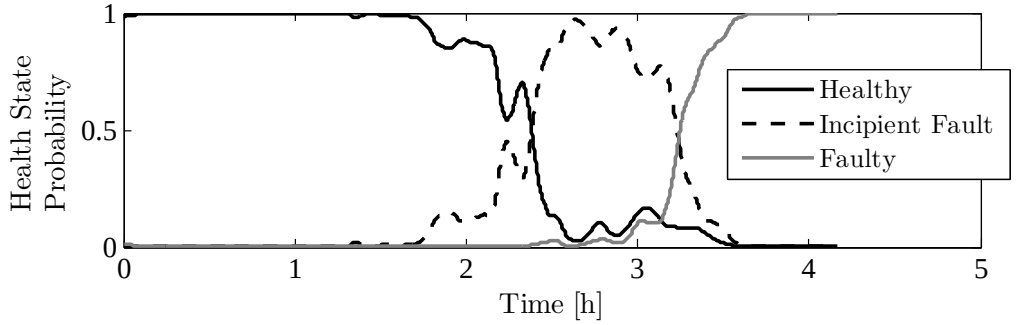

Figure 2.10: Discrete health states of a degrading component

Kim et al. [123] evaluated the use of support vector machines with probability estimates to identify the current health state. Five health states were selected as optimal in tracking degradation of rolling element bearings. The method used time- and frequency-domain features as inputs and health state identifier as target. The *RUL* was computed from historical *RUL* at each health state. Ocak et al. [124] proposed the use of Hidden Markov Models (HMM) to track the severity of bearing faults through probability estimates. The results showed that as the bearing approached the end of its life (approximately 10%), the HMM probability of normal condition dropped significantly. Chinnam and Baruah [125]

employed HMMs to identify the health states (Good → Bad → Worse) of a cutting tool from which the remaining useful life was estimated. The approach uses the loglikelihood of HMMs to identify the current health state and the transition points between different health states. Camci and Chinnam [126] employed HMMs to estimate health states of a cutting tool using observable sensor signals obtained during machining. The approach employed Monte Carlo simulation with state transition probabilities derived from hierarchical HMMs to estimate the remaining useful life [126]. Tamilselvan et al. [127] proposed the use of Deep Belief Networks (DBN) for health state classification of a degrading system using multi-sensor data. The approach was demonstrated with aircraft wing structures and aircraft engine health diagnosis.

Some systems have a physical health index that can be measured, for instance, wear can be used as the health index for systems that experience wear-induced failures. For such systems, it is possible to quantify the health index with continued operation of the system and define suitable thresholds. However, in most cases, due to the location of the components, it may be difficult to measure these quantities during operation. In such cases, training data can be obtained by measuring the health index simultaneously with other condition monitoring data at predetermined intervals. Machine learning algorithms are then trained to map condition monitoring data/extracted features to the health index. Once a trained model is available, it can be applied together with condition monitoring data to monitor the health of similar systems in real-time. Pal et al. [128] proposed the use of neural networks for sensor data fusion for tool wear monitoring during turning process. Measurements of cutting strains in tool holder and motor current were taken for different machining conditions consisting of cutting speed, depth of cut and feed rate [128]. Flank wear was derived from photos taken concurrently with the condition monitoring measurements. The approach was found to be effective in predicting tool wear. Cus and Zuperl [129] proposed the use of neural networks and adaptive neural fuzzy inference system (ANFIS) to predict tool wear and tool breakage from cutting force signals. The approach was evaluated on a milling machine with cutting forces measured using a piezoeletric force transducer while tool wear was periodically measured using a tool microscope. The approach was found to be effective in classifying the different health states including tool breakage and tool wear.

In some systems the condition monitoring data or performance monitoring data can be used as the health index. For components like fuel cells and lithium-ion batteries, the charging and discharge capacities can be used to track degradation of a system. Algorithms such as particle filters, that can be adapted to predict future trajectories, are usually suited for these systems. Tang et al. [130] developed a robotic test rig for developing and evaluating realtime PHM and Automated Contingency Management techniques for autonomous vehicles. Particle filter method was implemented on the prognosis server of the system and used to track aging of a battery. The charge capacity of the battery was used as the health index with end of life (EoL) defined as the time when the capacity falls below the critical limit. Saha et al. [131] compared the performance of three algorithms; autoregressive integrated moving average (ARIMA), extended Kalman filter (EKF) and particle filter in estimating the *RUL* of batteries. ARIMA and EKF have been found to have unsatisfactory performance in prognostics due to various sources of errors such as modeling inconsistencies, system noise and degraded sensor fidelity [131]. Gearboxes can be monitored using oil debris mass (ODM) [132]. He et al. [132] evaluated the use of ODM and a health index computed from vibration signals to estimate the *RUL* of gears

through particle filter approach. An ARIMA model was fitted to the experimental data to define the state transition function while a double exponential smoothing function was used to define the measurement function [132]. The results showed that the particle filter produces good RUL predictions.

2.3 Performance Evaluation

Prognostics and health management (PHM) aims at increasing availability, reliability and safety by avoiding catastrophic failures through fault diagnostics and prognostics. However, the process is complicated by uncertainties in future operating conditions, sensor noise, model inaccuracies, manufacturing variability and false diagnosis [133]. This calls for performance evaluation of the PHM methods before they can be deployed in critical technical systems. Performance evaluation also plays a big role in selection of algorithms for a given application. There is no clear definition of performance metrics for prognostic methods from standardization organizations but various authors have proposed various metrics to assist in standardizing the evaluation of prognostic methods [133–136]. Uckun et al. [134] presented a critical review of current research methods in PHM and made a comparison with standard research approaches in medicine which is more established. The study revealed that the current PHM research is based on ad-hoc experimental methods without a clear statistical rigor. A proposal for the steps required to bring PHM to maturity was made [134]. Saxena et al. [135] presented a survey of prognostic metrics already in use in various disciplines such as medicine, nuclear, aerospace, automotive and electronics. The study analyzed the similarities and differences of the prognostic metrics in the different disciplines and which metrics could be borrowed for PHM methods. Saxena et al. [133] proposed prognostic metrics which have the capability of incorporating probabilistic uncertainty estimates from prognostic algorithms. Guidelines for selection of methods based on distribution characteristics were also proposed [133]. Zhou et al. [136] presented a hierarchical method of comparison and selection of prognostic metrics for performance evaluation of algorithms for use in aircraft PHM. Figure 2.11 shows the functional classification of prognostic metrics [133]. Accuracy is a measure of error in the predicted RUL relative to the actual RUL while precision quantifies the length of 95% confidence bounds relative to the predicted RUL at any given time. Precision is normalized between 0 and 1 with the perfect value being 1 [137]. Return on investment (ROI) assesses the benefits of deploying a PHM system while cost and savings refers to the total cost of installing and operating the PHM system and the savings realized from the system [137].

2.3.1 Prognostic Performance Metrics

The following are the prognostic metrics that are relevant to this work. Other commonly used metrics can be found in [133, 138].

a) Relative error e_r defined as the relative deviation of the predicted remaining useful lifetime RUL_p from the actual remaining useful lifetime RUL_a and is given by

$$e_r = \frac{RUL_a - RUL_p}{RUL_a} \times 100\%. \tag{2.5}$$

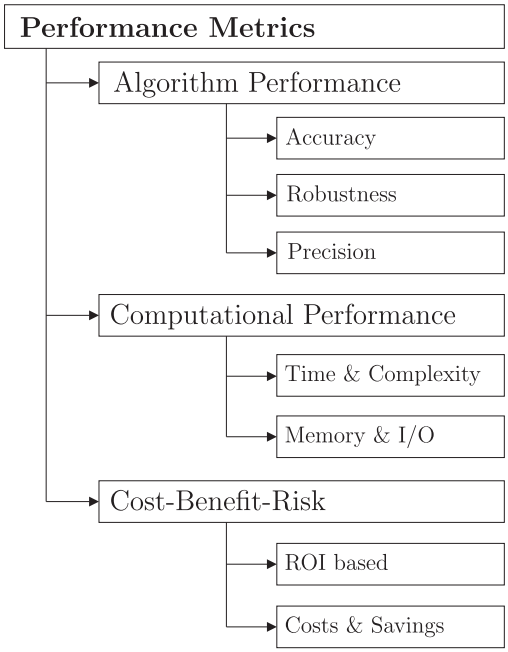

Figure 2.11: Functional classification of prognostic metrics [133]

The perfect score is zero and most accuracy based metrics are derived from error. A negative error means late prediction where a system or component fails before the predicted time while a positive error means early prediction. Late prediction is undesirable in prognostics, since the system may fail before the scheduled maintenance.

b) False positives (FP) assesses unacceptable early predictions at specified time instances. The user must set acceptable range for early prediction [133]. Very early predictions result in excessive lead time consequently minimizing the usable lifetime of a system or component. The perfect score is zero. FP is given by

$$FP = \begin{cases} 1 & \text{if } e_r > e_{FP} \\ 0 & \text{otherwise,} \end{cases} \tag{2.6}$$

where e_r is the relative error given by Equation 2.5 and e_{FP} is the allowable error for early predictions.

c) False negatives (FN) assesses unacceptable late predictions at specified time instances and similarly, the user must set the acceptable range for late prediction [133]. FN is given by

$$FN = \begin{cases} 1 & \text{if } e_r < -e_{FN} \\ 0 & \text{otherwise,} \end{cases} \tag{2.7}$$

where e_{FN} is the allowable error for late predictions. Figure 2.12 shows the acceptable range of RUL prediction at specified prediction instances. Predictions outside the acceptable range are either FP or FN.

d) Prognostic horizon (PH) is defined as the difference between time index when the prediction first meets the specified performance criteria (allowable relative error) and

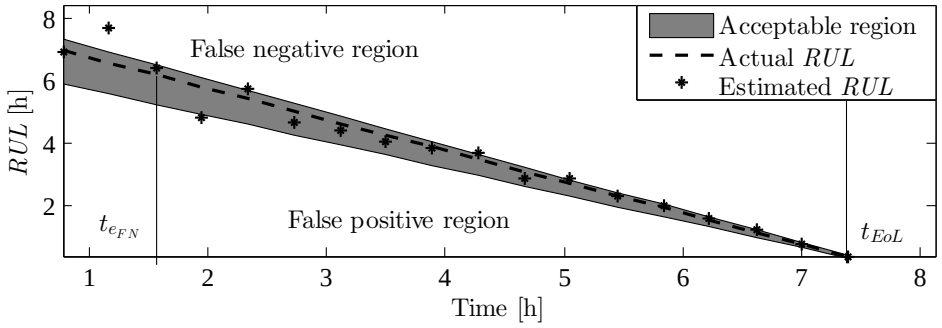

Figure 2.12: Acceptable RUL prediction range

the time at the end of life (t_{EoL}) of the component with

$$PH = t_{EoL} - t_{(e_{FP},e_{FN})}, \tag{2.8}$$

where $t_{(e_{FP},e_{FN})}$ is the time index when the prediction first falls within the allowable relative error, as shown in Figure 2.12.

e) Mean absolute percentage error (MAPE) is the average absolute percentage error of L units at the same prediction horizon.

$$\text{MAPE} = \frac{1}{L}\sum_{l=1}^{L}|e_r|. \tag{2.9}$$

The perfect score is also zero. MAPE can also be used to evaluate a single unit, where L is taken as the total prediction horizons in the lifetime of the unit.

2.3.2 Uncertainty in Prognostics

A good prognostic method should be capable of specifying the confidence levels associated with the RUL predictions. Inherent physical randomness and failure processes, uncertainty in model selection and propagation, sensor noise, etc., often lead to inaccurate predictions [139, 140]. It is therefore important to identify the sources of uncertainty for a specific technical system, quantify the uncertainties and integrate them into the prognostic method. Typical sources of uncertainty include:

1. Physical variability such as variability in material properties, manufacturing errors, variability in loading conditions, variability in initial health state, etc.
2. Measurement uncertainty resulting from sensor inaccuracies, measurement noise, data reduction effects and few training data sets
3. Future usage uncertainty where the future operating conditions are unknown and may vary from the estimated conditions at which the RUL is predicted
4. Model uncertainty where model selection inaccuracies as well as parameter selection uncertainties affect the RUL prediction

The uncertainties are usually integrated in form of probability distribution, especially in model based prognostics. The process involves identification and quantification of

the uncertainties, uncertainty propagation from input to output and sensitivity analysis to evaluate how the uncertainties influence the output [140]. For methods that integrate incremental learning, as more training data becomes available during operation, the effect of uncertainties on RUL prediction reduces and the probability distribution becomes narrower as seen in Figure 2.13.

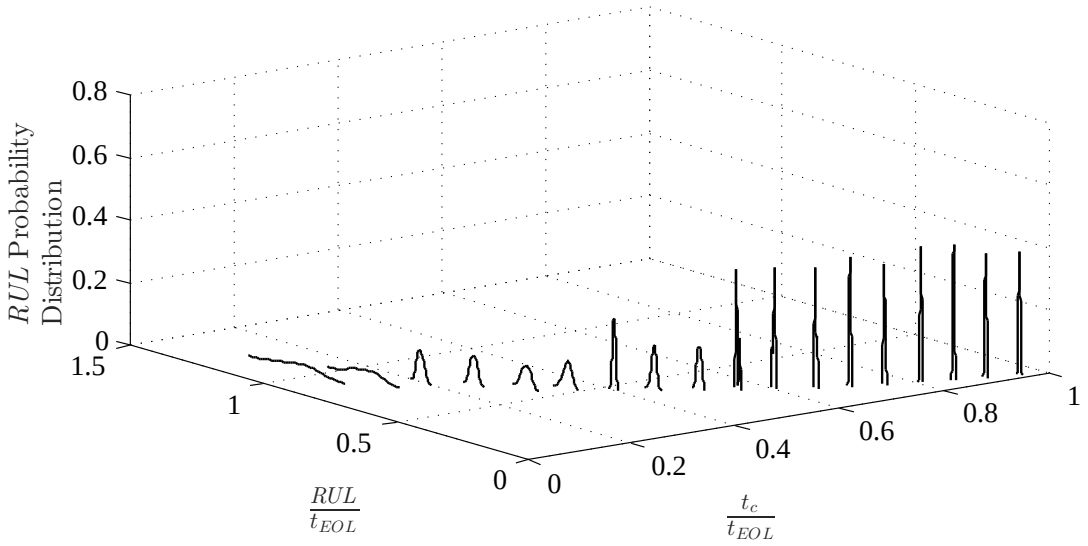

Figure 2.13: Probability distribution of RUL at specified prediction times

A number of approaches to incorporate uncertainties in prognostics have been proposed. Gu et al. [141] presented a method for uncertainty analysis of prognostics of electronic components subject to random vibration. The proposed approach employs a sensitivity analysis to identify the dominant input variables that affect the model output. Distributions of these input variables are then used in Monte-Carlo simulations to quantify damage. Liu et al. [142] proposed the use of Weibull distribution to quantify uncertainty in Lithium-ion battery prognostics by fitting the probability distribution to the output of multiple models of monotonic echo state network. The feasibility of the approach was evaluated with direct and indirect battery capacity measurements. Zhao et al. [143] proposed an integrated prognostic approach for gear RUL prediction based on finite element method for gear stress analysis, gear dynamic model for dynamic load calculation and a damage propagation model described using Paris' Law. The approach incorporates material and model uncertainties. Sun et al. [140] proposed a cognitive framework for uncertainty management in prognostics. The study proposed the use of fault state feature to quantify uncertainty and Monte Carlo methods for uncertainty propagation.

2.3.3 Confidence Intervals

In prognostics, confidence intervals CI consist of a range of RUL values that act as good estimates of the unknown value of RUL at each given prediction time [144]. The CI are calculated based on a sample population N_T of similar systems with available failure

times t_{EoL}. At any given operating time t_c, and assuming that the distribution of failure times of similar systems is Gaussian, the $(1-\alpha)100\%$ *CI* of the mean *RUL* can be calculated as [144]:

$$\overline{RUL} = \frac{1}{N_T}\sum_{i=1}^{N_T} t_{EoL,i} - tc, \tag{2.10}$$

$$CI_{(1-\alpha)\%} = \left[\overline{RUL} - \zeta_{\frac{\alpha}{2}}\frac{\sigma}{N_T}, \overline{RUL} + \zeta_{\frac{\alpha}{2}}\frac{\sigma}{\sqrt{N_T}}\right], \tag{2.11}$$

where $\overline{RUL}$ is the mean *RUL* of the sample population at the current prediction time t_c, σ is the standard deviation of *RUL* of the sample population, α is the level of significance and ζ is the critical value for $(1-\alpha)100\%$ confidence level. Typical values of ζ for a sample population with known standard deviation are shown in Table 2.2. The width of the *CI* depends on the sample size and the type of distribution and may be used as an indication of whether the estimated value falls within or it deviates from possible intervals of the sample population.

Table 2.2: Typical values of ζ for $(1-\alpha)100\%$ confidence level for a sample population with Gaussian distribution [145]

Confidence level	α	ζ
90%	0.10	1.645
95%	0.05	1.960
98%	0.02	2.326
99%	0.01	2.576

2.4 Conclusion

Selection of prognostic approaches as well as the algorithms largely depends on the technical system and the condition monitoring information available. In most applications of machine learning algorithms, there is no clear guideline on how to handle condition monitoring data for training and testing/online prognostics. There is also little information in literature on suitable approaches and algorithms to apply for a given technical system and this work aims at bridging this gap by creating a database of prognostic approaches as well as performance evaluation of these approaches on different kinds of condition monitoring data to serve as a guide for algorithm/method selection. Typical information necessary for method selection include:

- Type of condition monitoring data available such as, continuous type or single value type.
- Distribution of lifetimes of similar units.
- Degradation mechanism, for instance continuous degradation or degradation through discrete stages.

- Availability of measurable health index.
- Availability of mathematical models that describe failure.
- Operating conditions.

3 Data Preprocessing and Feature Extraction

The stages in a prognostic approach can be extracted from the key elements of a prognostics and health management system described in Chapter 1. Figure 1.2. The first step is the acquisition of condition monitoring data through a network of sensors. The data is then preprocessed and relevant features are extracted. Since not all features may contain useful information on the health of a system, a subset of features is selected and used as input to the prognostic method. Health indices or states are also defined at this stage, for use as the target in the prognostic method. This chapter discusses the first stages of a prognostic approach, which is mainly data acquisition and preprocessing.

3.1 Condition Monitoring Data

The success of a prognostic method depends on the information contained in the data acquired through sensors. Therefore, it is important to collect data that contains the most information about the condition or health of various components within the system. In addition, in order to reduce the cost and complexity of the system, it is important to identify critical components that need to be monitored and also evaluate whether operational data of a system could be used to indicate the condition of the system, thereby reducing the number of additional sensors in the system. Once sensor data that contains indicators of fault progression is acquired, it is pre-processed and features are extracted to reduce the dimensionality of the data. The choice of condition monitoring data depends on the system in consideration and the possible sensor data that can be acquired without interrupting operation while ensuring the most cost effective installation of the measuring units. In some cases where it is difficult to mount sensors for data acquisition or in structural health monitoring, active sensing is applied, where the system is excited at predetermined time intervals and the response is measured and analyzed to determine the condition of the system. In fluid machinery, a wide range of sensors for the acquisition of process data (pressure, temperature, flow rate, etc.) and fluid data (fluid condition, contamination, water content, etc.) are used for condition monitoring. The combined use of process data and fluid data together with machine data such as cycle times, energy consumption, etc., are processed and used for fault diagnosis and prognosis beyond the conventional threshold monitoring. The acquired data falls into three categories [146]:

i) Single value type, where the only one sample is recorded at each prediction interval. Typical examples include quasi-static condition monitoring data such as temperature, pressure, load.

ii) Continuous signal type, where at each prediction interval, the signal is sampled for a predetermined duration at a specified sampling frequency, e.g. vibration, force, electric voltage, electric current.

iii) Multi-dimensional type, e.g. thermal images, X-Ray images.

This section briefly describes a number of commonly used sensor data for condition monitoring of technical systems which are relevant to this work and are used in later stages.

3.1.1 Temperature Measurements

Temperature can be used as the condition monitoring data for systems with relative motion giving rise to wear-related failures. Wear is approximately proportional to frictional energy which is usually dissipated in form of heat leading to temperature rise [147]. By tracking the change in temperature with time, it is possible to track degradation and consequently conduct prognosis on such systems. Figure 3.1 shows a typical run-to-failure temperature curve of a ball bearing [148].

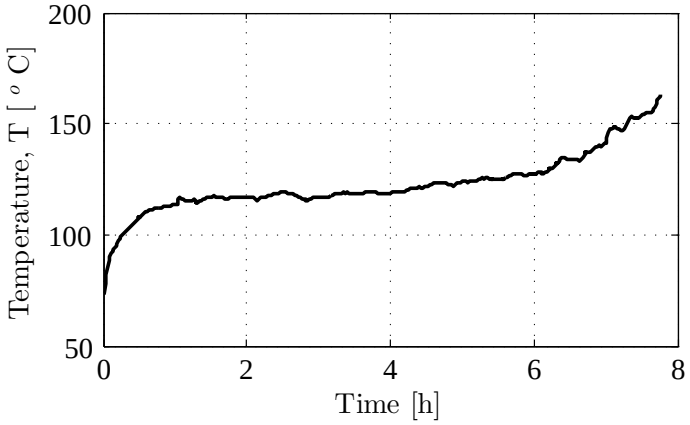

Figure 3.1: Run-to-failure temperature curve of a ball bearing

3.1.2 Performance Index Monitoring

In some systems, diagnosis and prognosis can be achieved through monitoring the performance of the technical system. Since degradation of components leads to loss of performance, performance monitoring data can be processed to diagnose faults as well as predict when a system is likely to fail. For instance, irreversible degradation of proton exchange membrane (PEM) fuel cells can be linked to their measured output voltage. By tracking the changes in the output voltage and defining suitable failure thresholds, the RUL of these systems can be estimated [149]. Figure 3.2 shows aging of a PEM fuel cell tracked using measured output voltage [150].

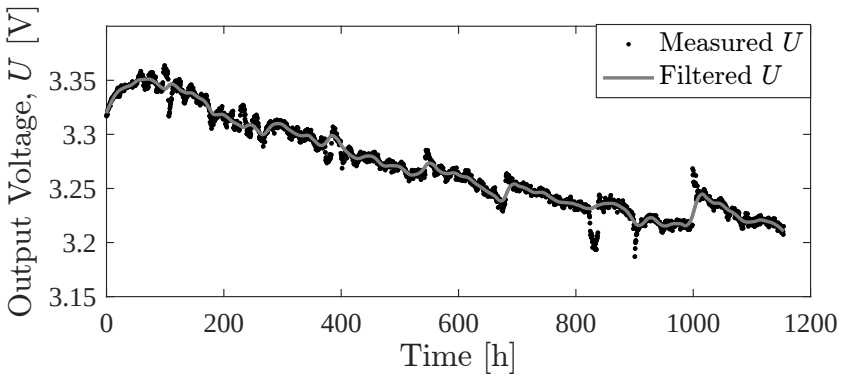

Figure 3.2: Aging of a PEM fuel cell tracked using output voltage

3.1.3 Vibration Measurement

Vibration measurement is the most common condition monitoring data for dynamic systems, especially rotating machinery, since vibrations are usually the first sign that a system is prone to failure. The vibrations are recorded for a specified period of time at predetermined intervals depending on typical lifetime or expected lifetime of the system. The vibration signals can be analyzed for fault diagnosis, for instance using frequency spectrum to identify fault characteristic frequencies as seen in Figure 2.2. Figure 3.3(a) shows run-to-failure vibration data of a rolling element bearing. The data consists of concatenated signals that are sampled at predetermined intervals up to the end of life of the bearing. Figure 3.3(b) is the root mean square (RMS) value computed for each sampled signal up to the end of life, to show fault progression [148].

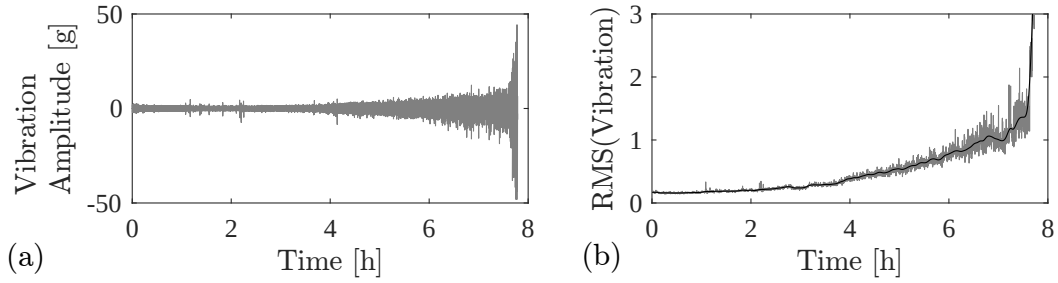

Figure 3.3: Vibration data of ball bearing (a) concatenated signals and (b) corresponding RMS value showing fault progression with continued operation

3.1.4 Force Measurements

Dynamic force can be used to monitor the condition of dynamic systems such as machine tools. A force transducer is mounted on the system subjected to dynamic loading and as the system degrades, the degradation manifests itself in the measured force as seen in Figure 3.4. Figure 3.4(a) shows concatenated signals that are sampled during each cutting

cycle of a milling machine tool, while (b) shows the corresponding RMS value computed for each sampled signal. The use of force measurement as the condition monitoring data has been widely used in machine tools for monitoring tool wear.

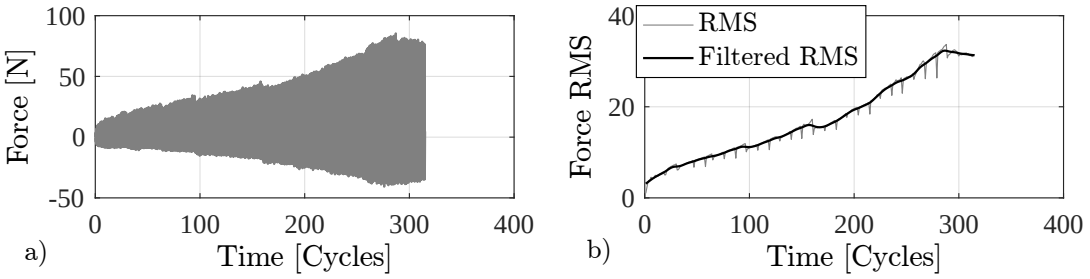

Figure 3.4: Run-to-failure force data of a milling machine tool (a) concatenated signals and (b) corresponding RMS value showing tool wear progression with continued operation

3.1.5 Electric Current Signatures

Motor current signal analysis (MCSA) is gaining popularity as condition monitoring data for electromechanical drive systems. Motor current measurements are obtained from current sensors embedded onto the system for control of motor operation, for example by a frequency inverter or for performance monitoring. The MCSA is especially used to detect broken rotor bars or motor bearing faults which result in changes in the air-gap [151, 152]. Recently, further investigations on the use of MCSA for detecting faults in the downstream equipment are being conducted [153, 154]. Figure 3.5(a) shows the raw current signals of healthy and faulty bearings obtained from a Permanent Magnet Synchronous Motor while Figure 3.5(b) shows clustering of features for a healthy and faulty bearings [155]. From the raw signals, it is not easy to distinguish between a healthy and faulty bearing. However, the different health states can be clearly observed from extracted features.

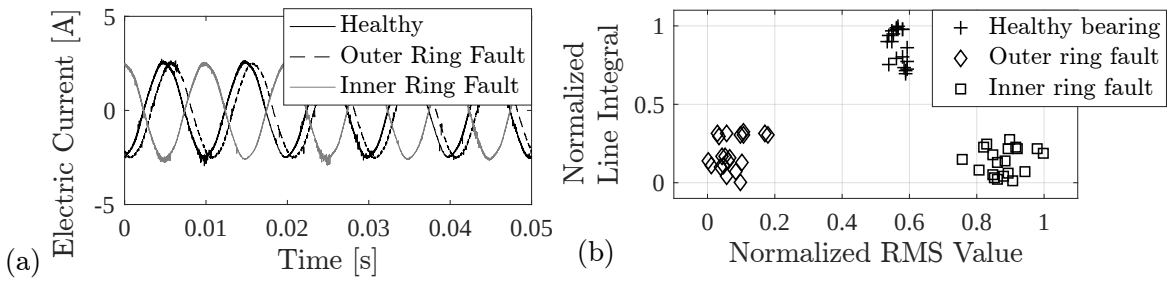

Figure 3.5: Application of MCSA for condition monitoring of an electromechanical drive system (a) raw current data (b) extracted features for different health states

3.1.6 Active Sensing

For systems which undergo random excitation, active sensing may be applied periodically to assess the health of the technical systems. In this case, the system is excited through a set of actuators and the response is measured using a set of sensors. With constant excitation, the response can be processed for fault diagnosis and prognosis. Typical application of this approach is structural health monitoring of systems such as wind turbines, bridges, composite structures for aerospace industry, etc. The approach can also be extended to monitoring piezoelectric transducers through the concept of self-sensing [156]. Figure 3.6 shows application of active sensing using piezoelectric transducers to detect delamination in fibre-reinforced composites subjected to fatigue cyclic loading [157]. Active sensing was conducted at intervals of 50,000 cycles. Figure 3.6(c) shows progressive reduction of the RMS value of the sensor signal as delamination increases.

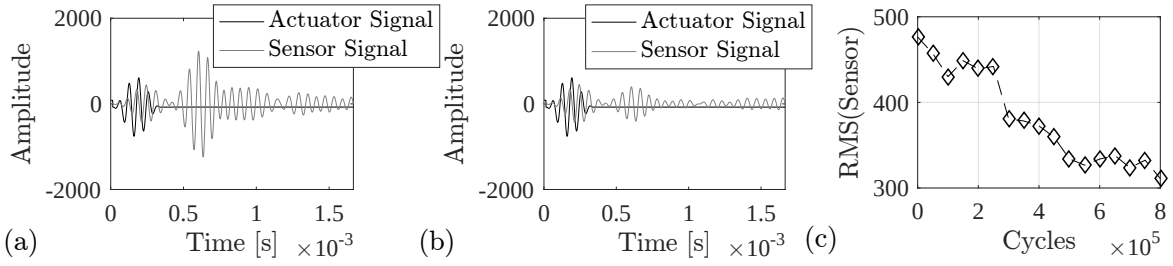

Figure 3.6: Sensor and actuator signals applied to a fibre-reinforced composite for (a) healthy (b) delaminated states and (c) progression of degradation

3.2 Data Preprocessing

Real world data is highly susceptible to noise, missing values, outliers and inconsistency. The quality of the data being processed for condition monitoring affects the fault diagnostic and prognostic results. In order to improve on the quality of the raw data and consequently of the prognostic results, the raw data is preprocessed through filtering to remove noise and data cleaning. Data cleaning involves filling in missing values, removal of outliers and resolving inconsistencies.

3.2.1 Wavelet Signal Denoising

Most of the acquired data for condition monitoring is usually corrupted by noise which should be removed for better condition monitoring results. Vibration is the most widely used condition monitoring data for rotating machinery. The measured signal is made up of the true vibration signal which consist of narrow band signals spaced by the fault characteristic frequencies and wide band noise [158]. Wavelets can be used for signal denoising to remove high frequency sub bands which contain the most of the noise and very

little signal information. Unlike the band-width based filter methods, wavelet denoising does not alter the nature of the signal and hence no important information on the state of a system is lost [159]. Wavelet denoising involves the decomposition of a signal into frequency components using discrete wavelet transform (DWT)as follows [160]:

1) The raw signal is decomposed into approximate and detailed coefficients using discrete wavelet transform
2) The detailed coefficients are suppressed using a threshold criteria and value.
3) The signal is reconstructed by applying inverse wavelet transform to the original approximate coefficients and suppressed detail coefficients.

Figure 3.7 shows raw and wavelet denoised vibration signals of a rolling element bearing at different health states [161, 162]. Figure 3.7(a) shows the vibration signals of a healthy bearing while Figures 3.7(b) and (c) shows the vibration signals of a bearing with fault on the outer ring (OR) and inner ring (IR) respectively.

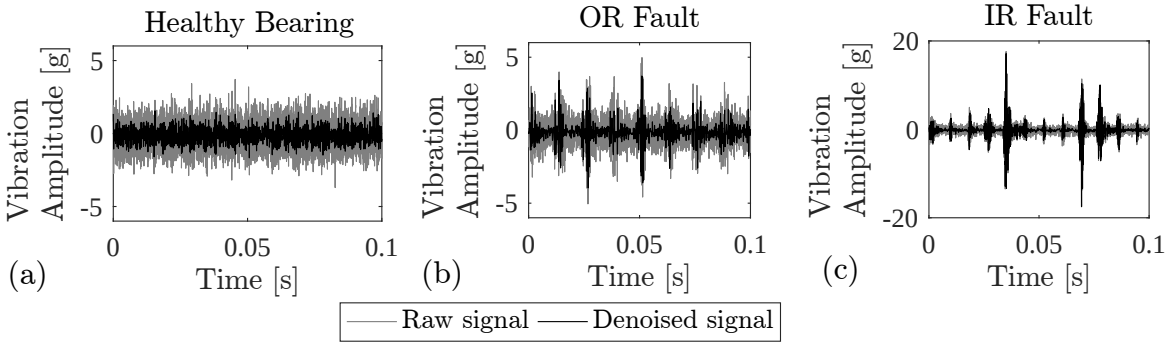

Figure 3.7: Raw and denoised ball bearing signals for different health states (a) healthy bearing, (b) bearing with outer ring (OR) fault and (c) bearing with inner ring (IR) fault

From Figure 3.7(b)-(c), the periodic impulses arising from bearing faults are enhanced by denoising. The frequency of these impulses depends on the bearing characteristic frequency which can be computed from bearing geometry and operating speed. Table 3.1 shows the bearing characteristic frequencies for the data in Figures 3.7 - 3.9. BPFI is the ball pass frequency of inner ring, BPFO is the ball pass frequency of outer ring, BSF is the ball spin frequency and FTF is the fundamental train frequency (frequency of the cage).

Figures 3.8 and 3.9 show the frequency spectrum of the raw and denoised vibration signals. It can be observed that the amplitude at the bearing characteristic frequencies are increased by denoising.

Another important feature used in fault diagnosis and prognosis in rotating machinery is the kurtosis which represents the fourth statistical moment of the distribution of data

Table 3.1: Bearing characteristic frequencies and operating conditions

Bearing information	BPFO	3.245× shaft speed
	BPFI	4.755× shaft speed
	BSF	2.556× shaft speed
	FTF	0.594× shaft speed
Operating conditions	Nominal speed	25 Hz
	Load	1200 N
	Sampling rate	97.6 kHz/ 48.8 kHz
	Record length	6 s/ 3 s

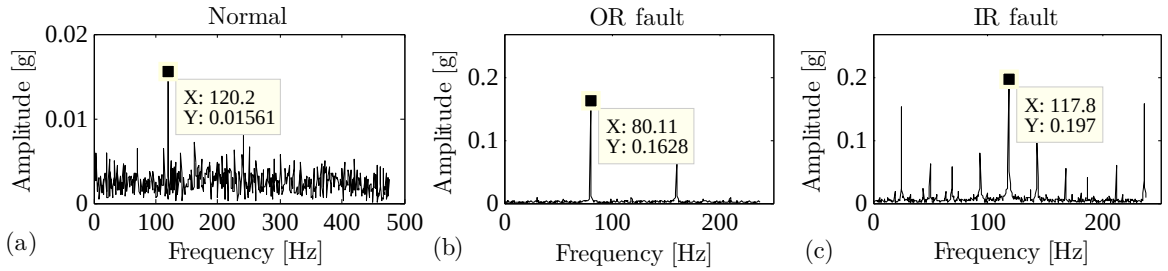

Figure 3.8: Frequency spectrum of a ball bearing at different health states obtained from raw vibration signals

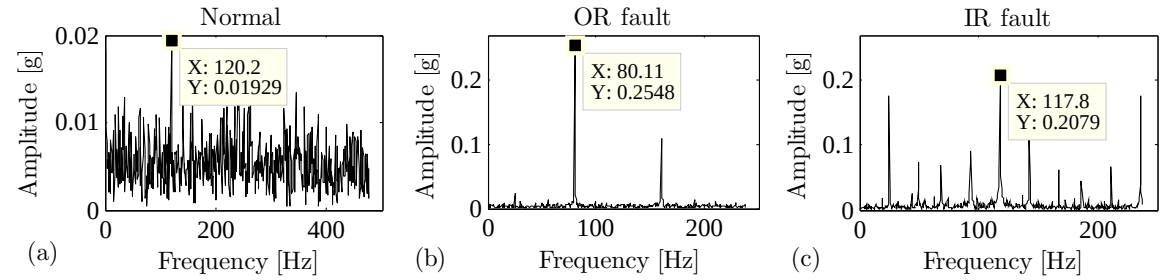

Figure 3.9: Frequency spectrum of a ball bearing at different health states from denoised vibration signals

about the mean [163] and is given by

$$\text{Kurt} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{s_i - \mu}{\sigma}\right)^4, \tag{3.1}$$

where μ is the mean of the signal and σ is the standard deviation of the data. It is used to measure the impulsive characteristic of a signal and as such it is an important feature for detecting and tracking evolution of rolling element bearing faults. The kurtosis of a healthy bearing is ≈ 3.0 [163]. As seen in Table 3.2, the maximum amplitude of the frequency spectrum and kurtosis of the denoised signals are higher than the raw signals, further emphasizing on the importance of denoising the signals.

Table 3.2: Comparison of features extracted from raw and denoised signals

Feature	Bearing condition	Raw signal	Denoised signal
FFT amplitude	Normal	0.0156	0.0193
	OR fault	0.163	0.255
	IR fault	0.197	0.208
Kurtosis	Normal	3.01	3.47
	OR fault	5.08	11.79
	IR fault	27.96	35.57

3.2.2 Outlier Removal and Filtering

Single valued condition monitoring data or features extracted from signals may contain outliers, which are values that are distant from other observations or inconsistent with other observations and may lead to false alarms being raised by the condition monitoring system. The outliers may be due to variability or data acquisition errors. Various methods for detecting and removing outliers have been proposed [164]. The condition monitoring data or feature may also contain noise which affects regularization of machine learning algorithms and therefore should be removed (feature filtering/smoothing). The most common method for smoothing is the moving average which involves creating a series of averages of different subsets of the full data set. The disadvantages of smoothing methods based on moving average is that it lags behind the trend and hence the end of the feature which represents the current health condition is biased to the historical data values. A more suitable approach for smoothing condition monitoring data and indices is the kernel based smoother which operates like a weighted moving average. The weights in the average depend on a kernel function $K(t)$, usually Gaussian [165].

Consider a signal $s(t)$ with n data points and let $s(t_i)$ denote a sample point, where $i = 1, 2, 3, ...n$. The estimate of the filtered data point, $s_f(t_i)$ is given as

$$s_f(t_i) = \frac{\sum_{j=1}^{n} sk_j \cdot s(t_i)}{\sum_{j=1}^{n} sk_j}, \tag{3.2}$$

where,

$$sk_j = K\left(\frac{t_i - t_j}{h_w}\right) \tag{3.3}$$

is a Gaussian Kernel function given by

$$K\left(\frac{t_i - t_j}{h_w}\right) = \frac{e^{-\frac{1}{2}\left(\frac{t_i - t_j}{h_w}\right)^2}}{\sqrt{2\pi}}, \tag{3.4}$$

with the filtered signal s_f and the smoothing window length h_w.

Figure 3.10 shows the noisy voltage with outliers (peaks) and smoothed voltage used as condition monitoring data for a PEM fuel cell [150].

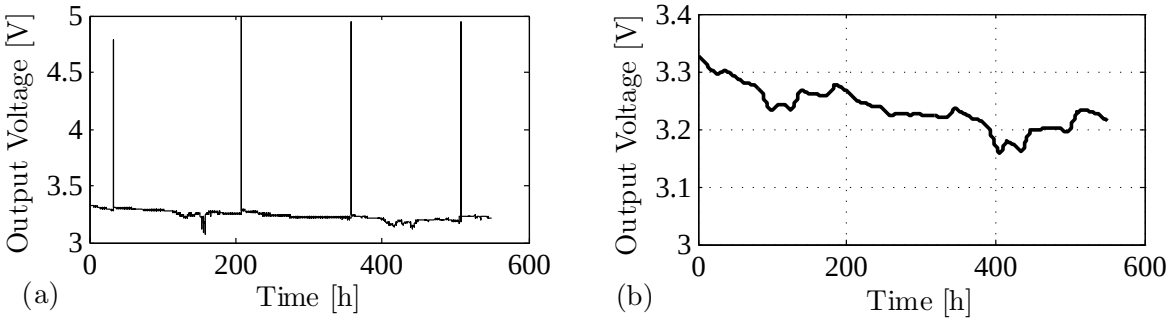

Figure 3.10: Condition monitoring data of a PEM fuel cell (a) raw data and (b) data filtered with a kernel smoother

3.3 Feature Extraction

In most technical systems, condition monitoring data is acquired for a certain duration at predetermined intervals leading to huge amounts of data. In order to reduce the data set and obtain better generalization of machine learning algorithms in both diagnosis and prognosis, it is important to extract features from the condition monitoring data (usually filtered or denoised data). Feature extraction also reduces the complexity and computational requirements of the machine learning algorithms. For waveform type and image type condition monitoring data, the features can be extracted in the following domains:

1. time domain features,
2. frequency domain features,
3. time - frequency domain.

3.3.1 Time Domain Features

These features are computed directly from the waveform/image data and contain statistical information pertaining to the health of the system [166] and usually represent energy, amplitude and distribution of the condition monitoring data. These features have been successfully employed in fault diagnosis of various technical systems such as bearings, gears, electromechanical drives [62, 166, 167]. Table 3.3 shows a list of time domain features commonly used in fault diagnosis and prognosis.

Figure 3.3(b) shows an increasing trend of the RMS value of vibration signal with damage propagation and hence it is a good feature for prognostics (Data obtained from [148]). Figure 3.11 shows the distribution of impulse factor for different health states of a ball bearing and hence its suitability for fault diagnosis.

Time synchronous averaging (TSA) methods have also been employed to isolate signals of different components in a system, for instance in gearboxes [166]. However, one must have prior knowledge of the repetitive frequencies of the desired signals, for instance gear mesh frequencies and bearing characteristic frequencies [132, 166]

Table 3.3: Time domain features

Feature	**Equation**
RMS value	$\text{RMS} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} s_i^2}$
Variance	$\text{variance} = \frac{1}{n}\sum_{i=1}^{n}(s_i - \bar{s})^2$
Skewness	$\text{Skewness} = \frac{\sum_{i=1}^{n}(s_i-\bar{s})^3}{(n-1)\sigma^3}$
Entropy	$\text{Entropy} = -\sum_{i=1}^{n} s_i \log_2(s_i)$
Kurtosis	$\text{Kurtosis} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{s_i-\mu}{\sigma}\right)^4$
Peak value	$\text{Peak} = \max(\lvert s_i \rvert)$
Peak to peak	$\text{Peak to Peak} = \max(s_i) - \min(s_i)$
Crest factor	$\text{Crest factor} = \frac{\text{Peak}}{\text{RMS}}$
Shape factor	$\text{Shape factor} = \frac{\text{RMS}}{\frac{1}{n}\sum_{i=1}^{n}\lvert s_i \rvert}$
Clearance factor	$\text{Clearance factor} = \frac{\text{Peak}}{\left(\frac{1}{n}\sum_{i=1}^{n}\lvert s_i \rvert\right)^2}$
Line integral	$\text{Line integral} = \sum_{i=1}^{n}\lvert s_{i+1} - s_i \rvert$
Impulse factor	$\text{Impulse factor} = \frac{\text{Peak}}{\frac{i}{n}\sum_{i=1}^{n}\lvert s_i \rvert}$

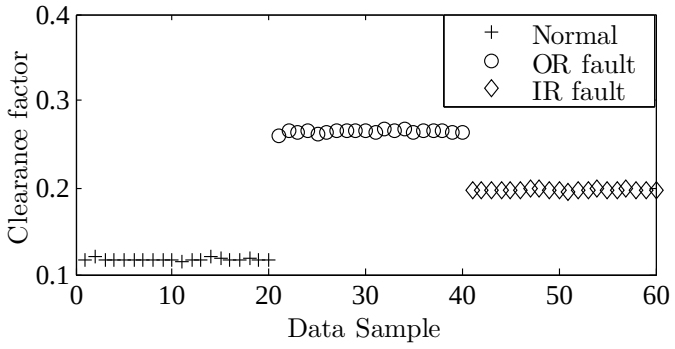

Figure 3.11: Time domain features showing feature distribution for fault diagnosis

3.3.2 Frequency Domain Features

A visual observation of a signal may not reveal the periodic components of the signal associated with faults. However, the Discrete Fourier Transform (DFT),

$$S(m\omega_s) = \sum_{i=0}^{n-1} s_i e^{-jm\omega_s k}, \quad m = 1, 2...n, \tag{3.5}$$

where S is the signal in frequency domain, s is the signal in time domain, $\omega_s = \frac{2\pi}{n}$ is the angular frequency, can be computed to see if the signal has characteristic frequency components [168]. Therefore, extraction of frequency features involves transforming the signal from time domain to frequency domain. These features are important in identifying characteristic frequencies of a system associated with various types of faults, as well as resonance frequencies of the system. Fast Fourier Transform (FFT) is the most widely used fault diagnosis technique for identifying frequency features from condition monitoring data [166]. Once in the frequency domain, features in Table 3.4 may be extracted for fault diagnosis and prognosis.

Table 3.4: Frequency domain features

Feature	**Equation**
Maximum amplitude	Maximum Amplitude = $\max(S)$
Frequency at maximum amplitude	Frequency = $\omega_s\|_{\max(S)}$
Energy of signal	$E_s = \sum_{m=1}^{n} \|S_m\|^2$

Figure 3.12(a) shows the maximum amplitude of the frequency spectrum of vibration signals of a ball bearing at different health states which shows separation of different health states and hence it is a good feature for fault diagnosis. Figure 3.12(b) shows the evolution of the maximum amplitude of the frequency spectrum with degradation of a bearing when run to failure. Since this feature shows a trend consistent with degradation of the bearing, it is suitable for prognostics.

3.3.3 Time-Frequency Domain Features

The disadvantage of frequency-domain features is that only the frequency components are captured and therefore the features may not show how the frequency components change with time or the location of these frequency components in the signal. They are therefore not suited for non-stationary signals, that is, signals whose frequency components are changing with time. Some technical systems such as bearings and gears generate non-stationary vibrations in presence of faults and require feature extraction techniques capable of presenting the features in both time and frequency domain [169].

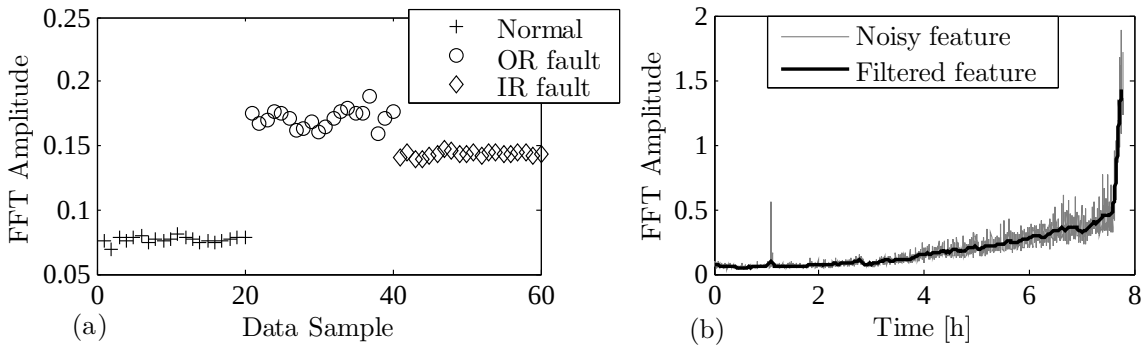

Figure 3.12: Frequency domain features showing (a) feature distribution for fault diagnosis and (b) feature evolution with damage progression

Various time-frequency methods for analyzing non-stationary signals in condition monitoring have been proposed [35, 159, 169, 170]. Of these methods, discrete wavelet transform (DWT) and empirical mode decomposition (EMD) have received a lot of attention due to their high resolution in both time and frequency domains [35, 171]. In DWT, described in section 3.2.1, the signal is decomposed into a sequence of wavelet coefficients from which statistical and frequency features can be extracted. EMD also involves decomposing a signal into a series of components called intrinsic mode functions (IMFs). Each IMF must satisfy the following rules [172]:

1. The number of extrema and the number of zero crossing should be equal to 0 or differ by at most 1.

2. The mean value of the envelope defined by local maxima and the envelope defined by local minima should be zero.

EMD decomposes the signal into these IMFs as follows [172]:

1. Identify all local extrema in the data and connect the local maxima using a cubic spline to form the upper envelope. Connect the local minima to form the lower envelope.

2. Obtain the mean of the envelope of local maxima and envelope of the local minima, $m_1(t)$ and subtract it from the original signal,

$$h_1(t) = s(t) - m_1(t). \tag{3.6}$$

   If $h_1(t)$ is an IMF, it becomes the first component of $s(t)$, if not, $h_1(t)$ becomes the original signal and the process is repeated k times until an IMF is obtained or a stopping criteria (e.g when the standard deviation of components from two consecutive trials is 0.1) is reached. This process is called sifting and the resulting component is the IMF,

$$imf_1(t) = h_{1k}(t). \tag{3.7}$$

3. The process is then repeated to obtain subsequent IMFs, $imf_2(t)....imf_{nf}(t)$. The process is stopped when $imf_{nf}(t)$ becomes monotonic, which is referred to as a residue, $r_{nf}(t) = imf_{nf}(t)$.

4. By summing up all the IMFs, the original signal is obtained as follows

$$s(t) = \sum_{j=1}^{nf} imf_j(t) + r_{nf}, \tag{3.8}$$

where $imf_i(t)$ are intrinsic mode functions and r_{nf} is the residue [172].

The IMFs consist of different frequency components and can be used to isolate characteristic frequencies of various faults from the condition monitoring data as shown in Figure 3.13. In Figure 3.13, the original vibration signal is decomposed into 13 IMFs (only 4 IMFs and the residue are shown). Relevant statistical and frequency features can then be extracted from the IMFs.

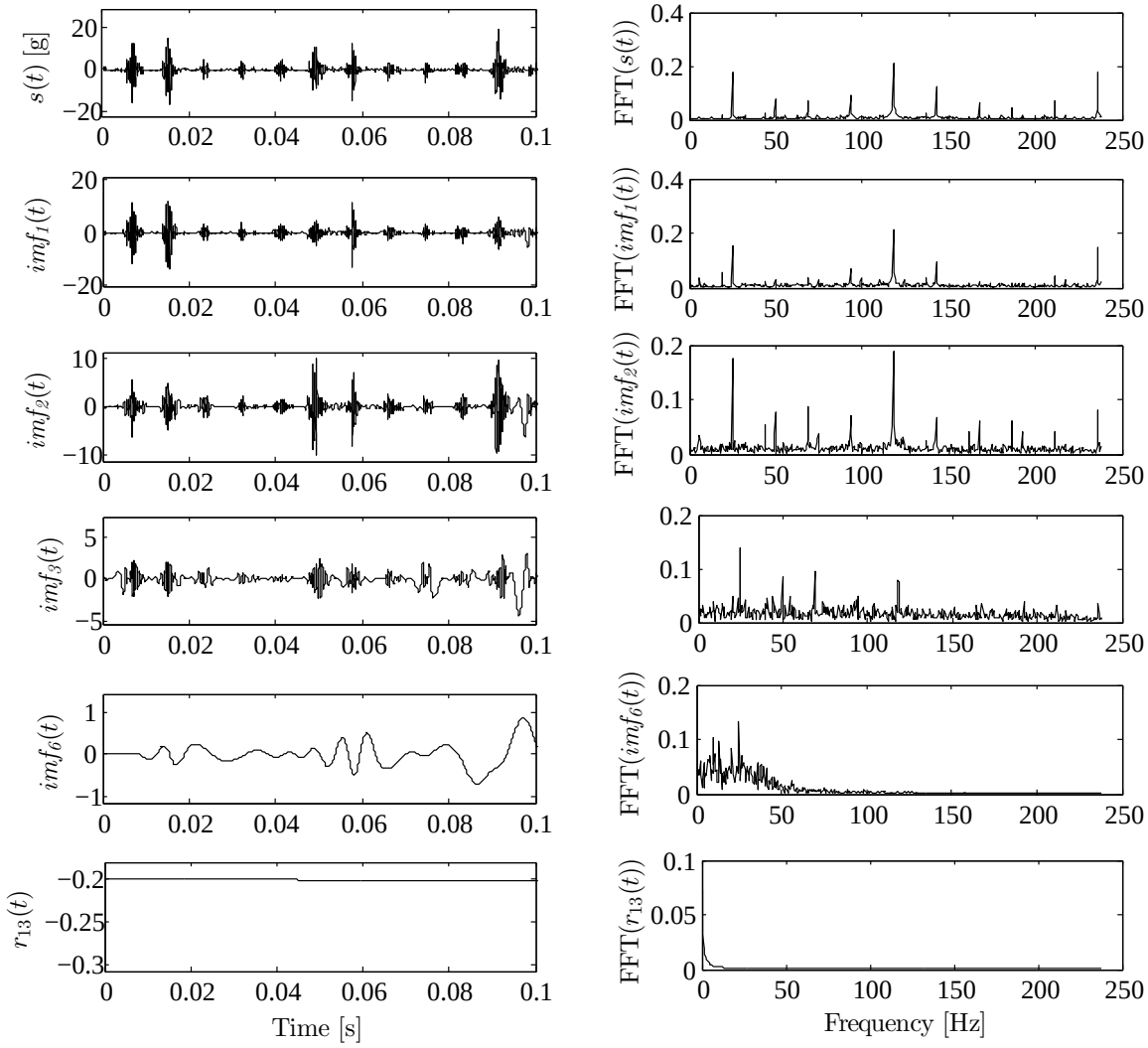

Figure 3.13: EMD of a vibration signal of a bearing with inner ring fault and corresponding frequency spectrum (shaft frequency ≈ 25 Hz, BPFI=118.2 Hz)

3.4 Feature Selection

Not all extracted features may contain useful information on the condition of a technical system and in order to improve the performance of the diagnostic and prognostic algorithms (classification or regression approaches), it is important to select a subset of features that contain the most information relating to faults or damage propagation. Feature selection helps to improve accuracy, avoid overfitting as well as reducing computation time or algorithms by selecting only relevant features and discarding redundant features. Feature selection approach depends on the application and may fall under two categories: classification and regression. There has been considerable effort to develop algorithms for automatic feature selection, mostly focused on feature selection for fault diagnosis. In this case, features are selected based on their capability to discriminate between different classes (fault categories or health states). Linear discriminant analysis (LDA) which is based on the assumption that different classes generate data based on Gaussian distributions has been employed in feature selection for fault classification [173]. Distance evaluation technique, which computes the largest distance separating data between classes, is another feature selection technique that has been employed to select the optimal features that represent the different health states of a degrading component [123]. Camci et al., [174] proposed a feature evaluation method for effective bearing prognostics based on separability value. The features were divided into time segments and the separability of the segments based on 25th and 75th percentile distributions computed. The overall separability value of each feature was then computed as a feature evaluation value. The use of separability of features as a method of feature selection can also be found in [175]. Benkedjouh et al., [176] employed isometric feature mapping reduction technique to find a small number of features that represent a large number of observations. Other methods of feature selection can be found in [56, 79, 177]. Tran and Yang [78] presented a method for feature selection based on classification and regression trees. Most of the methods presented were found to have high computational requirements and in some cases have poor generalization in that a different subset of features are selected for the same data set at different test runs. In this work, two approaches for feature selection have been proposed and are presented in the following sections.

3.4.1 Feature Selection for Health State Classification

Suitable features for fault or health state classification should provide a good separation between different classes. In this section, a new feature selection method based on maximum separation distance between different health states is presented [155].

Given a feature set of $j = 1, 2, ...Q$ features in $c = 1, 2, ...N_c$ classes or health states, the feature selection is performed as follows:

1. Normalize the features between 0 and 1.
2. Compute the mean of each normalized feature x_j within class c as follows

$$m_{jc} = \frac{1}{n} \sum_{i=1}^{n} x_{ijc}. \tag{3.9}$$

3. Compute the mean of the squared Euclidean distance between each feature data point, i and the mean of the corresponding feature in each class,

$$d_j = \frac{1}{N_c^2 n} \sum_{k=1}^{N_c} \sum_{c=1}^{N_c} \sum_{i=1}^{n} \left(x_{ijk} - m_{jc} \right)^2. \tag{3.10}$$

4. Normalize the separation distance with the maximum feature separation distance to produce a performance evaluation criteria

$$\bar{d}_j = \frac{d_j}{\max(\underline{d})}. \tag{3.11}$$

5. Select distance with a performance greater than a predetermined threshold. The threshold can be defined by evaluating the classification accuracy for each combined set of features.

The performance of this approach was evaluated with experimental data for condition monitoring of defects and damages of rotating machinery obtained from the Chair of Design and Drive Technology, University of Paderborn, as described in the next section.

3.4.1.1 Experimental Set-up

The test rig is a modular system for diagnosing various defects in rotating machinery components such as gearboxes, electrical machines and rolling element bearings and shafts [155]. The electric motor current and vibration signals of test bearings are measured for bearings at different health conditions such as normal, inner ring and outer ring pitting or fracture. Figure 3.14 shows the test rig consisting of several modules: a drive motor, a torque-measuring shaft, a bearing module, a flywheel and a load motor.

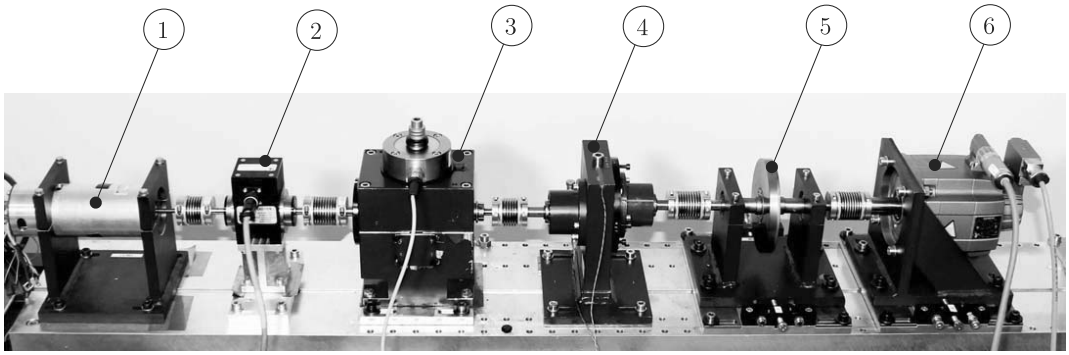

Figure 3.14: Modular testrig for fault diagnosis of rotating machinery: (1) drive motor, (2) torque measuring shaft, (3) rolling element bearing module, (4) gear module, (5) flywheel, (6) load motor [155]

Table 3.5: Operating conditions

Designation	**Rotational Speed [RPM]**	**Load Torque [Nm]**	**Radial Load [N]**
$N09\_M07\_F10$	900	0.7	1000
$N15\_M01\_F10$	1500	0.1	1000
$N15\_M07\_F04$	1500	0.7	400
$N15\_M07\_F10$	1500	0.7	1000

Several types of damages were produced and categorized by their type and severity. These damages consist of indentation and pitting. The pitting size is categorized by the length of the damaged surface in the rolling direction as shown in Table 3.6. Data was acquired at four operating conditions shown in Table 3.5.

Table 3.6: Grouping of bearing damages and severity for both inner ring and outer ring faults

Designation	**Damage type and severity**
P1	small pitting ($<$ 2 mm)
P2	medium pitting (2 - 5 mm)
P3	large pitting ($>$ 5 mm)
I	indentation

Two electric current signals and one vibration signal were recorded at a sampling frequency of 64 kHz for a period of 4 seconds. For each bearing condition and operating condition, 20 samples were recorded, with each sample recorded after dismantling and reassembling the bearing module. This ensures that assembly variability expected in real practice is taken care of.

A total of 31 features were extracted from each signal as follows:

1. 12 time domain features shown in Table 3.3.
2. 7 frequency domain features (Table 3.4) consisting of:
   - Maximum amplitude of FFT of each signal, corresponding frequency and energy of FFT signal.
   - Maximum power spectral density (PSD) amplitude and corresponding frequency.
   - Maximum amplitude of FFT of envelope signal and corresponding frequency.
3. 12 time-frequency domain features extracted from the first six IMFs of EMD. The 12 features consist of signal energy and clearance factor of each of the six IMFs.

Three samples were randomly selected for feature extraction using the proposed method. Figure 3.15 shows the performance evaluation of the features extracted from (a) electric current signals and (b) vibration signal.

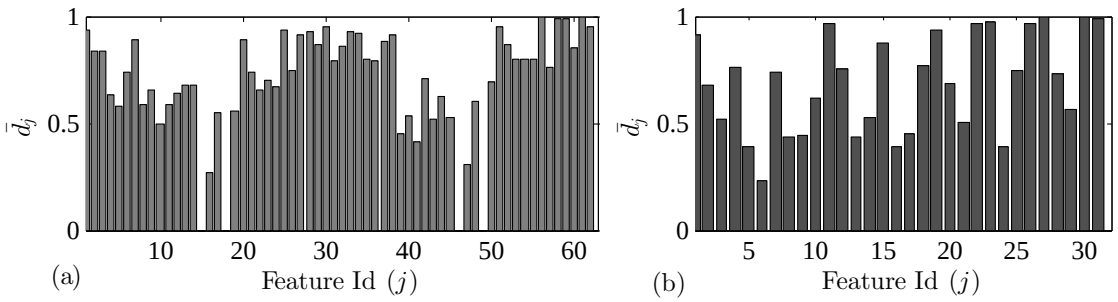

Figure 3.15: Normalized separation distance of features extracted from (a) electric current signals and (b) vibration signals

A plot of the two features showing the maximum separation distance between classes for the two types of signals is shown in Figure 3.16. Features from the vibration signals show a better separation between classes compared to the motor current signals. However, motor current signals could still be used for diagnostics and prognostics since the different health states can be clearly observed from the features.

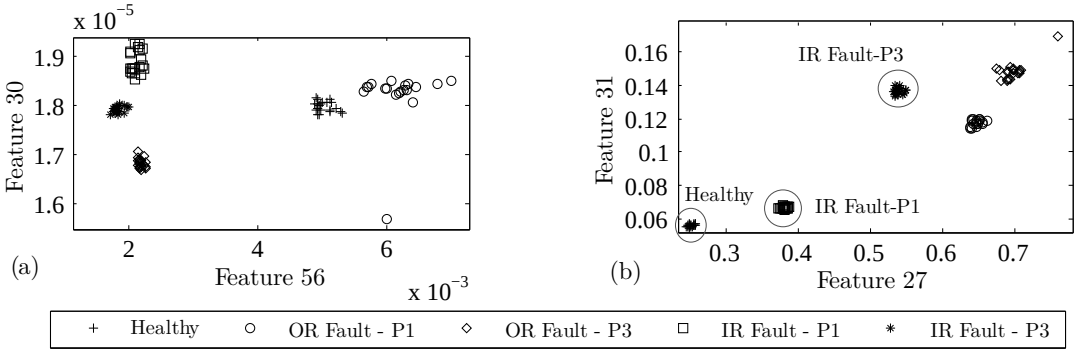

Figure 3.16: Clustering of different health states from (a) motor current signals and (b) vibration signals

3.4.2 Feature Selection for Regression

Regression involves analyzing the relationship between variables and in the context of prognostics, this process involves estimating a function that represents the health index, degradation or remaining useful lifetime of a technical system. The process involves mapping condition monitoring data or features extracted from the data to a function and hence feature selection involves selecting features that accurately represent the health index. In this work, kernel based extreme learning machines (ELM) is employed for feature selection due to its robust prediction and fast training and prediction times. The features are first evaluated individually on their ability to estimate the health index. The input to the ELM method is the feature x_j while the target vector y is the health index or RUL. The performance of the features is based on the mean square error (MSE)

between the actual health index and the predicted health index is computed as follows:

$$\mathrm{MSE}_j = \frac{1}{n}\sum_{i=1}^{n} y_i^2 - y_{ijp}^2, \tag{3.12}$$

$$p_j = \frac{1}{\mathrm{MSE}_j}, \tag{3.13}$$

$$\bar{p}_j = \frac{p_j}{\max(\underline{p})}. \tag{3.14}$$

This yields performance values within the range, $0 \leq \bar{p}_j \leq 1$ and the selection criteria can be evaluated by iteratively selecting features above a minimum $\bar{p}_j$ and computing the MSE of the predicted health index.

3.4.2.1 Extreme Learning Machines (ELM)

Extreme learning machines is a relatively new simple learning algorithm for single-hidden layer feedforward neural network (SLFN) which was first proposed by Huang in 2005 [178]. Figure 3.17 shows the structure of an SLFN with radial basis function (RBF) hidden neurons. x_j is the input vector at the input neuron j, a_l is the input weight connecting the hidden neuron l and the input neurons, b_l is the bias of the hidden neuron, β_l is the output weight of the hidden neuron and y is the output [178].

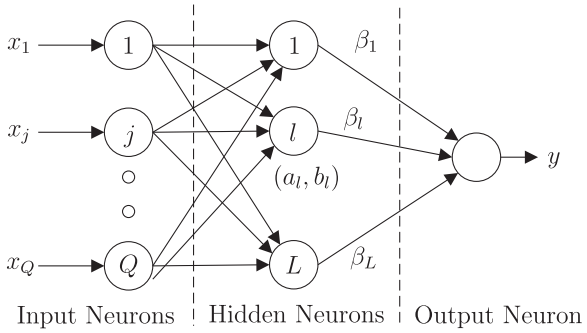

Figure 3.17: Structure of ELM [178]

In ELM, the input weights and hidden layer biases of SLFN are randomly generated, while the output weights linking the hidden layer to the output layer are determined through simple generalized inverse operation of the hidden layer output matrices [178]. The ELM learning process is extremely fast compared to other machine learning algorithms such as support vector machines and artificial neural networks with back propagation [178] and involves three steps:

1. The input weights a_l and hidden node bias b_l are randomly generated and used with an activation function to calculate the hidden layer output,

$$G_l = \sum_{i=1}^{Q} g_j(a_l, b_l, x_j), \tag{3.15}$$

where g_i is the activation function which can be sigmoid, sine, radial basis function (RBF), etc. In this work, the RBF was found to yield the best performance for most applications,

$$g_j(a_l, b_l, x_j) = \exp\left(-\frac{b_l|x_j - a_l|}{\gamma^2}\right), \tag{3.16}$$

where γ is the RBF parameter. The network output is given by

$$y = \sum_{l=1}^{L} \beta_l G_l. \tag{3.17}$$

2. Given n training samples $\{\underline{x}_i, y_i\}$, where $i = 1, 2..n$, the output weights are calculated as follows

$$\underline{\beta} = \mathbf{H}^{\dagger}\underline{y}, \tag{3.18}$$

where $\mathbf{H}^{\dagger}$ is the Moore-Penrose generalized inverse of the hidden layer output matrix $\mathbf{H} = [\mathbf{G}^T(\underline{x}_i)....\mathbf{G}^T(\underline{x}_n)]^T$.

To improve generalization performance so that the ELM can accurately predict new input data, a regularization parameter is added. In this case, the output weights are obtained as follows

$$\underline{\beta} = \left(\frac{\mathbf{I}}{C} + \mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\underline{y}, \tag{3.19}$$

where C is a regularization parameter and $\mathbf{I}$ is a unit matrix.

3. The learned model consists of the input and output weights and the hidden node biases which can be used with input data $\underline{x}$ to predict an output y_p, in a same way as the general approach for supervised machine learning algorithms shown in Figure 2.5.

3.4.2.2 Application Example

To evaluate the proposed feature extraction method, run-to-failure condition monitoring data for a high-speed CNC milling machine provided for the 2009 PHM Society Conference was used [179]. Condition monitoring data was recorded from three types of sensors:

1. Dynamometer to record force measurements in three directions.
2. Accelerometer for recording vibrations in three directions.
3. Acoustic emission sensors for recording the acoustic emissions data.

Data was acquired at 50 kHz for an average duration of 4 seconds during each cutting cycle. Table 3.7 shows the operating conditions of the machine.

In addition to the condition monitoring data, the flank wear of a 6mm ball nose tungsten carbide tool was measured after each cycle using a microscope system to provide the ground truth wear of the tool. The wear was measured from three flutes of the tool and the maximum wear of the three measurements shown in Figure 3.18 is taken as the health index. Six data sets are provided but only three that contain the wear measurements are used in this study.

Table 3.7: Milling machine operating conditions

Item	Value
Spindle speed	10,400 RPM
Feed rate	1555 mm/min
Y depth of cut (radial)	0.125 mm
Z depth of cut (axial)	0.2 mm

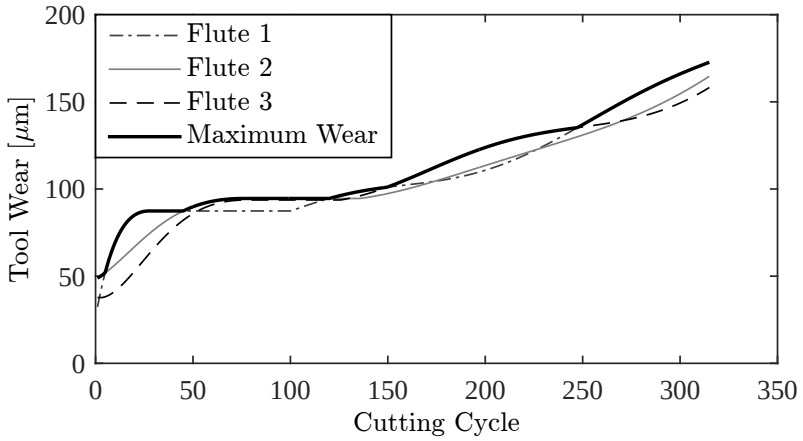

Figure 3.18: Tool wear for test 1

31 features are extracted from each signal, yielding a total of 217 features. Each feature is evaluated on its ability to estimate the tool wear. Half of the samples are randomly selected and used to train the ELM algorithm while the other half is used for testing and validation. The performance of the method is evaluated as described in section 3.4.2.

Figures 3.19-3.21 show the performance of each sensor data as well as a plot of the best and worst performing feature. The best performing features show an increasing trend consistent with degradation while the worst performing feature does not show a consistent trend. However, some of these features may be suitable with classification.

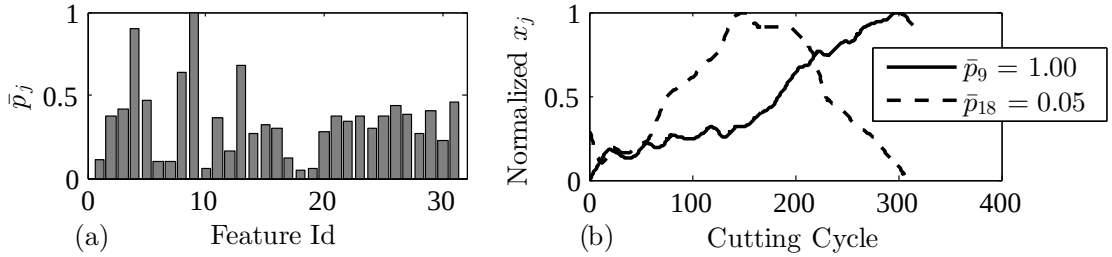

Figure 3.19: Performance of features from accelerometer in X direction (a) Performance for each feature, (b) A plot of best and worst features

To determine the threshold of $\bar{p}_j$ for feature selection, $\bar{p}_j$ was varied within the range $0 \leq \bar{p}_j \leq 1$ and all features with $\bar{p}_j$ above each value were selected for training using 60%

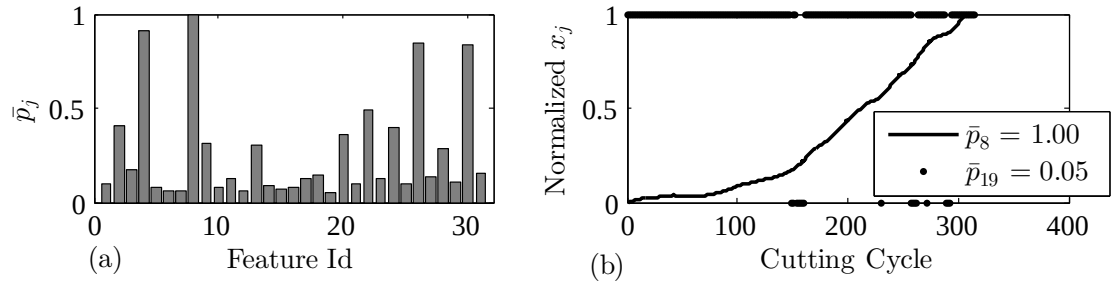

Figure 3.20: Performance of features from dynamometer in X direction (a) Performance for each feature, (b) A plot of best and worst features

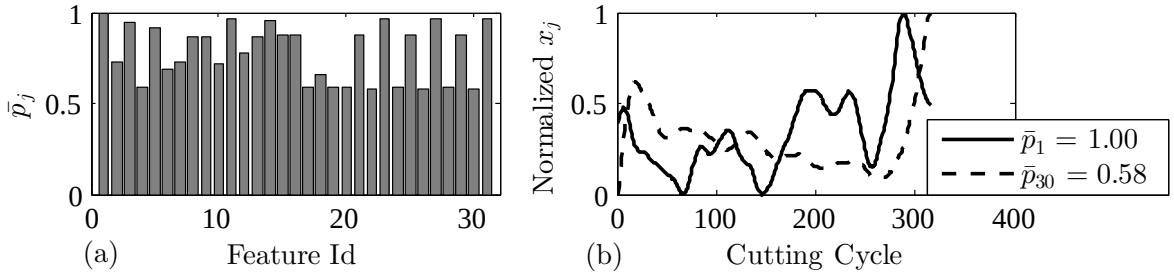

Figure 3.21: Performance of features from acoustic emission data (a) Performance for each feature, (b) A plot of best and worst features

of randomly selected samples and 40% for testing. The MSE between the actual and the predicted wear of the test data was computed. For this application, the minimum MSE was found by selecting features with $\bar{p}_j \geq 0.65$ as shown in Figure 3.22(a).

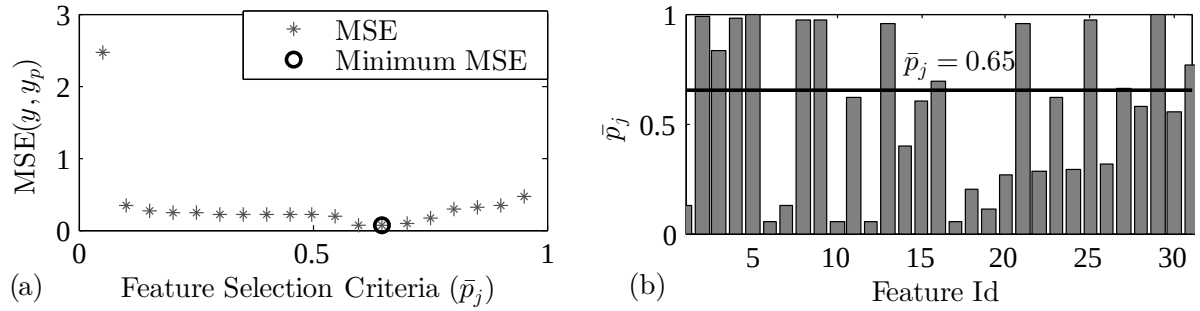

Figure 3.22: Feature selection criteria (a) MSE between predicted and actual wear for features with $0 \leq \bar{p}_j \leq 1$ and (b) selected features with $\bar{p}_j \geq 0.65$

Figure 3.22(b) shows the features with $\bar{p}_j \geq 0.65$ for accelerometer Z, while Table 3.8 shows the number of features selected from each sensor.

To demonstrate the feasibility of this approach, for each trial, two data sets were selected for training and one for testing. Figure 3.23 shows a comparison of the actual tool wear

Table 3.8: Number of features selected per sensor

Sensor	Number of features
Accelerometer X	2
Accelerometer Y	7
Accelerometer Z	10
Dynamometer X	4
Dynamometer Y	8
Dynamometer Z	2
Acoustic Emission	21

and predicted tool wear using all features as input and using selected features as input. Using selected features as input to ELM results to a better estimation of the tool wear.

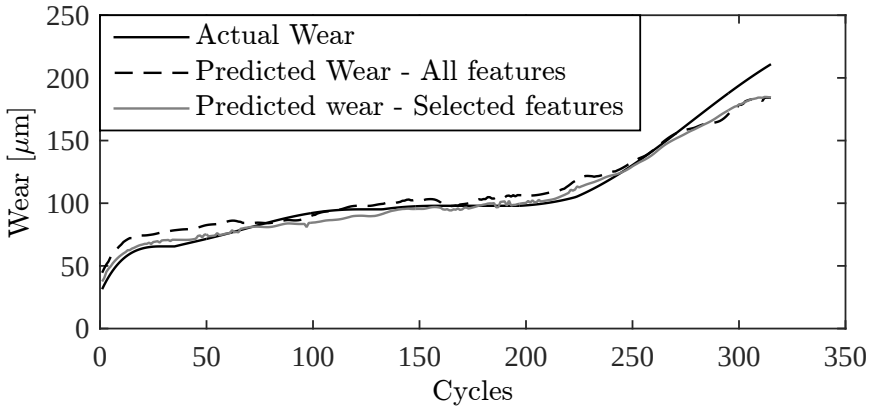

Figure 3.23: Comparison of actual and predicted tool wear with all features and with selected features as input to ELM for test data set 2

Table 3.9 shows the MSE between actual and predicted tool wear. The MSE obtained with feature selection is much lower, thus emphasizing on the need for feature selection.

Table 3.9: Comparison of MSE of predicted tool wear using ELM with all features and ELM with selected features as input

Test	All features	Selected features
1	128.13	69.21
2	80.44	44.90
3	166.86	103.36

Another approach for feature selection is to set the number of features to use, Q_f. The $\bar{p}_j$ is then ordered in a descending manner and the first Q_f features are selected. However, this approach may not provide the optimum features to use for a given application.

4 Development and Evaluation of Prognostic Approaches

The following sections outline different methodologies that have been evaluated on their suitability for fault diagnosis and prognosis of technical systems. The methods can be categorized into five approaches depending on the condition monitoring data available:

1) Health state estimation.
2) Application of machine learning algorithms to map extracted features to remaining useful lifetime (RUL).
3) Application of machine learning algorithms to map extracted features to a health index HI.
4) Propagating a health index HI to a threshold.
5) Model-based approach utilizing temperature measurements.

4.1 Health State Estimation

Some technical systems undergo a number of discrete health states before failure. For instance, a ball bearing pitting fault may begin on a race of a rolling element bearing which propagates to the rolling element and on to other bearing components before failure. This evolution of the fault may exhibit a number of health states which can be identified using classification algorithms. If run to failure data of similar systems is available, unsupervised machine learning algorithms such as k-means and self-organizing maps (som) of neural networks can be used to discretize the condition monitoring data/features into a number of clusters (health states). The clustered data is then used to train supervised machine learning algorithms for classification to identify the current health state from features extracted from condition monitoring data. This is done in two phases:

1. Training, where the machine classification algorithm uses labeled data to learn the underlying behavior between different health states. This results to a classification model with the necessary weights, biases and parameters.
2. Testing or online condition monitoring phase where unlabeled features from condition monitoring data of a similar unit are used as the input to the algorithm for classification.

Machine learning algorithms that produce health state probabilities as an output are better suited for this prognostic approach since the health state probabilities can be combined with the historical lifetimes of similar systems at each health state to estimate the RUL. Such algorithms include support vector machines (SVM), random forests

(RF) and probabilistic neural networks (PNN). In this study, only SVM and RF are implemented since they have been found to result in high classification accuracies [180].

If the health states of the training data are already known, for instance through FMEA, then this approach can be used for both diagnosis and prognosis. It may also be integrated in multilevel dependability concept for self-optimizing mechatronic systems [6], as well as low level prognosis where only the current health state is indicated and when the system reaches the critical health state, then an alarm for shut-down or maintenance planning is raised. Figure 4.1 shows the workflow of this approach for long-term prognostics.

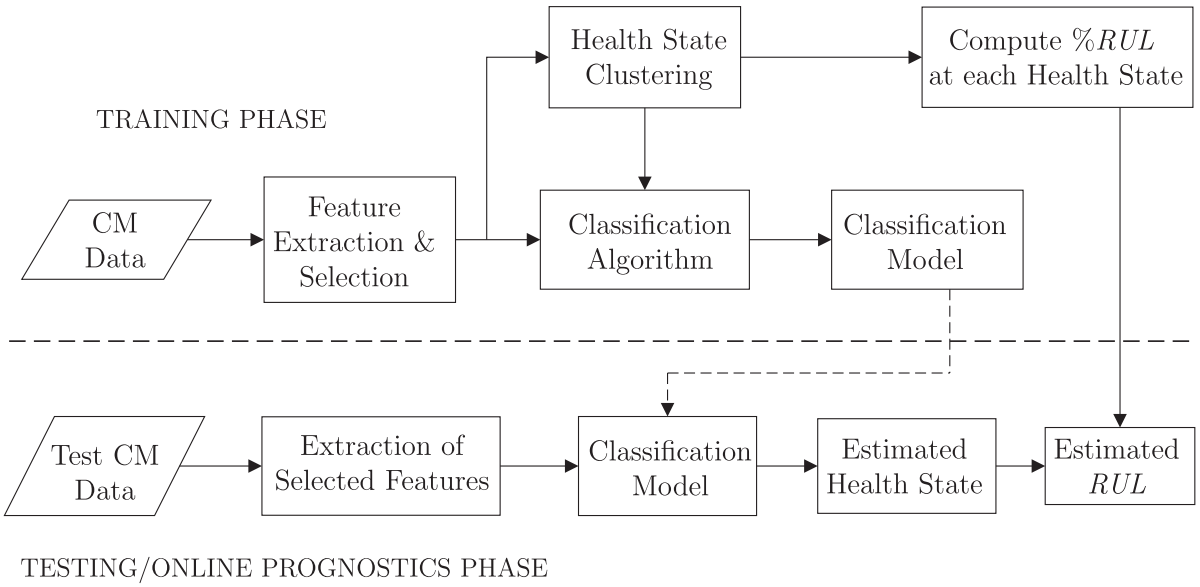

Figure 4.1: Work flow of health state estimation approach to prognostics

The following subsections discuss the background of support vector machines (SVM) and random forests (RF), which are implemented with this approach. When run-to-failure data of a system is not labeled, then the number of possible health states is unknown and should be determined prior to clustering the data into the health states. After the discussion of SVM and RF, a method for identifying the number of possible health states in the data is discussed. The approach is then evaluated with run-to-failure data of ball bearings and wear data of a high speed milling machine.

4.1.1 Support Vector Machines (SVM)

Support Vector Machines (SVM) is a supervised machine learning technique for solving binary data classification problems introduced in 1995 by Vapnik et al. [181]. The binary classification problem is solved by finding a hyperplane that separates the data into its two classes as shown in Figure 4.2 [104]. The planes that are parallel to the hyperplane and lie on the border of each class are called bounding planes. Data points that lie on the bounding planes are called support vectors [104]. Given n training samples $\{\mathbf{x}_i, y_i\}$,

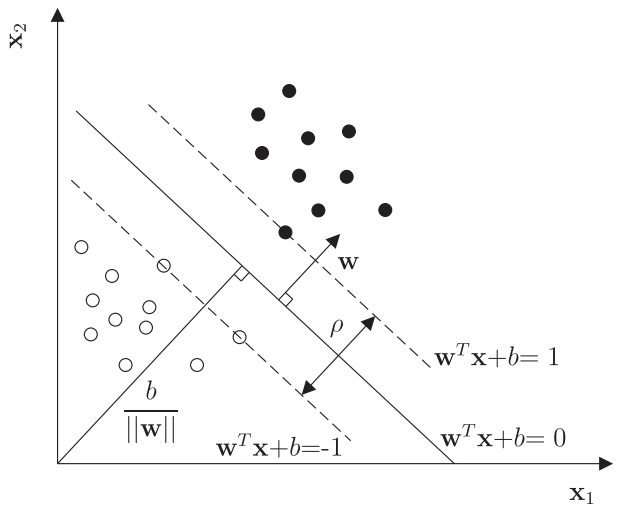

Figure 4.2: SVM hyperplane for linearly separable data

$i = 1, 2...n$, where each sample has Q features, ($\mathbf{x}_i \in R^Q$) and a class label with one of the binary values ($y_i \in \{-1, 1\}$), a hyperplane is a $n-1$ dimensional space described by

$$\mathbf{w}^T\mathbf{x} + b = 0, \tag{4.1}$$

where, $\mathbf{w}$ is a vector orthogonal to the hyperplane and b is a constant. Such a hyperplane $(\mathbf{w}, b)$ that separates data defines the function

$$f(\mathbf{x}) = \text{sgn}(\mathbf{w}^T\mathbf{x} + b), \tag{4.2}$$

for correctly classifying the data [182]. The hyperplane is chosen such that the bounding planes are a functional distance of at least 1 from the hyperplane, giving rise to the constraints

$$\mathbf{w}^T\mathbf{x} + b \geq 1 \text{ when } y_i = +1, \tag{4.3}$$

$$\mathbf{w}^T\mathbf{x} + b \leq -1 \text{ when } y_i = -1, \tag{4.4}$$

$$\text{or in compact form: } y_i(\mathbf{w}^T\mathbf{x} + b) \geq 1 \quad \forall i. \tag{4.5}$$

The problem of SVM involves maximizing the margin (distance between the two bounding planes) separating the two classes of data. The margin between linearly separable data is given by

$$\rho = \frac{2}{\parallel \mathbf{w} \parallel}, \tag{4.6}$$

whose derivation is given in Appendix A. Maximizing the margin involves minimizing $\parallel \mathbf{w} \parallel$ [181],

$$\min_{\mathbf{w},b} \quad f(\mathbf{w}) = \frac{\parallel \mathbf{w} \parallel^2}{2}, \tag{4.7}$$

subject to the distance constraint in Equation 4.5. Since the problem involves optimization subject to inequality constraints, Lagrange Multipliers are introduced so that the

problem becomes [181],

$$f(\mathbf{w}, b, \alpha) = \frac{1}{2} \parallel \mathbf{w} \parallel^2 - \sum_{l=i}^{n} \alpha_i \Big[y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1 \Big], \tag{4.8}$$

where α_i is a Lagrangian multiplier, one for each data point. The solution to this quadratic programming problem (QP) is obtained by maximizing f with respect to $\alpha \geq 0$ and minimizing it with respect to $(\mathbf{w}, b)$ [183]. The optimal solution can be obtained by differentiating Equation 4.8 with respect to $\mathbf{w}, b$ and setting the derivatives to 0. This leads to

$$\frac{\partial f(\mathbf{w}, b, \alpha)}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = 0, \tag{4.9}$$

$$\frac{\partial f(\mathbf{w}, b, \alpha)}{\partial b} = -\sum_{i=1}^{n} \alpha_i y_i = 0, \tag{4.10}$$

which leads to the optimal solution

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i, \tag{4.11}$$

and

$$b = y_i - \mathbf{w}^T \mathbf{x}_i. \tag{4.12}$$

Substituting Equations 4.11 and 4.12 into Equation 4.8 leads to the dual problem

$$F(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \mathbf{x}_j, \tag{4.13}$$

subject to:

$$\sum \alpha_i y_i = 0, \tag{4.14}$$

$$\alpha_i \geq 0 \quad \forall i. \tag{4.15}$$

The optimization problem involves computing the inner products $\mathbf{x}_i^T \mathbf{x}_j$ between all training points and finding α_i, $i = 1, 2...n$ such that $f(\alpha)$ is maximized. Each $\alpha_i > 0$ indicates that the corresponding $\mathbf{x}_i$ is a support vector and the decision function will have the form

$$f(\mathbf{x}) = \operatorname{sgn} \left(\sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b \right). \tag{4.16}$$

As seen in Equation 4.16, the decision function relies on an inner product between the test point $\mathbf{x}$ and the support vectors $\mathbf{x}_i$ [182].

4.1.1.1 Non-Linearly Separable Data

For data that is not linearly separable, a kernel function $z = \phi(\mathbf{x})$ can be used to map the features into a higher dimensional feature space and constructing a hyperplane in this space [183]. The kernel function is selected in such a way that the new training data

$\{\phi(\mathbf{x}_i), y_i\}$ is separable by a hyperplane as shown in Figure 4.3 [182]. To set up the new

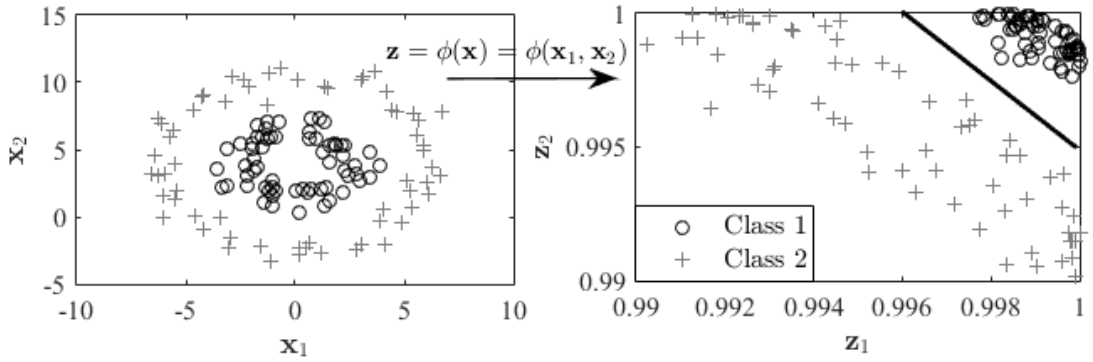

Figure 4.3: Transforming input data to a new feature space where it is linearly separable

optimization problem, all the occurrences of $\mathbf{x}$ are replaced by $\phi(\mathbf{x})$ so that the inner product becomes

$$K(\mathbf{x}_i\mathbf{x}_j) = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j), \tag{4.17}$$

and the QP problem becomes

$$f(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j), \tag{4.18}$$

subject to:

$$\sum \alpha_i y_i = 0, \tag{4.19}$$

$$\alpha_i \geq 0 \quad \forall i. \tag{4.20}$$

The decision function becomes

$$f(\mathbf{x}) = \text{sgn}\left(\sum_{i=1}^{n}\alpha_i y_i \phi(\mathbf{x}_i)^T\phi(\mathbf{x}) + b\right). \tag{4.21}$$

The kernel function allows the construction of a hyperplane in the higher dimensional feature space without explicitly performing calculations in this feature space [183]. Typical kernel functions used include:

- Linear, $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T\mathbf{x}_j$.
- Polynomial of power p, $K(\mathbf{x}_i, \mathbf{x}_j) = (1 + \mathbf{x}_i^T\mathbf{x}_j)^p$.
- Radial basis function (RBF), $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \parallel \mathbf{x}_i - \mathbf{x}_j \parallel^2)$
- Multi-layer perceptron, $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma_0\mathbf{x}_i^T\mathbf{x}_j + \gamma_1)$,

where γ is a tunable parameter. Figure 4.4 shows the effect of the kernel parameter of a radial basis function in transforming the input features to a higher dimensional space where the data is linearly separable.

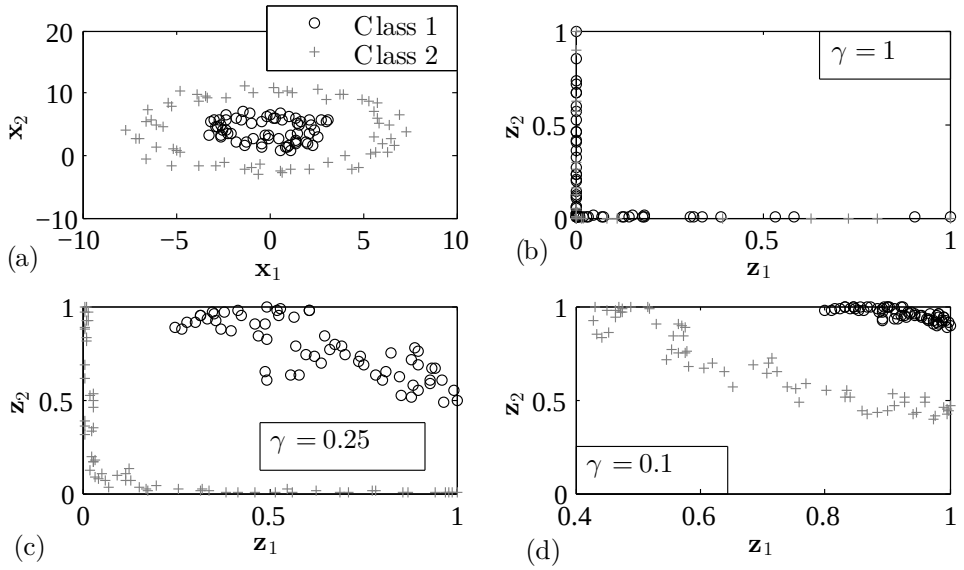

Figure 4.4: Effect of RBF kernel parameter on separation margin of data that is not linearly separable

4.1.1.2 Soft Margin SVM

In most applications, the training data set contains noise and may not be linearly separable. In such a case, slack variables ζ_i may be introduced to allow misclassification of difficult or noisy examples as shown in Figure 4.5 [183].

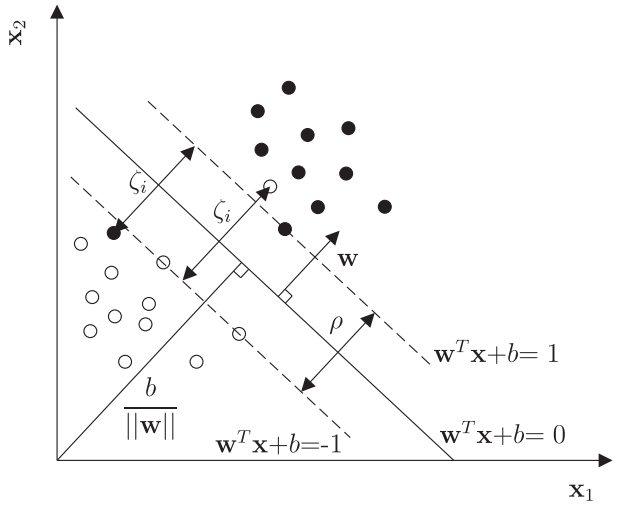

Figure 4.5: Soft margin classification

The new formulation of the optimization problem incorporating slack variables becomes

$$\min f(\mathbf{w}) = \quad f(\mathbf{w}) = \frac{\| \mathbf{w} \|^2}{2} + C \sum_{i=1}^{n} \zeta_i, \tag{4.22}$$

subject to

$$y_i(\mathbf{w}^T\mathbf{x} + b) \geq 1 - \zeta_i \text{ and } \zeta_i \geq 0 \quad \forall i. \tag{4.23}$$

C is referred to as a regularization parameter that controls trade-offs between maximizing the margin and minimizing misclassification and can be tuned during training to provide better regularization. A very large value of C may lead to over-fitting while a small value may lead to high misclassification rate during training and consequently during testing [183].

4.1.1.3 Multi-Class SVM

Multi-class classification is achieved by constructing and combining several binary classifiers. Pairwise coupling method, in which N_{svm} SVM classifiers are constructed, is commonly used for multi-class classification of data with N_c classes, where,

$$N_{svm} = \frac{N_c(N_c - 1)}{2}. \tag{4.24}$$

A binary classifier is built for each pair as shown in Table 4.1.

Table 4.1: Pairwise method for multi-class classification using binary classifiers

(1,2)	(1,3)	(1,4)	..	(1,N_c)
	(2,3)	(2,4)	..	(2,N_c)
		(3,4)	..	(3,N_c)
				($N_c - 1, N_c$)

Class probability estimates are computed through voting rule [184].

4.1.1.4 Optimal Tuning of SVM Parameters

For each SVM application there are at least two parameters that require tuning, the cost parameter C and the kernel function parameter γ. In this work, the optimal parameters of SVM were obtained through combined use of differential evolution (DE) and particle swarm optimization (PSO) methods. DE is used to find the search limit of the parameters while PSO is used to obtain the optimum parameters within the search limit. Details of DE and PSO are given in Appendix B. During the optimization process, 3-fold cross validation is employed, where the training data is partitioned into three parts and for each iteration, two parts are used for training and the other part is used for testing and validation. The objective function of the optimization process is to reduce the cross

validation error e_{cv} between the target y and predicted output y_p, computed as follows

$$e_{ci} = \begin{cases} 0 \text{ if } y_{pi} = y_i, \\ 1 \text{ if } y_{pi} \neq y_i, \end{cases} \tag{4.25}$$

$$e_{cv} = \frac{1}{n_{cv}} \sum_{i=1}^{n_{cv}} e_{ci}, \tag{4.26}$$

where e_{ci} is the classification error for sample i and n_{cv} is the number of samples in the testing and validation partition. Figure 4.6 shows the workflow of parameter tuning for multi-class SVM. Once the optimum parameters have been found, the parameters are used together with the whole training data to train the SVM which results in a classification model that can be used with test data or for online health state classification as described in Figure 4.1. The termination criteria for optimization can be set as desired CV error (ideal is 0) or maximum number of iterations.

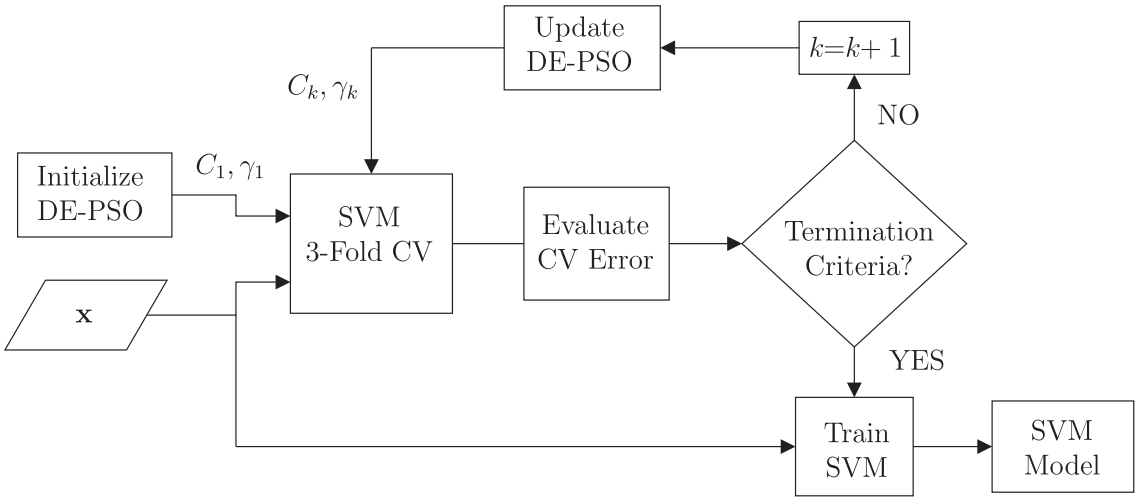

Figure 4.6: SVM parameter tuning through DE and PSO

4.1.2 Random Forests

Random forests consist of an ensemble of randomized decision trees used for classification and regression applications [185]. A random forest consists of a set of trees M_k where each tree consists of split nodes and leaves as illustrated in Figure 4.7.

A tree is grown using training data by recursive splitting at the split nodes where each feature is evaluated and, depending on the value of the feature, it is passed to the left or right child [185]. Each leaf stores the statistics of the feature passed on during training. The terminal nodes are the decision nodes and each terminal node is dominated by one of the classes. New observations are classified by passing $\mathbf{x}$ down to the terminal node of the tree and a class is assigned by majority vote.

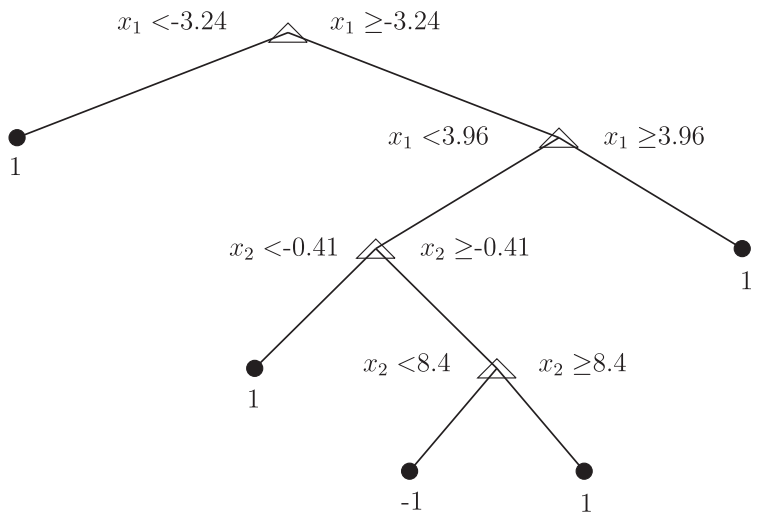

Figure 4.7: A sample decision tree for classification

The measures for selecting the best split are often based on the amount of information at the child node. Examples of such measures include [186]:

$$\text{Entropy}(k) = -\sum_{c=1}^{N_c} p(c|k) \log_2 p(c|k), \tag{4.27}$$

$$\text{Gini}(k) = 1 - \sum_{c=1}^{N_c} \Big[p(c|k)\Big]^2, \tag{4.28}$$

$$\text{Classification error}(k) = 1 - \max_c [p(c|k)], \tag{4.29}$$

where $p(c|k)$ is the probability of class c at node k.

Given an original feature vector $\mathbf{x} \in R^Q$ with Q features $\mathbf{x}_1 ... \mathbf{x}_n$, the random forest with M_k trees is constructed as follows [187]:

1. Each tree is constructed from a random selection of $m = \sqrt{Q}$ features $\{\mathbf{x}_{ij}\}_{j=1}^{m}$.
2. For each select the best split based on the maximum amount of information at the child node. The amount of information at the child can be obtained by computing the entropy at the nodes.
3. Predict data by aggregating the predictions of M_k decision trees, majority votes for classification and average for regression.
4. Probability estimates for classification can be computed from the number of votes for each class from M_k trees.

4.1.3 Identifying the Number of Health States of Unlabeled Run-to-Failure Data

Unsupervised machine learning algorithms such as k-means and self-organizing maps (SOM) neural networks for clustering unlabeled data require prior knowledge of the number of clusters. For run-to-failure data that is not labeled, the optimum number of

health states within the data can be determined through k-means clustering algorithm using the extracted features as input in a recursive process as follows:

- The number of health states N_c is varied from 2 to N_{cm} and the similarity between the clusters for each set number of health states is determined.
- For each set number of health states, k-means algorithm is used to cluster data into N_c health states. The centroid ct_{ij} of each feature at each health state is computed, where $i = 1, 2.., N_c$ and $j = 1, 2.., Q$.
- For each feature, the centroids are normalized between [0,1] and the mean for each health state is obtained as follows

$$ct_i = \frac{1}{Q} \sum_{j=1}^{Q} ct_{ij}. \tag{4.30}$$

- The pairwise difference between the resulting centroids is then obtained as follows

$$\Delta ct_{jk} = ct_j - ct_k, \quad j = 1, 2.., N_c - 1, \text{ and } k = j + 1. \tag{4.31}$$

  This produces an $(N_c - 1) \times N_c$ upper matrix.

- For each column k of the upper matrix, all the elements with $|\Delta ct_j| \leq th$ are located, where th is a predetermined threshold. These elements indicate a similarity between clusters within a set number of health states. The number of these elements with non-similar or unique indices N_u is obtained as follows

$$sc_k = \begin{cases} 0 \text{ if } j_k = j_{k+1}, \\ 1 \text{ if } j_k \neq j_{k+1}, \end{cases} \tag{4.32}$$

$$N_u = \sum_{k=1}^{N_c - 1} sc_k, \tag{4.33}$$

  where sc is a value for unique indices. The threshold can be set by taking the absolute value of the difference between standard deviation and the mean of ct.

- The identified number of health states is then obtained by subtracting the number of similar health states from the initial number of health states

$$N_{c,opt} = N_c - N_u. \tag{4.34}$$

- The most frequently identified number of health states in N_{cm} is selected as the optimum number for a given data set.

To demonstrate the feasibility of this method of determining the optimum number of health states in unlabeled data, rolling element data with three known health states (healthy and outer ring (OR) fault at two severity levels), described in section 3.4.1.1 were used. The extracted features from vibration signals were used as input. Figure 4.8(a) shows the clustering of the data within the feature space while Figure 4.8(b) shows the histogram of the number of health states. The count is the number of times that the number of health states N_c is identified from N_{cm}. The method correctly identifies 3 health states in this data.

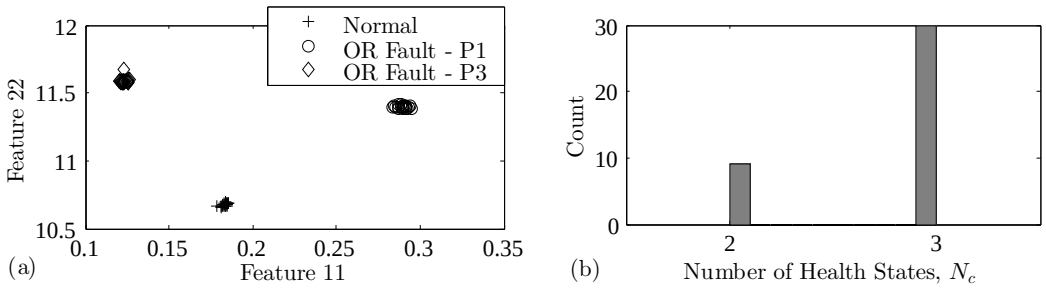

Figure 4.8: Identification of optimum number of health states for bearing data with 3 health states (a) Clustering of data within feature space and (b) distribution of identified health states

4.1.4 Remaining Useful Life Estimation from Health State Probabilities

The output of the machine learning algorithms is the health state and the health state probability P_i, which is the probability that a data point belongs to health state i. This section describes a method for estimating the remaining useful lifetime (RUL) from the health state probabilities. The nomenclature of computing the RUL from the health state probabilities is shown in Figure 4.9. RUL_i is the remaining useful lifetime at the transition point between health state $i-1$ and i, t_c is the current time at which the RUL is being computed, t_{HS_i} is the duration spent at the current health state. As the system degrades, the probability of the current health state decreases while the probability of the next health state increases.

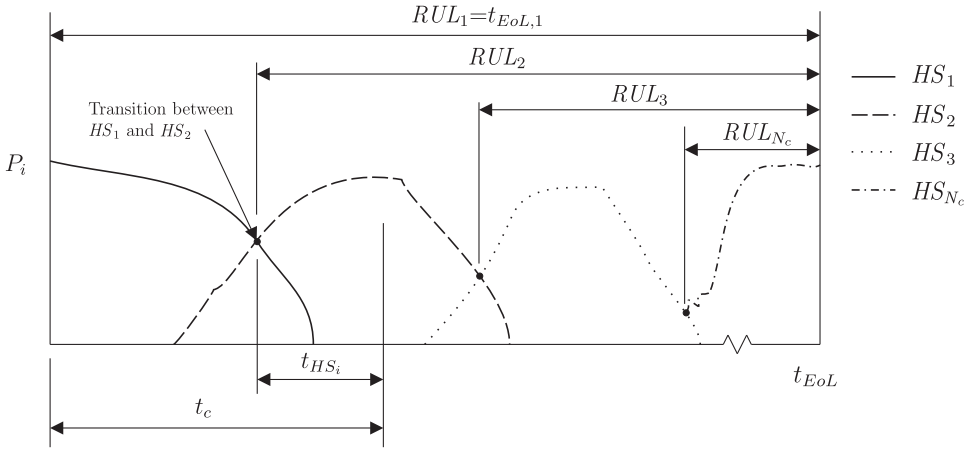

Figure 4.9: Nomenclature for computing RUL

A fleet of units operating under similar conditions can be assumed to undergo through similar health states until failure occurs. Therefore, it can be assumed that the units spend the same percentage of lifetime in each health state. Based on these assumptions,

the RUL of a test data set is computed based on the historical percentage RUL of the training data sets at each health state as follows

$$RUL = \begin{cases} t_{EoL,1} \sum_{i=1}^{N_c} P_i L_i - t_{HS_i}, & \text{if } HS_i = 1, \\ \bar{t}_{EoL} \sum_{i=1}^{N_c} P_i L_i - t_{HS_i}, & \text{if } HS_i > 1, \end{cases} \tag{4.35}$$

where $L_i = \frac{RUL_i}{t_{EoL,1}}$ is the percentage RUL at health state i, $t_{EoL,1}$ is the time at end of life of the training data, HS_i is the current health state and $\bar{t}_{EoL}$ is the mean predicted time to end of life computed as follows

$$\bar{t}_{EoL} = \frac{1}{HS_i - 1} \sum_{i=2}^{HS_i} \frac{t_c}{1 - L_i}. \tag{4.36}$$

4.1.5 Application Examples

The feasibility of the health state estimation approach to prognosis is demonstrated on two types of data sets:

1. ball bearing data
2. tool wear of a high speed milling machine

4.1.5.1 Health state estimation of ball bearings

Run to failure data of ball bearings obtained from the 2012 IEEE PHM challenge was used [148]. The data consists of two vibration signals sampled at 25.5 kHz for 0.1 seconds at intervals of 10 seconds, obtained from a test rig suited for highly accelerated degradation tests for rolling element bearings [148]. In addition, temperature measurements for some tests were provided. The temperature was sampled at 10 Hz for 60 seconds each minute. The operating conditions of the bearings are shown in Table 4.2.

Table 4.2: Operating conditions of bearing (Designation: 61804-2RS1)

Variable	**Rated Value**	**Utilized value**
Speed	13000 RPM	1800 RPM
Radial Load	3.1 kN	4 kN

31 features described in section 3.3 were extracted from each vibration sensor giving rise to 62 features. The optimum number of health states was determined using the method for identifying optimum number of health states, described in section 4.1.3, with $Q = 30$ and $N_{cm} = 40$. Figure 4.10 shows the distribution of the number of clusters identified from training data sets. 5 health states were found to have the highest count (number of times N_c is identified within N_{cm}) and hence selected as the optimum number of health states. k-means algorithm was then used to cluster the features into 5 health states.

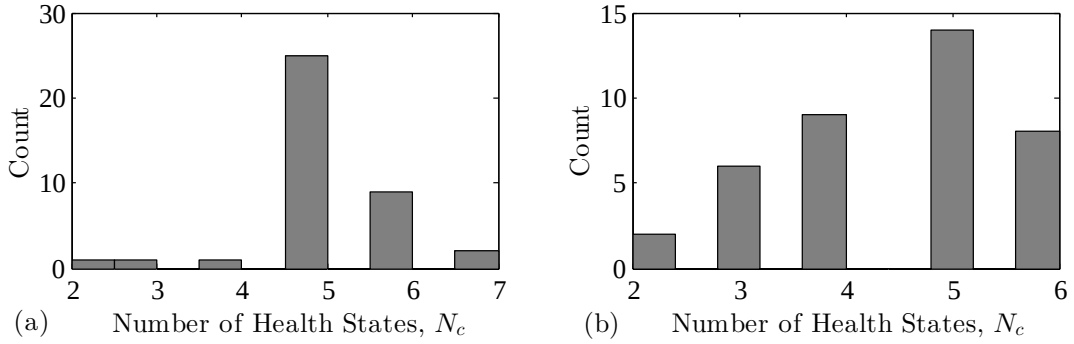

Figure 4.10: Histogram of number of health states for training data sets (a) bearing 1_1 and (b) bearing 1_2

With the health states identified from the training data, 23 features were selected using the maximum separation distance method described in section 3.4.1. Figure 4.11 shows the performance measure for each feature which represents the ability to distinguish different health states. With a selection threshold of $\bar{d}_j \geq 0.6$, 23 features from both sensors were selected.

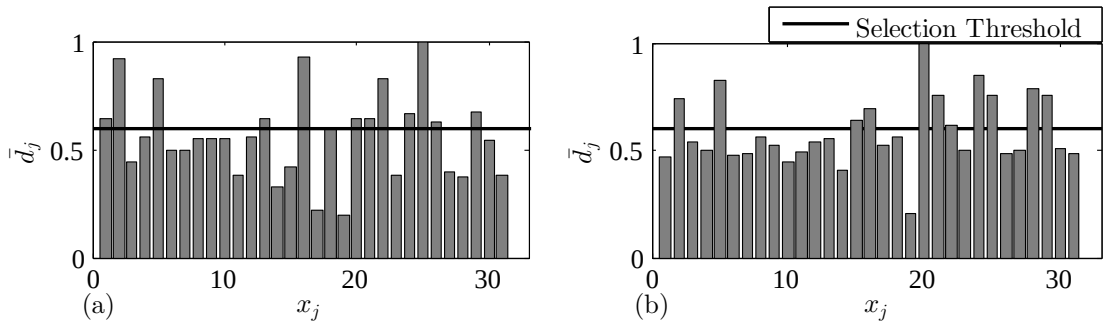

Figure 4.11: Performance of each feature on separability of health states (a) sensor 1 and (b) sensor 2

The selected features are used as inputs to the machine learning algorithms while the health state label is used as the target for training. The output from the trained model is the health state probability from which the remaining useful lifetime (RUL) can be estimated using the method described in section 4.1.4. Figure 4.12(a) and (c) show the health state probability of training data set 1_1 from support vector machines (SVM) and random forests (RF) with a number of trees $M_k = 100$ respectively, while (b) and (d) are comparisons of predicted RUL and actual RUL at specified time intervals ($t_c = (1, 2, ...19) \times 0.05 t_{EOL}$).

From Figure 4.12(b) and (d), 14 and 12 predictions out of 19 respectively fall within the acceptable error region of $\pm 10\%$. This indicates that the proposed method would be fairly accurate for similar systems. The trained models were then used with the test data of similar bearings in order to evaluate the performance and regularization of the method

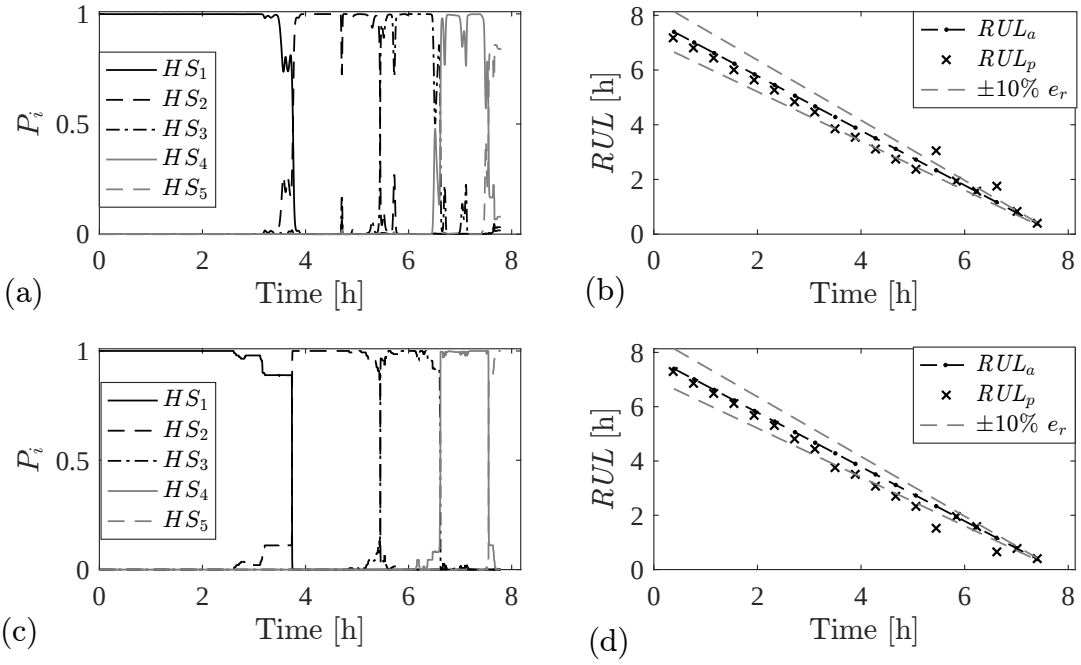

Figure 4.12: Health state probabilities of training bearing 1_1 obtained from (a) SVM and (c) RF and the corresponding predicted *RUL* at specified prediction times (b) and (d)

for unknown data, similar to online prognosis. The selected features of the test data are used as input to the trained models which gives the health state probability as the output. *RUL* is estimated using equation 4.35. Figure 4.13 shows the health state probabilities of test bearing 1_3 and 1_4 at different prediction times, estimated using SVM with optimally tuned parameters, while Figure 4.14 are the health state probabilities of the same bearings estimated using RF. As the bearing degrades, the probability of the next health state increases while the probability of the current health state decreases.

From Figures 4.13(b),(d) and Figures 4.14(b),(d) the predicted *RUL* improves as the bearing changes to health state 2 and onwards since the estimated $\bar{t}_{EOL}$ is updated as more information is made available. Both the SVM and RF methods were evaluated with all the training and test bearings and Table 4.3 summarizes the performance of the SVM-based health state estimation method while Table 4.4 is the performance of the RF-based health estimation method. *RUL* is estimated at 19 prediction intervals at increments of $0.05t_{EOL}$. In both cases, majority of the predictions fall within the acceptable error region of $\pm 10\%$. *FP* refers to the total number of false positives, while *FN* is the total number of false negatives, which are predictions outside the acceptable region (as defined in section 2.3). Considering all the test cases, SVM method has a mean MAPE of 11.89 with 78 out of 133 predictions within the acceptable region. Random forests has a MAPE of 12.68 with 75/133 acceptable predictions. This is a good performance considering the low number of training data sets available. The *FP* and *FN* can be reduced by increasing the number of training data sets to cover the high variation in lifetime of the bearings.

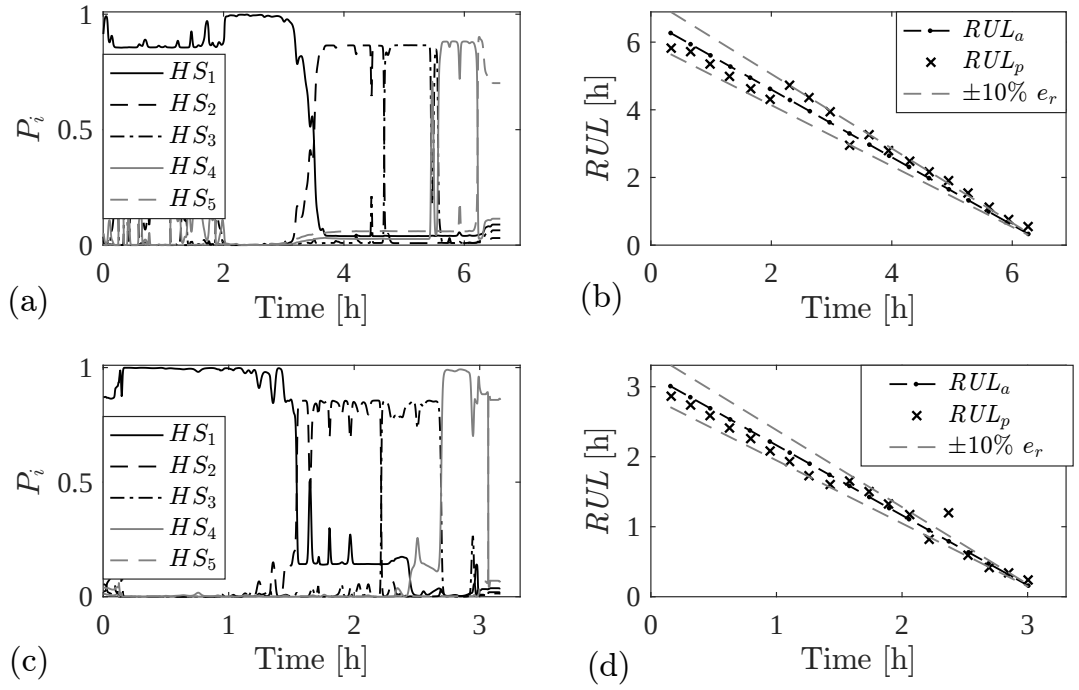

Figure 4.13: Health state probabilities and corresponding RUL predictions at specified prediction times for test bearing 1_3 - (a) and (b), and test bearing 1_4 - (c) and (d), estimated using SVM with optimally tuned parameters

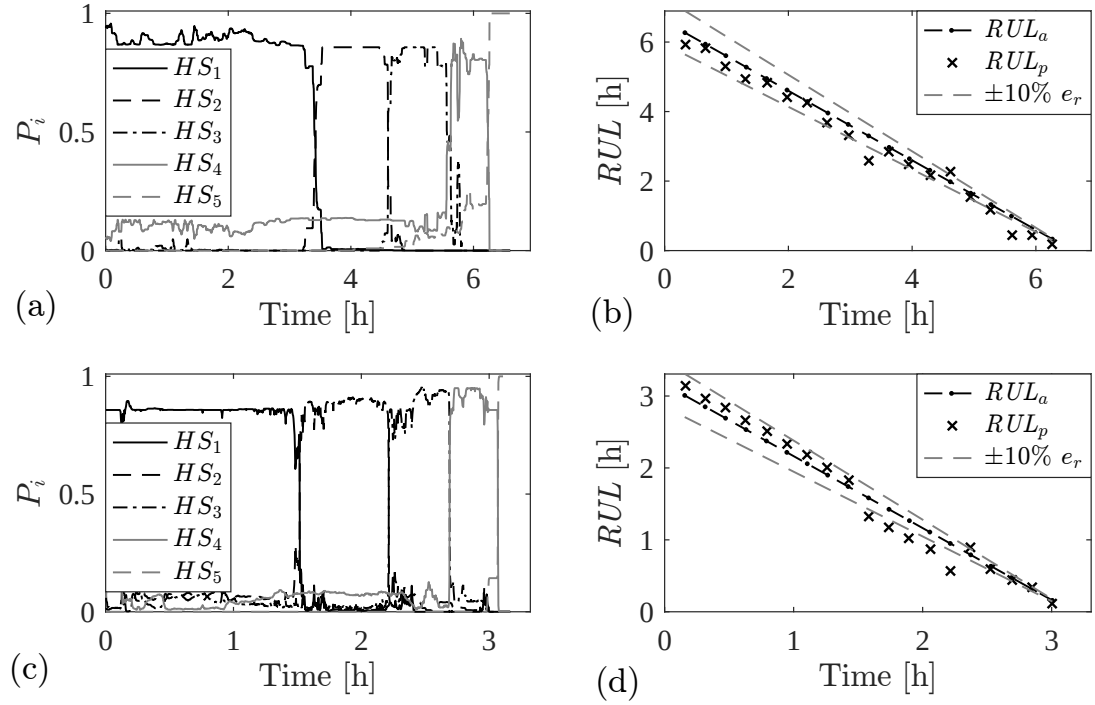

Figure 4.14: Health state probabilities and corresponding RUL predictions at specified prediction times for test bearing 1_3 - (a) and (b), and test bearing 1_4 - (c) and (d), estimated using RF

Table 4.3: Performance evaluation of the SVM-based health estimation method based on prognostic metrics

	Early Predictions		Late Predictions		
Test	FP	Mean e_r [%]	FN	Mean e_r [%]	MAPE
1_1	3	11.71	2	-38.59	7.36
1_2	6	21.83	2	-39.06	14.82
1_3	1	20.81	3	-27.73	8.81
1_4	2	13.08	2	-49.24	11.08
1_5	6	15.74	5	-34.34	15.92
1_6	7	16.82	2	-44.89	14.18
1_7	4	18.84	4	-31.95	12.92

Table 4.4: Performance evaluation of the RF-based health estimation method based on prognostic metrics

	Early Predictions		Late Predictions		
Test	FP	Mean e_r [%]	FN	Mean e_r [%]	MAPE
1_1	7	20.58	0	0	8.89
1_2	6	19.32	7	-13.56	13.48
1_3	4	38.48	1	-15.04	12.87
1_4	6	24.40	2	-10.90	11.93
1_5	8	15.20	3	-25.90	13.87
1_6	5	15.36	2	-65.98	15.16
1_7	5	26.34	1	-27.99	12.54

4.1.6 Health state estimation of tool wear

In this application, condition monitoring data of a high speed milling machine described in section 3.4.2.2 was used. Three sets of data that contain the ground truth wear data are used for evaluation of the proposed methods. For each data set, 31 features per sensor were extracted from the raw data giving a total of 217 features. Failure threshold was set at a wear value of 170×10^{-3} mm to define the end of life of the tool. The method for identifying the optimum number of clusters in unlabeled data, described in section 4.1.3, was used to determine the optimum number of health states, with $Q = 60$ and $N_{cm} = 40$. 4 health states were identified as the optimum health states as shown in Figure 4.15(a). k-means algorithm is then used to cluster the training data into (k=4) health states. Figure 4.15 shows clustering of features within the four health states.

Maximum separation distance feature selection method for health state classification described in section 3.4.1 was used to select suitable features from each sensor, with a selection criterion of $\bar{d}_j \geq 0.8$. Figure 4.16 shows the performance of the features for each sensor. A total of 82 features were selected.

The three data sets were used for evaluation of the method in a three-fold cross evaluation manner where two data sets are used for training and the other data set for testing. The selected features are used as inputs to the machine learning algorithms while the health state label is used as the target.

Figure 4.17 shows the health states and corresponding RUL at 19 specified prediction

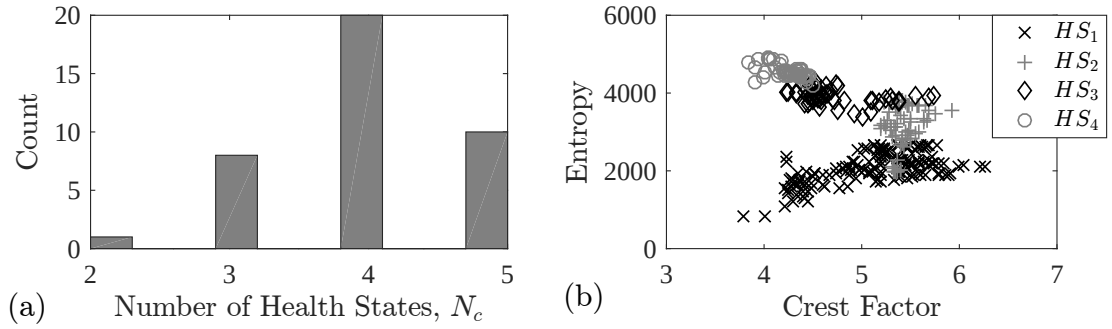

Figure 4.15: Health states for data set 1 (a) optimum number of health states and (b) clustering of features into four health states

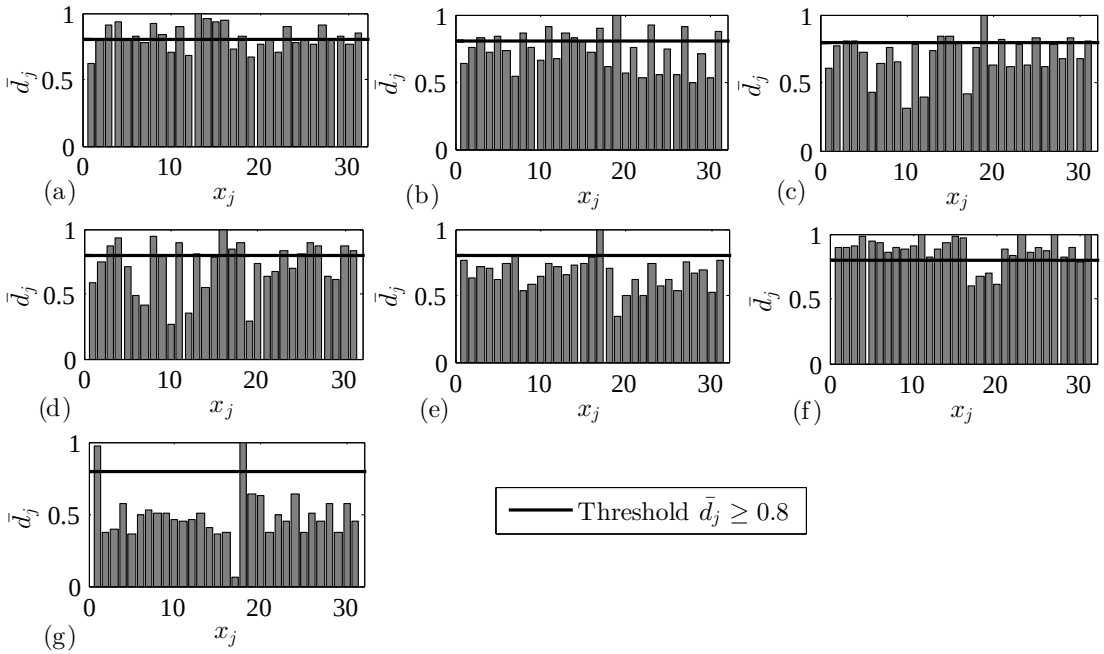

Figure 4.16: Performance evaluation of features extracted from data set 1 based on maximum separation distance between health states, (a)-(c) acceleration data, (d)-(f) force data and (g) acoustic emission data

times for each of the three data sets obtained from SVM while Figure 4.18 are the results from the RF. Both approaches estimate the *RUL* within an average MAPE of 20. However, RF based method has a better performance with a mean MAPE of 9.75 compared to SVM with a mean MAPE of 12.38. In addition, random forest produces fewer false positives and false negatives.

Table 4.5 summarizes the performance of the SVM-based health state estimation method while Table 4.6 is the performance of the RF-based health estimation method.

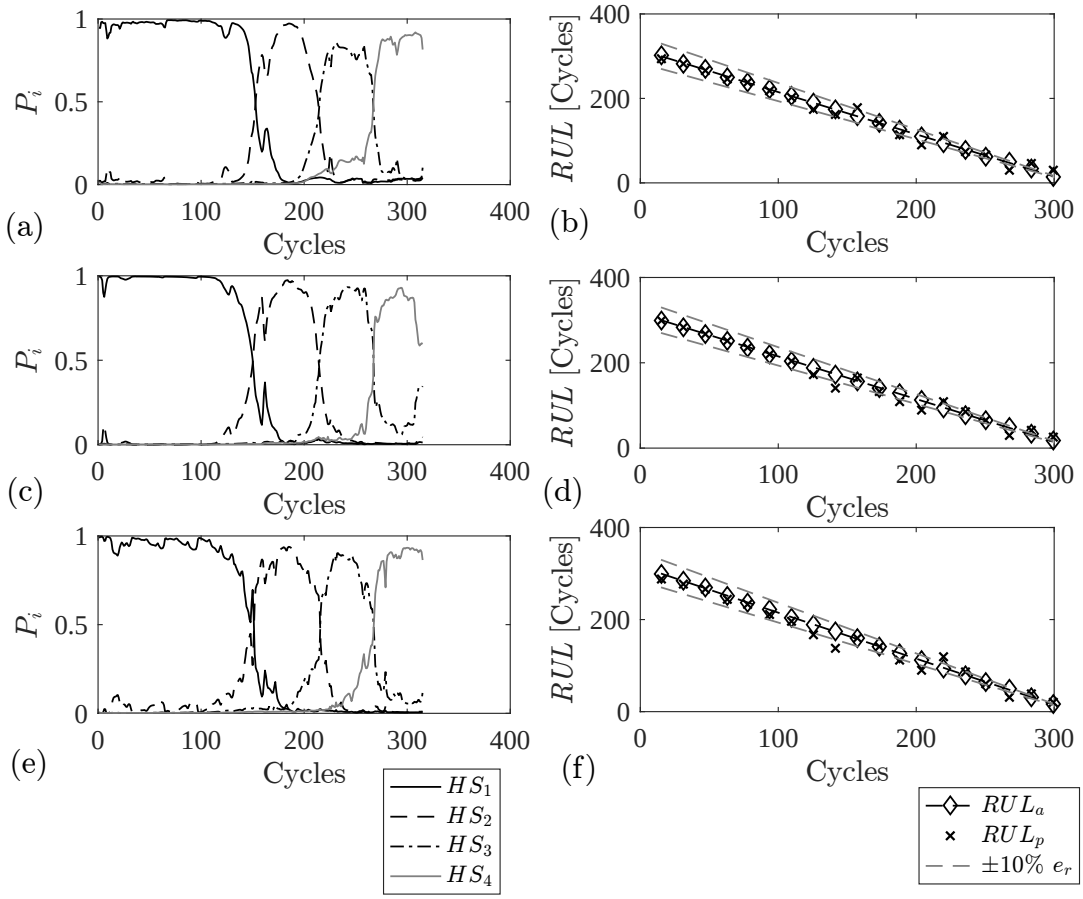

Figure 4.17: SVM-based Health state probabilities and RUL of the training data sets: data set 1 - (a) and (b), data set - 2 (c) and (d) and data set - 3 (e) and (f)

Table 4.5: Performance evaluation of the SVM-based health state estimation method based on prognostic metrics

	Early Predictions		Late Predictions		
Test	FP	Mean e_r [%]	FN	Mean e_r [%]	MAPE
1	3	23.17	4	-41.05	13.91
2	4	22.86	3	-32.29	11.36
3	5	20.63	3	-27.56	11.87

Table 4.6: Performance evaluation of the RF-based health state estimation method based on prognostic metrics

	Early Predictions		Late Predictions		
Test	FP	Mean e_r [%]	FN	Mean e_r [%]	MAPE
1	6	19.48	2	-10.77	9.79
2	4	19.76	1	-18.30	9.11
3	5	18.65	2	-28.35	10.33

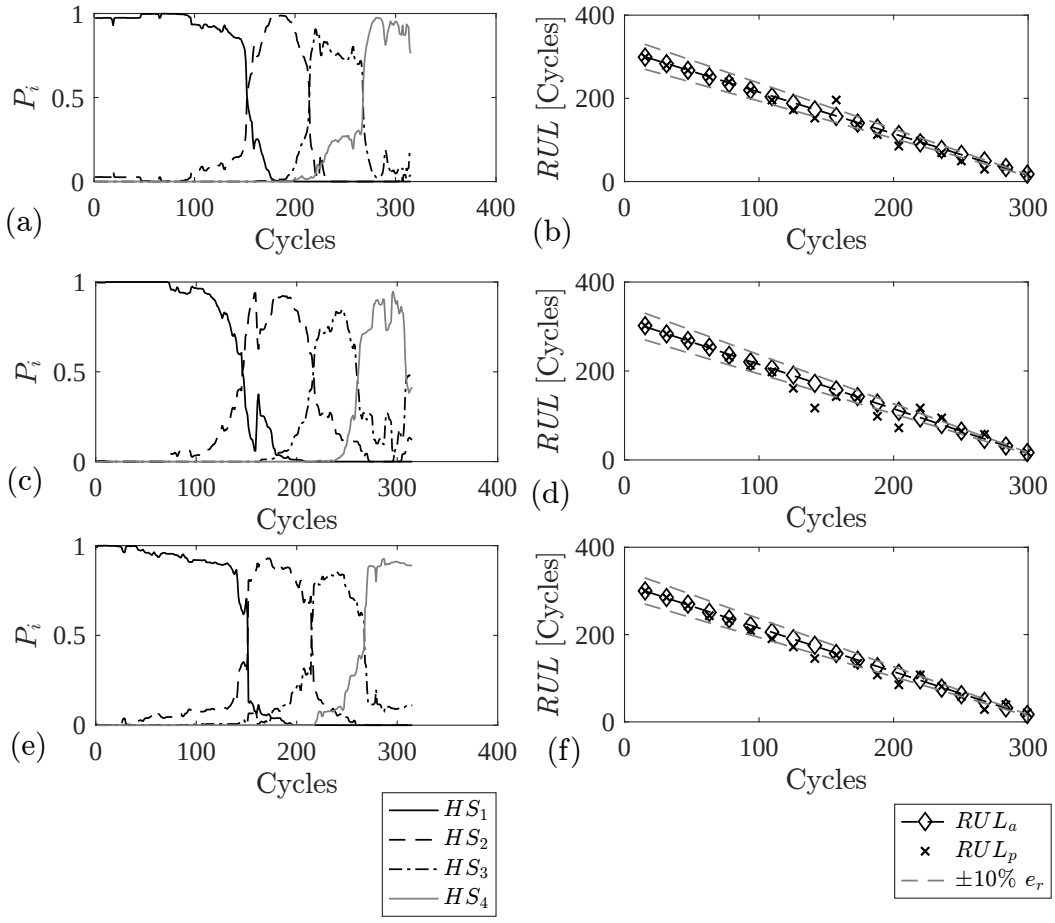

Figure 4.18: RF-based Health state probabilities and RUL of the training data sets: data set 1 - (a) and (b), data set - 2 (c) and (d) and data set - 3 (e) and (f)

4.2 Mapping Extracted Features to Remaining Useful Lifetime

This approach involves the use of machine learning (ML) algorithms to map features extracted from raw condition monitoring data to the remaining useful lifetime of a fleet of systems operated at similar operating conditions. The process involves function approximation and regression machine learning algorithms are therefore recommended. For satisfactory results, the systems/condition monitoring data should fulfill the following conditions:

- The systems should have narrow distribution of lifetimes.
- The operating conditions should have very small standard deviations.
- The extracted features should display monotonic change (increasing or decreasing).

The workflow of this method is shown in Figure 4.19. A health index derived from normalized *RUL* is used as the target. The training process results in a regression model that can be used with input features from similar systems for testing or online prognostics. In order to improve trendability of the extracted features, an autoregressive model is applied to each extracted feature. This serves as a smoothing technique as well as enhancing the degradation trend of the feature.

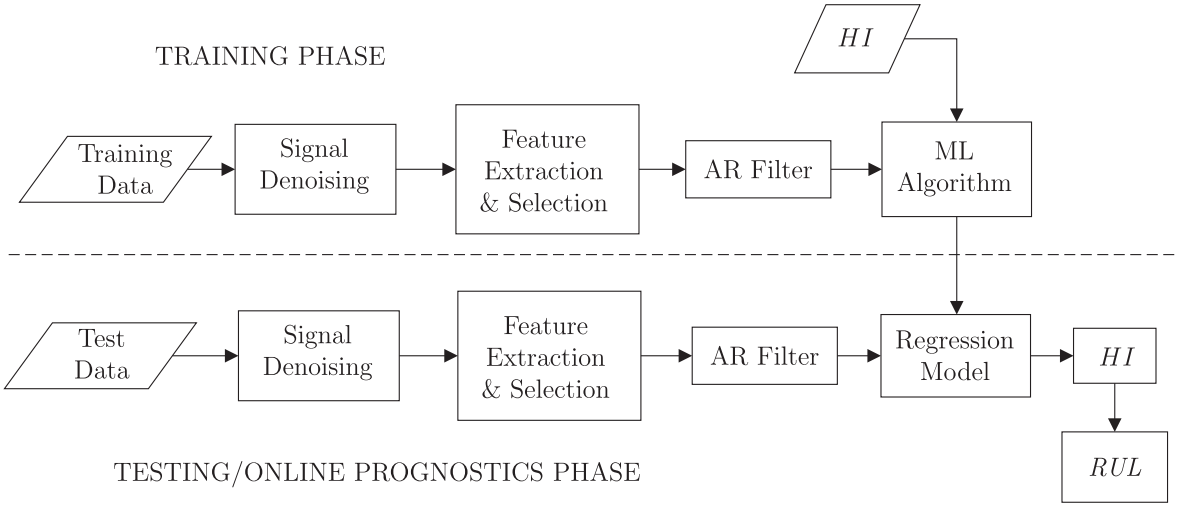

Figure 4.19: Workflow of the proposed method utilizing machine learning algorithms to map extracted features to the *RUL*

4.2.1 Autoregressive (AR) Model

AR model represents a time series in which the next value in the sequence is predicted based on a certain number of previous values. The AR model parameters may contain

important information regarding the condition of a component [188]. The following model is established to each of the extracted features x to obtain degradation trend:

$$x_i = \sum_{k=1}^{m} a_k x_{i-k} + e_i, \qquad i = 1, 2...n \tag{4.37}$$

where a_k are the model parameters, m is the model order, e_i is the residual of the model and n is the number of data points in x. In this work, the model parameters were determined using the Yule-Walker method [189]. The performance of the AR model depends on the choice of the model order m. In this study, the modified Akaike information criteria, AIC was employed to determine the optimum m[190]:

$$AIC(m) = log(\hat{\sigma}_m) + \frac{2m}{n}, \tag{4.38}$$

where,

$$\hat{\sigma}_m = \frac{1}{(n-m)} \sum_{i=m+1}^{n} (x_i - \sum_{k=1}^{m} a_k x_{i-k})^2, \tag{4.39}$$

The denominator $n - m$ in Equation 4.39 is used to reduce bias of low values of the model order. In this study, the model order was varied from 1 to 100 and the model order m yielding the minimum AIC was selected. Figure 4.20(a) shows the AIC for different values of m for an extracted feature in Figure 4.20(b). Model order $m = 16$ yields the lowest AIC and hence it is selected as the optimum model order for use in Equation 4.37. Figure 4.20(c) shows the resulting feature after application of AR. An increasing trend can be observed from the feature, indicating suitability of the method for processing features to be used for prognosis.

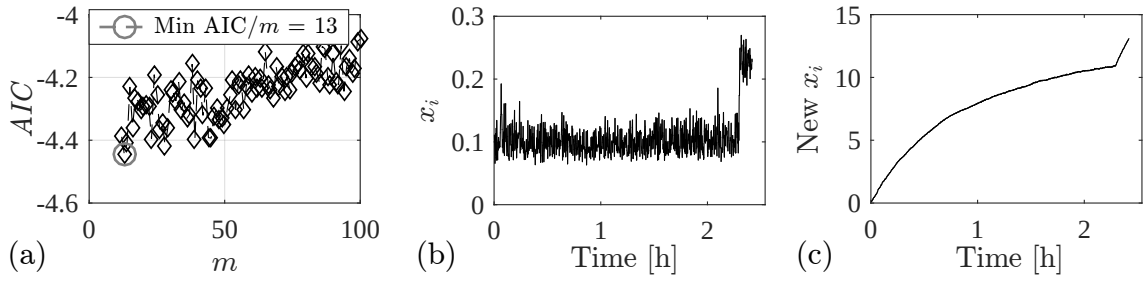

Figure 4.20: Estimating the AR model order m using Akaike information criteria (a) AIC for different model order (b) extracted feature before application of AR and (c) new feature after application of AR

The AR features are used as the input to machine learning algorithms. Normalized RUL of the training data set is used as the health index, HI or target. The normalized health index is used in order to account for the varying lifetimes of the different units. The RUL at the current time is then computed through Equation (4.40) which is derived from Figure 4.21. Given the current time, t_c, and the current health index, HI_c, the estimated

remaining useful lifetime RUL_p can be obtained by similar triangles as follows

$$RUL_p = t_{EoL} - t_c = \frac{t_c}{1 - HI_c} HI_c. \tag{4.40}$$

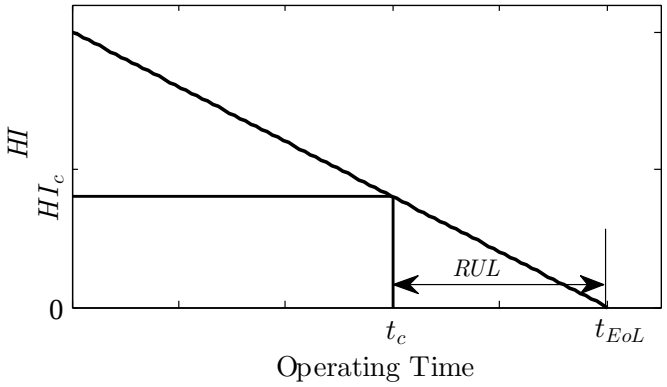

Figure 4.21: Estimating the RUL from the current time and normalized RUL

4.2.2 Application Examples

4.2.2.1 Ball Bearing Data

Features extracted from the raw data were selected using the method for feature selection for regression described in section 3.4.2, using a threshold of 0.65. 42 out of 62 features were selected and used as the input to the machine learning algorithms, while the normalized health index was used as the target. This resulted to a regression model consisting of weights, biases and parameters used in the mapping process. For testing, the selected features of the test data were used as input to the trained model and the current HI was obtained as the output. Equation 4.40 was then applied to obtain the RUL at the current time. Two data sets were used for training while all the data sets, including the training data sets were evaluated in the testing stage.

Figure 4.22 shows the estimated and actual RUL of bearing 1_3 at specified operating times estimated using extreme learning machines (ELM), random forests (RF) and support vector machines (SVM).

The results for three algorithms already described previously (ELM, RF and SVM) are presented in Tables 4.7, 4.8 and 4.9 respectively. ELM has a better performance overall with fewer FP and FN. All the algorithms produce predictions within a MAPE of 20.

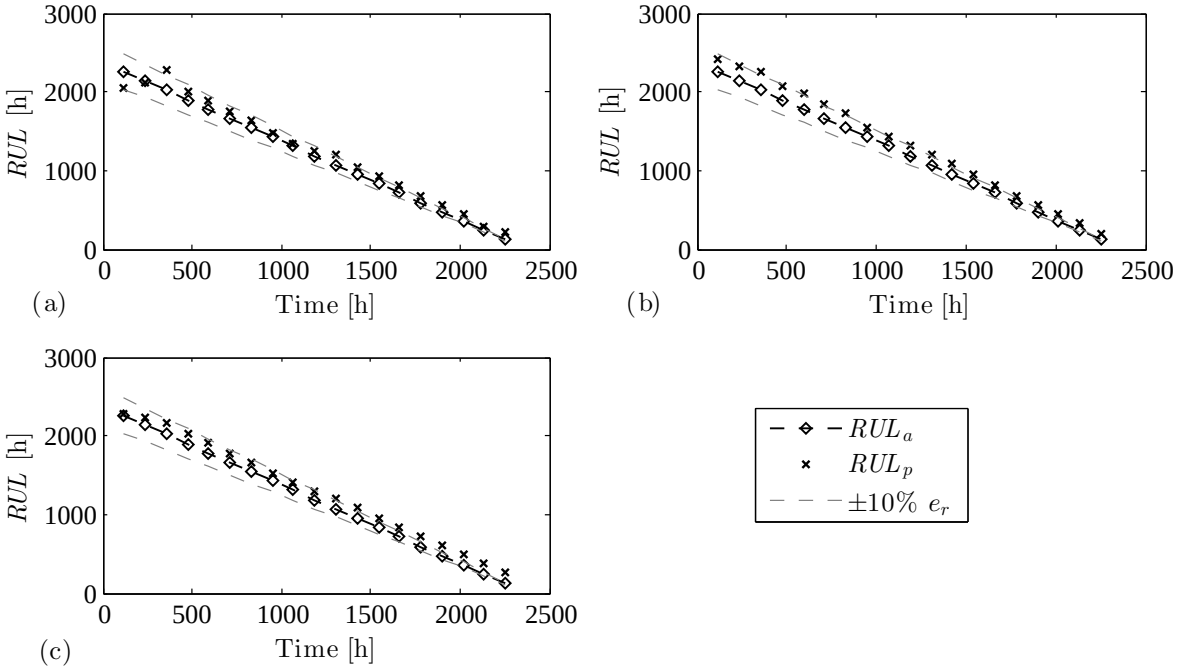

Figure 4.22: Estimated and actual *RUL* of bearing 1_3 at specified operating times: (a) ELM, (b) RF and (c) SVM

Table 4.7: Performance evaluation of the ELM-based *RUL* estimation approach

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1_1	9	14.47	0	0.00	11.09
1_2	7	14.97	3	-60.82	17.98
1_3	0	0.00	10	-21.86	13.76
1_4	3	20.02	4	-43.65	14.67
1_5	9	12.86	2	-26.85	12.31
1_6	17	25.02	0	0.00	21.51
1_7	5	21.60	3	-30.62	13.39

Table 4.8: Performance evaluation of the RF-based *RUL* estimation approach

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1_1	6	11.68	6	-11.73	9.37
1_2	0	0.00	10	-40.65	23.39
1_3	0	0.00	14	-20.05	16.99
1_4	0	0.00	6	-45.40	17.09
1_5	0	0.00	6	-21.65	9.74
1_6	2	17.47	3	-28.34	8.80
1_7	8	22.03	3	-31.82	16.33

Table 4.9: Performance evaluation of the SVM-based *RUL* estimation approach

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1_1	1	21.78	6	-23.96	13.04
1_2	0	0.00	6	-41.45	16.34
1_3	0	0.00	9	-36.25	20.68
1_4	3	23.61	9	-26.98	18.24
1_5	9	28.11	0	0.00	16.22
1_6	16	14.78	0	0.00	13.38
1_7	0	0.00	2	-22.30	5.95

4.2.2.2 Milling Machine Cutting Tool Data

Similarly, features extracted from the raw signals were selected using the method for feature selection for regression described in section 3.4.2. 54 features (from Table 3.8) were used as input to the machine learning algorithms and the normalized *HI* as the target. Training and testing was done in a 3-fold manner, with two data sets used for training and the other for testing. Figure 4.23 shows the estimated and actual *RUL* of test set 1 of milling machine cutting tool at specified operating times estimated using ELM, RF and SVM.

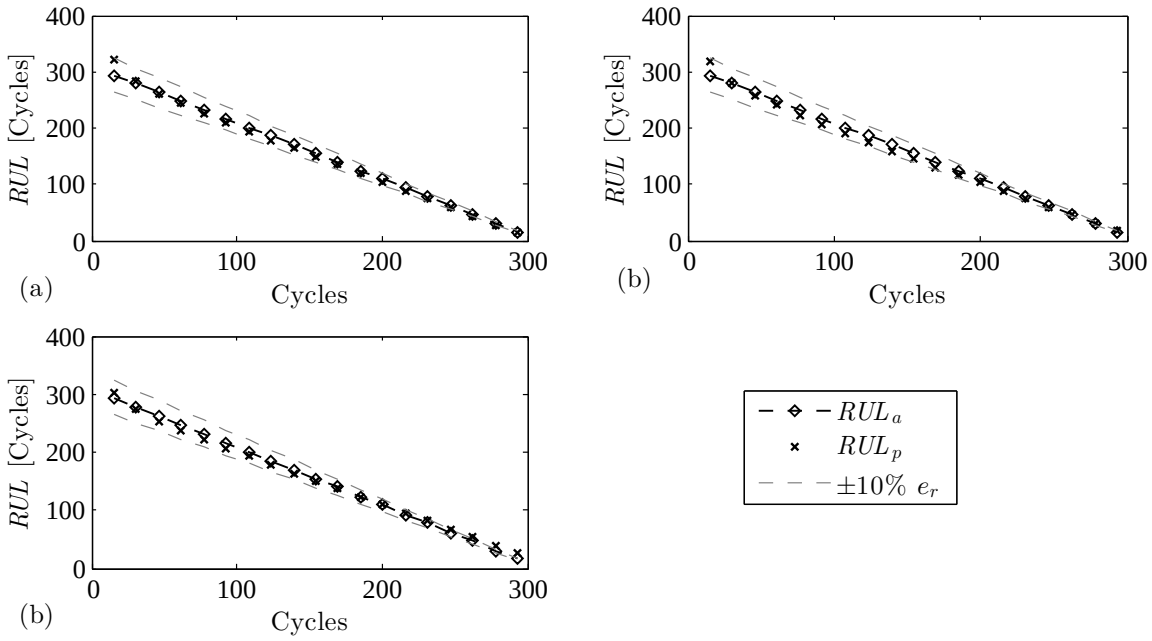

Figure 4.23: Estimated and actual *RUL* of test data set 1 of milling machine cutting tool at specified operating times: (a) ELM, (b) RF and (c) SVM

Tables 4.10 - 4.12 summarize the performance of the three algorithms in estimating *RUL*. All the algorithms yield results with a MAPE of less than 20. Given the limited number

of data sets available, the approach is quite promising on this kind of systems.

Table 4.10: Performance evaluation of the ELM-based *RUL* estimation approach for milling machine cutting tool

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	6	22.86	0	0	11.02
2	6	20.20	13	-30.34	11.62
3	1	20.47	9	-25.07	13.55

Table 4.11: Performance evaluation of the RF-based *RUL* estimation approach for milling machine cutting tool

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	10	12.78	2	-54.21	15.14
2	2	18.86	0	0.00	4.53
3	0	0	8	-36.91	15.31

Table 4.12: Performance evaluation of the SVM-based *RUL* estimation approach for milling machine cutting tool

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	14	16.67	2	-45.18	17.85
2	0	0.00	5	-39.47	11.54
3	0	0.00	4	-38.75	10.00

4.3 Mapping Extracted Features to a Health Index

Some technical systems have a physical health index which is measurable and may be used to represent the ground truth degradation of the system. A good example of such a system is tool wear monitoring in machine tools. Monitoring tool wear is important since it reduces maintenance costs by optimizing maintenance planning, increases the

availability of the machine tool, helps in product quality control since a worn out tool reduces the quality of a product and also increases safety. In such systems, the tool wear can be measured with a microscope, concurrently with condition monitoring data. Since it would be costly to set-up a tool wear system for each machine tool, a number of run-to-failure tests can be conducted and the data used for training machine learning algorithms to map extracted features from condition monitoring data to the ground truth wear. The trained model can then be used with condition monitoring data of similar systems to monitor wear. The workflow of this approach is similar to Figure 4.19 but the ground truth health index is used as the target. Machine learning algorithms are then trained to map the extracted features to the health index. In the testing stage, the selected features of the testing data sets are used as input to the trained model which produces the predicted/estimated health index at the current time. Once the current health index is estimated, the remaining useful lifetime can be estimated in two ways:

1. Using similarity based measure where the similarity between the predicted health index of the test data set and the health index of the training data sets is computed through MSE. Weights w are then generated from the mean square error (MSE) and used with the RUL^T of the training data at the predicted value of HI to obtain a weighted RUL of the test data set. Given N_T training data sets,

$$w_k = \left[\frac{1}{n}\sum_{i=1}^{n}\left(HI_{aki} - HI_{pki}\right)^2\right]^{-1}, \tag{4.41}$$

$$W = \sum_{k=1}^{N_T} w_k, \tag{4.42}$$

$$RUL_p = \frac{1}{W}\sum_{k=1}^{N_T} w_k RUL_k^T, \tag{4.43}$$

   where n is the number of data points in the test data.

2. Propagating the health index to a threshold using methods such as particle filter (section 4.4.1), extended/unscented kalman filters and multi-step ahead autoregressive methods.

4.3.1 Application Example

To evaluate the performance of this approach, three data sets for monitoring tool wear of high speed milling machine described in section 3.4.2.2 were used. For each case, two data sets were used for training and the other for testing so that $N_T = 2$. Features were selected using the method for feature selection for regression described in section 3.4.2 and used as input to machine learning algorithms. The health index HI was taken as the maximum tool wear as described in the application example in section 3.4.2.2 and used as the target for the machine learning algorithms. Figure 4.24 shows the predicted health index at the current time and the RUL^T of the training data at the current health index, used in estimating the RUL_p of the test data through weighted approach.

Figure 4.25 shows a comparison of predicted remaining useful lifetime RUL and actual remaining useful lifetime HI of data set 2 as the test data set using extreme learning machines (ELM). Majority of the predictions lie within the 10% error bound.

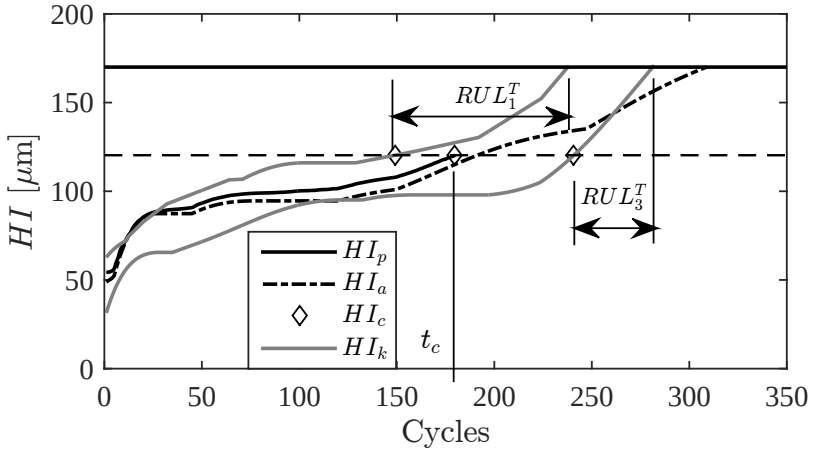

Figure 4.24: ELM-based predicted health index of data set 2 and RUL^T of training data sets (1 and 3) at the current health index, HI_c. HI_a is the actual health index of the test data

.

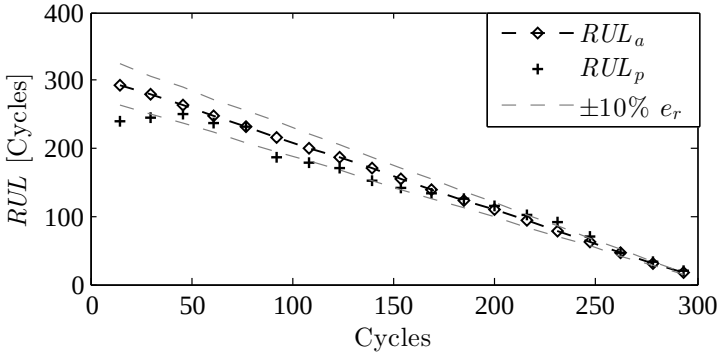

Figure 4.25: ELM-based estimated RUL at specified operating times for data set 2

Tables 4.13-4.15 show the performance of the proposed approach using different machine learning algorithms.

Table 4.13: Performance evaluation of the ELM-based health index prediction

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	9	17.59	2	-29.64	13.23
2	6	24.65	3	-17.67	11.91
3	10	28.34	0	0.00	15.46

This approach yields the best results with random forests, producing a mean MAPE of 12.22 for the three data sets. However, the MAPE is still outside 10% margin which could be attributed to the low number of training data sets. The performance could be improved by employing methods that propagate the predicted health index to the

Table 4.14: Performance evaluation of the RF-based health index prediction

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	6	16.80	1	-38.52	10.74
2	7	23.20	2	-25.16	12.74
3	0	0.00	9	-19.94	11.45

Table 4.15: Performance evaluation of the SVM-based health index prediction

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	9	20.92	3	-38.11	17.56
2	2	11.77	5	-24.03	10.30
3	0	0.00	6	-21.37	8.63

threshold.

4.4 Propagating Health Index to a Threshold

If the condition monitoring data consists of a measurable health index or performance index, then the health index can be propagated from the current value to a predetermined threshold, from which the *RUL* is estimated as shown in Figure 2.8(a). Methods that account for various uncertainties during the propagation are best suited for this approach, where the *RUL* is represented as a distribution as shown in Figure 2.8(b). In this work, particle filter method which has been extensively used for model based prognostics of non-linear systems [132, 149, 191–193], is employed.

4.4.1 Particle Filter Approach

Particle filter is a general Monte Carlo (Sampling) method for estimating the state of a system that changes over time using a sequence of noisy measurements obtained from the system [114]. The state of the system is considered to evolve according to

$$q_k = f\left(q_{k-1}, t_{k-1}, t_k\right) + n_k, \tag{4.44}$$

where q_k is the state of the system at time k and f is the transition function that propagates q_{k-1} to q_k, and n_k is the process noise. The state vector is assumed to be unobservable and its information is only obtained through noisy measurements of its observation o_k which is obtained by

$$o_k = g(q_k) + \nu_k, \tag{4.45}$$

where g is the observation model and ν_k is the measurement noise. The filtering process involves the estimation of the state vector at time k, given all the measurements up to time k, denoted by $o_{1:k}$. From a Bayesian inference, this problem involves recursively calculating the posterior distribution $p(q_k|o_{1:k})$ which is done in two steps [114].

1. Prediction step, where the distribution $p(q_k|o_{1:k-1})$ is computed from the filtering distribution $p(q_{k-1}|o_{1:k-1})$ at time $k-1$ as follows:

$$p(q_k|o_{1:k-1}) = \int p(q_k|q_{1:k-1})p(q_{k-1}|o_{1:k-1})dq_{k-1}, \tag{4.46}$$

   where $p(q_{k-1}|o_{1:k-1})$ is assumed to be known due to recursion and $p(q_k|q_{k-1})$ is given in Equation (4.44) [114]. The distribution $p(q_k|o_{1:k-1})$ is known as a prior over q_k before receiving the most recent measurement o_k. Equation 4.46 is known as Chapman-Kolmogorov equation whose derivation can be found in [194].

2. Update step, where the prior is updated with the new measurement o_k using Bayes' rule to obtain the posterior over q_k

$$p(q_k|o_{1:k}) = \frac{p(o_k|q_k)p(q_k|o_{1:k-1})}{p(o_k|o_{1:k-1})}. \tag{4.47}$$

The computations in the prediction and update steps Equations (4.46-4.47) can be done using approximation methods such as Monte Carlo sampling [114].

When using this approach for prognostics, there is no new measurement available and hence the update step is not carried out. The system state is propagated using the state model until a predefined threshold is reached and this defines the end of life of the system, t_{EoL}. The RUL is then calculated for $i = 1, 2, ...N_p$ particles as follows

$$RUL_i = t_{EoL,i} - t_c. \tag{4.48}$$

The overall RUL is obtained from the distribution of the RUL obtained from the N_p particles. A typical approach is by taking a certain percentile (e.g. 45%) of the distribution. This increases the probability of early predictions and lowers that of late predictions. This approach has been employed in prognostics of various technical systems such as batteries [192, 195], fuel cells [149, 196], gears [132] and bearings [193].

4.4.2 Application Example

4.4.2.1 Remaining Useful Lifetime of PEM Fuel Cell

In this part, data sets provided by FCLAB Research Federation as part of the IEEE PHM 2014 Data Challenge [150] to evaluate the health states and remaining useful lifetime of proton exchange membrane (PEM) fuel cells are utilized. The challenge involved two activities:

1. State of Health (SOH) estimation.
2. Remaining Useful Lifetime (RUL) prediction.

This work discusses the method used in the second part of the challenge, that is, prediction of the remaining useful lifetime. Three kinds of data were available for the challenge: polarization and electrochemical impedance spectroscopy (EIS) parameters and aging parameters [150]. The EIS experiments were conducted at predetermined intervals, a

process referred to as characterization of the fuel cells. In estimating the remaining useful lifetime, only the aging parameters were utilized. The following two data sets were provided:

1. Run-to-failure data from fuel cell 1 (FC1) which was operated in a stationary regime. This data was considered as the learning data set.
2. Truncated run-to-failure data from fuel cell 2 (FC2) which was operated under dynamic current. This data was considered as the testing data set and the challenge was to estimate the remaining useful lifetime from the truncation time (550 hours) to the time that the initial voltage drops to predefined thresholds.

Data Description

As mentioned before, only the ageing data of both the training and testing data sets was used in this study. The data was drawn from two 5-cell PEM fuel cell stacks with an active area of 100 cm^2. FC1 was operated under a constant current of 70 A while FC2 was operated under a ripple current of 70 A with a 7 A oscillation at a frequency of 5 kHz. Condition monitoring data like power loads, temperatures, hydrogen and air stoichiometry rates, etc., were acquired. Figure 4.26 shows sample ageing parameters for FC1, where T_{in} H2 is the inlet temperature of hydrogen and T_{out} H2 is the outlet temperature of hydrogen, I is the current drawn from the cells and U_{tot} is the stack voltage of the cells.

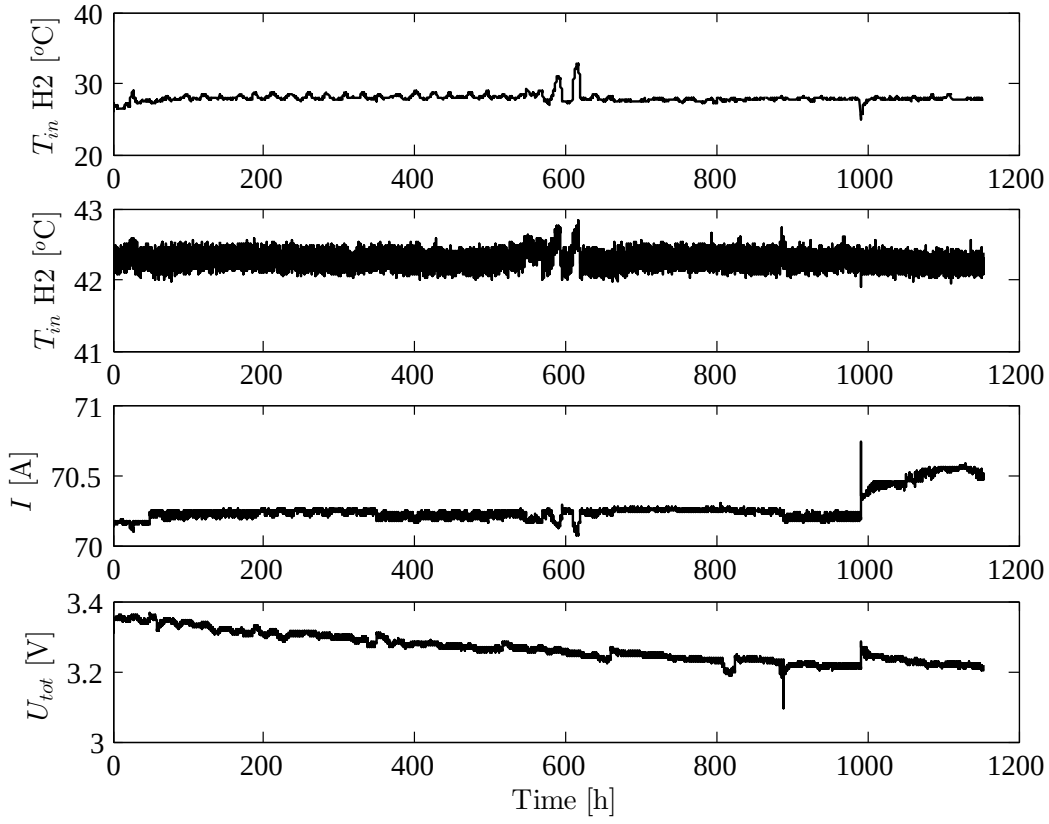

Figure 4.26: Sample monitoring parameters for FC1.

From Figure 4.26, it is evident that the stack voltage drops with time, indicating degradation phenomena. The voltage was therefore taken as the health index of the system *HI*. A suitable threshold can be defined as a percentage drop from the initial value of the *HI* when the fuel cell is new. At any given operational time, t_c, the *RUL* can be estimated as the remaining time before the threshold is reached, as shown in Figure 4.27. The focus of the challenge was to estimate *RUL* before a certain amount of power loss is reached and since voltage is directly proportional to power, in this context, the *RUL* is defined as the time before a certain amount of voltage drop (drop in *HI*) is reached.

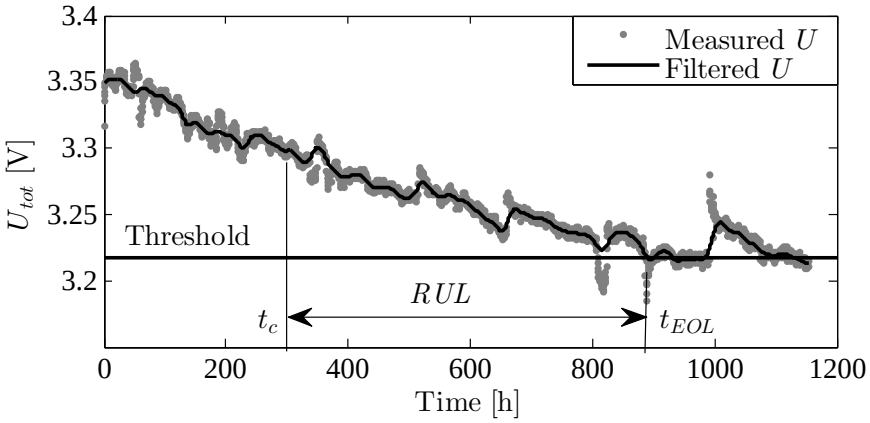

Figure 4.27: Estimating the *RUL* based on the current time and failure threshold

Data Preprocessing

From Figure 4.28(a), one can observe that the raw data is noisy and contains large peaks. Therefore the first step in processing the data was to remove the peaks and filter the noise. A kernel based smoother described in section 3.2.2 was employed to filter the noisy data. The filtered data (Figure 4.28(b)) was taken as the health index. After filtering,

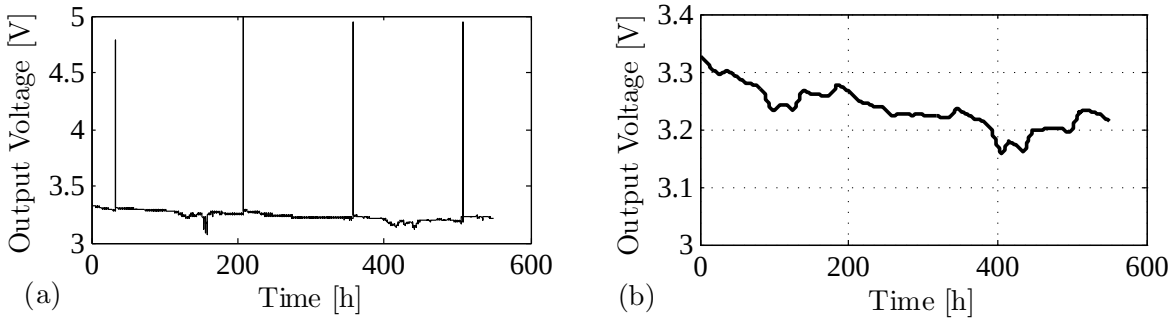

Figure 4.28: Condition monitoring data of a FC2 (a) raw data and (b) data filtered with a kernel smoother

the data was interpolated using cubic spline method so as to obtain samples at a uniform time interval of 0.5 hours.

Data Evaluation Challenges

Some of the challenges experienced in the data processing and analysis include:

- Threshold values depend on how the data is filtered and the filtering window used. This means that the estimated RUL also depends on how the data is preprocessed.
- In the test data, some threshold voltages are reached before the truncated time. These voltages were ignored.
- The degradation process showed different degradation trends after each characterization. In addition, the self-healing effect observed after each characterization was not uniform.

State Equation Selection

For the transition function in Equation 4.44, the following equations were tested with the training data and the equation parameters were adjusted accordingly:

1. Exponential equation:

$$f(HI_{k-1}, t_{k-1}, t_k) = \exp(-\beta \cdot (t_k - t_{k-1})) \cdot HI_{k-1} \tag{4.49}$$

2. Pure logarithmic equation:

$$f(HI_{k-1}, t_{k-1}, t_k) = -\alpha \cdot \ln(t_k/t_{k-1}) + HI_{k-1} \tag{4.50}$$

3. Log-linear equation:

$$f = -\alpha \cdot \ln(t_k/t_{k-1}) - \beta \cdot (t_k - t_{k-1}) + HI_{k-1} \tag{4.51}$$

4. Linear equation:

$$f = -\beta \cdot (t_k - t_{k-1}) + HI_{k-1} \tag{4.52}$$

5. Polynomial equation:

$$f = -\alpha \cdot (t_k - t_{k-1})^2 - \beta \cdot (t_k - t_{k-1}) + HI_{k-1} \tag{4.53}$$

Parameters α and β in the state equation need to be initialized and this can be done through a parameter sensitivity analysis using the training data. The observation equation was defined as

$$o_k = HI_k + \nu_k, \tag{4.54}$$

where nu is the observation noise computed from the training data.

Parameter Initialization

In order to initialize the model parameters and evaluate the effectiveness of the method in RUL estimation, the training data was truncated at time, $t_c = 550$ hours. A sensitivity analysis was conducted and the first series of results was used in parameter initialization. Since the initial distributions have to be defined first, it was assumed that for the initial state HI_0, the distribution is centered on the initially measured value with a range of ± 0.1 V. The unknown state equation parameters were also initialized with distributions taken as 1% of the nominal value. However, if more data sets are available for training, then the distribution of various parameters can be computed from the data sets. Data was fitted to the model to identify a suitable range of these parameters. Through sensitivity analysis, the number of particles was set to 2000 and predictions were made every 0.5 hours.

Remaining Useful Lifetime Estimation

The state models defined in section 4.4.2.1 were tested with the training data on their ability to estimate the remaining useful lifetime. The failure threshold was set to 96% of the initial voltage. This threshold was based on typical values for PEM fuel cells [149]. Figure 4.29(a) and (b) shows the predicted HI and the RUL distribution respectively, using the particle filter method. The overall RUL_p is taken as the 45_{th} percentile of the distribution in Figure 4.29(b) to ensure early predictions. This means the area to the left of the 45^{th} percentile is equal to 45% of the total distribution.

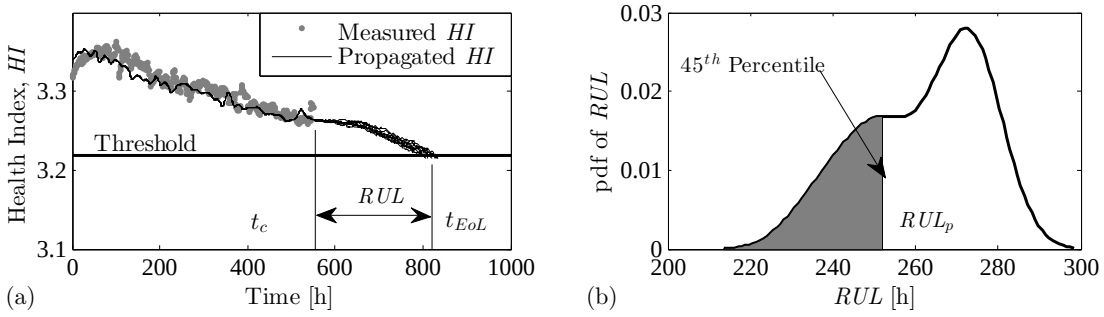

Figure 4.29: RUL estimation with exponential equation as the state model (a) HI trajectories and (b) RUL distribution

Table 4.16 shows the results of estimated remaining useful lifetime (RUL_p) at a failure threshold on 96%, with the error calculated using Equation 2.5. The approach produces estimations within an error margin of 14% which is a good reference for prognostics considering the limited number of training data sets available. However, the predictions can be improved through parameter adaption depending on the rate of degradation, as described in the next section.

Table 4.16: *RUL* estimation using particle filter with different state equations. RUL_a = 258 hours

State equation	RUL_p	e_r [%]
Exponential	237.0	8.13
Pure logarithmic	241.5	6.40
Log-linear	222.5	13.75
Linear	249.5	3.29
Polynomial	254.0	1.55

Parameter Adaptation

Figure 4.30 shows the self-healing effects where the voltage abruptly increases after characterization (EIS measurements). In addition, one can observe different degradation trends after each characterization. The model presented in Figure 4.29 does not cater for these effects.

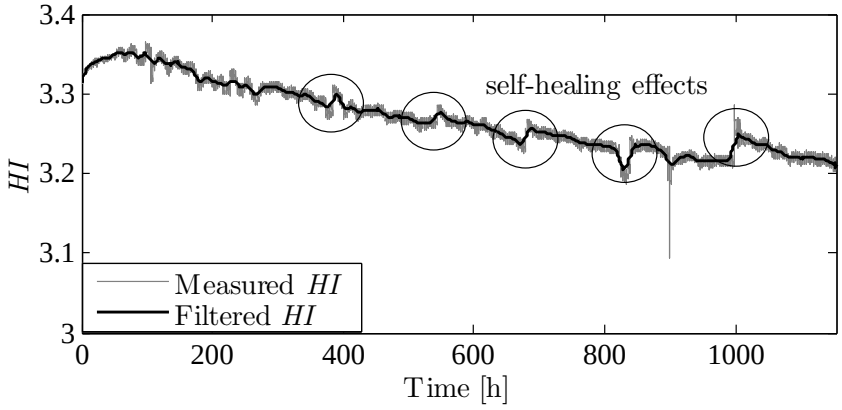

Figure 4.30: Self-healing effects of the PEM fuel cells after characterization

To address these limitations, a self-healing factor was introduced after each characterization step and then the state equation parameters were adapted to the changing degradation trend. This process was conducted at the prediction stage of the particle filter. Parameter adaptation was achieved by computing the difference in average gradient between any two characterization intervals. The self-healing factor, sf was computed by taking the average fractional increase in voltage after each characterization. Figure 4.31 shows the workflow of parameter adaptation and integration of self-healing factors sf. t_c is the current time, t_{ch} is the characterization time, $HI_k^{(i)}$ is the system state for particle i. A stopping criteria can be defined based on the failure threshold.

Figure 4.32 shows the predicted *HI* with integration of a self-healing factor and state model parameter adaptation. From Figure 4.32, it is evident that the proposed method is able to capture the degradation trend of the fuel cells, including adapting to the changing rate of degradation after characterization.

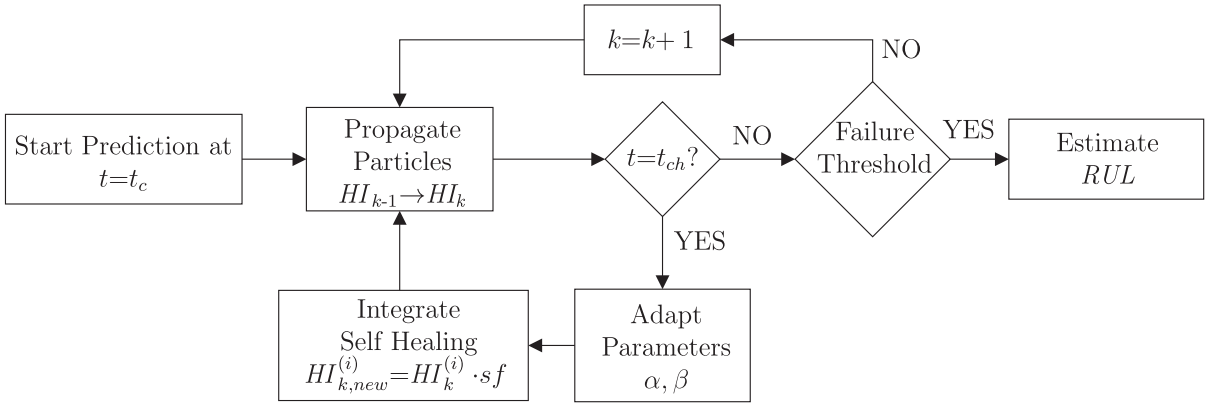

Figure 4.31: Workflow of method with parameter adaption and integration of self-healing factors

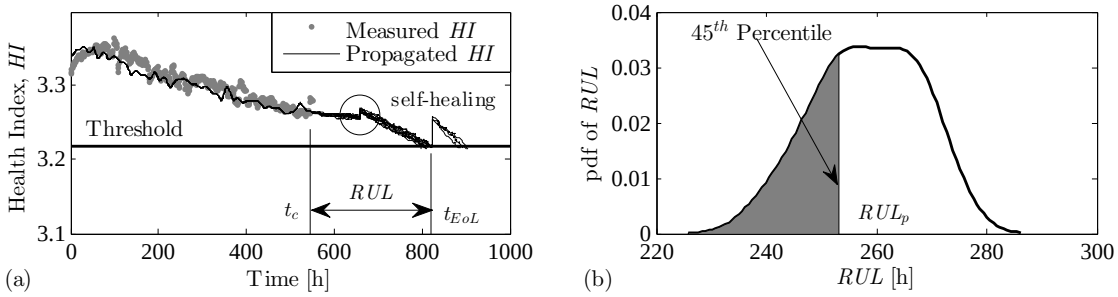

Figure 4.32: RUL estimation with integration of self-healing factors and parameter adaptation (a) HI trajectories and (b) RUL distribution

Table 4.17 shows the performance of the particle filter with integration of self-healing factor and parameter adaptation. A comparison between Tables 4.16 and 4.17 shows a relatively great improvement in the RUL prediction.

Table 4.17: RUL estimation with integration of self-healing and parameter adaptation. $RUL_a = 258$ hours.

State model	RUL_p [h]	e_r [%]
Exponential	253.5	1.74
Pure logarithmic	247.5	4.07
Log-linear	247.5	4.07
Linear	253.0	1.94
Polynomial	256.0	0.78

Performance Evaluation with Test Data

After parameter identification and adaptation, the models were applied to the test data and the RUL at the given current time and for predetermined failure thresholds was

evaluated. Figure 4.33 shows sample RUL estimation using polynomial state model. The predicted remaining useful lifetime RUL_p is taken as the 45^{th} percentile of the distributions as shown in Figure 4.33(b). This allows early predictions in most of the cases.

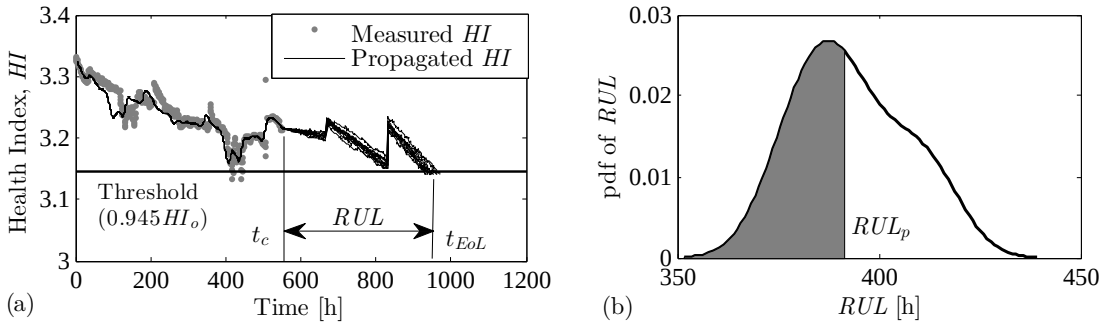

Figure 4.33: RUL estimation with integration of self-healing factors and parameter adaptation (polynomial state model) (a) HI trajectories and (b) RUL distribution

In order to exploit the benefits of individual state equations for propagating HI, a weighted ensemble of the results from the five state equations was applied as follows:

$$RUL = \frac{\sum_{k=1}^{5} w_k \cdot RUL_k}{\sum_{k=1}^{5} w_k}. \tag{4.55}$$

RUL_k is the estimated remaining useful lifetime using state model k and w_k is the weight of state equation k, computed from the errors in Table 4.17 as shown in Equation 4.56. The weight of each state model could also be calculated based on the mean square error obtained during the training stage.

$$w_k = 1 - \frac{e_{rk}}{100}. \tag{4.56}$$

Table 4.18 shows the estimated RUL at different failure threshold values and associated percentage errors, computed using Equation 2.5. The failure threshold was taken as a specified percentage power drop, in this case, percentage drop of the initial value of the health index (HI_o). Apart from the first threshold, which is too close to the current time t_c, all other predictions are very accurate (within 1% error). This is very good performance and hence it can be concluded that the proposed approach is suited for this kind of systems.

In prognostics, good performance of algorithms relates to early predictions and as such late predictions were penalized more in the scoring of the challenge. The score accuracy of a RUL estimate was defined as follows [150]:

$$A_{FT} = \begin{cases} e^{\ln(0.5)\cdot\frac{e_r}{5}} & \text{if } \%\ e_r \leq 0, (FN), \\ e^{-\ln(0.5)\cdot\frac{e_r}{20}} & \text{if } \%\ e_r \geq 0, (FP). \end{cases} \tag{4.57}$$

Table 4.18: Estimated RUL_k at different failure thresholds and associated percentage errors.

Failure threshold	RUL_a **[h]**	RUL_p **[h]**	e_r [%]
3.5% of HI_o	021.4442	062.0146	-189.19
4.0% of HI_o	194.1917	194.0457	0.075
4.5% of HI_o	209.7127	209.0492	0.316
5.0% of HI_o	384.3280	381.5899	0.713
5.5% of HI_o	386.7023	389.0192	-0.618

where A_{FT} is the score at a failure threshold

$$FT \in \{3.5; 4.0; 4.5; 5.0; 5.5\%\}.$$

The final score for all RUL estimates at the various thresholds, FT was taken as the mean of all A_{FT}. A score of 0.77 out of the maximum 1.0 was achieved with this approach, which was the top score in the challenge. Considering that only one data set was used for training, the performance of the method is very good. However, with more training data sets, the performance can be improved since the distribution of various parameters can be computed.

4.4.2.2 Ball Bearing Data

Temperature measurements can be used to monitor the degradation of systems that undergo wear. Although temperature measurements have been recognized as a condition monitoring data, their application in estimating the remaining useful life has not been fully realized. One of the identified limitations of temperature in machinery diagnosis and prognosis is the inability to identify faults at the development stage. However, this limitation can be overcome by strategic positioning of the temperature sensors. A number of sensors for condition monitoring of technical systems have been developed. A wireless temperature sensor for condition monitoring of bearings operating through thick metal plates was proposed by [197]. The sensor consists of a temperature-sensitive permanent magnet which is attached to the inner ring of the bearing, thus allowing the bearing temperature to modulate the produced magnetic field. Joshi et al. [198] demonstrated the application of radio telemetry for bearing cage temperature measurement for use in condition monitoring of bearings. The cage telemetry was found to capture faults such as loss of lubrication much faster than housing thermocouple. Brecher et al. [199] demonstrated the use of a customized telemetry system for measuring a bearing's inner ring temperature for high speed applications. The analysis showed that the inner ring temperature was vital in accurately monitoring the health of the bearing. These developments could prove useful in enhancing fault identification and estimation of remaining useful life in bearings as well as reducing the overall cost of the condition monitoring system. The temperature of the bearing increases rapidly during initial operation and then increases slowly to a steady state temperature as shown in Figure 4.34. As the bearing nears failure, the rate of temperature rise increases.

In this section, the approach described in section 4.4.1 can be implemented with tem-

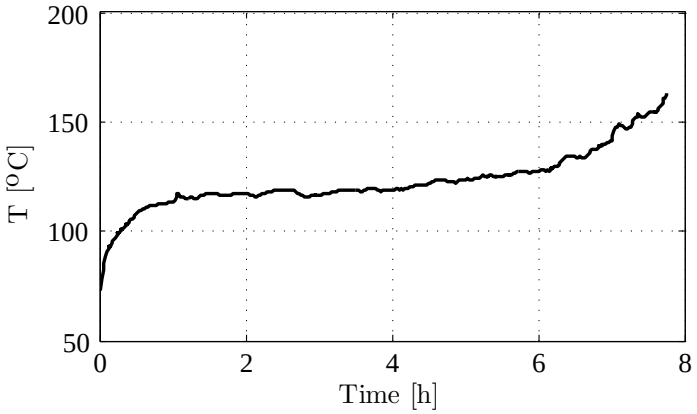

Figure 4.34: Typical run-to-failure temperature curve of a ball bearing

perature measurements as the health index. In this case, the system state is defined as $q_k = HI_k$. HI is obtained by normalizing the temperature change within the range $0 \leq HI \leq 1$ as shown in Equation (4.58)

$$HI = \frac{\Delta T - \Delta T_{min}}{\Delta T_{max} - \Delta T_{min}}. \tag{4.58}$$

In this study, two state equation were selected based on the temperature trend. The first part of the curve was approximated using a logarithmic equation while the second part was approximated using an exponential equation. The transition point was taken as the point where the rate of change of the health index (filtered using a kernel-based smoother) is zero, that is, $\frac{dHI}{dt} = 0$. Figure 4.35 shows the selection of state equations based on training data set from bearing 1_1.

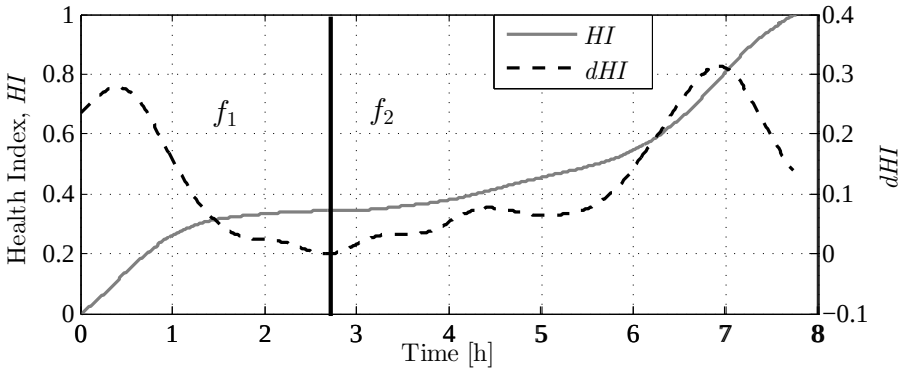

Figure 4.35: Health index with different state models

The selected state equations are shown below:

$$f_1 = \alpha \cdot \ln\left(\frac{t_k}{t_{k-1}}\right) + HI_{k-1}, \tag{4.59}$$

$$f_2 = HI_{k-1} \cdot \exp(\beta(t_k - t_{k-1})), \tag{4.60}$$

where, α and β are state equation parameters to be fitted from the training data.

To evaluate the performance of the approach on the training data and the suitability of the selected state equation parameters, the available training data is truncated at different fractions of the component's lifetime. The health index is then computed and propagated until it reaches a threshold. The RUL is then calculated from the trajectories of the HI as shown in Figure 4.36(a). The overall RUL_p is taken as the 45^{th} percentile of the RUL distribution as shown in Figure 4.36(b). As stated earlier, this approach increases the probability of early predictions.

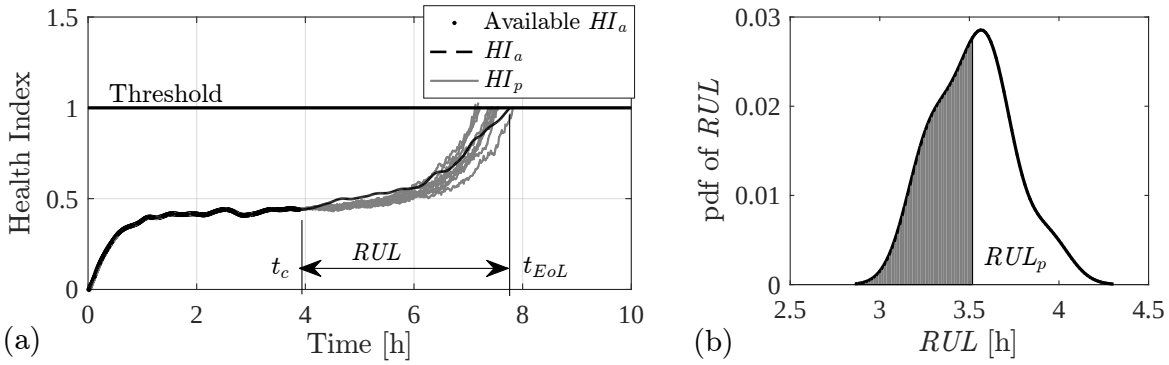

Figure 4.36: RUL estimation for bearing 1_1 at $t_c = 0.5t_{EoL}$ (a) HI trajectories and (b) RUL distribution

Figure 4.37 shows the performance of the approach based on the training data at different prediction intervals. At the early stages of prediction, the approach performs poorly, with a number of predictions falling outside the allowed error margin. This is because the rate of degradation changes at different stages of the bearings lifetime.

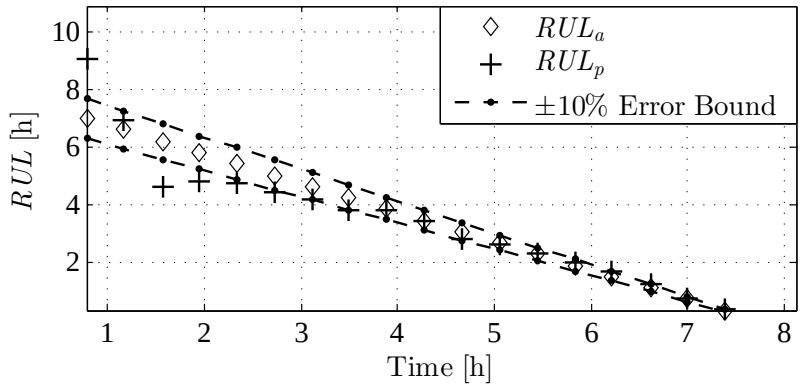

Figure 4.37: Estimated RUL at different prediction intervals without parameter adaptation

Most of the particle filter approaches employed in literature use single state equation

parameters when propagating the state of the system. However, as seen in Figure 4.37, due to the non-linearity of the degradation trend, it is difficult to obtain parameters that are able to track degradation of the systems accurately throughout its lifetime. The parameters are accurate at certain degradation stages and inaccurate at others. This limitation can be addressed by adapting the parameters to the rate of degradation. Figure 4.38 shows the RUL estimation at specified prediction intervals, with state equation parameter adaptation based on the rate of change of the health index. With this approach, the estimated RUL is approximately within the allowable $\pm 10\%$ error bounds at all degradation stages.

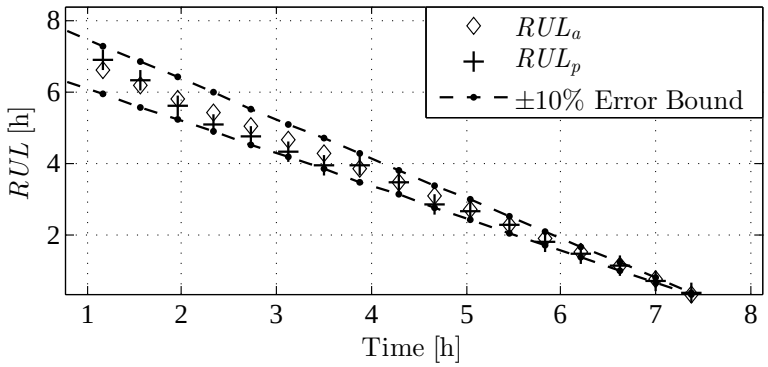

Figure 4.38: Estimated RUL at different prediction intervals with parameter adaptation

Once suitable state equation parameters have been identified, the method is then applied to truncated test data or data acquired in real-time. This involves normalizing the change in temperature of the test data using Equation 4.58 together with the maximum and minimum values obtained from the training data. The particles are propagated with resampling until the available data is exhausted after which the model is used to propagate the health index up to the threshold. Figure 4.39 shows the RUL estimated through this approach for bearing 1_7. The overall RUL_p at each prediction interval is taken as the 45^{th} percentile of the RUL distribution as shown in Figure 4.39(b).

The method was applied to other test bearings and a performance analysis conducted and presented in Table 4.19. With this approach, a mean absolute percentage error (MAPE) less than 20, computed for all the prediction intervals was achieved in all the data sets. Apart from data set 4, the approach results in fewer false negatives FN, which is a good attribute for prognostic methods. The approach takes into account measurement and state propagation uncertainties as well as the changing rate of degradation. The prediction accuracy also increases as more data becomes available due to resampling.

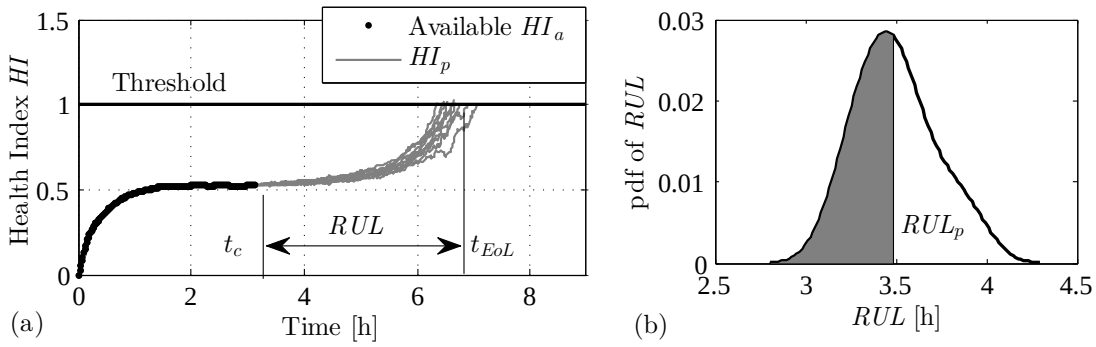

Figure 4.39: RUL estimation for bearing 1_7 at $t_c = 0.5t_{EoL}$ (a) HI trajectories and (b) RUL distribution

Table 4.19: Performance evaluation of the PF-based HI propagation using temperature measurement

	Early Predictions		Late Predictions		
Test	FP	Mean e_r [%]	FN	Mean e_r [%]	MAPE
1_1	3	26.66	0	0.00	6.17
1_2	8	21.10	6	-31.92	19.75
1_4	0	0.00	18	-14.55	13.90
1_5	15	18.19	1	-24.79	16.63
1_6	16	19.41	0	0.00	17.10
1_7	0	0.00	5	-62.53	18.30

4.5 Model Based Approach for Systems that Undergo Wear

This section discusses a simplified model based prognostic approach for systems that experience wear-induced failures. The approach is derived by correlating wear with temperature rise due to frictional heating and uses run-to failure temperature measurements to obtain coefficients which can be used with the temperature measurements at any given time to estimate the remaining useful life. Typical application includes rolling element bearings, journal bearings, shafts supported under journal bearings, friction clutches, gears, etc. Due to lack of sufficient temperature measurements for prognostics, only ball bearings are discussed in this section.

During operation, rolling element bearings encounter resistance to rotation which consist of rolling and sliding friction. This resistance occurs at the rolling contacts, contact areas between the rolling elements and the cage, as well as the guiding races [147]. The frictional forces perform work which is dissipated in form of heat, consequently increasing the bearing temperature. The frictional heat generated depends on the applied load, rotational speed, the type and size of bearing, properties and quantity of lubricant as well as the rate of heat dissipation. The rise in temperature reduces the viscosity of the lubricant which leads to a reduction in the lubricant film thickness. This results to higher asperity contact, increased heat generation due to increased friction and consequently increased wear [198]. Wear results to continued loss of geometric accuracy of the rolling and gradual development of other faults such as micro-pitting [147]. Since it is assumed

that wear can be prevented by proper attention to the bearing, no considerable effort has been made to estimate the remaining useful life of bearings related to wear and change in temperature [147]. Johnson [200] investigated the temperature produced by frictional heating in sliding contact by examining the temperature produced in a half space by a heat source which moves on the surface. The maximum temperature occurs towards the rear of the heated zone which has the longest exposure, as shown in Figure 4.40 [200].

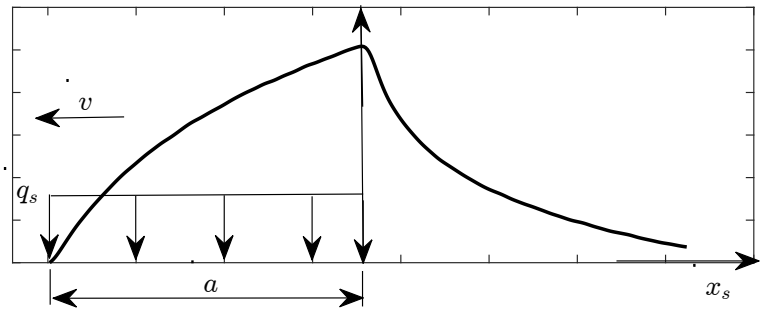

Figure 4.40: Temperature distribution due to uniform moving heat source, where q_s is the heat source, ΔT is the temperature difference, a is the half contact length and v is the velocity of the heat source

For a bearing rotating at constant speed, frictional heat is generated at all contact points leading to an overlap in the maximum temperature throughout the bearing. This would result to an almost constant temperature distribution. Therefore, the temperature will not be a function of position but the factors mentioned previously. The bearing operating temperature will also depend on the balance between the heat generated and the heat removed from the bearing through conduction, convection and radiation.

When a bearing is run continuously to failure, at constant speed, initially the rate of temperature change increases rapidly then decreases to an almost constant value. Just before failure, the temperature change again increases rapidly. This behavior can be observed in Figure 4.34 and is consistent with degradation of bearings, where degradation is high initially, followed by gradual degradation and finally high degradation towards failure. This indicates that temperature can be used to effectively track degradation of components occasioned by wear.

Assuming that the heat removal rate is constant for a given system, then it is possible to track bearing degradation from temperature measurements. Since wear is approximately proportional to the work done by the frictional forces which give rise to frictional heat, then it may be assumed to be directly proportional to the temperature rise in the component. This relationship can be formulated as

$$\frac{\Delta \dot{m}}{\Delta A} \propto \frac{\Delta P_R}{\Delta A} \propto (T(t) - T_o), \tag{4.61}$$

where $\Delta \dot{m}$ is the material removal rate, ΔP_R is frictional power, $T(t)$ is the current temperature, T_i is the room temperature and ΔA is the contact area. Taking the in-

itial temperature as the reference for temperature change and considering that wear is approximated by the material removal rate yields

$$m_{EoL} = \rho \Delta A \Delta z = \int_0^{t_{EoL}} \dot{m}(t)dt = k \int_0^{t_{EoL}} \Delta T(t)dt, \tag{4.62}$$

where ρ is the density of the material, ΔA is the contact area, Δz is the approximate wear depth, t_{EoL} is the time at the end of life of the component, k is proportionality constant, m_{EoL}, is the allowable mass that can be removed through wear before a component is considered to have failed. The ratio of allowable mass to the proportional constant can be obtained from the training data as by

$$\frac{m_{EoL}}{k} = \int_0^{t_{EoL}} \Delta T(t)dt. \tag{4.63}$$

Considering the current time, t_c, Equation 4.63 can be rewritten in cumulative form as

$$\frac{m_{EoL}}{k} = \int_0^{t_c} \Delta T(t)dt + \int_{t_c}^{t_{EoL}} \Delta T(t)dt. \tag{4.64}$$

During testing or online prognosis, the second term of Equation (4.64) is unknown. This term is proportional to the remaining allowable wear before the component fails. Therefore this factor can be referred to as the remaining wear coefficient. Equation (4.64) can be rearranged as:

$$\int_{t_c}^{t_{EoL}} \Delta T(t)dt = \frac{m_{EoL}}{k} - \int_0^{t_c} \Delta T(t)dt. \tag{4.65}$$

The term $\int_{t_c}^{t_{EoL}} \Delta T(t)dt$ can be used as a health index HI, defined as follows

$$HI = \frac{\int_{t_c}^{t_{EoL}} \Delta T(t)dt}{\frac{m_{EoL}}{k}}. \tag{4.66}$$

Division by $\frac{m_{EoL}}{k}$ normalizes the health index so that HI is within the range $0 \leq HI \leq 1$, with $HI = 1$ for a healthy component and $HI = 0$ for a failed component. Figure 4.41 shows the health index of the two bearings used to generate the training data sets. The health index can be approximated by a 2^{nd} order polynomial.

The term $\frac{m_{EoL}}{k}$ is computed from the training data sets which contain run-to-failure temperature measurements. The health index of the test bearing at the current time is computed using Equation (4.66), with $\frac{m_{EoL}}{k}$ obtained from the training data sets. To obtain the time to end of life, t_{EoL} of the test bearing, a polynomial curve of order 2 is fitted to the calculated health index and extrapolated to the point where the health index is zero. The RUL is calculated as shown in Figure 4.42(a). Figure 4.42(b) shows the RUL estimation at specified prediction intervals. The method has performs poorly during the early lifetime of the component when less data for fitting the HI is available. However, towards the end of life, the approach is very accurate.

The method was applied to other test bearings and a performance analysis conducted and presented in Table 4.20. With this approach, a mean absolute percentage error (MAPE) less than 20 for all the data sets was achieved. The approach is particulary very accurate towards the end of life of the components and majority of the predictions that are within

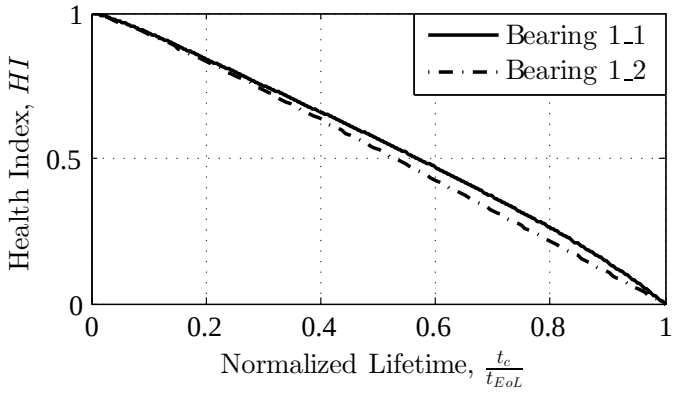

Figure 4.41: Health index computed from the training bearings

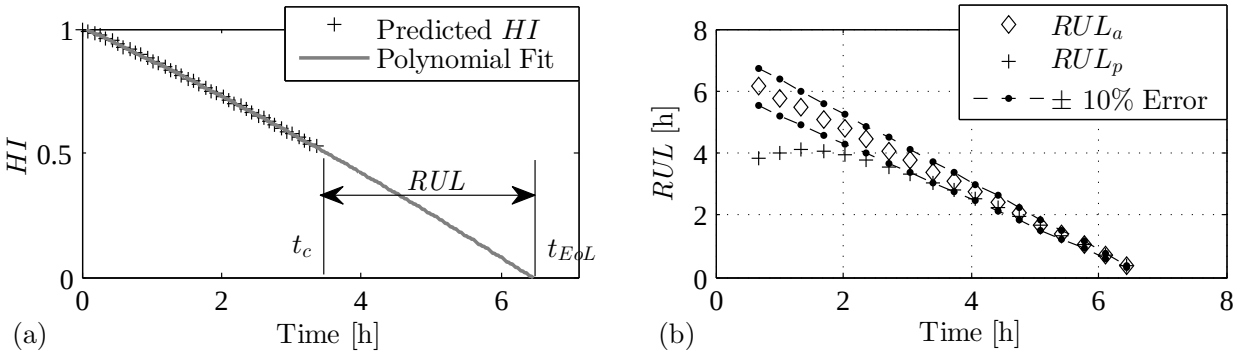

Figure 4.42: RUL prediction for bearing 1_6 at $t_c = 0.5t_{EoL}$ (a) HI trajectory and (b) estimated RUL at specified prediction intervals

the allowable error margin are located within the second half of the component's lifetime. Compared to the HI propagation method discussed in section 4.4, this approach performs better with an overall MAPE of 14.37 compared to 15.31 and very few false negatives. The performance can be improved if more training data sets are available to compute the distribution of $\frac{m_{EoL}}{k}$.

Table 4.20: Performance evaluation of the proposed model-based approach using temperature measurement

	Early Predictions		Late Predictions		
Test	FP	Mean e_r [%]	FN	Mean e_r [%]	MAPE
1_1	7	30.14	0	0.00	13.43
1_2	10	23.02	1	-25.00	15.56
1_4	8	25.40	0	0.00	11.81
1_5	13	21.23	0	0.00	17.21
1_6	10	21.97	0	0.00	13.60
1_7	9	26.04	1	-10.53	14.65

4.6 Ensemble of Methods

Different methods perform differently depending on the training data available and also the degradation level at the time of prediction. For instance, the model based method utilizing temperature measurements, is very accurate for predictions made within the second half of the component's lifetime. In order to improve prediction accuracy and build a more robust prognostic method, an ensemble of methods can be built by combining predictions of several methods based on a weighted average. The weights can be generated through performance evaluation of the training data in cases where only few data sets are available or by evaluating the training data sets in a cross validation manner in cases where more training data sets are available. The weight of each method can be taken as the average score computed through Equation 4.57 at different prediction intervals. This results in a weight of 1 for accurate predictions, in which $e_r = 0\%$. Table 4.21 shows the weights of each method computed from the training data sets of the ball bearings while Table 4.22 shows the weights of the methods for the milling cutter data sets. Although the mean absolute percentage error (MAPE) of the model-based approach utilizing temperature is lower than some of the other methods, majority of the predictions are early thereby giving it a higher weight.

Table 4.21: Prognostic method weight computed from ball bearing data sets

Prognostic method	MAPE	w_i
Health State Estimation - SVM	11.50	0.68
Health State Estimation - RF	11.18	0.57
Mapping features to *RUL* - ELM	14.40	0.66
Mapping features to *RUL* - RF	16.38	0.42
Mapping features to *RUL* - SVM	14.49	0.47
Propagating HI (temperature)	12.96	0.60
Model-based approach (temperature)	14.50	0.58

Table 4.22: Prognostic method weight computed from milling machine data sets

Prognostic method	MAPE	w_i
Health State Estimation - SVM	12.37	0.64
Health State Estimation - RF	10.58	0.65
Mapping features to *RUL* - ELM	10.75	0.51
Mapping features to *RUL* - RF	14.45	0.52
Mapping features to *RUL* - SVM	12.01	0.53
Mapping features to *HI* - ELM	12.52	0.60
Mapping features to *HI* - RF	12.20	0.56
Mapping features to *HI* - SVM	12.43	0.56

The resulting RUL is computed as follows

$$RUL_{ens} = \frac{\sum_{i=1}^{n_m} w_i RUL_i}{\sum_{i=1}^{n_m} w_i}, \tag{4.67}$$

where RUL_i is the RUL estimated by method i, w_i is the weight of method i and n_m is the number of methods. Performance evaluation of the ensemble of methods is shown in Table 4.23 for ball bearing data sets. Various combinations of the methods were evaluated and for the ball bearing data sets, combination of the five best algorithms was found to yield the best results, with a mean MAPE of less than 10% and 101/133 predictions within the acceptable error region of ±10%. The ensemble of methods outperforms all individual methods and it also integrates the advantages of each of the combined methods to yield a robust approach.

Table 4.23: Performance evaluation of the ensemble of 5 best methods for ball bearing data sets

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1_1	1	10.26	0	0.00	3.72
1_2	0	0.00	4	-35.57	10.36
1_3	0	0.00	5	-22.00	8.95
1_4	0	0.00	4	-17.01	7.11
1_5	1	16.41	1	-24.17	6.55
1_6	9	14.02	2	-50.57	14.12
1_7	0	0.00	5	-27.48	11.04

The performance evaluation of ensemble of methods for the milling machine cutting tool data sets is shown in Table 4.24. Similarly, various combinations of the methods were evaluated and a combination of four best methods was found to yield the best performance with a mean MAPE of 9.58 and 36/57 predictions within allowable error region. The ensemble of methods also outperforms all the individual methods.

Table 4.24: Performance evaluation of the ensemble of 4 best methods for milling machine cutting tool data sets

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	2	16.45	2	-24.32	7.54
2	3	14.73	2	-31.60	8.70
3	11	16.47	1	-17.79	12.52

4.7 Prognostic Method Selection

Selection of the most suitable method for a given application depends on a number of factors as listed in conclusion of the literature review in section 2.4.

In cases where run-to-failure condition monitoring data is available, a number of approaches may be applicable. However some approaches perform better than others depending on the type of system. Table 4.25 summarizes the performance of different approaches on ball bearing data. *FP* and *FN* are the total number of false positive and false negatives for all the data sets. A_m is the mean score of all tests, based on specified prediction time intervals computed using Equation 4.57. Health state estimation approach based on support vector machines (SVM) with optimally tuned parameters has the best performance for individual algorithms, with a performance score of 0.62 out of the maximum 1. The method has fewer total number of false positives and false negatives. This is attributed to the high classification accuracy of SVM with tuned parameters. This approach is suited for systems such as bearings, gears, pumps, etc., that have several failure modes that can be identified from condition monitoring data. For systems that undergo wear and for which temperature sensors can be incorporated to track temperature rise from dissipated frictional heat, then a model-based approach is best suited. This approach is particularly very accurate towards the end of life of the system, when maintenance decision is required. This approach is also less costly in terms of instrumentation and computational requirements. In order to incorporate the benefits of individual algorithms, an ensemble of methods based on a weighted approach is necessary. The ensemble method has the best overall performance with a mean performance score of 0.64, a mean MAPE of 8.83 and 101/133 predictions within the allowable error region. This shows that the proposed methods are excellent for prognostics considering the limited number of data sets used for training and hence can be used as reference for prognostics at the industry level.

Table 4.25: Summary of the performance of different approaches on ball bearing data sets

Method	*FP*	*FN*	MAPE	A_m
Health State Estimation - SVM	**29**	**20**	**12.27**	**0.62**
Health State Estimation - RF	41	16	12.67	0.59
Mapping features to *RUL* - ELM	48	22	14.92	0.56
Mapping features to *RUL* - RF	16	48	14.53	0.44
Mapping features to *RUL* - SVM	29	32	14.83	0.48
Propagating HI (temperature)	42	30	15.31	0.50
Model-based approach (temperature)	57	2	14.37	0.62
Ensemble (5 best methods)	**11**	**21**	**8.83**	**0.64**

Table 4.26 summarizes the performance of different approaches on milling machine cutting tool data. Random forest (RF) based health state estimation method provides the overall best performance with a score of 0.66. This could be attributed to the several wear states that the tool undergoes before it fails as observed on different contours of the

wear data in Figure 3.18. Other methods such as SVM-health state estimation method and extreme learning machines (ELM)-based method of mapping extracted features to a health index derived from *RUL* also produce good performance. The ELM-based health index estimation approach is the easiest to implement and if a physical health index is not available, then a virtual health index can be defined from the data using data fitting methods. The methods can be improved further using more training data sets as this increases the probability of covering more failure modes.

Table 4.26: Summary of the performance of different approaches on milling machine cutting tool data sets

Method	*FP*	*FN*	MAPE	A_m
Health State Estimation - SVM	12	10	12.38	0.60
Health State Estimation - RF	**15**	**5**	**9.74**	**0.66**
Mapping features to *RUL* - ELM	13	21	12.32	0.53
Mapping features to *RUL* - RF	12	10	11.66	0.55
Mapping features to *RUL* - SVM	14	11	13.13	0.53
Mapping features to *HI* - ELM	25	5	13.53	0.61
Mapping features to *HI* - RF	13	12	11.65	0.57
Mapping features to *HI* - SVM	11	14	12.16	0.56
Ensemble (4 best methods)	**16**	**5**	**9.59**	**0.66**

For systems where a health index is used as the condition monitoring data, such as fuel cells, lithium ion batteries, electronic components, etc., then the best approach is to propagate the *HI* to a predetermined threshold using a method such as particle filter, that can incorporate various uncertainties and also adapt to the changing rate of degradation.

5 Prognostics of Piezoelectric Transducers

Piezoelectric transducers are used in a wide range of applications and in most cases they are subjected to electromechanical cyclic loading. This often leads to accumulated fatigue damage and eventually to failure. For systems with high throughput, unforeseen failures often lead to long machine downtime leading to loss of production time and revenue. The reliability of these transducers is therefore an important aspect in their application [201]. Condition monitoring which comprises continuous or periodic monitoring of the health of technical systems and utilizing this information to predict failures and estimate the remaining useful life RUL can be used to increase reliability, safety and availability [8]. This is achieved by using the RUL estimates to schedule appropriate maintenance or to adaptively control the reliability for high precision positioning of actuators.

Traditionally, condition monitoring of piezoelectric transducers involves periodic characterization of health indices through impedance measurements, which requires additional measuring equipment, increases downtime and decreases availability. The cost of a condition monitoring system can be reduced by use of quantities such as driving electric current or voltage which are normally used for control and performance monitoring purposes. Ronkanen [202] evaluated the use of electric current measurement in control and monitoring applications. The study showed that current measurements can be used for applications such as displacement control, force estimation and to detect self-heating in actuators without the use of additional sensors [202]. Self-sensing approach using electric current has also been used in detection of cavitation in ultrasound transducers [156].

In this chapter, methods proposed in Chapter 4 are extended to utilize self-sensing capability of piezoelectric transducers to estimate their current health state and RUL. The impedance of a piezoelectric transducer operating at constant conditions changes with degradation. Therefore, if the transducer is excited with a constant amplitude voltage, the electric current is expected to change with degradation of the transducer. The electric current can be used as condition monitoring data to track degradation. The proposed approach uses machine learning algorithms to map features extracted from electric current to a health index (as detailed in section 4.3) derived from velocity measurements. Similarity based methods (described in section 4.3) and propagating HI with particle filters (described in section 4.4) are used to estimate the RUL from the current health index. The feasibility of the approach is demonstrated using run-to-failure experiments of piezoelectric bimorph benders.

5.1 Experimental Set-up

A set-up for electric cyclic loading of piezoelectric bimorph actuators (type M1876 from Johnson Matthey) was developed. The bimorph is clamped between two aluminum bars on one end while the other end is free as shown in Figure 5.1. It is dynamically excited with a sinusoidal voltage (V_{pp} = 80 V, f_e = 200 Hz), which is generated by a signal generator (Wavetek 395) and amplified by an amplifier (Piezomechanik LE250/2). Electric current is measured using a non-intrusive current probe (Tektronix TM5003) while the displacement of the free end is measured using a laser vibrometer (Polytec OFV 3001+OFV 512). The data are recorded for 0.5 s at intervals of 1 minute and at a sampling frequency of 20 kHz using a data acquisition (DAQ) board (Measurement Computing USB-202). Fifteen run-to-failure experiments were conducted.

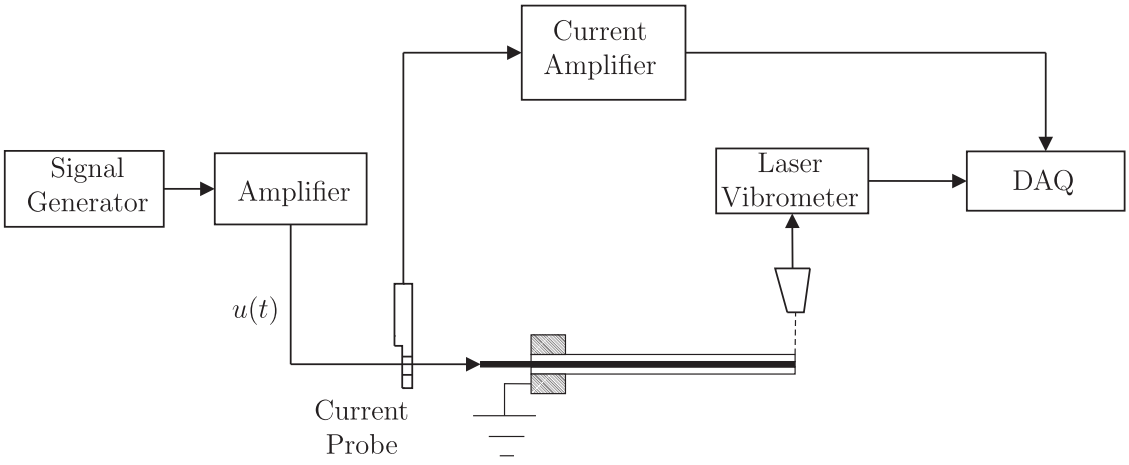

Figure 5.1: Experimental set-up for electrical cyclic loading of a piezoelectric bimorph actuator

Table 5.1 shows the material and geometric data of the piezoelectric bimorphs utilized in the experiments. The bimorph is rated at 150 V, but at frequencies near resonance, displacements is dynamically amplified. Therefore at these frequencies, the bimorph should be operated at much lower voltages, approximately 15 V [203]. With the mentioned excitation voltage, the degradation of the bimorph is accelerated so that failure occurs within 20 hours or after approximately 15 million cycles.

Typical application of the set-up in Figure. 5.1 is in textile knitting machines where the bimorphs are used for controlling the position of the needle during knitting [204]. The main mode of failure observed with this set-up was development and evolution of surface and internal micro cracks in the piezoelectric ceramic of the bimorph as shown in Figure 5.2.

Table 5.1: Specifications of the Bimorphs

Parameter	Value
Total length of piezoelectric layers	45±0.1 mm
Effective length of bender	38±0.5 mm
Beam width	2.1±0.1 mm
Resonance frequency	∼250 Hz

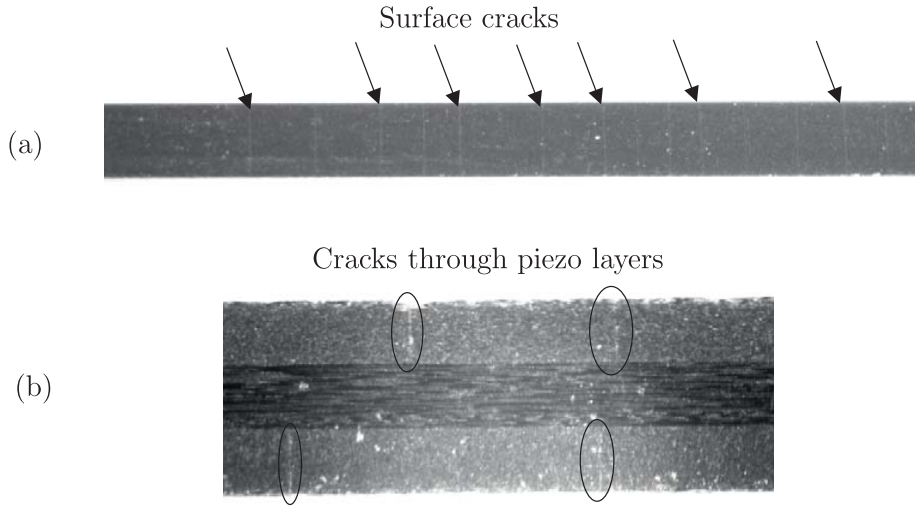

Figure 5.2: A bimorph showing development of surface and internal cracks in the bimorph after 2×10^7cycles (a) top view and (b) a section of the side view (magnification of 30×)

5.2 Identification of Health Indices

In order to develop a suitable condition monitoring system, it is important to identify measurable quantities that give an indication of the health of a component. The traditional approach is to use admittance or impedance measurements to characterize various properties of the piezoelectric device. Measurement of performance indicators such as displacement or force developed by the piezoelectric device is another approach which is easier to implement in online condition monitoring.

5.2.1 Admittance Measurement

An impedance analyzer (HP 4192A) was used to measure the electrical admittance of the piezoelectric bimorph over a range of excitation frequencies between 200-300 Hz at an amplitude of 1 V. The resonance and anti-resonance frequencies of the piezoelectric bimorph in this set-up are expected to lie within this range. The measurements were conducted after every hour (approximately 7.2×10^5 cycles). Figure 5.3(a) shows the admittance over the specified frequency range after specified operation cycles. The admittance decreases with continued operation, which corresponds to an increase in the impedance of the piezoelectric bimorph. Therefore, the equivalent resistance, R_m due to internal damping of the bimorph can be used to track degradation as shown in Figu-

re 5.3(b). The equivalent resistance is computed from the admittance by

$$R_m = \frac{1}{\max(\mathrm{Re}(Y_{el}))}, \tag{5.1}$$

where Y_{el} is the admittance.

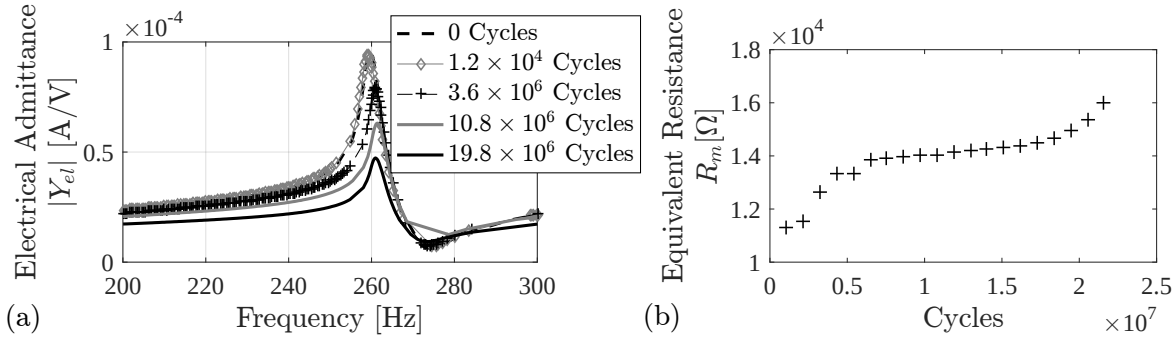

Figure 5.3: Health index identification of piezoelectric bimorph (a) admittance measurement at specified operation cycles and (b) change in equivalent resistance with continued operation

Another indicator that shows a trend consistent with degradation is the minimum phase of the admittance which is also proportional to the internal damping of the piezoelectric bimorph. A plot of the phase of the admittance over the excitation frequencies is shown in Figure 5.4(a), while Figure 5.4(b) shows the trend of the minimum phase over the lifetime of a bimorph.

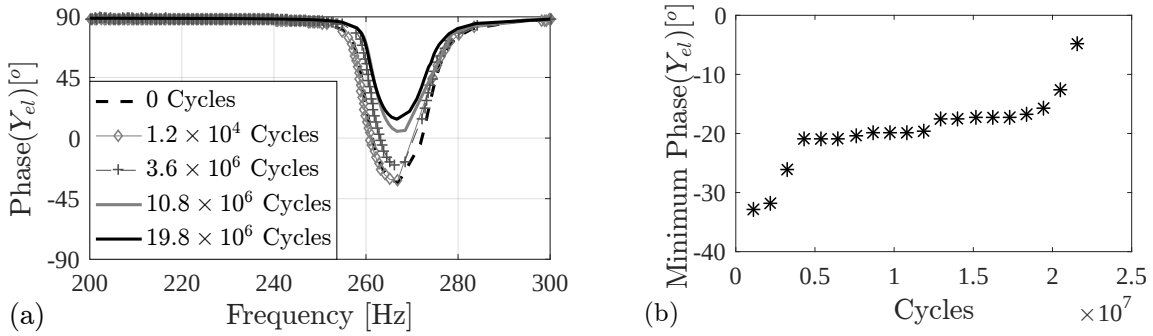

Figure 5.4: Tracking degradation through minimum phase of electrical admittance (a) phase measurement and (b) trend of minimum phase with continued operation

The disadvantage of admittance or impedance measurements taken over a range of frequencies for tracking degradation of piezoelectric bimorphs is that it may require in-

terruption of production during operation and hence it may not be suitable for online condition monitoring. In addition, it is difficult to define failure threshold based on these quantities without assessing the performance of the transducer. A more realistic approach would be to use a performance index such as displacement or force measurements as the health index. The quantity to use depends on the application of the piezoelectric bimorph.

5.2.2 Displacement Measurements

Displacement of the free end is one of the measurable performance indices that can be used to track degradation of piezoelectric transducers. In this work, the velocity of the free end which is proportional to the displacement was measured using a laser vibrometer. Figure 5.5(a) shows changes in velocity after specified operation cycles for a bimorph excited by a sinusoidal voltage set at a constant value at the signal generator. The peak to peak velocity can be used to define a health index as

$$HI = \frac{v_A}{v_{Ao}}, \tag{5.2}$$

where v_A is the peak to peak velocity, $v_{Ao} = v_A(t = t_o)$ is the initial peak to peak velocity. Figure 5.5(b) shows the trend of the health index of the lifetime of a piezoelectric transducer.

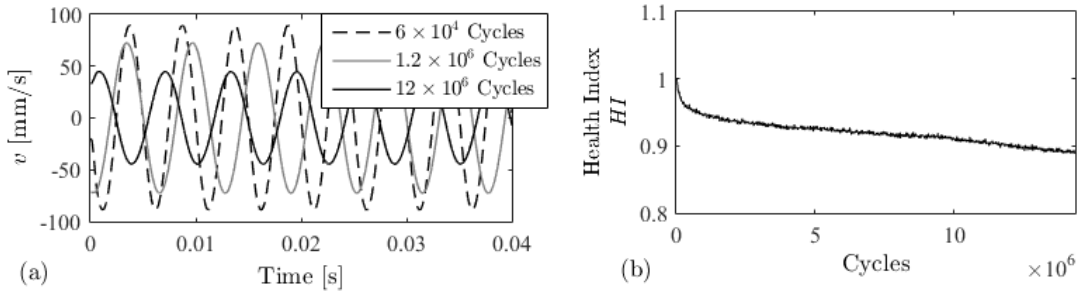

Figure 5.5: Change in velocity with degradation (a) raw velocity measurements and (b) health index derived from velocity

The normalization of the health index with the excitation voltage prevents a false alarm due to changes in the excitation voltage, especially for systems that require precise displacement where the excitation voltage is varied to ensure a constant displacement is attained. The end of life of a transducer can be defined as a predefined percentage drop of the health index, which depends on the application of the transducer. In this work, the failure threshold was set at 10% drop of the health index.

5.2.3 Electric Current Measurements

In most applications of piezoelectric transducers, driving electric current and voltages are measured for control purposes. As seen in Figure 5.3, the admittance changes with degradation, therefore, if one of these quantities is held constant, then the other quantity may be tracked to represent degradation under constant load. In this application, the amplitude of the driving voltage is held constant and the electric current is measured using a non-intrusive electric current probe. Figure 5.6(a) shows changes in current after specified operation cycles. The trend in the electric current change throughout the lifetime of a piezoelectric transducer can be observed from features extracted from the raw electric current signals, as seen in Figure 5.6(b). This trend appears to be consistent with degradation.

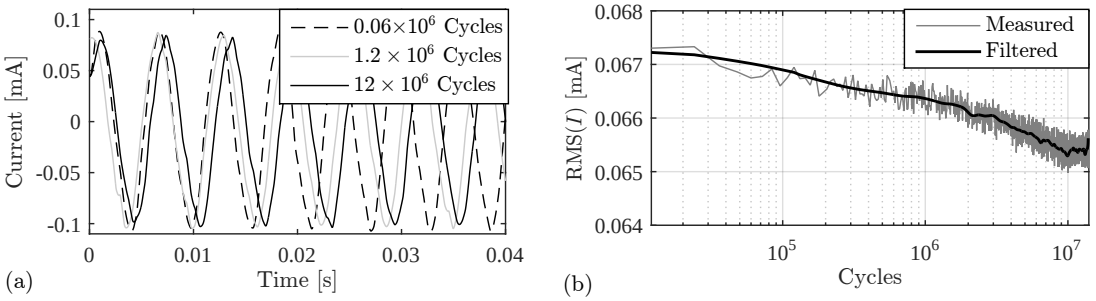

Figure 5.6: Change in electric current with degradation (a) raw electric current measurements and (b) RMS value of the current with continued usage

5.3 Prognostics Approach

In actual practice, the physical health index may not be measured easily. However, measurable quantities such as the driving electric current or voltage can be used together with machine learning algorithms to approximate the health index from which the remaining useful life can be estimated. Figure 5.7 shows the workflow of the proposed approach which is an extension of the method described in section 4.3. The approach involves two steps: training, where a machine learning algorithm is trained to map extracted features to the health index and testing, where the learned model is used with extracted features from the test data to estimate the current health index of the transducer. The estimated health index is then either propagated to a predetermined threshold using particle filter, from which the RUL is determined or the RUL is computed through similarity approach.

5.3.1 Health Index Prediction

Nine run-to-failure data sets ($N_T = 9$) were used to train three machine learning (ML) algorithms: extreme learning machines (ELM), random forests (RF) and support vector

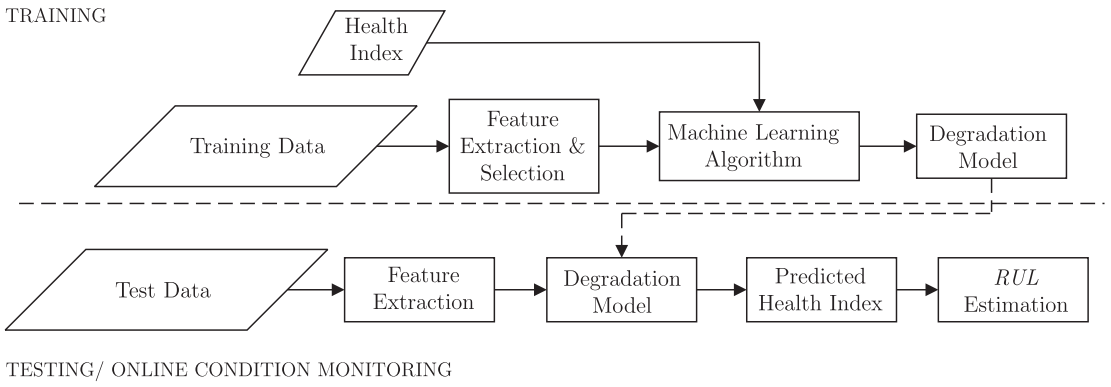

Figure 5.7: Workflow of the prognostic approach for piezoelectric transducers

machines (SVM). The data sets consisted of Q input features x_k^{tr} extracted from current measurements using methods described in section 3.3 and selected using methods described in section 3.4.2. The data sets also contain velocity measurements which were used as health indices HI^{tr} and as ground truth representation of degradation. This resulted in nine models for each algorithm, each consisting of weights, biases and parameters used in mapping the features to the health index. The use of multiple models ensures that various uncertainties such as loading and manufacturing variability are taken into consideration. Six data sets were used for testing. For each test data set, extracted features x_k^{te} were used as inputs to the trained models which have predicted health index as output. Nine trajectories of the predicted HI_{pi}, corresponding to $i = 1, 2, ...N_T$ models are produced. The final HI_p is obtained by combining all HI_{pi} using weighted approach. The weights w_i are obtained through a similarity measure computed from the mean square error (MSE) between the training features x^{tr} and test features x^{te} as follows

$$w_i = \frac{1}{Q}\sum_{k=1}^{Q}\left(\left[\frac{1}{n}\sum_{j=1}^{n}\left(x_{ijk}^{tr} - x_{jk}^{te}\right)^2\right]^{-1}\right), \tag{5.3}$$

$$W = \sum_{i=1}^{N_T} w_i, \tag{5.4}$$

$$HI_p = \frac{1}{W}\sum_{i=1}^{N_T} w_i \cdot HI_{pi}, \tag{5.5}$$

where n is the number of data points in the test data. Figure 5.8is a comparison of predicted health index with ELM, RF and SVM and the actual health index of a sample test bimorph obtained according to Equation 5.2. The two indices overlap each other but a slight difference between the different algorithms can be observed. The overlap indicates that the algorithms accurately predict the health index.

The performance of the algorithm in predicting the health index was evaluated by computing the mean square error (MSE) and correlation coefficient between the predicted and actual HI. Table 5.2 shows the performance of the algorithm. A correlation coefficient, r_c close to 1 and MSE close to 0 shows high similarity between the predicted and actual HI. The approach is accurate in estimating the HI, as can be observed from the performance metrics in Table 5.2.

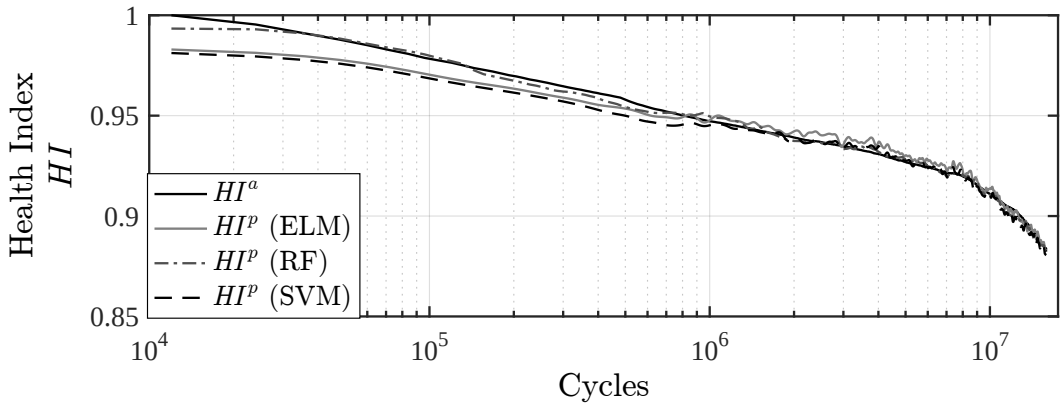

Figure 5.8: Comparison of predicted HI_p and actual HI_a health index for test data set 1

Table 5.2: Performance evaluation of the approach in predicting HI

Test	ELM		RF		SVM	
	MSE $[\times 10^{-6}]$	r_c	MSE $[\times 10^{-6}]$	r_c	MSE $[\times 10^{-6}]$	r_c
1	6.62	0.990	2.76	0.997	8.84	0.991
2	11.30	0.996	10.47	0.997	11.08	0.996
3	8.14	0.993	8.62	0.999	6.86	0.994
4	6.46	0.994	2.42	0.998	5.32	0.995
5	7.85	0.991	2.57	0.998	7.44	0.992
6	9.09	0.993	3.14	0.998	16.3	0.993

5.3.2 Remaining Useful Lifetime Estimation

In prognostics and health management of technical systems, the RUL is estimated at specified intervals of operation. This information is then used to either schedule appropriate maintenance or control the reliability of the technical systems. In this work, two methods, similarity based method (described in section 4.3) and particle filter (described in section 4.4), are used to estimate the RUL.

5.3.2.1 Similarity Based Method

The similarity between the features of the test data set and those of the training data sets is computed through MSE. The similarity weights obtained in section 5.3.1 are used with the RUL of the training data at the predicted value of HI to obtain a weighted RUL of the test data set as follows

$$RUL_p = \frac{1}{W}\sum_{i=1}^{N_T} w_i RUL_i^{tr}. \tag{5.6}$$

The confidence intervals were calculated as described in section 2.3.3 using normal distribution of the lifetimes, since the lifetimes of the training data sets were found to be normally distributed as shown in Figure 5.9.

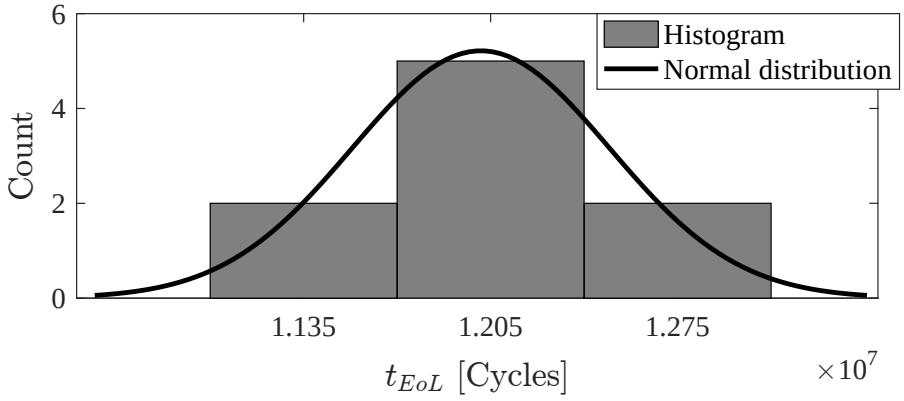

Figure 5.9: Normal distribution of t_{EoL} of the training data sets

Figure 5.10 shows a comparison of estimated *RUL* of test data set 1 based on similarity method with *HI* predicted using (a) extreme learning machines (ELM), (b) random forests (RF) and (c) support vector machines (SVM).

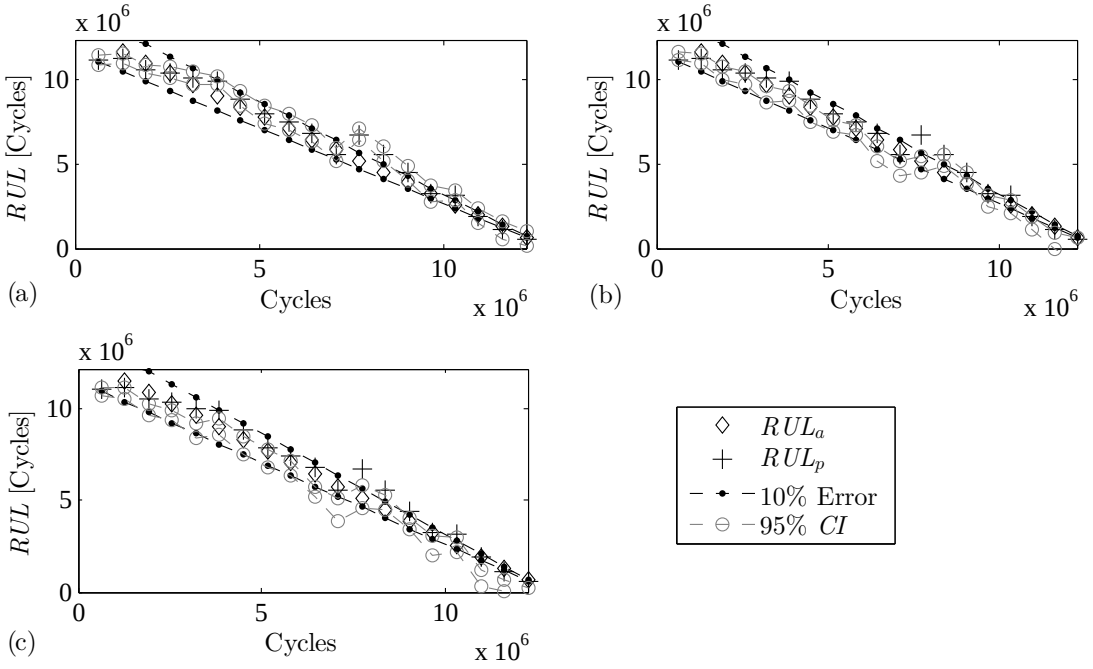

Figure 5.10: *RUL* predictions at specified time intervals for test data set 1 with *HI* predicted through, (a) ELM, (b) RF and (c) SVM

The performance of the methods was evaluated and is presented in Tables 5.3, 5.4 and 5.5. The mean MAPE for ELM, RF and SVM are 8.9, 9.4 and 9.7 respectively, all less than 10. Therefore, it can be concluded that this approach is a good reference for prognostics as long as there is sufficient (above ten) training data sets.

An ensemble of the *HI* obtained from the three methods based on a simple mean was

Table 5.3: Performance evaluation of the method with the health index estimated using ELM

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	1	19.69	1	-23.7	6.35
2	2	17.84	6	-14.31	8.50
3	0	0.00	4	-20.31	7.34
4	0	0.00	4	-19.93	6.74
5	2	14.00	3	-44.42	11.29
6	10	20.58	1	-11.02	13.16

Table 5.4: Performance evaluation of the method with the health index estimated using RF

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	2	28.56	1	-12.06	6.87
2	1	53.52	5	-14.45	10.90
3	0	0.00	9	-17.26	10.31
4	0	0.00	5	-20.13	7.35
5	0	0.00	4	-42.02	11.37
6	6	18.26	0	0.00	9.82

Table 5.5: Performance evaluation of the method with the health index estimated using SVM

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	3	26.99	0	0.00	8.14
2	4	26.93	3	-16.59	11.57
3	1	61.55	2	-11.05	7.78
4	1	47.13	4	-16.22	7.62
5	1	12.00	3	-34.64	8.79
6	10	22.31	0	0.00	14.15

also evaluated and results are presented in Table 5.6. The mean MAPE for the ensemble is 8.7, which is very close to that of the three algorithms. This shows that the algorithms (ELM, RF and SVM) perform more or less the same with this approach. However, the ensemble exploits the advantages of each algorithm and hence resulting in a more robust prognostic method.

5.3.2.2 Particle Filter Method

In this method, particle filter is used to propagate the health index from the current time stamp to a threshold from which the *RUL* is estimated. The current health index is taken as the mean of the *HI* obtained from the three machine learning algorithms. The training data was used to fit a state model for propagating the health index. The

Table 5.6: Performance evaluation of the method with the health index estimated using ensemble of ELM, RF and SVM

	Early Predictions		Late Predictions		
Test	*FP*	Mean e_r [%]	*FN*	Mean e_r [%]	MAPE
1	3	17.21	0	0.00	6.35
2	2	30.15	3	-16.15	10.21
3	1	18.07	5	-15.53	7.86
4	0	0.00	4	-18.22	6.11
5	1	11.55	3	-43.04	9.87
6	10	18.24	0	0.00	11.97

state equation and its parameters were identified through particle swarm optimization (PSO) by fitting various state equations to the training data. The lowest mean square error (MSE) between the actual health index and the fitted health index was used as the selection criteria for the state equation. A two-degree exponential equation given below was found to be the best fit for the health index:

$$HI_{k+1} = HI_k \left[\exp\left(-\alpha t_k\right) + \exp\left(\frac{\beta}{t_k + 1}\right) \right], \tag{5.7}$$

with the model parameters obtained as: $\alpha = 9.5 \times 10^{-5}$ and $\beta = 0.016$.

The observation equation was taken as

$$o_k = HI_k + \nu_k, \tag{5.8}$$

where ν_k is the observation noise, computed from the training data sets.

Figure 5.11 shows a comparison of the actual and the fitted *HI* for training data set 4. As seen in Figure 5.11, the two curves overlap each other indicating that the fitted equation correctly represents the behavior of the data. The MSE between the actual *HI* and fitted *HI* for this data set was found to be 1.94×10^{-5}.

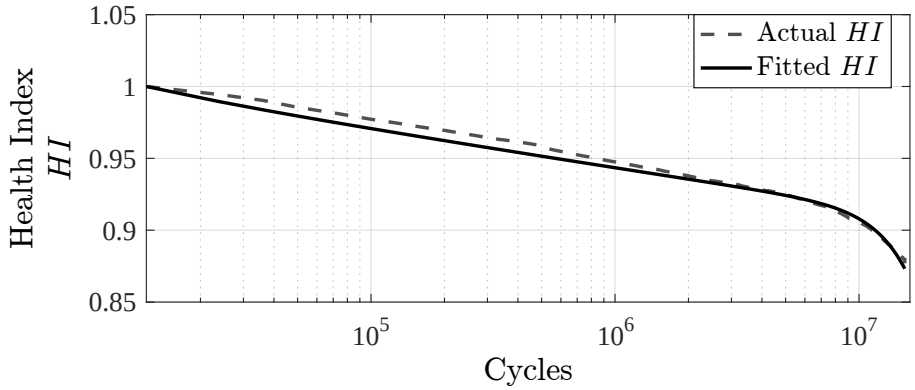

Figure 5.11: Comparison of the actual *HI* and fitted *HI* for training data set 4, obtained through particle swarm optimization (PSO)

The available predicted HI_p is used as the observation or measurement during the update step. The particles are initialized with the initial HI_{po}, which was found to be distributed around 1 with a standard deviation of 0.015. The HI_p is propagated until the threshold is reached from which the RUL is estimated as shown in Figure 5.12(a). Figure 5.12(b) shows the distribution of RUL from the particles. The overall RUL is obtained as the 45^{th} percentile of the RUL distribution, as shown in Figure 5.12(b). The 45^{th} percentile of the distribution is the RUL_p at which the area to the left is 45% of the distribution. This ensures early prediction in most cases.

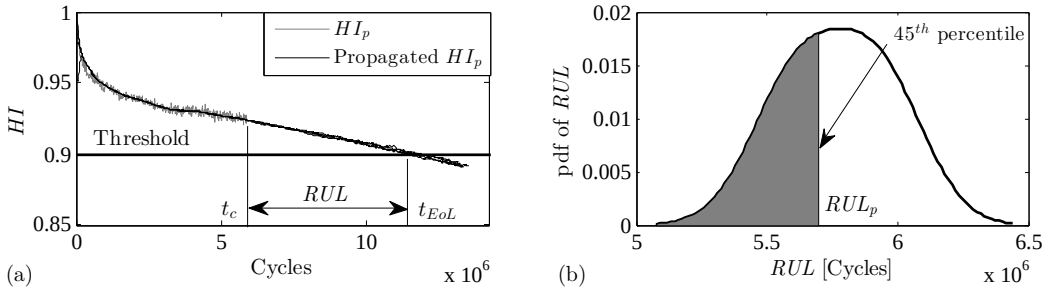

Figure 5.12: Estimated RUL at a specific operating time for test data set 1 (a) HI trajectories of a specified number of particles and (b) PDF of RUL for a specified number of particles

The RUL estimation was conducted at specific prediction intervals and a sample performance is shown in Figure 5.13. The confidence interval was computed based on the standard deviation of the RUL of the training specimens at the specified prediction intervals. The confidence interval lies within the 10% error margin as can be observed from Figure 5.13.

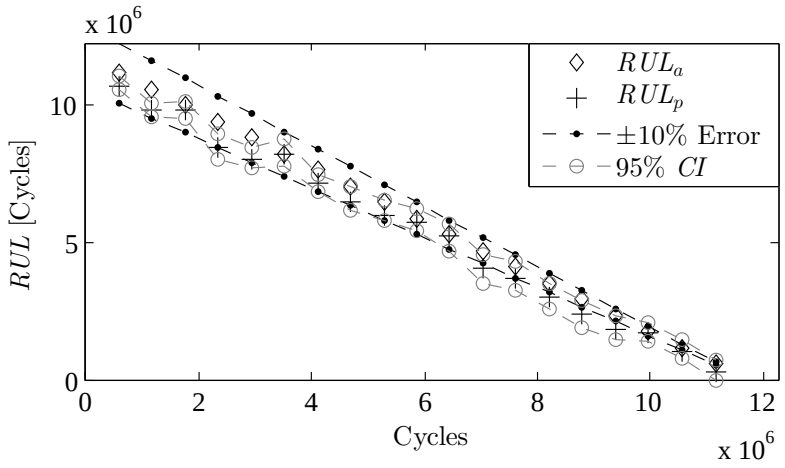

Figure 5.13: Estimated RUL at specified intervals of operating time for test data set 1

Table 5.7 shows the performance of the particle filter approach for all the test data sets. The mean absolute percentage error (MAPE) for all the prediction intervals lies within ±15% indicating the approach can be used as a good reference for scheduling maintenance or for controlling the reliability of the transducers.

Table 5.7: Performance evaluation of the method with RUL estimated using particle filter method

	Early Predictions		Late Predictions		
Test	FP	Mean e_r [%]	FN	Mean e_r [%]	MAPE
1	3	21.31	1	-12.98	6.77
2	2	31.32	2	-13.82	8.73
3	0	0.00	4	-14.09	6.07
4	0	0.00	3	-18.67	5.85
5	2	11.70	2	-53.43	9.10
6	11	17.08	0	0.00	12.15

In estimating the health index from the features, SVM shows the best performance and has the least mean square error (MSE), as seen in Table 5.2. However, RF and ELM also perform relatively well. The accuracy of the RUL estimations depends on the accuracy in predicting the HI. Table 5.8 compares some of the performance metrics for the different methods presented. The average computation time per prediction does not include the training time since it is assumed that during online prognosis, the trained model is already available. The mean MAPE is for all the six test data sets and the performance evaluation value A_m takes into account early and late predictions. The highest score when all predictions are timely is 1.

Table 5.8: Performance Evaluation of different methods in RUL estimation

Method	FP	FN	mean MAPE	A_m	Average computation time per prediction [s]
ELM	15	19	8.89	0.598	0.78
RF	9	24	9.43	0.586	0.14
SVM	20	12	9.68	0.618	1.07
Ensemble	17	15	8.73	0.613	1.99
PF	18	12	8.11	0.646	88.00

Particle filter method shows the overall best performance but has more computational requirements in terms of time and memory.

6 Conclusion and Future Work

6.1 Conclusion

The ability to estimate the current health state of a technical system and to accurately estimating its remaining useful life is very important in condition based maintenance. The information on the health condition of a system is not only useful in scheduling maintenance but also in optimizing inventory and resource utilization. This information can also be used to ensure the mission objective of autonomous and intelligent systems is achieved. In the case of machine tools, the health state of the system can also be used for indirect quality control of products since the quality of a product is affected by the condition of the tools. Given the importance of condition monitoring of technical systems, this work focussed on development of four approaches for condition monitoring of technical systems with the aim of predicting when a system is likely to fail within reasonable accuracies. The approaches depend on the type of system and the condition monitoring data and how it is processed. The main aim of this work was to develop a guide for selecting the appropriate approach and method for prognostics of a given system. This was done by evaluating the performance of selected approaches and algorithms with condition monitoring data of different systems.

For continuous signal-type condition monitoring data, such as vibration signals, signal denoising through discrete wavelet transform (DWT) is an important step in preprocessing the data. Denoising removes the high frequency sub-bands of a signal which contain the most noise and thereby ensuring that the denoised signal has the most information on the condition of the system. It was demonstrated that denoising the signals enhances the extracted features for fault diagnosis and prognosis. Features extracted from the denoised signals are used together with machine learning algorithms to predict the current health state as well as the remaining useful life. Since not all features contain information on the health state of a system, it is important to select relevant features in order to reduce dimensionality of the data and consequently reduce the computational requirements. Feature selection also reduces over-fitting of machine learning algorithms and hence improves the generalization ability of the algorithms. In this work, two methods for feature selection have been proposed. The maximum separation distance method is proposed for systems with discrete health states or for diagnoses. In this case, the separation distance between any two classes is computed and features that show maximum separation of classes are selected as input to machine learning algorithms. The method was validated with vibration and electric current data of ball bearing with different faults. For regression or function approximation (for instance health index approximation), the use of extreme learning machines to evaluate the performance of each feature on approximating the health index is best suited for feature selection. This allows the selection of features that closely approximate the health index.

For single valued condition monitoring data or in some cases data used for performance monitoring of technical systems, preprocessing involves the removal of outliers as well as filtering out the noise using smoothing algorithms such as kernel based regression smoother and moving average smoother. The kernel smoother provides better results since it does not have the bias at the end point of a feature/health index as opposed to the moving average approach. The condition monitoring data is then directly used as the health index and propagated to a predetermined threshold.

A large number of machine learning algorithms have been proposed in literature. However, there is no clear guideline on which approach to use with the machine learning algorithms for prognostics. In this work, a number of approaches based on the type of system being monitored and the condition monitoring data available, have been proposed:

- For systems or components that display several discrete health states before failure, such as ball bearings and gearboxes, machine learning algorithms can be trained to identify the current health state through health state probabilities. Once the current health state is identified, the health state probabilities are used with the historical percentage of the remaining useful life of similar systems to estimate the remaining useful life. In this work, a method for identifying the optimum number of health states has been introduced. The approach involves the use of k-means clustering algorithm with randomly sampled features to cluster data into a number of clusters and a similarity index between sorted elements of the clusters is computed. The number of clusters with the highest similarity index is selected as the optimum number of health states. The approach was validated with vibration data of ball bearings with three health states or conditions: healthy, small pitting on outer ring and large pitting on outer ring. Once the optimum health states have been identified, the k-means algorithm is used to cluster the training data from which the health state label/class is identified. The selected features are used as inputs to machine learning algorithms and the health state as the target. The result from the training process is a health state classification model which can be used online or with test data of similar systems. Two machine learning algorithms were evaluated based on this approach:

  1. support vector machines (SVM) with optimally tuned parameters
  2. random forests (RF)

  The methods were evaluated with run-to-failure data of ball bearings and milling machine cutting tools with both methods yielding a mean absolute percentage error (MAPE) of less than 20 and with majority of the predictions falling within acceptable error. SVM produces the best overall performance.

- If a system is operating continuously at constant or near constant operating conditions, then machine learning algorithms can be used to map extracted features to a health index that represents the fractional remaining useful life of the system. This involves function approximation and the utilization of machine learning algorithms for regression. In this work, three state of the art algorithms for regression: extreme learning machines (ELM), random forests (RF) and support vector machines (SVM) are evaluated with the ball bearing and milling machine cutting tool data. In both types of data, the algorithms estimate the *RUL* with a MAPE less than

20, which is a good reference for prognostics. However, ELM method has better predictions, with the majority of the predictions being early predictions.

- When a system has a measurable health index in addition to condition monitoring data, then machine learning algorithms can be employed to map the extracted features from condition monitoring data to the health index. The approximated health index is then used with either similarity based approach or particle filter to estimate the RUL. Similarly, this involves function approximation and requires regression machine learning algorithms. ELM, RF and SVM were evaluated with milling machine data and found to yield MAPE of less than 20. The performance of this approach compares very well with health-state based methods but has the advantage of being less complex since the selection of health states is not required.
- In some systems, a performance index which is usually measured for control purposes, for instance voltage measured in fuel cells for purposes of regulating power usage, could also be used as condition monitoring data. In systems or components that have relative motion and hence wear is a failure mode, temperature could be used as a condition monitoring data. In this case, the data is usually single valued and can be used with a method for propagating the health index from the current value to a predetermined threshold, from which the *RUL* is determined. In this work, particle filter approach was used for this approach. The method was adapted to cater for changing degradation rates and self-healing effects observed in some systems such as fuel cells. The approach was evaluated with run-to-failure data of PEM fuel cells and temperature measurements for ball bearings. The approach was found to yield high accuracies when used with voltage measurements of fuel cells. The performance with temperature measurements of ball bearings was moderate. This is because temperature is very sensitive of the surroundings and may require an additional sensor for reference.
- For systems that undergo wear and for which temperature sensors can be incorporated to track temperature rise from dissipated frictional heat, then a model-based approach best suited. This approach correlates wear with heat dissipated due to friction from which a health index is computed. This approach is particularly very accurate towards the end of life of the system, when maintenance decision is required. This approach is also less costly in terms of instrumentation and computational requirements.
- In order to incorporate the benefits of individual methods, then an ensemble of the methods based on a weighted average can be constructed. Various combination of the methods can be evaluated in order to obtain the best combination for a given approach. The ensemble proves to be a robust approach for prognostics and outperforms all individual algorithms.

To demonstrate the applicability of some of the methods for online prognosis, laboratory experiments on degradation of piezoelectric bimorph benders were conducted. Electric current, normally used for control purposes, was used as the condition monitoring data while displacement was measured as a health index for ground truth degradation. The use of machine learning algorithms to map the extracted features to the health index was evaluated. Random forests machine learning algorithm was found to yield the best results in predicting the health index. To estimate the *RUL*, particle filter method and similarity based methods were evaluated. Particle filter approach was found to yield the

best performance of a mean MAPE less than 10 and an overall performance of 0.64 out of the possible 1. However, the approach has higher computational requirements in terms of computation time and memory and may not be applicable for systems with short lifetimes.

6.2 Future Work

Although a number of prognostic methods have been developed, evaluated and validated using various data sets, a number of research issues have been identified.

- Some technical systems are operated under non-stationary conditions such as speed and load, which may vary with time either in a periodic or stochastic manner. Most condition monitoring data do not show any observable change with degradation as a result of the varying operating conditions and as such it would be very difficult to conduct diagnosis and prognosis using the methods already proposed here and in literature. In addition, without knowledge of future operating conditions, it becomes a challenge to quantify the remaining useful lifetime. An approach for incorporating the non-stationary operating conditions so that a degradation trend is clearly captured is therefore necessary.
- Most mechatronic systems are very complex and consist of many interacting components. In order to monitor all critical components, many sensors are required, thereby increasing the cost of the condition monitoring system. In order to reduce the number of sensors and consequently the cost, strategic sensor placement should be evaluated. This can be done by combining model-based and data driven methods to identify critical points where sensors can be mounted to provide the maximum information about a system.
- The prognostic methods proposed have been validated using run-to-failure laboratory experiments. A way of adapting the methods with real-life equipment without having to run the equipment to failure is necessary. This could be done by correlating the real-life sensor data with the experimental data and determining transfer/mapping factors that could be used with the methods for diagnostic and prognostic purposes.
- There is need for prognostic methods for systems without run-to-failure condition monitoring data but for which prognostics would improve reliability and safety, for instance wind turbine components. In most cases only the baseline data is available. Prognostic methods that detect and quantify deviation from baseline data can be used together with predefined thresholds to estimate when the system is likely to fail.

References

[1] V. Jurca, T. Hladik, and Z. Ales, "Optimization of preventive maintenance intervals," *Maintenance and Reliability*, vol. 3, pp. 41–44, 2008.

[2] D. Nguyen and M. Bagajewicz, "Optimization of preventive maintenance scheduling in processing plants," in *18th European Symposium on Computer Aided Process Engineering - ESCAPE 18*, 2008.

[3] W. Wang, "An intelligent system for machinery condition monitoring," *IEEE Transactions on Fuzzy Systems*, vol. 16, pp. 110–122, 2008.

[4] K. Goebel, A. Saxena, J. Celaya, and I. Roychoudhury, "Tutorial: Introduction to prognostics," in *First European Conference of the Prognostics and Health Management*, 2012.

[5] J. L. Godwin and P. Matthews, "Classification and detection of wind turbine pitch faults through SCADA data analysis," *International Journal of Progno*, vol. 16, 2013.

[6] C. Sondermann-Wölke and W. Sextro, "Integration of condition monitoring in self-optimizing function modules applied to the active railway guidance module," *International Journal on Advances in Intelligent Systems*, vol. 3, pp. 65–74, 2010.

[7] T. Meyer and W. Sextro, "Closed-loop control system for the reliability of intelligent mechatronic systems," in *Second European Conference of the Prognostics and Health Management Society*, 2014.

[8] G. Vachtsevanos, F. Lewis, M. Roemer, A. Hess, and B. Wu, *Intelligent Fault Diagnosis and Prognosis For Engineering Systems.* John Wiley & Sons, Inc, 2006.

[9] E. D. Otilia, G. Rafael, F. D., M. Eugenia, and Z. Noureddine, "Review of prognostic problem in condition-based maintenance," in *European Control Conference, Budapest: Hungary*, 2009.

[10] D. Tobon-Mejia, K. Medjaher, and N. Zerhouni, "The ISO 13381-1 standard's failure prognostics process through an example," in *IEEE Prognostics & System Health Management Conference, PHM'2010*, 2010.

[11] A. K. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical Systems and Signal Processing*, vol. 20, pp. 1483–1510, 2006.

[12] D. Tobon-Mejia, K. Medjaher, N. Zerhouni, and G. Tripot, "Hidden markov models for failure diagnostic and prognostic," in *Prognostics and System Health Management Conference, PHM"11*, 2011.

[13] J. K. Kimotho, T. Meyer, and W. Sextro, "PEM fuel cell prognostics using particle filter with model parameter adaptation," in *Prognostics and Health Management (PHM), 2014 IEEE Conference on*, 2014.

[14] L. Wiliem, D. Hargreaves, R. F. Stapelberg, and P. K. D. V. Yarlagadda, "Development of real-time data filtering for SCADA system," *Journal of Achievements in Materials and Manufacturing Engineering*, vol. 21, no. 2, pp. 9–92, 2007.

[15] C. S. Byington and A. K. Garga, *Data Fusion for Developing Predictive Diagnostics for Electromechanical Systems.* Chapter 23 of CRC Press Handbook of Multisensor Data Fusion, 2^{nd} ed., 2001.

[16] R. Isermann, "Model-based fault-detection and diagnosis - status and applications," *Annual Reviews in Control*, vol. 29, pp. 71–85, 2005.

[17] M. Nyberg, *Model Based Fault Diagnosis Methods, Theory, and Automotive Engine Applications*. PhD thesis, Linköping University, Sweden, 1999.

[18] S. Anwar and L. Chen, "An analytical redundancy-based fault detection and isolation algorithm for a road-wheel control subsystem in a steer-by-wire system," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 5, pp. 2859–2869, 2007.

[19] A. D. Pouliezos and G. S. Stavrakakis, "Real time fault monitoring of industrial processes," in *International Series on Microprocessor-Based and Intelligent Systems Engineering*, Springer Netherlands, 1994.

[20] A. Varga and D. Ossmann, "LPV model-based robust diagnosis of flight actuator faults," *Control Engineering Practice*, vol. 31, pp. 135–147, 2014.

[21] S. Narasimhan, I. Roychoudhury, E. Balaban, and A. Saxena, "Combining model-based and feature-driven diagnosis approaches - a case study on electromechanical actuators," in *21st International Workshop on Principles of Diagnosis*, 2010.

[22] C. Byington, M. Watson, D. Edwards, and P. Stoelting, "A model-based approach to prognostics and health management for flight control actuators," in *IEEE Aerospace Conference*, 2004.

[23] M. Daigle, M. Foygel, and V. Smelyanskiy, "Model-based diagnostics for propellant loading systems," in *2011 IEEE Aerospace Conference*, 2011.

[24] F. Caccavale, P. Cilibrizzi, F. Pierri, and L. Villani, "Actuators fault diagnosis for robot manipulators with uncertain model," *Control Engineering Practice*, vol. 17, pp. 146–157, 2009.

[25] G. K. Fourlas, S. Karkanis, G. C. Karras, and K. J. Kyriakopoulos, "Model based actuator fault diagnosis for a mobile robot," in *IEEE International Conference on Industrial Technology (ICIT)*, 2014.

[26] T. Hsiao and M.-C. Weng, "A hierarchical multiple-model approach for detection and isolation of robotic actuator faults," in *2010 American Control Conference*, 2010.

[27] A. Wahrburg, B. Matthias, and H. Ding, "Cartesian contact force estimation for robotic manipulators - a fault isolation perspective," in *9th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, 2015.

[28] L. Xiangyang and C. Wanqiang, "Rolling bearing fault diagnosis based on physical model and one-class support vector machine," *ISRN Mechanical Engineering*, 2014.

[29] V. T. Do and U.-P. Chong, "Signal model-based fault detection and diagnosis for induction motors using features of vibration signal in two dimension domain," *Journal of Mechanical Engineering*, vol. 57, pp. 655–666, 2011.

[30] W. Wang and A. K. Wong, "A model-based gear diagnostic technique," tech. rep., Airframes and Engines Division Aeronautical and Maritime Research Laboratory, 2000.

[31] J. Park, J. M. Ha, B. D. Youn, S. H. Leem, J.-H. Choi, and N. H. Kim, "Model-based fault diagnosis of a planetary gear using transmission error," in *Annual Conference of the Prognostics and Health Management Society*, 2014.

[32] J. Treetrong, "Electric motor fault diagnosis based on parameter estimation approach using genetic algorithm," in *Proceedings of the International Conference of Engineers and Computer Scientists*, 2010.

[33] Z. Peng and F. Chu, "Application of the wavelet transform in machine condition monitoring and fault diagnostics: a review with bibliography," *Mechanical Systems and Signal Processing*, vol. 18, pp. 199–221, 2004.

[34] H. Zhang, H. Zhou, X. Shi, J. Huang, J. Sun, and L. Huang, "Research on rolling bearing fault diagnosis with adaptive frequency selection based on LabVIEW," *International Journal of Control and Automation*, vol. 3, pp. 93–100, 2014.

[35] C. Yi, J. Lin, W. Zhang, and J. Ding, "Faults diagnostics of railway axle bearings based on IMF's confidence index algorithm for ensemble EMD," *Sensors*, vol. 15, pp. 10991–11011, 2015.

[36] F. Pan, S. R. Qin, and L. Bo, "Development of diagnosis system for rolling bearings faults based on virtual instrument technology," *Journal of Physics*, vol. 48, pp. 467–473, 2006.

[37] N. Sawalhi, R. Randall, and H. Endo, "The enhancement of fault detection and diagnosis in rolling element bearings using minimum entropy deconvolution combined with spectral kurtosis," *Mechanical Systems and Signal Processing*, vol. 21, no. 6, pp. 2616–2633, 2007.

[38] S. Guoji, S. McLaughlin, X. Yongcheng, and P. White, "Theoretical and experimental analysis of bispectrum of vibration signals for fault diagnosis of gears," *Mechanical Systems and Signal Processing*, vol. 43, pp. 76–89, 2014.

[39] W. T. Thomson and R. J. Gilmore, "Motor current signature analysis to detect faults in induction motor ddrive - fundamentals, data interpretation and industrial case histories," in *Proceedings of the Thirthy-Second Turbomachinery Symposium*, 2003.

[40] C. Lessmeier, C. Piantsop Mbo'o, C. I., D. Zimmer, and H. K., "Untersuchung von bauteilschäden elektrischer antriebsstränge im belastungsprüfstand mittels statorstromanalyse," in *Aachener Kolloquium für Instanthaltung, Diagnose und Anlagenüberwachung*, 2012.

[41] A. Alwodai, T. Wang, Z. Chen, F. Gu, R. Cattley, and A. Ball, "A study of motor bearing fault diagnosis using modulation signal bispectrum analysis of motor current signals," *Journal of Signal and Information Processing*, vol. 4, pp. 72–79, 2013.

[42] S. Lee and S. Kim, "Internal combustion engine sound-based fault detection and diagnosis using adaptive line enhancers," *Journal of Automobile Engineering, Proc. IMechE*, vol. 222, pp. 593–605, 2008.

[43] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[44] A. Datta, C. Mavroidis, and M. Hosek, "A role of unsupervised clustering for intelligent fault diagnosis," in *ASME International Mechanical Engineering Congress and Exposition*, 2007.

[45] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, ch. Chapter 8: Cluster Analysis: Basic Concepts and Algorithms, pp. 487–568. Pearson, 2005.

[46] X. Liu, L. Ma, S. Zhang, and J. Mathew, "Using fuzzy c-means and fuzzy integrals for machinery fault diagnosis," in *In Proceedings International Conference on Condition Monitoring, Cambridge, England*, 2005.

[47] G. Yu, C. Li, and S. Kamarthi, "A cluster-based wavelet feature extraction method for machine fault diagnosis," *Applied Mechanics and Materials*, vol. 10-12, pp. 548–552, 2008.

[48] A. Widodo, D. Satrijo, M. Huda, G.-M. Lim, and B.-S. Yang, "Application of self organizing map for intelligent machine fault diagnostics based on infrared thermography images," in *Sixth International Conference on Bio-Inspired Computing: Theories and Applications*, 2011.

[49] D. Solomatine, L. See, and R. Abrahart, *Practical Hydroinformatics Computational Intelligence and Technological Developments in Water Applications*, ch. Data-Driven Modelling: Concepts, Approaches and Experiences, pp. 17–30. Springer Link, 2008.

[50] D. Bodden, N. Clements, B. Schley, and G. Jenney, "Seeded failure testing and analysis of an electromechanical actuator," in *IEEE Aerospace Conference*, 2007.

[51] Y. Yu, Y. Dejie, and C. Junsheng, "A roller bearing fault diagnosis method based on EMD energy entropy and ANN," *Journal of Sound and Vibration*, vol. 294, pp. 269–277, 2006.

[52] B. Samanta, K. R. Al-Balushi, and S. A. Al-Araimi, "Bearing fault detection using articial neural networks and genetic algorithm," *EURASIP Journal on Applied Signal Processing*, vol. 3, pp. 366–367, 2004.

[53] B. Li, G. Goddu, and M.-Y. Chow, "Detection of common motor bearing faults using frequency domain vibration signals and a neural network based approach," in *Proceedings of the American Control Conference*, 1998.

[54] X. Chen, J. Zhou, J. Xiao, X. Zhang, H. Xiao, W. Zhu, and W. Fu, "Fault diagnosis based on dependent feature vector and probability neural network for rolling element bearings," *Applied Mathematics and Computation*, vol. 247, pp. 835–847, 2014.

[55] M. Unal, M. Onat, M. Demetgul, and H. Kucuk, "Fault diagnosis of rolling bearings using a genetic algorithm optimized neural network," *Measurement*, vol. 58, pp. 187–196, 2014.

[56] K. Zhang, Y. Li, P. Scarf, and A. Ball, "Feature selection for high-dimensional machinery fault diagnosis data using multiple models and radial basis function networks," *Neurocomputing*, vol. 74, pp. 2941–2952, 2011.

[57] M. Ahmed, F. Gu, and A. Ball, "Fault classification using an artificial neural network based on vibrations from a reciprocating compressor," in *Computing and Engineering Researchers' Conference, University of Huddersfield*, 2010.

[58] Z. Du, B. Fan, X. Jin, and J. Chi, "Fault detection and diagnosis for buildings and HVAC systems using combined neural networks and subtractive clustering analysis," *Building and Environment*, vol. 73, pp. 1–11, 2014.

[59] J.-D. Wu and C.-H. Liu, "An expert system for fault diagnosis in internal combustion engines using wavelet packet transform and neural network," *Expert Systems with Applications*, vol. 36, pp. 4278–4286, 2009.

[60] M. Sangha, J. Gomm, D. Yu, and G. Page, "Fault detection and identication of automotive engines using neural networks," tech. rep., Control Systems Research Group, School of Engineering, Liverpool John Moores University, 2005.

[61] J. Rafiee, F. Arvani, A. Harifi, and M. Sadeghi, "Intelligent condition monitoring of a gearbox using artificial neural network," *Mechanical Systems and Signal Processing*, vol. 21, pp. 1746–1754, 2007.

[62] N. Saravanan, V. K. Siddabattuni, and K. I. Ramachandran, "Fault diagnosis of spur bevel gear box using artificial neural network (ANN), and proximal support vector machine (PSVM)," *Applied Soft Computing*, vol. 10, pp. 344–360, 2010.

[63] J. Sun, Y. Chai, C. Sub, Z. Zhua, and X. Luo, "BLDC motor speed control system fault diagnosis based on LRGF neural network and adaptive lifting scheme," *Applied Soft Computing*, vol. 14, pp. 609–622, 2014.

[64] S. Moosavi, A. Djerdi, Y. Ait-Amirat, and D. Khaburi, "Ann based fault diagnosis of permanent magnet synchronous motor under stator winding shorted turn," *Electric Power Systems Research*, vol. 125, pp. 67–82, 2015.

[65] C. Xu, H. Zhang, D. Peng, Y. Yu, C. Xu, and H. Zhang, "Study of fault diagnosis of integrate of d-s evidence theory based on neural network for turbine," *Energy Procedia*, vol. 16, pp. 2027–2032, 2012.

[66] I. Eski, S. Erkaya, S. Savas, and S. Yildirim, "Fault detection on robot manipulators using artificial neural networks," *Robotics and Computer-Integrated Manufacturing*, vol. 27, pp. 115–123, 2011.

[67] X. Zhang, W. Chen, B. Wang, and X. Chen, "Intelligent fault diagnosis of rotating machinery using support vector machine with ant colony algorithm for synchronous feature selection and parameter optimization," *Neurocomputing*, vol. 167, pp. 260–279, 2015.

[68] F. Chen, B. Tang, T. Song, and L. Li, "Multi-fault diagnosis study on roller bearing based on multi-kernel support vector machine with chaotic particle swarm optimization," *Measurement*, vol. 47, pp. 576–590, 2014.

[69] X. Zhang and J. Zhou, "Multi-fault diagnosis for rolling element bearings based on ensemble empirical mode decomposition and optimized support vector machines," *Mechanical Systems and Signal Processing*, vol. 41, pp. 127–140, 2013.

[70] X. Zhang, Y. Liang, J. Zhou, and Y. Zang, "A novel bearing fault diagnosis model integrated permutation entropy, ensemble empirical mode decomposition and optimized svm," *Measurement*, vol. 69, pp. 164–179, 2015.

[71] L. Gao, Z. Ren, W. Tang, H. Wang, and P. Chen, "Intelligent gearbox diagnosis methods based on svm, wavelet lifting and rbr," *Sensors*, vol. 10, pp. 4602–4621, 2010.

[72] T. Praveenkumara, M. Saimurugan, P. Krishnakumarc, and K. Ramachandrand, "Fault diagnosis of automobile gearbox based on machine learning techniques," *Procedia Engineering*, vol. 97, pp. 2092–2098, 2014.

[73] F. Chen, B. Tang, and R. Chen, "A novel fault diagnosis model for gearbox based on wavelet support vector machine with immune genetic algorithm," *Measurement*, vol. 46, pp. 220–232, 2013.

[74] L. Ma, G. Chen, and H. Wang, "Gearbox fault diagnosis method based on svm trained by improved sfla," *Communications in Computer and Information Science*, vol. 462, pp. 257–263, 2014.

[75] H. Xue, H. Wang, P. Chen, K. Li, and L. Song, "Automatic diagnosis method for structural fault of rotating machinery based on distinctive frequency components and support vector machines under varied operating conditions," *Neurocomputing*, vol. 116, pp. 326–335, 2013.

[76] Z. Yunlong and Z. Peng, "Vibration fault diagnosis method of centrifugal pump based on emd complexity feature and least square support vector machine," *Energy Procedia*, vol. 17, pp. 939–945, 2012.

[77] A. Widodo and B.-S. Yang, "Application of nonlinear feature extraction and support vector machines for fault diagnosis of induction motors," *Expert Systems with Applications*, vol. 33, pp. 241–250, 2007.

[78] V. T. Tran and B.-S. Yang, "Machine fault diagnosis and condition prognosis using classification and regression trees and neuro-fuzzy inference systems," *Control and Cybernetics*, vol. 39, pp. 22–54, 2010.

[79] V. Sugumaran, V. Muralidharan, and K. I. Ramachandran, "Feature selection using decision tree and classification through proximal support vector machine for fault diagnostics of roller bearings," *Mechanical Systems and Signal Processing*, vol. 21, pp. 930–942, 2007.

[80] M. Amarnath, V. Sugumaran, and H. Kumar, "Exploiting sound signals for fault diagnosis of bearings using decision tree," *Measurement*, vol. 46, pp. 1250–1256, 2013.

[81] V. T. Tran, B.-S. Yang, M.-S. Oh, and A. C. C. Tan, "Fault diagnosis of induction motor based on decision trees and adaptive neuro-fuzzy inference," *Expert Systems with Applications*, vol. 36, pp. 1840–1849, 2009.

[82] V. T. Tran, B.-S. Yang, and M.-S. Oh, "Fault diagnosis of induction motors using decision trees," in *KSNVE Annual Autumn Conference*, 2006.

[83] H. Yang, J. Mathew, and L. Ma, "Intelligent diagnosis of rotating machinery faults - a review," in *3rd Asia-Pacific Conference on Systems Integrity and Maintenance*, 2002.

[84] Y. Lei, Z. He, Y. Zi, and Q. Hu, "Fault diagnosis of rotating machinery based on multiple ANFIS combination with GAs," *Mechanical Systems and Signal Processing*, vol. 21, pp. 2280–2294, 2007.

[85] L. Zhang, G. Xiong, H. Liu, H. Zou, and W. Guo, "Bearing fault diagnosis using multi-scale entropy and adaptive neuro-fuzzy inference," *Expert Systems with Applications*, vol. 37, pp. 6077–6085, 2010.

[86] M. Celik and R. Bayir, "Fault detection in internal combustion engines using fuzzy logic," *Journal of Automobile Engineering, Proc. IMechE*, vol. 221, pp. 579–587, 2007.

[87] P. Subbaraj and B. Kannapiran, "Fault detection and diagnosis of pneumatic valve using adaptiveneuro-fuzzy inference system approach," *Applied Soft Computing*, vol. 19, pp. 362–371, 2014.

[88] X.-H. Zhang and J.-S. Kang, "Hidden markov models in bearing fault diagnosis and prognosis," in *Computational Intelligence and Natural Computing Proceedings (CINC)*, 2010.

[89] H. Ocak and K. A. Loparo, "A new bearing fault detection and diagnosis scheme based on hidden markov modeling of vibration signals," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2001.

[90] V. Purushotham, S. Narayanan, and S. A. Prasad, "Multi-fault diagnosis of rolling bearing elements using wavelet analysis and hidden markov model based fault recognition," *NDT & E International*, vol. 38, pp. 654–664, 2005.

[91] Z. Li, Z. Wub, Y. Hea, and C. Fulei, "Hidden markov model-based fault diagnostics method in speed-up and speed-down process for rotating machinery," *Mechanical Systems and Signal Processing*, vol. 19, pp. 229–239, 2005.

[92] J.-S. Kang, X.-H. Zhang, and Y.-J. Wang, "Continuous hidden markov model based gear fault diagnosis and incipient fault detection," in *2011 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE)*, 2011.

[93] O. Geramifard, *Hidden Markov Model-Based Methods in Condition Monitoring of Machinery Systems*. PhD thesis, National University of Singapore, 2013.

[94] R. Li, D. He, and J. Zhu, "Investigation on full ceramic bearing fault diagnostics using vibration and ae sensors," in *2012 IEEE Conference on Prognostics and Health Management (PHM)*, 2012.

[95] J. Tian, M. H. Azarian, and M. Pecht, "Anomaly detection using self-organizing maps-based k-nearest neighbor algorithm," in *European Conference of the Prognostic and Health Management Society*, 2014.

[96] L. Jiang, T. Shi, and J. Xuan, "An intelligent fault diagnosis method of rolling bearings based on regularized kernel marginal fisher analysis," in *25th International Congress on Condition Monitoring and Diagnostic Engineering*, 2012.

[97] M. Safizadeh and S. Latifi, "Using multi-sensor data fusion for vibration fault diagnosis of rolling element bearings by accelerometer and load cell," *Information Fusion*, vol. 18, pp. 1–8, 2014.

[98] A. Bouguerne, A. Lebaroud, A. Medoued, and A. Boukadoum, "Classification of induction machine faults by k-nearest neighbor," in *7th International Conference on Electrical and Electronics Engineering (ELECO)*, 2011.

[99] R. Li, D. He, and E. Bechhoefer, "Investigation on fault detection for split torque gearbox using acoustic emission and vibration signals," in *Annual Conference of the Prognostics and Health Management Society*, 2009.

[100] L. Peng, B. Jiao, H. Liu, and T. Zhang, "The fault diagnosis of wind turbine gearbox based on improved KNN," *Advances in Energy Engineering (AEE)*, vol. 3, pp. 8–13, 2015.

[101] D.-F. Lin, P.-H. Chen, and M. Williams, "Measurement and analysis of current signals for gearbox fault recognition of wind turbine," *Measuremnt Science Review*, vol. 13, no. 2, pp. 89–93, 2013.

[102] C.-Y. Lee, C.-C. Kuo, R. Liu, I.-H. Tseng, and L.-C. Chang, "Detection of gearbox lubrication using PSO-based WKNN ," *Measurement Science Review*, vol. 13, no. 3, pp. 108–114, 2013.

[103] D. M. Himmelblau, "Applications of artificial neural networks in chemical engineering," *Korean Journal of Chemical Engineering*, vol. 17, no. 4, pp. 373–392, 2000.

[104] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[105] J. Hang, J. Zhang, and M. Cheng, "Application of multi-class fuzzy support vector machine classifier for fault diagnosis of wind turbine," *Fuzzy Sets and Systems*, 2015.

[106] J. B. Coble, *Merging Data Sources to Predict Remaining Useful Life - An Automated Method to Identify Prognostic Parameters*. PhD thesis, University of Tennessee, Knoxville, 2010.

[107] M. Petch and S. Kumar, "Data analysis approach for system reliability, diagnostics and prognostics," in *In Pan Pacific Microelectronics Symposium*, 2008.

[108] J. Yang, X. Ma, and Z. Yang, "Life prediction of scraper conveyor based on weibull distribution fault diagnosis theory," in *International Conference of Electrical, Automation and Mechanical Engineering (EAME 2015)*, 2015.

[109] Z. Chen and S. Zheng, "Lifetime distribution based degradation analysis," *IEEE Transactions on Reliability*, vol. 54, no. 1, pp. 3–10, 2005.

[110] A. S. Heng, *Intelligent Prognostics of Machinery Health Utilizing Suspended Condition Monitoring Data*. PhD thesis, Queensland University of Technology, Australia, 2009.

[111] T.-H. Fan and T.-M. Hsu, "Accelerated life tests of a series system with masked interval data under exponential lifetime distributions," *IEEE Transactions on Reliability*, vol. 61, no. 3, pp. 798–808, 2012.

[112] M. Daigle and K. Goebel, "A model-based prognostics approach applied to pneumatic valves," *International Journal of Prognostics and Health Management*, vol. 2, no. 2, 2011.

[113] C. Sankavaram, B. Pattipati, A. Kodali, K. Pattipati, M. Azam, S. Kumar, and M. Pech, "Model-based and data-driven prognosis of automotive and electronic systems," in *5th Annual IEEE Conference on Automation Science and Engineering Bangalore, India*, 2009.

[114] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle ffilter for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[115] C. S. Byington, M. J. Roemer, and M. J. Watson, "Prognostic enhancements to diagnostic systems (peds) applied to shipboard power generation systems," *Proceedings of ASME Turbo Expo 2004*, vol. 2, pp. 825–833, 2004.

[116] C. S. Kulkarni, *A Physics-Based Degradation Modeling Framework for Diagnostic and Prognostic Studies in Electrolytic Capacitors*. PhD thesis, Graduate School of Vanderbilt University, Nashville, Tennessee, 2013.

[117] R. Patrick-Aldaco, *A Model Based Framework for Fault Diagnosis and Prognosis of Dynamical Systems with an Application to Helicopter Transmissions*. PhD thesis, Georgia Institute of Technology, 2007.

[118] N. Bolander, H. Qiu, N. Eklund, E. Hindle, and T. Rosenfeld, "Physics-based remaining useful life prediction for aircraft engine bearing prognosis," in *Annual Conference of Prognostics and Health Management Society*, 2009.

[119] J. Luo, M. Namburu, K. Pattipati, L. Qiao, M. Kawamoto, and S. Chigusa, "Model-based prognostic techniques," in *Proceedings of the IEEE Systems Readiness Technology Conference*, 2003.

[120] E. Sutrisno, H. Oh, A. S. S. Vasan, and M. Pecht, "Estimation of remaining useful life of ball bearings using data driven methodologies," *2012 IEEE Conference on Prognostics and Health Management*, 2012.

[121] Z. Tian, L. Wong, and S. Nima, "A neural network approach for remaining useful life prediction utilizing both failure and suspension histories," *Mechanical Systems and Signal Processing*, vol. 24, pp. 1542–1555, 2010.

[122] W. Caesarendra, A. Widodo, and B.-S. Yang, "Combination of probability approach and support vector machine towards machine health prognostics," *Probabilistic Engineering Mechanics*, vol. 26, pp. 165–173, 2011.

[123] H.-E. Kim, A. C. C. Tan, J. Mathew, and B.-K. Choi, "Bearing fault prognosis based on health state probability estimation," *Expert Systems with Applications*, vol. 39, pp. 5200–5213, 2012.

[124] H. Ocak, K. A. Loparo, and F. M. Discenzo, "Online tracking of bearing wear using wavelet packet decomposition and probabilistic modeling: A method for bearing prognostics," *Journal of Sound and Vibration*, vol. 302, pp. 951–961, 2007.

[125] R. B. Chinnam and P. Baruah, "Autonomous diagnostics and prognostics in machining processes through competitive learning-driven HMM-based clustering," *International Journal of Production Research*, vol. 47, no. 23, pp. 6739–6758, 2009.

[126] F. Camci and R. B. Chinnam, "Health-state estimation and prognostics in machining processes," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 3, pp. 581–597, 2010.

[127] P. Tamilselvan, Y. Wang, and P. Wang, "Deep belief network based state classification for structural health diagnosis," in *IEEE Aerospace Conference*, 2012.

[128] S. Pal, P. S. Heyns, B. H. Freyer, N. J. Theron, and S. K. Pal, "Tool wear monitoring and selection of optimum cutting conditions with progressive tool wear effect and input uncertainties," *Journal of Intelligent Manufacturing*, vol. 22, pp. 491–504, 2011.

[129] F. Cus and U. Zuperl, "Real-time cutting tool condition monitoring in milling," *Journal of Mechanical Engineering*, vol. 57, pp. 142–150, 2011.

[130] L. Tang, E. Hettler, B. Zhang, and J. DeCastro, "A testest for real-time autonomous vehicle PHM and contigency management applications," in *Annual Conference of the Prognostics and Health Management Society*, 2011.

[131] B. Saha, K. Goebel, and J. Christopherson, "Comparison of prognostic algorithms for estimating remaining useful life of batteries," *Transactions of the Institute of Measurement and Control*, vol. 31, no. 3-4, pp. 293–308, 2009.

[132] D. He, E. Bechhoefer, P. Dempsey, and J. Ma, "An integrated approach for gear health prognostics," in *Proceedings of the 2012 AHS Forum*, 2012.

[133] A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel, "Metrics for offline evaluation of prognostic performance," *International Journal of Prognostics and Health Management*, vol. 001, 2010.

[134] S. Uckun, K. Goebel, and P. J. Lucas, "Standardizing research methods for prognostics," in *International Conference on Prognostics and Health Management*, 2008.

[135] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, and M. Schwabacher, "Metrics for evaluating performance of prognostic techniques," in *International Conference on Prognostics and Health Management*, 2008.

[136] Y. Zhou, J. Bo, Z. Jie, and G. Mingwei, "Performance metrics assessment method on aircraft prognostics and health management," in *Fifth Conference on Measuring Technology and Mechatronics Automation*, 2013.

[137] K. Goebel, A. Saxena, S. Saha, B. Saha, and J. Celaya, "Machine learning and knowledge discovery for engineering systems health management," in *Prognostic Performance Metrics*, ch. Chapter 5, pp. 147–178, Chapman & Hall/CRC Press, 2012.

[138] A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel, "Evaluating algorithm performance metrics tailored for prognostics," *IEEEAC*, 2008.

[139] S. Sankararaman, "Significance, interpretation, and quantification of uncertainty in prognostics and remaining useful life prediction," *Mechanical Systems and Signal Processing*, vol. 52-53, pp. 228–247, 2015.

[140] B. Sun, T. Liu, S. Liu, and Q. Feng, "A cognitive framework for analysis and treatment of uncertainty in prognostics," *Chemical Engineering Transactions*, vol. 33, pp. 187–192, 2013.

[141] J. Gu, D. Barker, and M. Petch, "Uncertainty assessment of prognostics of electronics subject to random vibration," in *AAAI Fall Symposium on Artificial Intelligence for Prognostics*, 2007.

[142] D. Liu, , W. Xie, H. Liao, and Y. Peng, "An integrated probabilistic approach to lithium-ion battery remaining useful life estimation," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 3, pp. 660–670, 2015.

[143] F. Zhao, Z. Tian, and Y. Zheng, "Uncertainty quantification in gear remaining useful life prediction through integrated prognostics method," *IEEE Transactions on Reliability*, vol. 62, no. 1, pp. 146–159, 2013.

[144] S. Sankararaman and K. Goebel, "Uncertainty in prognostics and health management: An overview," in *European Conference of the Prognostics and Health Management Society*, 2014.

[145] C. Walck, *Hand-book on Statistical Distributions for Experimentalists*. University of Stockholm, 2007.

[146] J. G. Ferreira, "Applying intelligent methods for condition assessment and diagnosis of electrical drive system," in *Advance Diagnosis of Electro-Mechanical Systems*, 2012.

[147] T. Harris and M. Kotzalas, *Advanced Concepts of Bearing Technology: Rolling Bearing Analysis*. CRC Press, 2006.

[148] P. Nectoux, K. Medjaher, E. Ramasso, B. Morello, and N. Zerhouni, "Pronostia: An experimental platform for bearing accelerated degradation tests," *IEEE International Conference on Prognostics and Health Management, Denver, CO, USA*, 2012.

[149] M. Jouin, R. Gouriveau, D. Hissel, M.-C. Pera, and N. Zerhouni, "Prognostics of pem fuel cell in a particle filtering framework," *International Journal of Hydrogen Energy*, vol. 39, pp. 481–494, 2014.

[150] R. Gouriveau, M. Hilairet, D. Hissel, S. Jemei, M. Jouin, E. Lechartier, S. Morando, E. Pahon, M.-C. Pera, and N. Zerhouni, "IEEE PHM 2014 data challenge: Oultine, experiments, scoring of results, winners," tech. rep., IEEE 2014 PHM Challenge, 2014.

[151] G. B. Kliman, W. J. Premerlani, B. Yazici, R. A. Koegl, and J. Mazereeuw, "Sensorless, online motor diagnostics," *IEEE Computer Applications in Power*, vol. 10, no. 2, pp. 39–43, 1997.

[152] S. Villwock, *Identifikationsmethoden für die automatisierte Inbetriebnahme und Zustandsüberwachung elektrischer Antriebe*. PhD thesis, University of Siegen, 2007.

[153] T. Herold, C. P. Mbo'o, and K. Hameyer, "Evaluation of the use of an electrical drive as a sensor for the detection of bearing damage," in *AIA-DAGA Conference on Acoustics*, 2013.

[154] D. Zhen, T. Wang, F. Gu, and A. D. Ball, "Fault diagnosis of motor ddrive using stator current signal analysis based on dynamic time warping," *Mechanical Systems and Signal Processing*, vol. 34, no. 1-2, pp. 191–202, 2013.

[155] C. Lessmeier, J. K. Kimotho, D. Zimmer, and W. Sextro, "Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification," in *European Conference of the Prognostics and Health Management Society*, 2016.

[156] P. Bornmann, T. Hemsel, W. Sextro, G. Memoli, M. Hodnett, and B. Zeqiri, "Self-sensing ultrasound transducer for cavitation detection," *Ultrasonics Symposium (IUS), 2014 IEEE International*, pp. 663–666, 2014.

[157] A. Saxena, K. Goebel, C. Larrosa, V. Janapati, S. Roy, and F. Chang, "Accelerated aging experiments for prognostics of damage growth in composite materials," in *The 8th International Workshop on Structural Health Monitoring*, 2011.

[158] B. Zhang, C. Sconyers, M. Orchard, R. Patrick, and G. Vachtsevanos, "Fault progression modeling: An application to bearing diagnosis and prognosis," in *2010 American Control Conference*, 2010.

[159] J.-H. Ahn, D.-H. Kwak, and B.-H. Koh, "Fault detection of a roller-bearing system through the emd of a wavelet denoised signal," *Sensors*, vol. 14, pp. 15022–15038, 2014.

[160] Y. Wang, W. Wu, Q. Zhu, and G. Shen, "Discrete wavelet transfom for nonstationary signal processing," in *Discrete Wavelet Transform - Theory and Applications* (J. Olkkonen, ed.), InTech, 2011.

[161] E. Bechhoefer, "Condition based maintenance fault database for testing of diagnostic and prognostics algorithms," tech. rep., Mechanical Failure Prevention Technology (MFPT), 2013.

[162] E. Bechhoefer, "MFPT bearing dataset." http://data-acoustics.com/measurements/bearing-faults/bearing-2/. Accessed: 2015-11-17.

[163] F. d. Lorenzo and M. Calabro, "Kurtosis: A statistical approach to identify defect in roliing bearings," in *2nd International Conference on Marine Research and Transportation*, 2007.

[164] E. M. Knorr, *Outliers and Data Mining: Finding Exceptions in Data.* PhD thesis, The University of British Columbia, 2002.

[165] M. Wand and M. Jones, *Kernel Smoothing: Monographs of Statistics and Applied Probability.* Chapman & Hall, 1995.

[166] H. Yang, J. Mathew, and L. Ma, "Vibration feature extraction techniques for fault diagnosis of rotating machinery : a literature survey," in *AsiaPacific Vibration Conference*, 2003.

[167] N.-T. Nguyen, J.-M. Kwon, and H.-H. Lee, "Fault diagnosis of induction motor using decision tree with an optimal feature selection," in *The 7th International Conference on Power Electronics*, 2007.

[168] J. Liljencrants, "Spectrum analysis using the Fast Fourier Transform (FFT)," *STL-QPSR*, vol. 10, no. 1, pp. 7–13, 1969.

[169] Z. Feng, M. Liang, and F. Chu, "Recent advances in time - frequency analysis methods for machinery fault diagnosis: A review with application examples," *Mechanical Systems and Signal Processing*, vol. 38, pp. 165–205, 2013.

[170] C. Junsheng, Y. Dejie, and Y. Yu, "A fault diagnosis approach for roller bearings based on EMD method and AR model," *Mechanical Systems and Signal Processing*, vol. 20, pp. 350–362, 2006.

[171] I.-S. Lee, "Fault diagnosis of induction motors using discrete wavelet transform and artificial neural network," *Communications in Computer and Information Science*, vol. 173, pp. 510–514, 2011.

[172] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society of London*, vol. 454, pp. 903–995, 1998.

[173] M. Bator, A. Dirks, U. Monks, and V. Lohweg, "Feature extraction and reduction applied to sensorless drive diagnosis," *22. Workshop, Computational Intelligence, Dortmund*, pp. 163–178, 2012.

[174] F. Camci, K. Medjaher, N. Zerhouni, and P. Nectoux, "Feature evaluation for effective bearing prognostics," *Quality and Reliability Engineering International*, 2012.

[175] K. Medjaher, F. Camci, and N. Zerhouni, "Feature extraction and evaluation for health assessment and failure prognostics," *Proceedings of First European Conference of the Prognostics and Health Management Society,*, 2012.

[176] T. Benkedjouh, K. Medjaher, N. Zerhouni, and S. Rechak, "Remaining useful life estimation based on nonlinear feature reduction and support vector machines," *Engineering Application of Artificial Intelligence*, vol. 26, pp. 1751–1760, 2013.

[177] B. Li, P.-L. Zhang, H. Tian, S.-s. Mi, D.-s. Liu, and G.-q. Ren, "A new feature extraction and selection scheme for hybrid fault diagnosis of gearbox," *Expert Systems with Applications*, vol. 38, pp. 10000–10009, 2011.

[178] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, pp. 489–501, 2006.

[179] X. Li, J. Zhou, S. Huang, S. Phua, K. Shaw, and M. Er, "Fuzzy neural network modmodell for tool wear estimation in dry milling operation," in *Annual Conference of the Prognostics and Health Management Society*, 2009.

[180] J. K. Kimotho, C. Sondermann-Wölke, T. Meyer, and W. Sextro, "Application of event-based decision tree and ensemble of data driven mmethod for maintenance action recommendation," *International Journal of Prognostics and Health Management*, vol. 4, 2013.

[181] V. Vapnik and C. Cortes, "Support vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[182] D. Boswell, "Introduction to support vector machines," August 6 2002.

[183] R. Burbidge and B. Buxton, "An introduction to support vector machines for data mining," in *Sheppee, M. (ed.) Keynote Papers, Young OR12, University of Nottingham, Operational Research Society*, pp. 3–15, 2001.

[184] T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *Journal of Machine Learning Research*, vol. 5, pp. 975–1005, 2004.

[185] J. Gall, N. Razavi, and L. V. Gool, *Outdoor and Large-Scale Real-World Scene Analysis*, vol. 7474, ch. An Introduction to Random Forests for Multi-class Object Detection, pp. 243–263. Springer Berlin Heidelberg, 2012.

[186] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, ch. Classification: Basic Concepts, Decision Trees, and Model Evaluation, pp. 145–205. Pearson, 2006.

[187] A. Liaw and M. Wiener, "Classification and regression by randomforest," *R News*, vol. 2/3, pp. 18–22, 2002.

[188] Y. Zhang, H. Zuo, and F. Bai, "Classification of fault location and performance degradation of a roller bearing," *Measurement*, vol. 46, pp. 1178–1189, 2013.

[189] P. Stoica, B. Friedlander, and T. Sonderstrom, "A high-order yule-walker method for estimation of the AR parameters of an ARMA model," *Systems and Control Letters*, pp. 99–105, 1988.

[190] S. Ayalew, M. C. Babu, and L. K. M. Rao, "Comparison of new approach criteria for estimating the order of autoregressive process," *IOSR Journall of Mathematics*, vol. 1, no. 3, pp. 10–20, 2012.

[191] Z. Li, X. Yan, Z. Guo, Y. Zhang, and C. Yuan, "Condition monitoring and fault diagnosis for marine diesel engines using information fusion techniques," *Electronics and Electrical Engineering*, vol. 7, no. 123, pp. 109–112, 2012.

[192] Y. Xing, Q. Miao, K. L. Tsui, and M. Petch, "Prognostics and health monitoring for lithium-ion battery," in *Intelligence and Security Informatics (ISI), 2011 IEEE International Conference on*, 2011.

[193] J. Wang and R. X. Gao, "Multiple model particle filter for bearing life prognosis," in *2013 IEEE Conference on Prognostics and Health Management (PHM)*, 2013.

[194] J.-Q. Sun, *Stochastic Dynamics and Control*. Elsevier, first ed., 2006.

[195] S. Lee, H. Cui, M. Rezvanizaniani, and J. Ni, "Battery prognostics: SOC and SOH prediction," in *Proceedings of the ASME 2012 International Manufacturing Science and Engineering Conference*, 2012.

[196] J. K. Kimotho and W. Sextro, "An approach for feature extraction and selection from non-trending data for machinery prognosis," in *Proceedings of the Second European Conference of the Prognostics and Health Management Society*, 2014.

[197] L. A. Gupta and D. Peroulis, "Wireless temperature sensor for condition monitoring of bearings operating through thick metal plates," *IEEE Sensors Journal*, vol. 13, no. 6, pp. 2292–2298, 2013.

[198] A. Joshi, S. Marble, and F. Sadeghi, "Bearing cage temperature measurement using radio telemetry," *Journal of Engineering Tribology*, pp. 471–481, 2001.

[199] C. Brecher, M. Fey, A. Hassis, and S. Bonerz, "High-speed rolling bearing test rigs with contactless signal transmission for measuring the inner ring temperature," in *European Telemetry and Test Conference, etc2014*, 2014.

[200] K. Johnson, *Contact Mechanics*. Cambridge University Press, 1985.

[201] T. Hemsel, P. Bornmann, T. Morita, and C. Sondermann-Wölke, "Reliability analysis of ultrasonic power transducers," *Archives of Applied Mechanics*, 2014.

[202] P. Ronkan, *Current Measurement in control and monitoring of piezoelectric actuators*. PhD thesis, Tampere University of Technology, Tampere, Finnland, 2008.

[203] Morgan Electro Ceramics, "Cantilever mounted PZT 5A bimorphs," tech. rep., Morgan Electro Ceramics.

[204] A. J. Schmid, "Overview on the application of piezoceramics bending actuators - piezoceramics at argillon," in *1st International Workshop on Smart Materials and Structures*, 2004.

[205] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Springer-Verlag Berlin Heidelberg, 2005.

[206] m. E. H. Pedersen and A. J. Chipperfield, "Simplifying particle swarm optimization," *Applied Soft Computing*, vol. 10, pp. 618–628, 2009.

[207] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks*, 1995.

A Derivation of the Support Vector Machines (SVM) Margin

Consider a plane (line in 2D) defined by $w_1x_1+w_2x_2+b=0$, or in vector form ($\mathbf{w^T x}+b=0$), where $\mathbf{w}=\begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$ is a vector normal to the plane, as shown in Figure A.1.

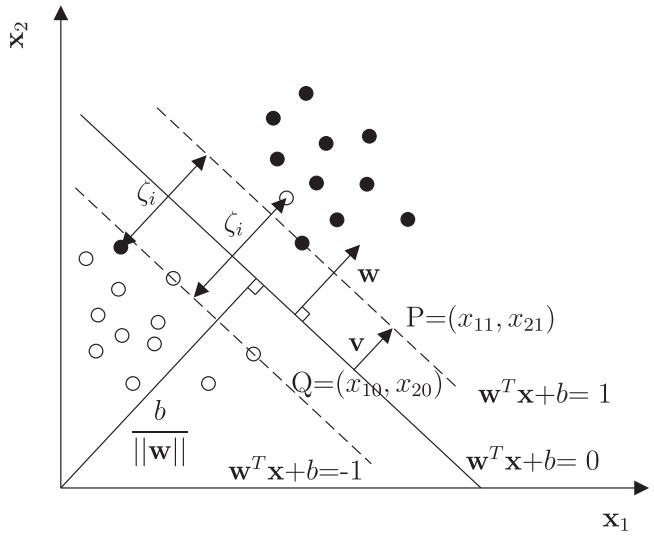

Figure A.1: SVM margin formulation

Since vector $\mathbf{v}=(x_{11}-x_{10}, x_{21}-x_{20})$ is parallel to vector $\mathbf{w}$, the distance from point Q to point P is the length of the projection of $\mathbf{v}$ onto the unit normal vector of $\mathbf{w}$. Therefore,

$$\|\mathbf{v}\| = \frac{\mathbf{v}\cdot\mathbf{w}}{\|\mathbf{w}\|} = \frac{w_1(x_{11}-x_{10})+w_2(x_{21}-x_{20})}{\parallel \mathbf{w} \parallel} = \frac{w_1x_{11}+w_2x_{21}-(w_1x_{10}+w_2x_{20})}{\parallel \mathbf{w} \parallel}. \tag{A.1}$$

Since point Q lies on the hyperplane,

$$w_1x_{10}+w_2x_{20}+b=0 \Rightarrow w_1x_{10}+w_2x_{20}=-b. \tag{A.2}$$

Equation A.1 reduces to:

$$\|\mathbf{v}\| = \frac{w_1x_{11}+w_2x_{21}+b}{\parallel \mathbf{w} \parallel}. \tag{A.3}$$

Since point P lies on the upper margin, $(w_1x_{11}+w_2x_{21}+b=1)$ and then the distance

from the hyperplane to the upper margin is given by:

$$\|\mathbf{v}\| = \frac{1}{\| \mathbf{w} \|}. \tag{A.4}$$

A similar derivation can be used for the lower margin. The total margin is thus given by:

$$\rho = 2\|\mathbf{v}\| = \frac{2}{\| \mathbf{w} \|}. \tag{A.5}$$

To maximize the margin, $\max \frac{2}{\|\mathbf{w}\|}$ is a difficult problem since the norm of $\mathbf{w}$ involves a square root. Therefore, the problem can be reformulated as minimizing the magnitude of the vector, $\mathbf{w}$ by

$$\min \frac{1}{2} \| \mathbf{w} \|^2 .$$

The square is introduced to simplify the minimization problem.

In order to cater for misclassification, the objective function $f(\mathbf{w})$ for SVM is formulated as:

$$\text{Minimize } f(w) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_i \zeta_i, \tag{A.6}$$

subject to:

$$\begin{aligned} \mathbf{w}^T x + b &\geq 1 - \zeta_i, \quad \text{if } y_i = +1, \\ \mathbf{w}^T x + b &\leq -1 + \zeta_i, \quad \text{if } y_i = -1, \end{aligned}$$

where C is the cost parameter and ζ_i is the slack parameter introduced to minimize misclassification in the data.

B SVM Parameter Optimization Techniques

As mentioned in chapter 4 section 4.1.1.4, there are two parameters for support vector machines that require tuning: the cost or regularization parameter, C and the kernel function parameter, γ, resulting to a 2-dimensional search space D. In this work, a combination of differential evolution (DE) and particle swarm optimization were used to tune these parameters for each application.

B.1 Differential Evolution

Differential evolution is a direct search method that utilize N_p population with D-dimensional individual vectors denoted by

$$\mathbf{z}(g) = z_{ij}, \quad i = 1, 2...N_p \text{ and } j = 1, 2..D, \tag{B.1}$$

where g indicates the generation to which the vector belongs [205]. In this work, the population consists of $z_i = [C_i \ \gamma_i]$. DE optimization consists of four steps: initialization, mutation, crossover and selection.

1. **Initialization:** The initial vector population is generated randomly within a predefined parameter space as follows

$$z_{ij} = z_{\min,j} + rn(z_{\max,i} - z_{\min,j}), \tag{B.2}$$

   where rn is a random number uniformly generated within the range [0,1].

2. **Mutation:** New parameter vectors are generated by adding weighted difference between two population vectors to a third vector called a mutant. For each target vector, a mutant vector is generated as follows

$$v_i(g+1) = z_{r_1}(g) + F \cdot (z_{r_2}(g) - z_{r_3}(g)), \tag{B.3}$$

   with random indices $r_1, r_2, r_3 \in \{1, 2..N_p\}$ and $F \in \{0, 2\}$ is the mutation control parameter that scales the difference between the vectors.

3. **Crossover:** A trial vector is generated between $z_i(g)$ and $v_i(g)$using a binomial crossover operator as follows

$$o_i(g) = \begin{cases} v_{ij}(g) & \text{if } rn \leq cr \text{ or } j = j_{rn} \\ z_{ij}(g) & \text{otherwise,} \end{cases} \tag{B.4}$$

where rn is a random number that is uniformly distributed within the range [0,1], cr is the crossover control parameter and j_{rn} is a random number generated within the range [1,D]

4. **Selection:** The better individual between trial vector $o_i(g)$ and the target vector $z_i(g)$ is selected based on the objective function as follows

$$z_i(g+1) = \begin{cases} o_i(g) & \text{if } f(o_i(g)) \leq f(z_i(g) \\ z_{ij}(g) & \text{otherwise.} \end{cases} \tag{B.5}$$

Once a new population is generated, the process of mutation, crossover and selection is repeated until a termination criteria is met. In this work, DE was implemented with $z_i = \{\gamma_i, C_i\}$, initialized within the range:

$$\begin{aligned} \gamma_i &\in \{2^{-15}, 2^3\} \\ C_i &\in \{2^0, 2^{15}\}. \end{aligned}$$

DE locates the range within which the optimum values lie. This range is then used to define the search region for PSO. The advantage of PSO is the ability to obtain the global optimum within a finer search space [206].

B.2 Particle Swarm Optimization

Particle swarm optimization was first proposed by Kennedy and Eherbert [207] to emulate the social behavior of a swarm of birds or insects (referred to as particles) that move in a search space. Particles interact with one another and together with their own experience, gradually move to better regions within the search space [207]. The particles are initially placed at random positions within the search space and move in random directions. The direction of each particle is then gradually changed so that it moves in the direction of its best previous positions and positions of its peers while searching within their vicinity with respect to a fitness function. Each particle i ($i = 1, 2, ...n$) is characterized by two factors: its position $z_i = (z_{i1}, z_{i2})$ and its velocity $v_i = (v_{i1}, v_{i2})$) [206]. The new position and velocity of particle i in the next iteration in D dimensional space is

$$v_{ij}(k+1) = \omega v_{ij}(k) + a_1 rn_{1j}(k)(p_{ij}(k) - z_{ij}(k)) + a_2 rn_{2j}(k)(p_{gj}(k) - z_{ij}(k)), \tag{B.6}$$

$$z_{ij}(k+1) = z_{ij}(k) + v_{ij}(k), \tag{B.7}$$

where p_{ij} is the particles personal best position attained so far, p_{gj} is the global best position attained so far by all particles, a_1 and a_2 are acceleration coefficients, rn_1 and rn_2 are random numbers uniformly distributed within [0,1] and ω is the inertia weight that controls the influence of the particles previous velocity in the current iteration.

The particles are first initialized within the range obtained by differential evolution. Then for each iteration, the velocity and position of each particle are calculated and compared with the previous personal best and global best until a termination criteria is reached.

Figure B.1 summarizes the application of PSO in tuning the parameters of SVM. The objective function is to minimize the cross-validation error during training of SVM.

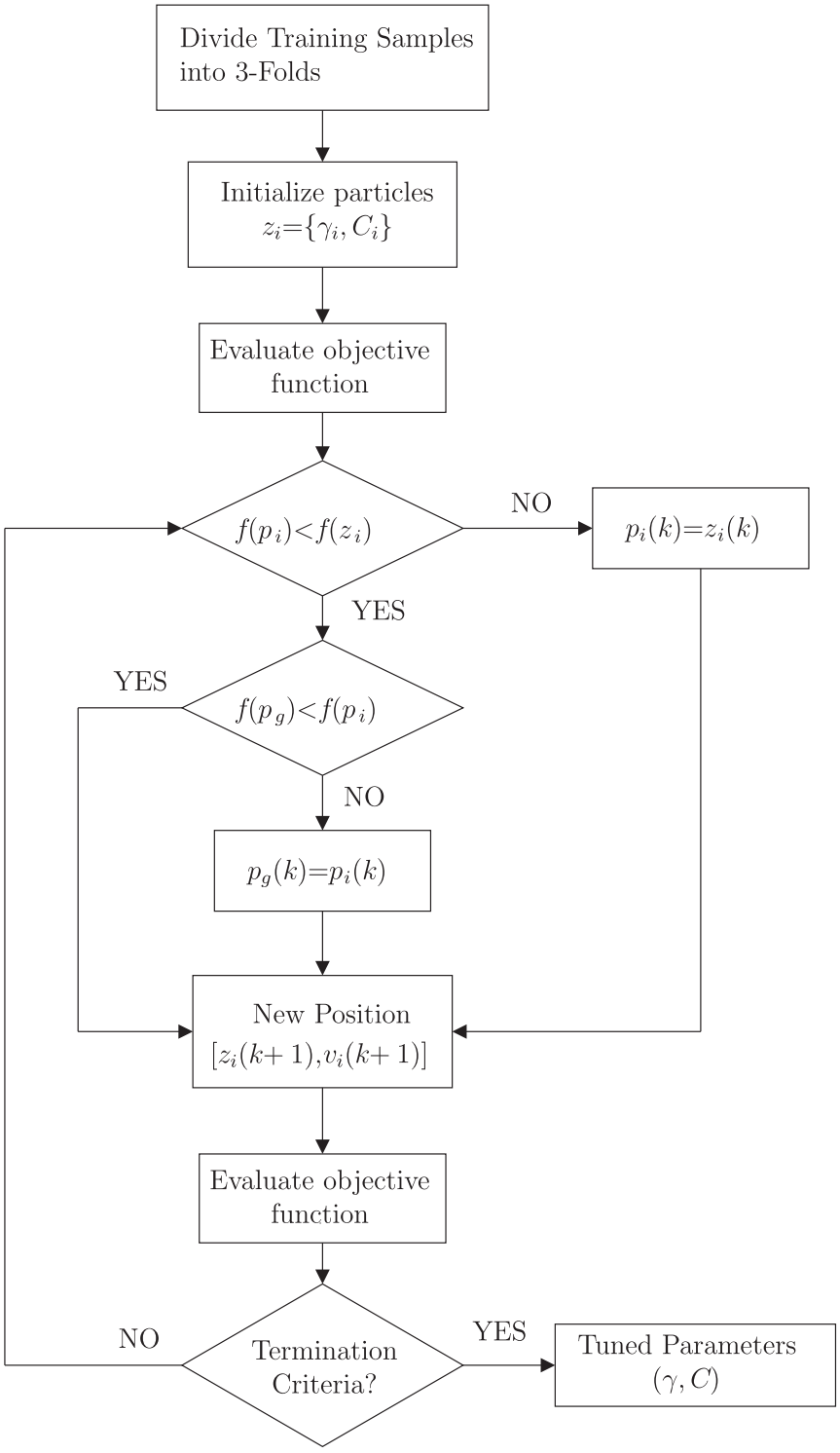

Figure B.1: Application of particle swarm optimization in tuning SVM parameters