12-2015

Fault Diagnosis and Failure Prognostics of Lithium-ion Battery based on Least Squares Support Vector Machine and Memory Particle Filter Framework

Mohammed Ali Lskaafi
University of Tennessee - Knoxville, mlskaafi@vols.utk.edu

Recommended Citation

Lskaafi, Mohammed Ali, "Fault Diagnosis and Failure Prognostics of Lithium-ion Battery based on Least Squares Support Vector Machine and Memory Particle Filter Framework. " PhD diss., University of Tennessee, 2015.
https://trace.tennessee.edu/utk_graddiss/3594

To the Graduate Council:

I am submitting herewith a dissertation written by Mohammed Ali Lskaafi entitled "Fault Diagnosis and Failure Prognostics of Lithium-ion Battery based on Least Squares Support Vector Machine and Memory Particle Filter Framework." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Industrial Engineering.

Rapinder Sawhney, Major Professor

We have read this dissertation and recommend its acceptance:

John E. Kobza, James L. Simonton, Jamie Coble

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Fault Diagnosis and Failure Prognostics of Lithium-ion Battery based on Least Squares Support Vector Machine and Memory Particle Filter Framework

A Dissertation Presented for
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Mohammed Ali Lskaafi
December 2015

Dedication

To my parents, Ali and Zahra for providing me with every opportunity and encouragement in life

Acknowledgements

This thesis is the result of four years of work. Obviously, several persons have been most helpful in the process. First, I would like to thank the members of my dissertation committee for taking time to review this work: Dr. Repinder Sawhney, Dr. John E. Kobza, Dr. James L. Simonton, and Dr. Jamie Coble. I would like to send my sincere gratitude to my supervisors: Prof. Repinder Sawhney, for all his help in carrying out the work presented in this thesis, with a fatherly patience.; Dr. Jamie Coble for her valuable feedback and for believing in me; and Dr. John E. Kobza and Dr. James L. Simonton for their help and recommendations. I would also like to acknowledge those who helped in ways other than academic advice. Firstly, I would like to thank Carla Arbogast, for all the help. I would also like to thank Ninad Pradhan, for his time and knowledge in getting this thesis finished. My thanks go to Sara Melton for her many hours of time, and for making the thesis better. I would like to extend my sincere thanks to Hodges Library for helping and providing open resources. I would also like to thank my parents, Ali and Zaher, who supported me at every stage, and for that last-minute proof reading. Finally, this dissertation work is supported by Saudi Arabian Culture Mission Fund for the Doctoral Program of Higher Education of Saudi.

To all those cited by name here and the many more I did not mentioned by name, I will forever owe you this achievement. Thank you all very much!

Mohammed Ali Lskaafi

1435 / Ramadan / 30

Abstract

A novel data driven approach is developed for fault diagnosis and remaining useful life (RUL) prognostics for lithium-ion batteries using Least Square Support Vector Machine (LS-SVM) and Memory-Particle Filter (M-PF). Unlike traditional data-driven models for capacity fault diagnosis and failure prognosis, which require multidimensional physical characteristics, the proposed algorithm uses only two variables: Energy Efficiency (EE), and Work Temperature. The aim of this novel framework is to improve the accuracy of incipient and abrupt faults diagnosis and failure prognosis. First, the LS-SVM is used to generate residual signal based on capacity fade trends of the Li-ion batteries. Second, adaptive threshold model is developed based on several factors including input, output model error, disturbance, and drift parameter. The adaptive threshold is used to tackle the shortcoming of a fixed threshold. Third, the M-PF is proposed as the new method for failure prognostic to determine Remaining Useful Life (RUL). The M-PF is based on the assumption of the availability of real-time observation and historical data, where the historical failure data can be used instead of the physical failure model within the particle filter. The feasibility of the framework is validated using Li-ion battery prognostic data obtained from the National Aeronautic and Space Administration (NASA) Ames Prognostic Center of Excellence (PCoE). The experimental results show the following: (1) fewer data dimensions for the input data are required compared to traditional empirical models; (2) the proposed diagnostic approach provides an effective way of diagnosing Li-ion battery fault; (3) the proposed prognostic approach can predict the RUL of Li-ion batteries with small error, and has high prediction accuracy; and, (4) the proposed prognostic approach shows that historical failure data can be used instead of a physical failure model in the particle filter.

Table of Contents

List of Tables

List of Figures

Abbreviations and Symbols

AE	Absolute Error
PCoE	Ames Prognostic Center of Excellence
Ahr	Ampere hour
BP	Backpropagation Neural Network
CBM	Condition Based Maintenance
CC	Constant Current
ERM	Empirical Risk Minimization
EE	Energy Efficiency
FDI	Fault Detection and Identification
FDD	Fault Detection and Diagnosis
JIT	Just in Time
JITP	Just–In-Time Point
LTI	Lead-time interval
LS-SVM	Least Square Support Vector Minchin
MAPE	Mean Absolute Percentage Error
MM	Mean Method
MSE	Mean Squared Error
M-PF	Memory Particle Filter
NASA	National Aeronautic and Space Administration
NN	Neural Network
OSA	Open System Architecture
PF	Particle Filter
PSO	Particle Swarm Optimization
PHM	Prognostic Health Management
QP	Quadratic programming
RBF	Radial basis function
RNN	Recurrent Neural Networks

γ	Regularization parameters
RUL	Remaining Useful Life
SIS	Sequential Important sampling
S	Standard deviation
SOC	State of Charge
SOH	State of Health
SLT	Statistical Learning Theory
SRM	Structural risk minimization
SSE	Sum Squares Error
SVM	Support Vector Machines
ToF	Time of Failure
VC	Vapik-Chervonek
WT	Working Temperature

Chapter 1

Introduction

1.1 Motivation and Problem Definition

Complex engineering systems, such as aircrafts, industrial processes, power plants and lithium-ion batteries are developed to perform specific functions in term of reliability, productivity, safety and availability. However, no matter how well a complex system is designed, the system will deteriorate. Thus, maintenance is introduced as an effective routine to sustain the reliability of the system. Maintenance has traditionally employed one of two maintenance philosophies: preventative or corrective maintenance. The common characteristic of preventative and corrective maintenance is that neither strategy takes the actual condition of the system into consideration before decided-upon maintenance activates. Therefore, preventive and corrective maintenance become a major expense in different industries; what's more, they lack the accuracy of failure rate at fixed intervals to avoid the catastrophic failures, are labor intensive, and minimize the system's availability. In fact, one third of the cost of maintenance is incurred unnecessarily due to bad planning, improper or misused preventive maintenance, and unavailable equipment that lead to decreased availability and increased maintenance cost in terms of labor and spare parts [183].

The designing of maintenance procedures for improving maintenance policy for achieving equipment reliability receives great attention from practitioners. In recent years, practitioners and engineers have preferred Condition Based Maintenance (CBM) over preventative and corrective maintenance to avoid high cost, increase availability and productivity, and improve performance. Condition-based maintenance (CBM) can be explained as a maintenance plan based on discrete measurement of the system or equipment during its operation. However, CBM is not effective for complex and dynamic systems, which are subject to multiple failure modes. [184].

Early fault diagnosis and failure prognosis increase system awareness during health monitoring of complex systems. The technology of diagnostics and prognostics aims to detect and isolate impending faults and predict their future progression based on current diagnostics and available measurement data. Reliable real-time fault diagnosis and ability to estimate Remaining Useful Life (RUL) can ensure the complex system will be safer and more reliable, improve its availability, and reduce downtime. This is useful when the system is expected to run for long-term.

1.2 Objective

The main objectives of this thesis are to develop a novel data driven framework based on online real-time Condition Based Maintenance and Prognostic Health Management (CBM/PHM) for a complex non-linear system within the domains of Lithium-ion battery (Li-ion). The framework must account for the following challenges:

1- Design a fault diagnosis and a failure prognosis for a Lithium-ion battery with small multidimensional physical characteristics, and low computational complexity.

2- Design a fault diagnosis to minimize the false alarm rates, and maximize fault detectability conditions.

3- Design a fault diagnosis which operates in an online fashion that has the ability to accumulate knowledge of nonlinear multiple mode processes, with less computational effort, and high generalization.

4- Design a method that allows the particle filter to estimate remaining useful life (RUL) in the absence of a physical failure model.

These pressing challenges have been formulated based on the problem discussion, where the focus is on the Li-ion batteries process of real time CBM/PHM. Most recent studies in this field focus either on fault diagnosis or failure prognosis application but not on both simultaneously. In this research, both technologies are to be implemented based on the need for real time application.

1.3 Assumptions

Problems of fault diagnosis and failure prognosis are diverse. In this research, the model is built based on addressing problems subject to following assumptions.

For fault diagnosis

1- It is assumed that observation data is collected from the system monitor, and monitoring measurement is representative of the state of the system, i.e., condition indictor is extracted from the sensor data, e.g, calibration, spectrometric data, power, currents, and voltages, or directly observed from the sensor.

2- It is assumed that sufficient observation data is available for optimizing the parameters of Least Square Support Vector Machine (LS-SVM). Also it is assumed that the optimization parameters have been established for all major systems' states of operation.

3- For the application purpose of lithium-ion battery, it is assumed that no partial charge and discharge events are happening during these scenarios; therefore a charge cycle initiates only after the battery is fully discharged [1].

For failure prognosis

1- It is assumed that a bank of nonlinear historical failure data related to the relevant system is available. This historical data explains the past behavior and can be used to predict the future.

2- It is assumed that this framework is applicable to incipient and abrupt failures.

3- It is assumed that the framework is applicable to different system as stand-alone or hybrid with different Fault Detection and Identification (FDI).

4- It is assumed that under normal operation condition (no-fault), the Memory Particle Filter (M-PF) framework is only waiting on an alert from the fault detection system. Once the fault is detected, then the M-PF becomes active.

5- It is assumed that the predefined threshold is available for the end of serviceable system life or End of Life (EoF), which is 30 % fade in rated capacity.

1.4 Analysis of Lithium-ion Battery Capacity degradation

Lithium-ion (Li-ion) battery was chosen as an example for a complex system because the internal state variables cannot be accessed by sensors or are hard to measure under operational conditions. [1, 2, 3, 4, 6, 7, 9]. Li-ion batteries exhibit high energy densities, long life time, and environmental friendliness. For these reasons, Lithium-ion batteries contribute to the advancement of technology and are widely used in many applications: from portable electronics and hybrid electric vehicles, to space and aircraft power systems. Failure of a lithium ion battery could lead to irreversible conditions, reduced performance, operational impairment, and under extreme conditions cause catastrophic failure. In order to maximum output from a Li-ion battery, and avoid catastrophic conditions, it is essential that any fault occurring in the battery be quickly detected and accurately diagnosed, and predicting the state of the battery under all operation conditions is necessary in order to prevent fatal failures [5,6,7]. Recently, capacity fade diagnosis and Remaining Useful Life (RUL) estimation of lithium-ion batteries has seen a growing interest among industry and the academic community as a hotspot and challenging problem in the fields of reliability.

1.4.2 Data Source and Capacity degradation

The proposed fault diagnosis and failure prognosis models are validated using Li-ion battery degradation data. The required Li-ion battery data set was obtained from the data repository of the National Aeronautic and Space Administration (NASA) Ames Prognostic Center of Excellence (PCoE) [1]. NASA‘s dataset includes 38 batteries sampled from a battery prognostics test (http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository). The test bed allows charging and discharging of batteries until failure. Various types of faults which occur in Li-ion batteries can be observed using this test bed.

The experiments used to build the dataset and to estimate the state of a battery used the following components: second generation Li-ion 1850 sized rechargeable batteries, power supply voltmeter programmable DC electronic loads, and thermocouple sensors. The batteries were tested using different operational charge and discharge sequences and different impedances at different temperatures [1, 3]. Several battery performance parameters were measured (Table 1).

The objective of these experiments was to be able to measure the capacity of the battery. Since battery capacity cannot be measured directly, indirect estimates of capacity based on the measured performance parameters was used by [1]. The 1850 Li-ion batteries were run through 3 different sequence operational charges, discharges and impedances at different temperatures 4, 24° C.

Table 1: Measured Li-ion Battery Parameters

Symbols	Description	Symbols	Description
ICi	Charge current in the ith cycle	WCi	Charge power in the ith cycle
*I*D*i*	Discharge current in the *i*th cycle	*W*D*i*	Discharge power in the *i*th cycle
*V*C*i*	Charge voltage in the *i*th cycle	η*i*	Energy efficiency in the *i*th cycle
*V*D*i*	Discharge voltage in the *i*th cycle	*Ci*	The capacity in the *i*th cycle
*E*C*i*	Charge time in the *i*th cycle	*i*	Cycle
*E*D*i*	Discharge time in the i*th* cycle	*wti*	Temperature in the *i*th cycle
T	Ambient temperature	*bti*	Working temperature in the *i*th
W	Power of the battery		

In the NASA database two experiments were carried out for individual lithium-ion cylindrical battery, in which the initial capacity ranges from 100% to 30 % of the rate value where below 30% or 1.4 Ampere hour (Ahr) the battery is considered to End of Life (EoL). Group I Batteries (Nos. 6 and 7 Fig.1.a) were tested with a constant current (CC) level of 2 A at ambient temperature of 24°C. Group II (Nos. 47 and 48, shown in Fig.1.b) were tested with a Constant Current (CC) level of 2 A at ambient temperature of 4°C. Fig. 1.b shows fade capacity doesn't always decrease monotonically (incipient fault), but interestingly the capacity experiences an abrupt fault during the cycle operation. Figure 1.a shows the capacity of batteries increasing quickly during the 90th cycle due to the self-discharge [9]. Further, Group I shows that the capacity of Li-ion batteries degrades slowly and there is regional regeneration phenomenon (See Fig.1.a). The capacity of the lithium-ion battery is significantly impacted by many factors such as current, voltage, and temperature due to the nature of the lithium-ion mechanism.

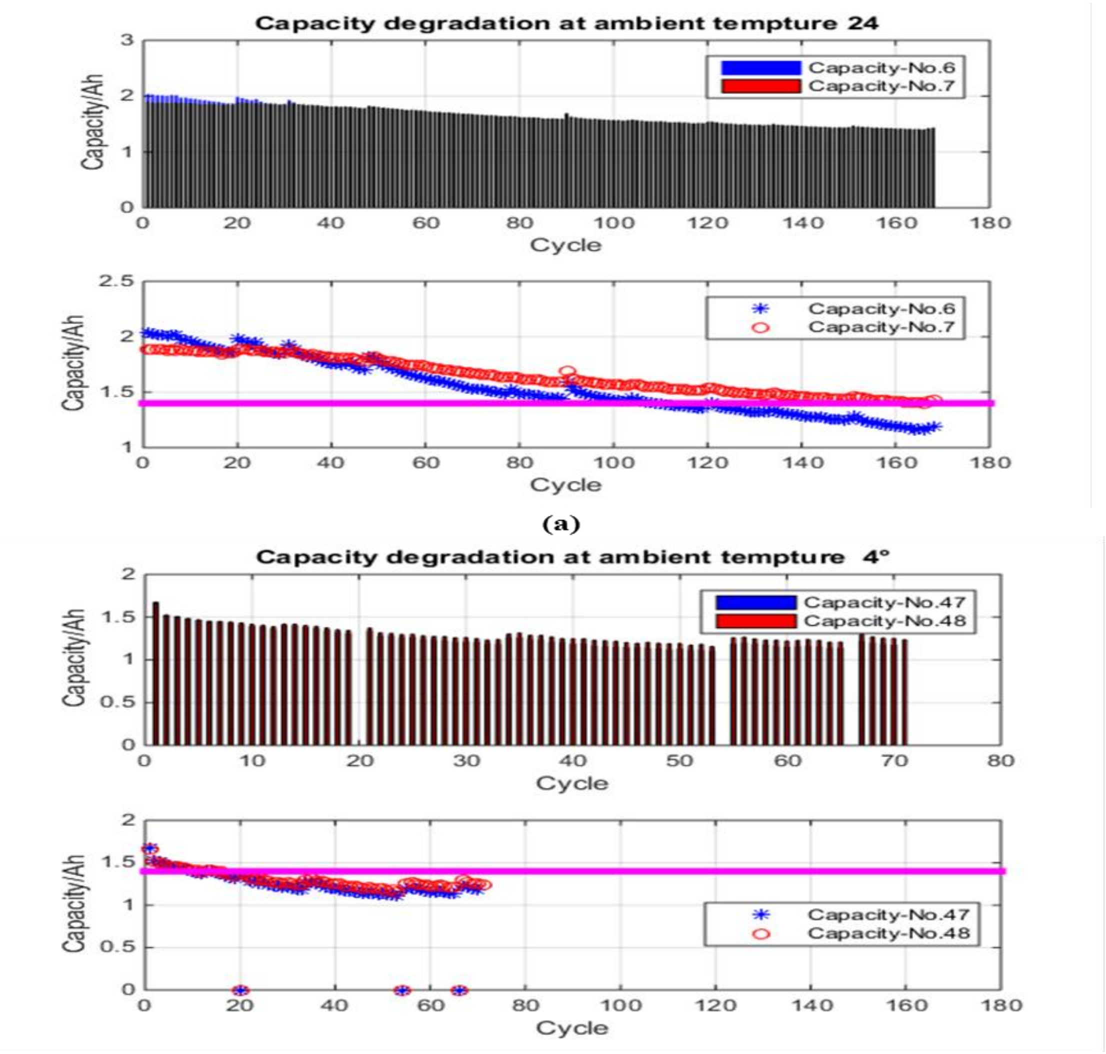
1.5. Data Analysis and Dimensionality Reduction

Traditionally, a performance fault diagnosis and failure prognosis of Li-ion battery has been characterized using the loss of capacity. There are several possible variables and several degradation mechanisms which affect capacity. This makes variable selection an important issue in fault detection and diagnostics and failure prognosis of Li-ion battery.

1.5.1 Overview of Data Analysis Techniques

The capacity will gradually decrease with aging due to temperature, time and cycle number as physical and chemical reactions occur during battery operation. Capacity fading refers to “the irreversible loss in the usable capacity of the battery.” However, failure point is identified as capacity of the battery reaching 80% of the initial capacity; after that, the battery will show an exponential decay of capacity, and it will soon fail [8]. There are two primary research directions for characterization of loss of capacity: State of Charge (SOC) and State of Health (SOH), both of which are hampered by improper variable selection and by poor performance in practical environments.

SOC requires extensive testing and data collection to build the relationship map between available voltage and available charge to estimate loss of battery capacity. SOH is primarily driven by usage data and uses pattern recognition to track battery aging rates. Though SOC and SOH methods based on Li- ion batteries have been studied for a long time, they are inaccurate in practical operating conditions. Moreover, most SOC and SOH estimation methods consider only the voltage, current, capacity, and neglect ambient temperature and battery temperature, which are important variables that link these physical qualities [1, 2, 3, 4,11,7, and 9].



(a)

(b)

Figure 1:Fade Capacity Degradation of Lithium-Ion Cylindrical Battery at Different Temperature of (a) 24°C and (b) 4°C.

1.5.2 Dimensionality Reduction Selection of Variables

For the proposed model, variables are selected based on an understanding of their physical significance for estimation of battery capacity. Battery data comprises several measurable variables, e.g. charge current, charge voltage, discharge current, and battery temperature; each of which is a data dimension. This multidimensional space can be greatly simplified by considering the relationship between these quantities.

One of the most important of these is the concept of electrical work, which is a function of current and voltage. The concept of electrical work is used in the definition of energy efficiency [10].

Energy Efficiency EE is defined as the percentage of energy used to meet the energy service required:

$$EE = \frac{W_D}{W_C} \times 100\ \% \qquad (1)$$

where

W_D is the energy efficiency during discharge that includes discharge current, discharge voltage, and discharge time, and

W_C is the energy efficiency during charging which includes charge current, charge voltage, and charge time.

Working Temperature WT is defined as the difference between the battery temperature during operation and the battery ambient temperature:

$$WT = \sum_{i=0}(wt_i \ \ - T\ ) \qquad (2)$$

Where

wt_i is the battery’s temperature that includes charging temperature and discharging temperature at ith the same cycle, and

T is the ambient temperature of the battery.

EE and WT, as seen in the above definitions, not only preserve all the physical quantities (current, time, temperature and voltage) of the battery, but also reduce dimensions of data to two variables and consequently reduce computational complexity [10].

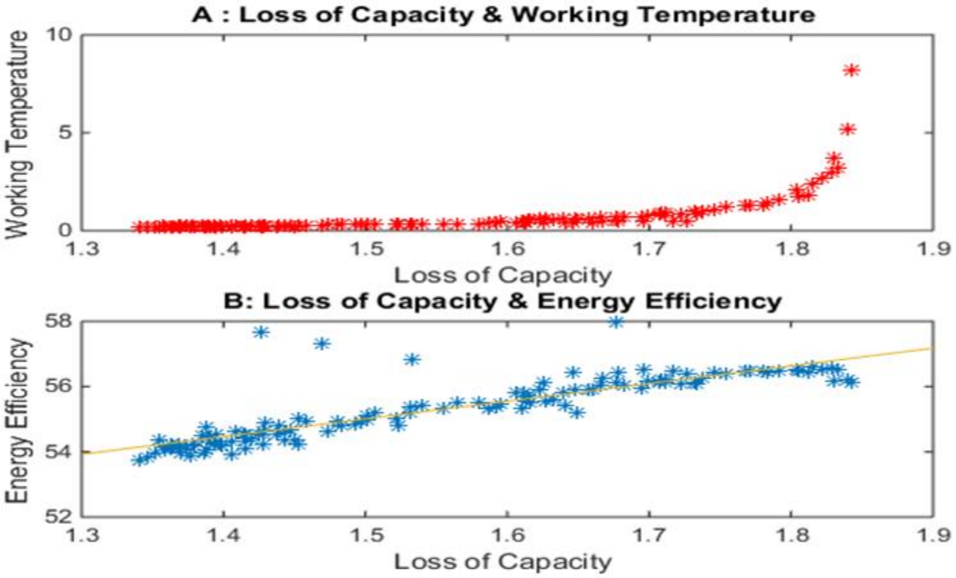


Figure 2:Analysis Loss Capacity of Lithium ion Battery at 24°c: (a) The Relationship Between Loss Capacity and Working Temperature at Different Rates; and (b) The Relationship Between Loss Capacity and Energy Efficiency at Different Rates

The results for the model selection of variable development are presented in the form of two plots. The Figure 2.a.b shows the relationship between EE, WT, and capacity loss of a battery at $24°C$. Figure 2.a shows the relationship between Working Temperature (WT) parameter and loss capacity. Under the same temperature, the loss capacity of battery will increase with the decrease of WT, so the working temperature of the battery must be considered in the judgment of loss capacity. Fig.2.b shows the relationship between Energy efficiency (EF) parameter and loss capacity of the battery at different rates. Fig.2.b shows the degree of linear correlation between the EF and capacity loss. Capacity of the battery decreases with the decrease in EF, and it is easy to notice that the EF shows outliers with time cycle resulting from self-discharge. In this research, we exploit these relationships to estimate the current and future fade capacities of the Li-ion battery

Fault diagnosis and failure prognosis based on LS-SVM uses EE and WT as input variables, and estimates capacity as output, where LS-SVM result will be used to train the Memory Particle Filter (M-PF) to estimate the remaining Useful Life (RUL).

1.6 Thesis Organization

This thesis is divided into four major chapters (see Figure 3):

Chapter 1 contains the motivation and problem definition, followed by the objective of research. The first chapter continues with highlighting the assumptions for fault diagnosis and failure prognosis. In particular, the chapter focuses on an analysis of Lithium-ion Battery capacity degradation, data analysis, and dimensionality reduction. Finally, the chapter concludes with the development of selection variables of energy efficiency and working temperature for NASA‘s dataset.

Chapter 2 outlines the scope of this thesis, and provides an introduction to the basic components of a real-time Prognostic Health Management (PHM) system. The diagnostic and prognostic modules used in the condition monitoring are discussed in great detail, and a brief literature review of the state-of-the-art is also provided, addressing key issue regarding data processing and sensors strategies that are specifically designed to CBM/PHM system. A brief review of Support Vector Machine (SVM) and Least Square Support Sector Machine (LS-SVM) is covered. The chapter concludes with all the theoretical needs for presenting the diagnosis and prognostics scheme.

Chapter 3 is an extensive treatment of LS-SVM based on adaptive threshold for fault diagnosis technology; the chapter begins with the general description of residual evaluation problem, and some of the drawbacks of fixed threshold are highlighted. Then we introduce Support Vector Machine (SVM) and briefly review off-line Least Square Support Vector Machine (LS-SVM) and Sequential LS-SVM based anomaly detector. Theoretical aspects of the adaptive threshold to tackle the problem of the fixed threshold are discussed. Finally, the framework is used to analyze the Li-ion battery degradation.

The result is analyzed and compared with fixed threshold by using different experiments on batteries.

Once an incipient failure is detected, prognostic health management become active. Memory-Particle Filter based on LS-SVM for real-time failure prognosis of nonlinear systems is presented in Chapter 4. Finally, the chapter concludes with a case study for estimating the Remaining Useful Life (RUF) of the Li-ion battery.

A brief review of performance metrics of prognosis scheme is highlighted in Chapter.5. The proposed algorithm is evaluated by comparing its performance with two popular techniques on the same data set, e.g. Recurrent Neural Networks (RNN), and Backpropagation Neural Networks.

The thesis concludes with summary of contribution, and suggested future work in Chapter.6.

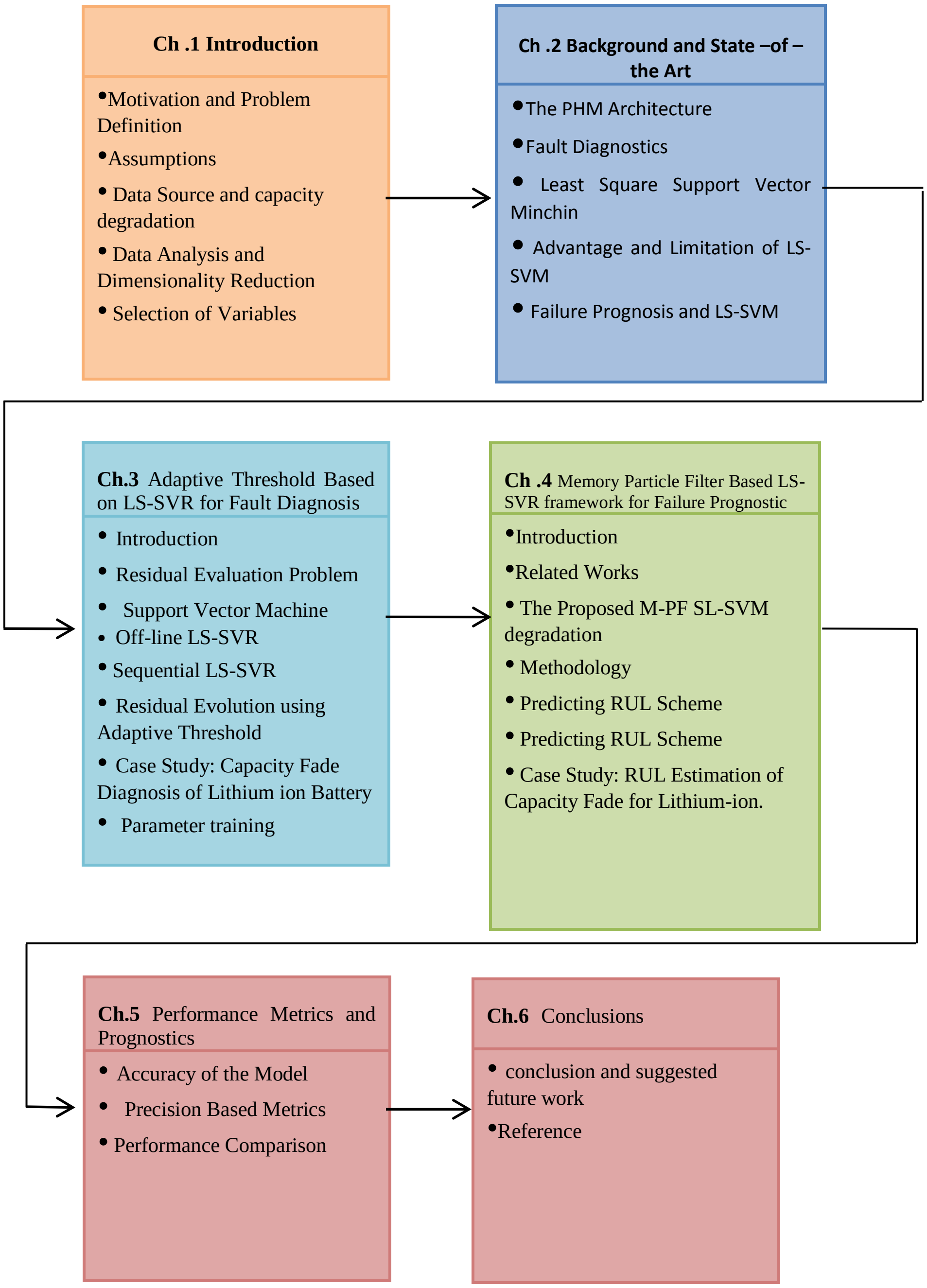

Ch .1 Introduction
•Motivation and Problem Definition
•Assumptions
• Data Source and capacity degradation
• Data Analysis and Dimensionality Reduction
• Selection of Variables
Ch .2 Background and State –of – the Art
•The PHM Architecture
•Fault Diagnostics
• Least Square Support Vector Minchin
• Advantage and Limitation of LS-SVM
• Failure Prognosis and LS-SVM
Ch.3 Adaptive Threshold Based on LS-SVR for Fault Diagnosis
• Introduction
• Residual Evaluation Problem
• Support Vector Machine
• Off-line LS-SVR
• Sequential LS-SVR
• Residual Evolution using Adaptive Threshold
• Case Study: Capacity Fade Diagnosis of Lithium ion Battery
• Parameter training
Ch .4 Memory Particle Filter Based LS-SVR framework for Failure Prognostic
•Introduction
•Related Works
• The Proposed M-PF SL-SVM degradation
• Methodology
• Predicting RUL Scheme
• Predicting RUL Scheme
• Case Study: RUL Estimation of Capacity Fade for Lithium-ion.
Ch.5 Performance Metrics and Prognostics
• Accuracy of the Model
• Precision Based Metrics
• Performance Comparison
Ch.6 Conclusions
• conclusion and suggested future work
•Reference

Figure 3: The Structure of the Thesis

Chapter 2

Literature Review

2.1 Prognostic Health Management Architecture & Just In-Time Maintenance

Just in Time (JIT) philosophy refers to a design and implementation inventory control policy with focus on elimination of waste. JIT emphasizes the importance of assets and availability of labor based on the quantity needed for specific times and specific amounts. Prognostic Health Management (PHM) refers to a capability to predict the Remaining Useful Life (RUL) by projecting the current health state of the equipment into the future based on an estimate of the future usage profiles [121]. RUL and JIT are closely linked in many elements. For example, both try to reduce the inventory and control supply based on where and when products are needed. Moreover, PHM and JIT aim to eliminate sudden failure, reduce schedule maintenance, and increase quality and utilization of plants and equipment. Finally, JIT and PHM aim to fulfill the customer needs; increase the availability and safety while reducing the cost and enhancing logistic and supply chains, which leads to increased market performance [11, 12].

In modern industry, the demand for accurate, timely, and robust incipient fault detection and diagnosis (FDD), and early prediction of the Remaining Useful Life (RUL) for a complex system has increased constantly. In order to fulfill these needs, this thesis outlines enhancing integration of a new online diagnostic and prognostic framework for complex system based on Open System Architecture for Condition-Based Maintenance (OSA-CBM).

2.1.1 OSA-CBM Development

Figure 4 provides an overview of the conceptual design of a seven layered prognostic health management architecture [12]. As shown in Figure 4, a system is monitored via a set of sensors usually integrated with a data acquisition device. The raw signal measurement at the sensors moves though the seven layered stages where data is filtered, denoised, and after that is processed using condition monitoring and health assessment to obtain a health indicator. The health indicator is then passed on to diagnostic and prognostic and decision support to provide recommended maintenance actions.

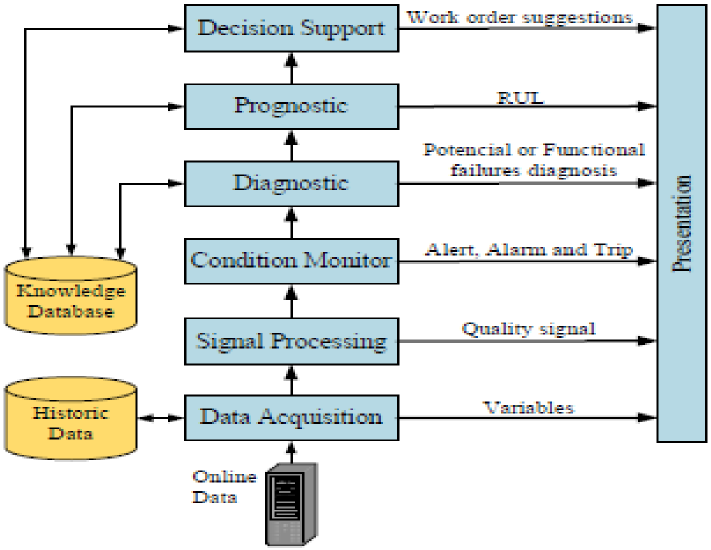


Figure 4: A top –level model of the OSA-CBM architecture [120]

The OSA-CBM architecture requires an appropriate set of sensors to monitor specific variables such as vibration, or measure specific quantities that relate to fault. The raw data collected by the sensor predominantly is not usable in raw form. Often the signals coming from sensors are very noisy: it may be a true signal distorted by other factors such as modulation, bias, low amplitude, very harsh environment, or sensor installation problems. Sometimes it is not possible to measure symptoms directly, or it might depend on another parameter. Moreover, sensors are subject to fault and anomaly which can cause corruption of a variety of faults and false alarms, so these problems will lead a pervasive lack of trust in the sensors. After the sensor's data is validated, the raw data must be prepared by preprocessing data-including filtration, denoising, and feature extracting -in order to enhance the data accuracy by removing unwanted noise, amplify the signal, and reduce the data volume, [24, 32]. Feature extracting here is the process of extracting useful information from raw signal data [18].

Clean sensor data and feature extraction are the most significant steps in OSA-CBM to identify, select, and extract the appropriate useful symptoms that relate to the abnormal operating condition. To enable the benefits of the feature extraction, the sensor data must be passed through different methods to obtain fault diagnostics and failure prediction [12,13]. From a practical viewpoint, OSA-CBM is a conceptual PHM architecture solution for the incorporation of diagnostic and prognostic technology from the equipment level to complex engineering system, PHM architecture presented in Figure 4 generic in design, but it can be applicable to various systems. Further, OSA-CBM architecture has the ability to integrate and share information from different resources, where the main goal of the of OSA-CBM is to recommend an accurate decision related to maintenance activities.

2.1.2 OSA-CBM Applications

This thesis implements real-time integrated Condition-Based Maintenance and Prognostic Health Management (CBM/PHM) architecture. The contribution of this thesis is the implementation of a diagnostic and prognostic data-driven approach. The system flowchart based on Least Square Support Vector Regression (LSSVR) and Memory-Particle filter (M-PF) is shown in Fig 5. This contribution is a generic framework, and applicable for a variety of complex engineering systems. Here CBM/PHM architecture uses LSSVM to extract features and perform self-diagnostics based on pattern recognition to predict reliable and accurate RUL. To maximize the effectiveness and efficiency of CBM/PHM Architecture, the framework adapts Bayesian estimation theory to measure degradation during field operation. The proposed approach involves two tasks: 1- Real-time early confident fault diagnosis based on adaptive threshold for incipient failure; and, 2- Prediction of the Remaining Useful Life (RUL) based on

Memory-Particle Filter (M-PF) of the filing system more accurately and precisely [11, 26].

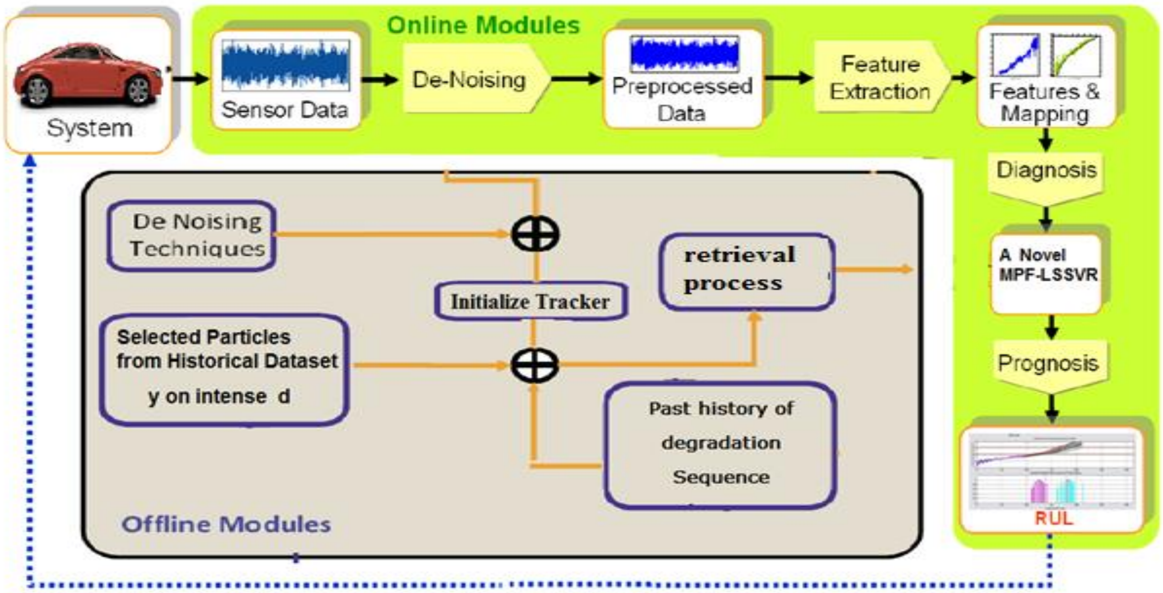


Figure 5 : OSA-CBM Architecture for Implementation of Fault Diagnosis and Failure Prognosis Algorithms [adapted and modified from NASA Ames Research Center]

The proposed architecture classifies the process based on their operation, i.e. online and offline (see Fig. 5). This thesis is primarily concerned with real-time diagnosis and prognostic learning algorithms. Therefore, the issues related to off-line processes are not discussed in a great detail. The off-line process is primarily concerned with a training model's parameter that involves the extraction of representative features from all the sensors, and a retrieval process from historical data, to determine the most important feature for system condition. [27,28].

The diagnosis strategy here is a challenging task due to the limited availability of knowledge about all the types of failure and the severity of the fault model. This makes data-driven approaches applicable because they do not require knowledge of the material properties, and behavior of the failure mechanisms [29].

2.2 Fault Diagnostics

The term fault diagnosing and detection involves three activities. First, fault detection involves identification of abnormal behavior or imminent faults throughout the operation system. Second, the fault isolation involves determining or locating which equipment has failed. A lot of research proposed on diagnostic systems focuses on fault detection and isolation activities exclusively: that is known as fault detection and isolation (FDI). The third activity of fault identification involves estimating and assessing the nature and severity of the fault, by either qualitative or quantitative approaches [36, 35,27]. A good comprehensive review of the fault diagnostics techniques and methods which have evolved in the last four decades [33, 37, 38], and extensive summaries of the history and in development of research in this area [36,39].

2.2.1 Fault Diagnosis Methods

Fault diagnosis procedures can be classified into several categories. For example Farrar et at.[40] suggest classifying them into two types of procedures: on-line and off-line at discrete intervals, depending on completion of the task and performance diagnostic. The other popular diagnostic category involves classification into one of two types: one based on system model fault identification, also known as model-based techniques, and another based on data-driven approaches, called model-free techniques [27, 34].

2.2.2 Model-based approaches

Model-based fault diagnosis includes state-space models, parametric models, and parity relations. Model-based fault diagnosis has been studied extensively in the literature in the form of survey papers and books [33, 117,118, 119]. Model-based approaches utilize a mathematical to model a system in form of a differential equation or equivalent transformation, and the system output can be measured numerically, based upon comparing a particular measurement of the model output and the actual system. When there is a difference, a potential fault is identified and the system declared to be at fault. The Model-based fault detection is a preformed fault diagnosis based upon using one of two producers: parameter estimation or residual evaluation. The difference between parameter estimation and residual evaluation generates a value called *residual signal*. The residual signal goes through a process called *residual evaluation,* which plays an important role in fault diagnosis. The residual signal is extracting the fault symptoms form system. From a practical viewpoint, several factors--such as process nonlinearity, large-scale system, high data dimensionality, and unavailability of good data--complicate the development of accurate, reliable and applicable mathematical models for explaining the physics-of-failure mechanisms [33, 40, 35, 34].

2.2.3 Data-driven approaches

Data-driven approaches to fault diagnosis rely on mapping the data received from the measurement space to the fault space; and it compares the pattern data under testing with other known free-fault reference since the behavior of the system continues a trend similar to ones that are previously known. In this situation, if the test data pattern deviates from the free-fault reference, then a potential fault is identified and the system declared to be faulty. This process is known a type of pattern recognition technique. A good comprehensive review of the data-driven approaches that have been applied to fault diagnostics can be found in [41, 24].

Jardine et al. [41] classified data-driven approaches are applied to fault diagnosis in two types: statistical approaches and artificial intelligence (AI) based approaches. Data-driven models can be implemented, without the need to understand the underlying complex physical mechanisms of a system. Further, data-driven approaches are applicable when data are sufficient. It is easy to transform high dimensional data to a lower dimension in order to capture the important information. Several data-driven approaches have been proposed in the literature, and have been applied to a variety of systems such as vehicles, sensors, aircraft engines, helicopters, and robotics, ranging from simple threshold techniques, as well as to more sophisticated algorithms such as moving window kernel PCA, artificial neural networks, wavelet analysis, and Fuzzy logic [26,32].

2.2.3.1 Artificial Neural Network

Neural Network (NN)-based approaches have been extensively implemented in fault diagnosis [34]. NNs have become more popular in the on-line fault diagnosis field because of their ability to approximate any continuous function without requiring any hypothesis underlying the model [42]. Unfortunately, NNs are the inherent “black box” devices of their operation; even though the neural network can generate solutions for many problems, it is impossible to explain and understand the reason behind its result or interpret the inner operation inside the algorithms. Further, NNs require relatively large volume data sets in the training phase, which can affect the generalization to new data. NN models have a tendency to overfitting; the algorithm doesn’t have rules to select the most suitable design such as the number of hidden layers , nodes and training parameters; and it is slow-learning and time-consuming with non-uniqueness solution due to convergence to local minima especially for large-order systems [43,34,35].

2.2.3.2 Support Vector Machine

From a theoretical point of view, Vapnik found that robust and accurate learning depended on two correlated crucial factors: insufficient volume data samples and improper structure strategy to reduce the number of unknown parameters; these factors can lead to the classification error problem in the machine-learning approaches such as a Neural network. Both problems are very common in fault diagnosis, and failure prognostic. Further, most traditional machine learning technique is based on Empirical Risk Minimization (ERM) structure strategy to minimize the misclassification. However, it has been shown that minimizing ERM can’t effectively reduce the actual risk, especially with a small training sample. Robust structure strategy to reduce the number of unknown parameters is very important and a big challenge, especially when the measurement cooperates with very high –level noise, and outliers [44, 45, 44].

Support Vector Machines (SVM) algorithms have been developed in recent years as an alternative to ERM, devoid of the various drawbacks of ERM techniques. SVMs are developed based on Statistical Learning Theory (SLT). SLT is a supervised learning framework for pattern recognition and provides a new technique for solving these problems [45]. The Support Vector Machine was first introduced by Vapnik and Lerner in 1963, was originally developed for classification [45]. Unlike neural networks, SVM associate to physical meaning of the training data and it is possible to explain and understand the reason behind its result. SVM technique needs small amount of training data sets for any estimation application [49,47]. According to literature in the last few years, Support Vector Machines have shown successful performance in many applications, such as Face Detection System[54], drug classification [53],web prediction [52],time series[51], and petroleum product[47].

2.2.3.3 Support Vector Machine structure

SVM is a nonlinear classification algorithm. Training sample is projected by kernel trick methods $K(x, x_i\ )$ into the higher-feature space. Thus, the conventional non-linear sample becomes linear in the high-dimensional space. In fact; the high-dimensional space is much larger than the original data space so hyper-plane separation of data is realized [55, 35]. The resulting decision function for a training sample x is given as follow:

$$f(x) = sgn\left(\sum_{i\in sv} y_i\, a_i\, K\left(x_{i,}x\right) + b\right) \qquad 2.1$$

Moreover, SVM is adapted ε-insensitive loss function; the ε- insensitive is robust to achieve better approximation of empirical classification error based on different noise distribution. From a theoretical viewpoint, the noise distribution usually is unknown. Vapnik [45] introduces the ε- insensitive loss function as best strategy for estimating unknown noise distribution in the engineering environment with a small amount of error.

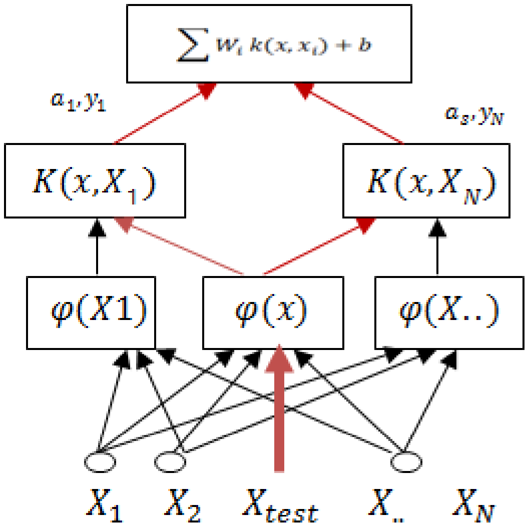


Figure 6: Graphical structure of SVC

The basic form of SVM classification carried out as Eq. (2.1). Figure 6 illustrates the structure of SVC classification with a single hidden layer process, comprised of different number nodes. The input nodes are fed the training sample to the hidden layer from bottom to top where $x_{i,} \in R^n$ is the input and the output is$y \in \{1,-1\}^l$. Only single hidden layer represents kernel function ($K\left(x_{i,}x\right)$) for making transformations into high dimension space. The kernel function $K(x.y)$ can be defined as

$$K(x.y) = \varphi(x)\varphi(y) \qquad 2.2$$

Dot products are added to the weights $w_i = y_i a_i$ which is used as a weights vector to determine which input are support vector, and b parameter is a constant term determining the shift of the hyper-plane but it doesn't change the decision function. Therefore, unlike NNs algorithms, the computational complexity of SVM doesn't depend on the dimensionality of the input space. . Moreover, SVM decision function depends only on a small subset of the training data call support vector.

2.3 Least Square Support Vector Minchin (LS-SVM)

Since Support Vector Machines (SVM) were introduced for classification and nonlinear function estimation, various types of SVM have been developed. SVM result is characterized by solving a convex optimization, or a more precise solving of a convex quadratic programming (QP) in dual space to separate the measurements into two classes of object by the optimal separating hyper-plane. The optimal separating is achieved by employing Vapink's epsilon-insensitive loss function. The equality and complexity of SVC does not relate with the dimension of the input space. The SVM's optimal separation in high-feature space is based on inequality constraints, and employs kernel condition in order to provide nonlinear transformation [45,56,57].

The major disadvantage of SVM is that it requires long training time and suffers from very high computation burden in the training phase because the solution of SVM is associated with constraining quadric programming. For a large number of training data, it requires large amounts of time and memory size, and it needs high computing to solve very large constraining Quadratic programming [45,56]. Suykens and Vandrwalle [58] evolved Least Square Support Vector Machine from SVM.

$$R_{LSSVM} = \frac{1}{2}\ \|w\|^2 + \bar{C}\ \sum_{i=1}^{N} e_i^2 \qquad 2.3$$

The Least Squares Support Vector Machine (LS-SVM) is a version of the principles of SVM for two class problems, LS-SVM reformulation of SVM, which involves adopting the 'Max Margin' idea; however, the ε-insensitive loss function is replaced by least squares loss function or Sum Squares Error (SSE) as it is used in Neural Network. Also, inequality constraints are converted to equability constraints. In this way, reformulation LS-SVM transforms the quadratic programming problem into a simple linear system. Eq.(2.3) shows the characteristic of LS-SVM, where the first term derives from the SVM to keep the feature of maximizing the margin. The second term is minimizing the training error that is characteristic of Gaussian loss function instead of Vapink's epsilon-insensitive loss function [59, 56, and 57,60]. In fact, these modifications are needed for addressing the high complexity and extensive memory requirements of the required quadratic programing that is associated with Vapink’s epsilon-insensitive loss function.

2.3.1 Advantage and Limitation of LS-SVM

There are a number of distinct advantages of LS-SVM over neural network approaches that make LS-SVM more reliable and robust, specifically when data is showing irregular or unknown distribution, which makes LS-SVM the preferred approach in the area of fault diagnosis and failure prognostics.

- **Solving complex nonlinear problem**

LS-SVM is capable of performing non-linear classification problems and regression estimation by kernel function (see section 2.2 for an introduction the kernel trick). The most common kernel functions in fault detection and failure prognosis field are Radial basis function (RBF), Polynomial, and Sigmoid function. The choice of kernel function type is very important since it is employed to measure the similarity of the feature and describe the relationship between input and support vectors. Every kernel function could produce unique measurements and results. [61, 62].

- **Capability to classify unseen sample**

Training set is projected by kernel function from lower dimensional space into the high-feature space to achieve maximum separation between two classes. The nearest data points that used to define the hyperplane are called support vectors. When the support vectors have been selected, the rest of the feature set is not required, as the support vectors can contain all the information-based need to define the classifier[49]. LS-SVM determines the separation hyper-plane. Capturing such a short distance from the separating hyper-plane to nearest data point of both classes means that LS-SVM’s hyperplane has the ability to correctly classify unseen data points better than traditional classification or distance based approaches such as decision trees and k-Nearest Neighbor (k-NN). [44, 60].

- **Ability to train small sample**

Unlike most of the classification approaches, LS-SVM has a unique advantage: strong generalization with ability to train a small number of samples. Eq. (2.7) describes the relationship between the actual risk R (w), and empirical risk $R_{emp}(w)$, which is defined as measure mean error rate for a finite observation training set

$$R(w) \leq R_{emp}(w) + \Phi(h/n) \qquad 2.7$$

Where Φ is the confident interval, n is the number of training sample, h is a measure of the model complexity, called Vapik-Chervonek (VC) dimension. Using a large amount of training data leads to a reduction in the error classification or adjusted $\Phi(h/n)$ to reduce the error and find the optimal hyper-plane classification. However, h has not been defined, and there are many algorithms introduced to estimate h. A common approach is to implement Structural Risk Minimization (SRM) based SVM [43,47, 46].

- **Implement as on-line learning algorithm**

The Least Square Support vector machine (LS-SVM) is capable of computing the global solution faster than standard SVM because it simplifies the complexity of quadratic programing (QP) by reformulating SVM to a simple linear system. This makes LS-SVM a practical method to implement LS-SVM as a real time classifier. A real-time online LS-SVM training algorithm has several advantages; for instance, it doesn’t require saving all historical data, needs small memory space, and is effective with very large and non-stationary data.

- **Free of Local Minima**

Unlike most classification algorithms, especially neural networks, LS-SVM doesn’t suffer from local minimum. Also LS-SVM can find a global optimum solution quickly, with less computational cost compared with conventional SVM [63, 61].

- **Generalization ability**

The generalization of algorithms can examine how to perform a learning ability from the given data set, so as to be able to produce useful information from a new sample as more data becomes available without updating the training algorithm. This capability is widely used in computer science and other fields. LS-SVM generalization ability is based on parameter of kernel function. This research selects Gauss RBF kernel function because it is robust and effective for a wide range of applications. Thus the generalization behavior of LS-SVM can be improved by finding the optimal value of parameters of the LS-SVM, and bandwidth of the RBF kernel [64]. More discussion about choosing the parameters of the LS-SVM will be highlighted in chapter 3.

2.3.2 Least square Support Vector Machine and Fault diagnosis

In recent years, LS-SVM has been successfully applied in the fault detection and diagnosis field. X. Long [68] proposed an anomaly detection of spacecraft based on LS-SVM in-orbit after feature extraction based on Principal Component Analysis (PCA). The non-standard or hydrid approach based on LS-SVM shows better fault identification and detection compared with standard LS-SVM. For example; T.S.Khawaja [69] applied online one-class Bayesian LS-SVM based only on the baseline data from health samples to the problem of fault diagnosis for monitoring a growing crack fault on a gearplate of the UH-60 Blackhawk gearbox aircraft. H.C. Dubey et al. [70] proposed a novel approach based on LSSVM classifiers model combined with modular topology for transmission of fault type classification. L. Min et al. [71] presented a hybrid framework of combining LS-SVM and Mahalanobis distance (MD) based on particle swarm optimization (PSO) to discover advances in the incipient faults in analog circuits. G. Yang et al. [72] proposed an effective multi-class fault diagnosis of noisy data for roller bearings by introducing a probability least square support vector machine. H.B.Zheng et al. [73] used both one-against-one and one-against-all multi-class LS-SVM classification schemes based on particle swarm optimization to improve the multi-class classification accuracy. To provide reliable incipient fault diagnosis performance, C.H.Wei et al. [74] developed a novel hybrid classifier for dissolved gas analysis of oil-immersed power transformers based on fault classification schemes using PSO-LS-SVM. Y. Zhang et al. [75] adopted the wavelet packet analysis (WPA) with LS-SVM for fault signals, the fault diagnosis ability realized by comparing the high performance of LS-SVM against probabilistic neural network (PNN). B.Long at el.[76]developed a reliable incipient fault novel diagnosis using multiple LS-SVM classification based on near-optimal feature vectors instead of a signal feature vector that was selected by Mahalabnobis distance based particle swarm optimization in analog circuits.

Table 2: Fault Detection and Diagnosis based on LS-SVM Comparison Works

Work Criteria	Online Learning	Training small sample	Detect incipient fault	Using Residual Error	Apply Adaptive threshold	Optimize parameters based PSO^*
X. Long et al.[68]						✓
T.S.Khawaja et al.[69]	✓		✓			
L.Min et al.[71]			✓			✓
G.Yang et al. [72]			✓			
H.B.Zheng et al.[73]						✓
C.H.Wei et al.[74]			✓			✓
Y. Zhang et al.[75]			✓			
B.Long at el[76]			✓			✓
Guo Su etal.[15]	✓			✓		
Xu Lishuang etal.[23]	✓					✓
Proposed method	✓	✓	✓	✓	✓	✓

Current data- driven approaches design based on LSSVM have focused on fault diagnosis, isolation, and failure classification of the interested system. However, they are unable to correctly tackle large magnitude errors when using a fixed threshold in the fault diagnosis for complex systems during the operation(see Table 2). To address this issue, a novel online Least Square Support Vector Machine (LS-SVM) is proposed in this research for fault diagnosis with an adaptive threshold by two steps. Firstly, residual error is obtained by comparing real- time LS-SVM function estimation output and the actual output of the system. Secondly, the adaptive threshold model is taken into account based on several factors including input, output model error, disturbance, and drift parameter. The fault detection and identification is combined with adaptive threshold in order to reduce the error fault, avoiding false alarms and improving the diagnostics of complex systems.

2.4 Failure Prognosis

Failure prognosis is key in enabling benefits from condition-based maintenance. However, the stable and effective prognostic model is highly reliant upon the accuracy of fault diagnosis. The primary motivation of failure prognosis is to model the progression of impending failure to estimate the Remaining Useful Life (RUL) or Time-of-Failure (ToF) of deteriorating equipment. Failure prognostics as part of maintenance provides prior notice and an effective mechanism for maintenance staff to avoid unexpected failure by providing sufficient lead-time to supply and replace spare parts before failure [77].

2.4.1 The Remaining Useful Life PDF

The time interval between early incipient fault detection and actual system failure is referred to as the lead-time interval (LTI). With sufficient lead-time interval (LTI), the system or equipment can operate safely with warning of upcoming failure. This enables maintenance stuff to prepare necessary workers and equipment, and improves logistics of spare parts, while maintaining a smaller inventory that leads us to implementing the principle of just-in-time (JIT) manufacturing. A prognostic was introduced first as a military application for the F-35 Joint-Strike Fighter (JSF), to enable the vision of autonomic logistics to satisfy and support military fighter aircraft [78, 77, 35].

The main objective of the prognostics is based on estimating Remaining Useful Life Probability Density Function (RULPDF), also known as Residual Life (RL), or the time left before the equipment fails [79]. Figure 7 illustrates the key concepts of the RUL PDF. The stated damage at current time t_0, whose value is between d_0 and d_1 with 95% confidence as shown in Figure 8. RUL prediction is made based on trend of observation and an estimate of the RUL PDF is generated as a bell-shaped curve (see Fig. 7 and 8).

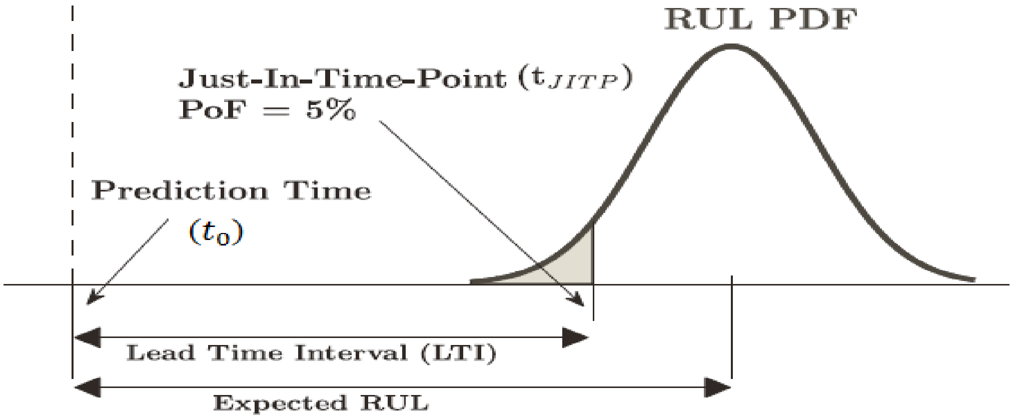


Figure7: Probability Density Functions for RUL [78]

The expected time of failure is shown in both Figures 7 and 8 as the middle of the distribution. Where the corrective maintenance actions have to be early, equipment is removed or replaced before attaining a high probability of the failure. The time chosen for maintenance action will avoid equipment failure and maximize the service-life. This time refers as Just–In-Time Point (JITP), and is illustrated in Figure 7 in terms of probability density functions for RUL. The JITP defines the latest point in time before which corrective maintenance actions must be carried out to avoid unexpected failure with 95 % probability that the equipment has not yet failed. The expected point of the failure is the point where extrapolation damage lines intersect with the failure threshold.

The time interval from present to the JITP is labeled as Lead –Time –Interval (LTI),so the probability of failure is equal to one minus probability of failure avoidance [78, 80, 81, 27, 35].

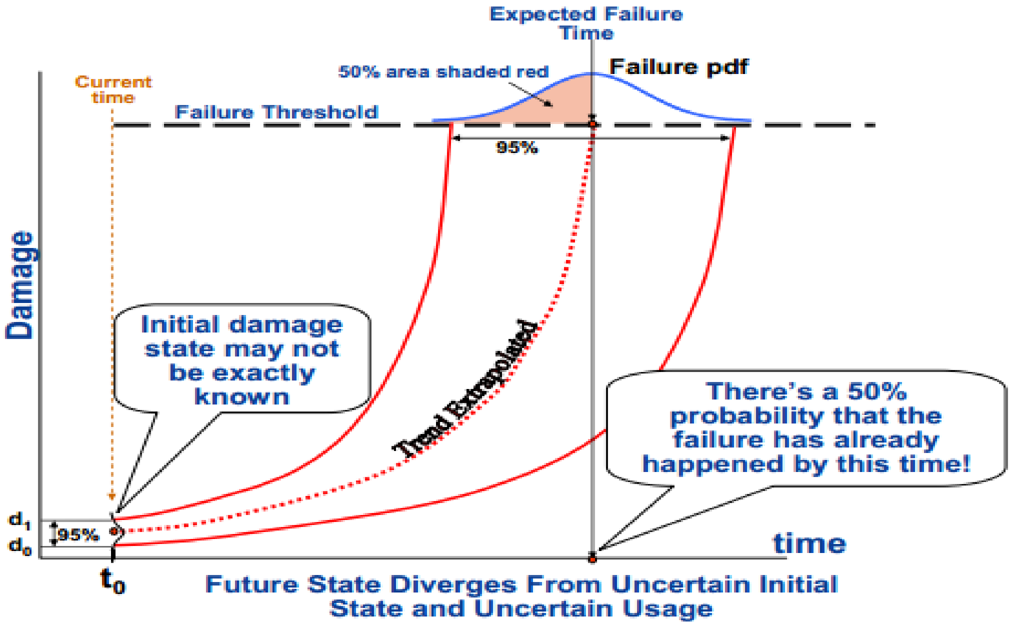

Expected Failure Time
50% area shaded red
Failure pdf
Current time
Failure Threshold
95%
Damage
Initial damage state may not be exactly known
Trend Extrapolated
There's a 50% probability that the failure has already happened by this time!
d_1
d_0
95%
t_0
time
Future State Diverges From Uncertain Initial State and Uncertain Usage

Figure 8: Damage Extrapolation and the Probability Density Function [78]

From theoretical perspective, the RUL PDF is a conditional PDF, changing continuously over time. In practice, when the system has not yet failed, RUL PDF should be recalculated at time t based on the new information; this involves renormalizing the PDF at each time so that its area equals one. As time passes, the variance of true RUL PDF becomes smaller and smaller and less uncertain, and PDF narrower and more stable, as fault condition progresses toward failure. For this reason, predicted value for failure point becomes more accurate and precise as time to failure decreases. This notion is illustrated in Figure 9 [80, 27, 81]. A complete comprehensive description of the probabilistic technique for predicating remaining useful life is provided by Engel et al. [80] and A. Hess[78].

2.4.2 Prognostic Techniques

Similar to diagnostics, prognostics can be classified into several categories based on varying perspectives. For example, J.Z. Sikorska [81] classified prognostics approaches into 4 main groups: 1- Knowledge-based models, 2- Life expectancy models, 3-Artificial Neural Networks, and 4- Physical models. X.-Sheng Si et al.[83] focused on statistical data-driven approaches for RUL estimation , and classified prognostics approaches based on direct health information (Regression, Wiener processes, Gamma processes and Monrovian-based models), and indirect health information (Stochastic filtering-based models, Covariate based hazard models, Hidden Markov model (HMM), and hidden semi-Markov model (HSMM).

Another classification is proposed by Jaw et al. [84], who classified prognostics into real-time and offline by using stored past operation information. Byington et al.[85] categorized prognostics as *horizontal* or *vertical* modules, respectively. Prognostic *horizontal* approach is based on failure diagnosis information, and damage-progression [40], while *vertical* is based on prior knowledge of life expectancy. Further, Byington et al.[85] proposes the three most common approaches to performing RUL prediction; these classifications summarize the range of possible applications to prognostics (see Fig.10), as moving from experience-based to model-based approaches, the accuracy performance and implementation cost is increased, whereas the range of capabilities of different approaches is decreased. However, the two primary approaches form the three levels of algorithms to pursue the prognostic problem: model-based and data-driven [41, 86, 83, 87].

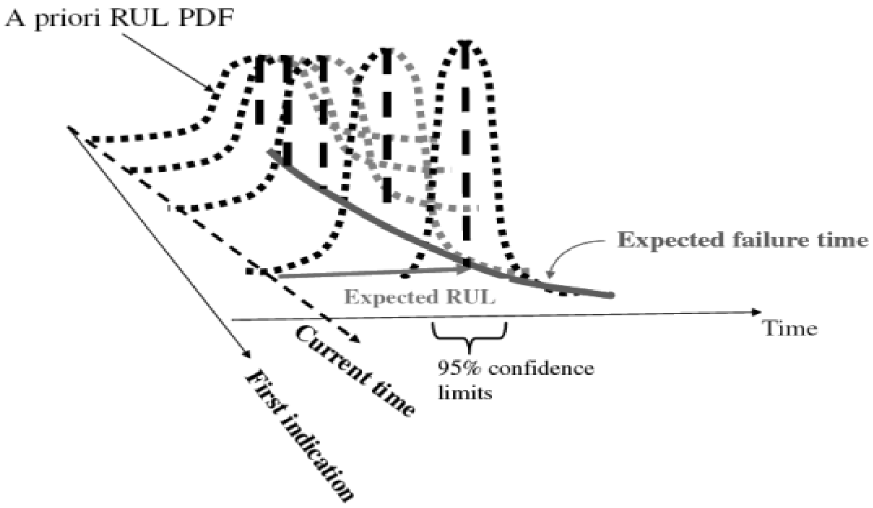


Figure9: Time Variance of the RUL PDF [80]

2.4.2.1 Model-Based Prognostic Approaches

Model-based prognostic approaches are the best methods when deriving an accurate mathematical model from first principles. Model-based prognostics can rely on extracting a true physical model that governs the system's failure degradation behavior (e.g., Crack growth) [88]. The most important benefit of the model-based approach is the ability to incorporate the physical understanding of system degradation. Another advantage is the ability to estimate and predict degradation under different loads and operating conditions. Model-based approaches can be fused with information obtained from data-driven approaches to enhance the prognostic ability and provide valuable information. [81].

The most popular prognostics based on physical models are state-space. State-spaces such as Kalman filters and other tracking filters can be implemented as a prognostic technique. Bayesian estimation provides a framework for dealing with various types of uncertainty in the prognostic problem. The current amount of degradation is modeled as random variable so the level of uncertainty can be propagated into the future.

The typical Bayesian filter is linear state-space model with Gaussian noise. However, various Bayesian filter techniques have been extended for modeling non-linear such as Extended Kalman Filter, and Particle Filter [81, 87, 70, 82, 86].

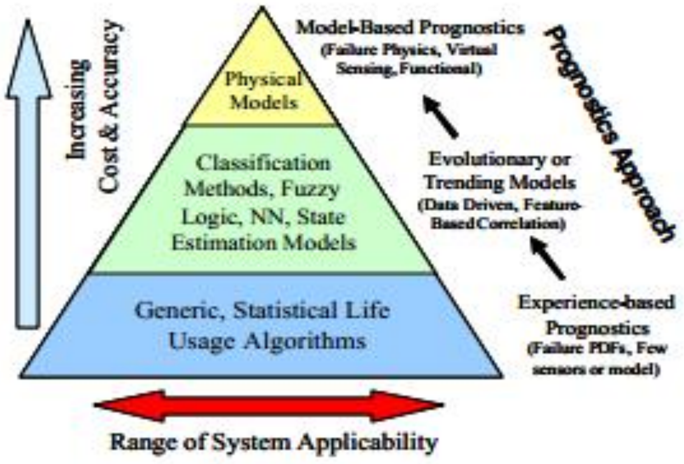


Figure10: Prognostic Technical Approaches [85]

In many practical applications, the process of building an accurate physical model is so complex because comprehensive understandings of all failure modes and characteristics system behavior under a range of operating conditions are not possible. Model-based approaches are built for very specific uses, and they are not easy to adapt to a different system [88]. As an alternative roadmap to failure prediction, RUL can employ the data-driven approaches which rely only on historical information and real-time measurement related to degradation of equipment (e.g, calibration, calorimetric data, spectrometric data, power, vibration and acoustic signal, temperature, pressure, oil debris, currents, voltages)[92].

2.4.2.2 Data-Based Prognostic Approaches

The data-driven approaches are based on the assumption that the statistical characteristics of data are relatively unchanged unless a malfunction occurs in the system.

The failure prediction strategies starts with diagnostic routine and appropriate dimensionality reductions of data acquisition, then applying feature extraction to obtain a system model parameter from the collected data by using appropriate signal processing. These features will be used to map system features into signal dimension degradation or health index, which is used to track the degradation behavior. Once the calculation of the current amount of degradation and diagnosis of the level of fault are made, next, the Remaining Useful Life (RUL) is predicted by using a range of techniques to extrapolate the current degradation into the future until it exceeds a predefined threshold [85,93]. The data-driven approaches are applicable when the data are sufficient. Moreover, it is not system specific and can be applied without prior knowledge or understanding of physical failure mechanisms [92]. With respect to model-based prognostics, this research has a goal of enhancing the data-driven framework of failure prediction by integrating Bayesian filters and data-driven approaches.

2.4.5 Failure Prognostics based on Data- Driven approaches

There are two types of data-driven approaches conventional time series and advance time series. Conventional time series approaches such as exponential smoothing, and autoregressive moving average model, which models two parts, an autoregressive (AR) and a moving average (MA)[116] [181,183]. Prognostics in these techniques rely on projecting the current level of degradation into the future with the assumption that there is some stability in the past patterns. The main advantage of this technique is simplicity and ease of implementation, and ARMA can be used for non-stationary situations. However, the reliance on past patterns can lead to inaccurate Prediction of RUL when at least one observation changes the pattern, and is less reliable in the long term. Advance time series approaches such as ANNs and SVMs are both applied in prognostics in which the current amount of degradation is extrapolated until it reaches predefined failure threshold (see [51, 29, 48, 94, 65, 95, 96, 97, 98, 99, 30]).

2.4.6 Model-Based and Failure Prognostics

A core limitation of ANNs and SVMs approaches are the absence of ability for uncertainty management, lack to provide probabilistic decision function. Therefore, much research has been conducted on Bayesian filters in failure prognostics [100,101,102,103,104], or hybrid Bayesian filters with data-driven approaches. For example, L. Pee[105] applied neural network Ensemble Multi-Layer Perceptron(EMP) and Radial Basis Functions(RBF) as the regression model, and incorporated them with Kalman filter; it provides a mechanism for ensemble multiple neural network model predictions over time. B. Saha et al. [107] integrated RUL prediction method using a hybrid approach: physics-based particle filter was used for system state estimation and parameter identification, and a data-driven predictor was used to estimate the future measurements. The aim of this framework is to incorporated strengths of each approach for determining the remaining useful life of batteries using both data-driven and model-based approaches. The Bayesian estimation approach is implemented as the Relevance Vector Machine and Particle filter (RVM-PF). The framework has significant advantages over classical techniques like Autoregressive Integrated Moving Average (ARIMA) and Extended Kalman Filtering (EKF). L. Yu et al. [108] developed data fusion framework for fault-proneness prognostics based on combined Support Vector Machine and Particle Filter (SVM-PF) in robot dead reckoning. The main goal is to extract weight fault probability parameters in order to select a suitable adaptive threshold for fault prediction. The framework is based on both a model-based and a data-driven approach. L. Chengliang et al.[109]developed a real-time novel based on Least Square Support Vector Machine-based Strong Tracking Particle Filter (LS-SVM-STPF); this method has the ability to track and predict abrupt failure type when salutatory stats occur in the system. Recursive Strong Tracking Filter (STF) is based on both suboptimal fading extended kalman filter (STEKF) and extended kalman filter based practical filter, where STEKF is used as a bridge with EKF-based practical filter . STF is used to update and produce important density, and improved the PF performance when salutatory state occurred in system. C. Xiongzi [110] presented a novel Least Square Support Vector Machine based

Particle filtering (PF-LSSVR) to address the time varying parameters problem for the long-term failure prognostic, and to overcome the limitation of both data-driven (LSSVR) and model-based Particle Filtering.

It is clear that Particle Filtering has gained popularity due to extremely effective ability to address real-time estimation and long-term failure prediction, and to provide probabilistic decision function as prognostic output(see Table 3). However, two major limitations prevent implementing this technique in the field of prognostics. First, it is reliant on accurate physical failure models that quantify the evolution of degradation over time. The physical model may not be available to a complex system, or usually is not applicable to a different system. The performance of PF tends to deteriorate if an inaccurate physical model is applied [111, 112]. Second, the efficiency of Important Sampling (IS), Project Sampling (PS), and Markov Chain Monte Carlo (MCMC) degenerates in high-dimensions, and they suffer extremely from approximate posterior density [113,114,115].

In this research, unlike previous failure prognostics, we develop an extension of the Particle Filter (PF) that uses real-time and historical data instead of an underlying explicit physical failure model. To address this problem, we describe a data-driven algorithm that combines real–time least Square Support Vector Machine based on Memory-Particle Filter (M-PF). Chapter.4 provides an overview of the Memory-Particle Filter based on Least Square Support Vector Machine. However, to the best of our knowledge, this is the first attempt to implement a Memory-Particle Filter (M-PF) for Prognostic Health Management (PHM), and failure prediction.

Table 3: Failure prognostic based on Particle Filter (PF) Comparison Works

Work Criteria	Online Learning	Training small sample	PF requires physical model	PF without physical model	Reduce degenerates problem	Abrupt Failure
E. Zio and Giovanni[104]			✓			
B. Saha et al.[107]	✓	✓	✓			
L. Yu et al[108]	✓		✓			
L. Chengliang et al.[109]	✓		✓			
C. Xiongzi et al.[110]	✓		✓			
Qiang Miao et al.[8]			✓			
Q.Miao[101]	✓		✓			
B. Saha and K. Goebel[2]	✓	✓	✓			
Proposed method	✓	✓		✓	✓	✓

Chapter 3

Adaptive Threshold Based on Least Square Support Vector Machine for Fault Diagnosis

An adaptive threshold scheme based on on-line Least Square Support Vector Machine (LS-SVM) is presented in this thesis. General description of residual evaluation problem is highlighted in section 3.2. The full mathematical description of the Support Vector Machine is presented in section 3.2. Least Square SVR (LS-SVM) function estimation is described in section 3.2.2. The real time modeling problem is considered in section 3.2.3. In section 3.4, we describe an adaptive threshold method. Finally, the real time LS-SVM applied to the real world test case of fade capacity of Lithium-ion battery is presented in Section 3.5. The chapter concludes with a performance comparison between the adaptive threshold and fixed threshold in the section 3.5.1, and LS-SVM‘s optimal parameters is presented in section 3.6.

3.1 Fault Diagnosis

Motivated by the increasing needs for high levels of operational safety, increased maintainability, and reliability in the presence of unexpected changes of a complex system, this thesis presents a practical approach to combine data-driven fault detection with an adaptive threshold for fault diagnosis. Recently, much attention has been devoted to diagnosis of both incipient and abrupt faults. Abrupt fault is typically represented as step-like deviation in the parameter of a system that occurs instantaneously. It is often an indication of imminent breakdown of the system. Incipient fault is modeled as slowly developing, and is represented by drift-type change that must be detected early enough in order to avoid more serious consequences. In abrupt-type failures, fast and early detection is the main objectives of fault diagnosis to avoid catastrophic consequences. On the other hand, incipient faults are very important in maintenance activities because deviation occurs slowly and develops over time, yet usually leads not to sudden failures, but to future unpredictable changes in the system. Therefore, the development of fast and early fault diagnosis schemes for incipient faults plays a crucial role in the reducing cost of maintenance activities [122,123].

One of the main difficulties in dealing with incipient faults is their small effect on the residuals, which can be hidden due to uncertainty [122]. For this reason, it is impossible for a scheme relying on a set of fixed thresholds to obtain all the fault data accurately. One potential solution is to consider an adaptive threshold technique.

3.2 General Description of Residual Evaluation Problem

During the condition monitoring of complex systems, a fault detection and identification system must be able to detect whether and where the fault has occurred. Residual generation is a procedure for extracting the fault symptoms from a complex system, and a threshold is applied to the residual for final decision-making. Generally speaking, the fixed threshold technique or the Mean Method (MM) is commonly used for determining the final decision of fault diagnosis. MM is usually based on average of entire data over a long period of time in order to minimize false alarm rates, and maximize the ability fault detection [125].

The residual will be greater than zero if there are any faults occurring in the system; else if the residual is close to zero, the system is considered to be in operation condition. Typically, the residual is derived without considering existing noise, modeling error, and disturbance; and under different operating conditions, these unexpected (uncertainty) influences will lead to non-zero residuals, resulting in false alarms. Therefore, a diagnostic system based on a non-zero residual error rule would either completely fail to detect and identify a fault, or may detect and identify the fault very late. A fixed threshold is therefore used in practical systems. The fixed threshold will directly affect the fault decision-making performance: too-high level of threshold leads to decreased sensitivity to fault detection. However, a small level of threshold increases the false alarm rate. For this reason, choosing the level of fixed threshold is not an easy task, and it is not reliable in practice. Last but not least, the fixed threshold will lead to increased maintenance cost, reduced resource availability, and declined productivity [126].

A potential solution is an adaptive threshold based on integrating the key factors in threshold model, such as residual error, system input, system output, and random disturbance. The problem of the fixed threshold is illustrated in Figure 11, with the model-based residual for fault detection. The residual is clearly a deviation from zero in the non-fault case; at the same time, the residual exhibits non-stationary features. Moreover, it can be noticed that the difference between the fixed threshold and the adaptive threshold, where the false alarm occurs at t_{F_A} and the fixed threshold also fails to detect the fault att_A. On the other hand, the adaptive threshold has the ability to avoid false alarms and handle the non-stationary quality of the residual to detect the fault at t_{F_A} [124,127].

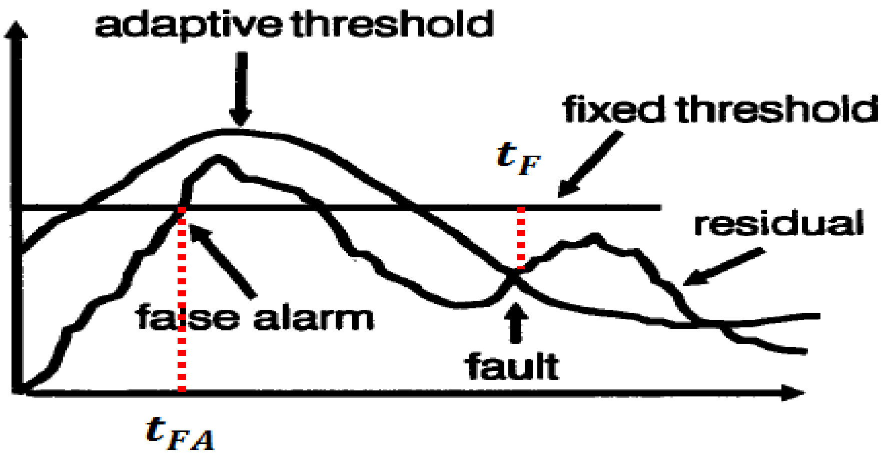


Figure 11: Adaptive Threshold Designs[149]

The ability to detect an incipient fault in a complex system in the presence of modeling uncertainties by adaptive threshold is referred to as “robust fault diagnosis.” This technique uses nonlinear data-driven approach based on sliding time window to monitor the lithium-ion battery whose operation condition changes at some unknown time due to a fault, where the estimation residual error of each fault is associated with an adaptive threshold.

3.2 Sequential LS-SVR based Anomaly Detector

3.2.1 Support Vector Machines (SVM) Principle

SVM is a relatively new computational learning method based on statistical learning theory. [49,94]. SVM was developed for classification based on structural risk minimization (SRM); unlike most of classical learning methods, such as neural network, the minimizing error is designed based Empirical Risk Minimization (ERM). The SVM has been applied successfully to condition monitoring and diagnostic problems, and shows highly accurate performance [65,49,66,43,128,129,130,76, 79,71,44].

When SVM is applied to classification, the samples are assumed have two classes, namely positive class and negative class. The label associated with each class is $y_i = 1$ for positive class and $y_i = -1$ for negative class, respectively. For linear data, it is possible to determine the hyper-plane $f(x) = 0$ that separates the given data [131,110].

$$f(x) = w^T x + b = \sum_{j=1}^{M} x_i w_j + b = 0 \qquad 3.1$$

Where $\boldsymbol{w}$ is m-dimensional vector, b is scalar, and both are used to define the position of the hyper-plane. However, the decision function is made using sign $f(x)$ to create separating hyper-plane that classify input data in either positive class or negative class. A distinct separating hyperplane should satisfy the constraints [49].

$$< w.x_i > +b \geq +1 \qquad for \quad y_i = +1 \qquad 3.2$$

$$< w.x_i > +b \leq -1 \qquad for \quad y_i = -1 \qquad 3.3$$

Or it can be presented in this form:

$$y(x) = sign[w^T x + b] \qquad 3.4$$

If the inequality in Eq. (3.2) and (3.3) can be separated from all training data, SVM will be a linearly separable. The optimal hyper-plane creates the maximum separating distance between the boundary and nearest data for each class. Then, boundary is placed in the middle of the separation. The distance between the two planes is defined as margin, and the data used to define the margin is known as support vector. The optimal margin of the two class data and support vector is represented in Figure12 [45,110].

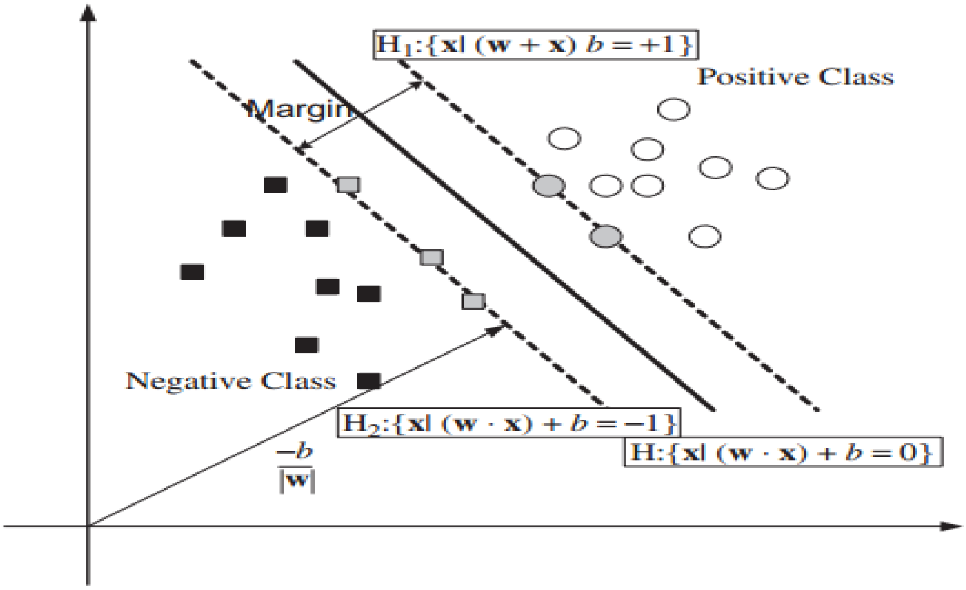

$H_1:\{\mathbf{x}|\ (\mathbf{w}+\mathbf{x})\ b=+1\}$
Positive Class
Margin
Negative Class
$H_2:\{\mathbf{x}|\ (\mathbf{w}\cdot\mathbf{x})+b=-1\}$
$H:\{\mathbf{x}|\ (\mathbf{w}\cdot\mathbf{x})+b=0\}$
$\frac{-b}{|\mathbf{w}|}$

Figure 12: SVM 'Max-Margin' Ideas[49]

From the geometry, the optimal hyper-plane can be found by solving the following constraint optimization. The maximum margin for SVM is obtained as follows:

$$\text{Minimize } \frac{1}{2} \parallel w \parallel^2 \qquad 3.5$$

$$\text{Subject to} \qquad y_i \leq w, x_i \geq -b \quad \leq \varepsilon \text{ ; } \leq w, x_i \geq +b - y_i \leq \varepsilon \qquad 3.6$$

The tacit assumption is that Eq. (3.5) cannot approximate all pairs $(x_i \ , y_i)$. In other words, the convex optimization is not feasible, and the assumption (3. 6) does not exist. We may allow some errors. Slack variable ξ_i is introduced to cope with infeasible constraints optimization, and allows an example to be misclassified. Hence, we present the formulation stated in Burges, C.J.C [132].

$$\text{Minimize } \frac{1}{2} \parallel w \parallel^2 + C \sum_{i=1}^{M} (\xi_i + \xi_i{}^*) \qquad 3.7$$

$$\text{Subject to} \qquad \begin{array}{c} y_i \leq w, x_i \geq -b \quad \leq \varepsilon + \xi_i \text{ ;} \\ \leq w, x_i \geq +b - y_i \leq \varepsilon + \ \xi_i{}^* \end{array} \qquad 3.8$$

Where $\xi_i$is measuring the distance between the margin and the examples x_i that lying on the wrong side of the margin, C is error penalty to determine the tradeoff between the fitness and training error, and ε is insensitive loss function. ε-Insensitive loss function is introduced as the best strategy for estimating loss function under real-life situations. The best loss function for estimate depends on the type of distribution of observation noise, e.g., Gaussian loss function is better under normal noise, whereas uniform loss function is better under uniform noise. However, the noise distribution usually is unknown and far from the Non-Gaussian especially in the field of health monitoring. Vapnik developed robust loss function by combining quadratic and linear loss functions [44,132, 133].

In order to solve the optimization problem in Eq. (3.7), the calculation can be solved in the dual Lagrangian space by introducing the Kuhn-Tucker condition Eq.(3.7). The solution of Eq. (3.7) [131] becomes

$$L(w, b, a) = \frac{1}{2} \parallel w \parallel^2 + C \sum_{i=1}^{M} a_i \ y_i \ (w. x_i + b) \sum_{i}^{M} a_i \qquad 3.9$$

To find the dual form of the problem, we need to minimize Eq. (3.9) by derivative of L to

a with respect to primal variables (w, b, ξ). We have the following saddle-point equations:

$$\frac{\partial L}{\partial W} = 0 \ , \ \frac{\partial L}{\partial b} = \ = 0 \qquad 3.10$$

Which replace into form

$$w = \sum_{i=1}^{M} a_i\, y_i x_i \quad \text{and} \quad \sum_{i=1}^{M} a_i\, y_i \ = 0 \qquad 3.11$$

From Eq. (3.11), we find that w is contained in the subspace spanned by the x_i. Using substitution Eq. (3.11) into Eq. (3.9), we obtain the dual quadratic optimization problem

Maximize

$$L(a) = \sum_i^M a_i - \frac{1}{2} \sum_i^M \sum_j^M \alpha_i a_j\, y_i\ (y_j x_i x_j) \qquad 3.12$$

Subject to $$a_i \geq 0 \ , \ i = 1, \dots\dots, M \ , \sum_{i=1}^{M} a_i\, y_i = \ 0 \qquad 3.13$$

Thus, by solving the dual optimization problem, one obtains the coefficient a_i which is required to express the w in Eq. (3.7), if input data cannot be separated by a hyper-plane in the input space, the data can be projected in the high dimensional feature spaces, where the linear classification is possible. This is achieved through use of nonlinear transformation vector $\boldsymbol{\phi}(\boldsymbol{x}) = (\boldsymbol{\phi}(\boldsymbol{x}), \dots\dots, \boldsymbol{\phi}(\boldsymbol{x}))$ to map the data from n-dimensional input space into high dimensional feature space, and the linear classification function in dual space is realized [131; 43]:

$$f(x) = sign\left(\sum_{j,i=1}^{M} a_i\, y_i \big(\boldsymbol{\phi}^T(\boldsymbol{x_i}).(\boldsymbol{x})\big) + \boldsymbol{b}) \right) \qquad 3.14$$

Transforming into high dimensional feature space creates the challenge of expensive computation due to large vector and over fitting. Kernel function can solve this challenge and perform dot product in one step with less computation effort. From a statistical viewpoint, kernels provide a way to manipulate various types of data as they are projected into high dimensional feature space by operating dot product in its original space. This leads to efficient algorithms with lower computation [131, 43].

$$f(x) = sign\left(\sum_{j,i=1}^{M} a_i\, y_i k(x_i, \boldsymbol{y_i}) + \boldsymbol{b}\right) \tag{3.15}$$

The most popular kernel functions are [140, 70]

Radial basis function(RBF)	$K(x,z) = \exp(-\frac{\lvert x-z\rvert^2}{2\sigma^2})$	3.15.a
Polynomial	$K(x,z) = (x^T z + c)^d$	3.15.b
Sigmoid function	$K(x,z) = \tanh(v. < x.z > + c)$	3.15.c

All kernel function must satisfy Mercer's theorem [60, 56] to be used as kernel function to compute a dot product in high- dimensional feature space. More discussion for selecting the appropriate parameter of kernel function will be discussed later in this chapter.

3.2.2 The off-line Least Square Support Vector Machine Function Estimation (LS-SVM)

SVM possesses the excellent properties of global optimization, high generalization, ability to train small samples, and ability to be successfully applied in a non-linear problem [64, 56]. However, the major drawback of SVM is that it requires quadric programming, which is time consuming and requires a large memory space [136, 137]. LS-SVM is reformulation of SVM, which involves adopting a max margin that searches for an optimum hyper-plane for separating the training data into two subsets (see Figure 13), and the ε-insensitive loss function is replaced by Sum Squares Error (SSE) as it is used in a Neural Network (NN). Also, inequality constraints are converted to equality constraints. Thus, LS-SVM becomes more efficient, very fast, and requires less computation time compared with standard SVM [135, 56, 58].

For N sample training data $\{x_i, y_i\}_{i=1}^{n}$ with input data x_i=R^n, and corresponding output $y_{\ i}$= R, the LS-SVM model considers non-linear function for function estimation

$$(x) = w^T \varphi(x) + b \qquad 3.16$$

Where w is the weight vector in primal weight space, b is base term, and $\varphi(x)$:$R^n \rightarrow R^{n_h}$ is a nonlinear function that maps the input data into a high dimension feature space. This makes non-linear inseparable data in primal space to be separable data in higher space, where n_h is dimensionality of the feature space, which can be determined by the optimization problem with given constraints. The LSSVM model is formulated as [131].

$$\underset{w,b,e}{Min} J_p(w,e) = \frac{1}{2} \parallel w \parallel^2 + \frac{1}{2}\gamma \sum_{i=1}^{N} e_i^2 \qquad 3.17$$

Subject to

$$y_k = w^T \varphi(x) + b + e_i \ , i = 1,2, \dots \dots N$$

Where J is the loss function, $e_i \in R$ is error variables allowing some tolerance of misclassification, b is bias value $b \in R$, and γ is a regularization parameter ($\gamma > 0$) determining tradeoff between the model's complexity and approximation accuracy. Smaller value γ can avoid over fitting, and obtain a smoother solution in the case of noise training sample, and a larger value can provide a more complex solution.

From the discussion above, it is easy to notice the main difference between standard SVM and LS-SVR at two points. First, inequalities are replaced by equality constraints. Second, square loss function is taken for error variable as Sum Square Error

(SSE). This reformulation greatly simplifies a problem such that LS-SVM solution becomes more efficient, and the solution follows directly from solving a set of linear equations rather than from a convex program [52].

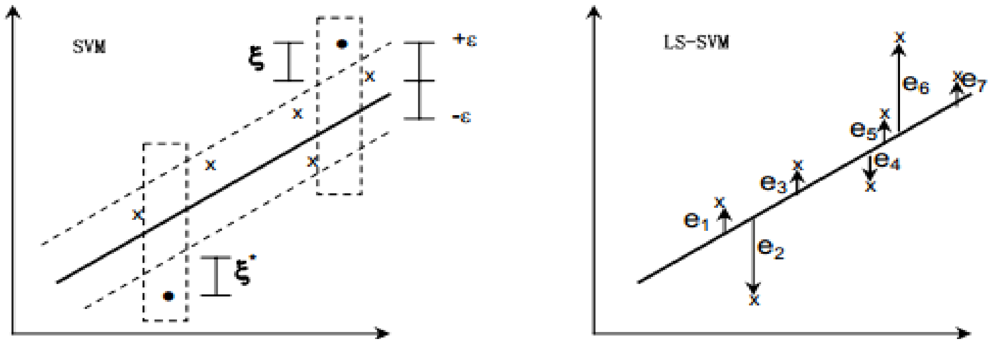


Figure13: Comparisons of SVM and LS-SVR. The solution of two classes can be expressed by data belonging to opposite classes. The purpose of sparse (dash line) of SVM is to minimize the number of support vectors. Whereas, the solution of LS-SVM is not sparse, as usually all data points become support vectors as shown in the Figure[188]

The solution of the optimization problem in Eq.(3.17) is established by Lagrangian multiplier $a_i \in R$. The Eq.(3.17) is formulated as

$$L(w,b,e,a) = \frac{1}{2} \parallel w \parallel^2 + \frac{1}{2} \gamma \sum_{i=1}^{N} e_i^2 - \sum_{k=1}^{N} a_i \{w^T \varphi(x_i) + b + e_i \text{-} y_i\} \qquad 3.18$$

Where a_i value is the Lagrange multiplier. The Karush Tucker conditions for optimality are given by

$$\begin{cases} \dfrac{\partial L_{LS-SVM}}{\partial W} = \ 0 \ \longrightarrow W = \sum_{I=1}^{N} a_k \varphi(x_{i)} \\ \dfrac{\partial L_{LS-SVM}}{\partial b} = 0 \rightarrow \sum_{i=1}^{N} a_k = 0 \\ \dfrac{\partial L_{LS-SVM}}{\partial e_i} = 0 \quad \longrightarrow a_i = \gamma e_i \,, (i = 1, \ldots., N \\ \dfrac{\partial L_{LS-SVM}}{\partial a_i} = 0 \rightarrow (w, \varphi(x_i) + b + e_i - y_{i=0} \end{cases} \tag{3.19}$$

These conditions are similar to standard SVM, only different for condition $a_i = \gamma e_i$. For loss sparseness property in LS-SVM see Figure 13, where ε- insensitive loss function and slack variables are replaced by error variables $e_i \ni \{e_1 \ldots . e_I\}$ to tackle the high complexity and extensive memory requirements of the required quadratic programing that associated with ε-insensitive loss function, where the error variables indicate the distances from each point to the regression function.

Eq. (3.19) can be written as the following linear equations set:

$$\begin{bmatrix} I & 0 & 0 & -Z^T \\ 0 & 0 & 0 & -1^T \\ 0 & 0 & 0 & -I \\ Z & 1 & I & 0 \end{bmatrix} \begin{bmatrix} w \\ b \\ e \\ a \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ b \end{bmatrix} \tag{3.20}$$

When incorporating the first three lines of Eq. (3.19) into the fourth line, w and e vanish. Equations containing only a and b are obtained. For simplicity, these equations can be expressed in a form of a matrix equation:

$$\begin{bmatrix} 0 & I_n^T \\ I_n & \Omega + \gamma^{-1} I_n \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix} \tag{3.21}$$

From [20], and utilizing the Mercer condition $\Omega_{ji} = \varphi(x_i)^T \varphi(x_i) \ = \ k(x, x_i) \ for \ i = 1, \ldots, N., j = 1, \ldots., N)$ we can reduce the complex calculations of inner product in the high-dimensional feature space. The typical choices of kernel function $K(\,x_i, x_i)$ in LS-SVM include linear kernel, polynomial kernel and RBF kernel, as in SVM. However, there is more emphasis on RBF kernel. Any kernel function must satisfy Mercer's condition [131; 58]. The solution of Eq. (3.19) can be obtained from (3.21) the solution a, b are

$$b = \frac{I_n^T\,(\Omega + \gamma^{-1} I_n)^{-1}\gamma}{I_n^T(\Omega + \gamma^{-1} I_n\ \ )^{-1} I_N} \qquad 3.22$$

$$a = \ (\Omega + \gamma^{-1} I_n)^{-1}\gamma - I_n b \qquad 3.23$$

By using the first part of equation Eq. (3.19) to replace the $\boldsymbol{w}$ in Eq. (3.16) and using property (Ω_{ji}), the desired LS-SVM regression model in dual space for function estimation can be written as [56,58, 94, 138, 60,63, 139,140].

$$f(x) = \sum_{i=1}^{N} a_i k(x, x_i) + b \qquad 3.24$$

3.2.3 Sequential LS-SVR based Anomaly Detector

Most existing algorithms for the LS-SVM require offline training with a fixed number of samples; i.e., these samples are delivered in a single batch, in which all examples are available at once [65, 51, 72, 63]. However, the offline training algorithm is not efficient for dealing with non-stationary, large volume data, time varying distributions, and tracking the dynamic change such fault in complex system [162,163,164]. We propose sequential Moving Window Least Squares Support Vector Machine (MWLS-SVM) technique. MWLS-SVM is used to identify the Li-ion battery parameters and their changes when measurement data involve a fault event. The MWLS-SVM method [48, 165, 162, 66] is a real time algorithm; whenever a new sample comes in window, the standard LS-SVM is updated by increasing the sample in training set with a new sample, and losing the old one as shown in Figure14, which keeps the number of samples constant. Here, an incremental updating algorithm and a decremental pruning algorithm are adopted to increment the training set and discard the oldest sample in order to decrease the computational complexity for adaptive tracking technique. In the on-line condition monitoring for a complex system, the training set changes with time.

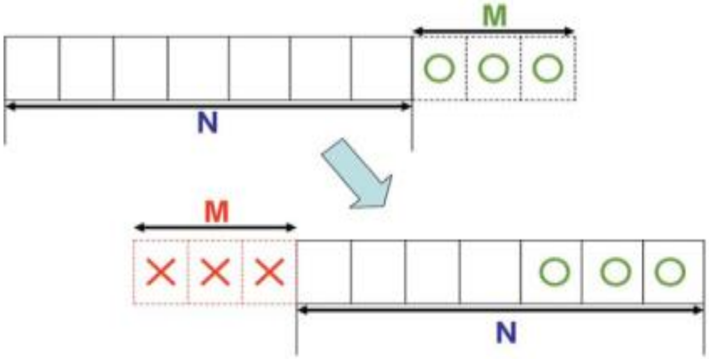


Figure 14: Diagram Representation Updating and Pruning for On-Line MWLS-SVM Learning Scheme Overview[163]

Let (X_{N+1}, y_{N+1}) be a new data pair , then incremental algorithms update the training set by adding the new data to the first (N data pair) so the training set becomes $\{(X_{N+1}, y_{N+1})\}_{K=1}^{N}$. Here, the incremental relation between the current model (X_N, y_N) data pairs and the optimal condition for next new model is (X_{N+1}, y_{N+1}).

Theorem 1 This theorem is a special case of the Sherman Woodbury Formula [167] in linear algebra, which is essential for Incremental learning algorithms. Given X is a symmetrical matrix that has $(n+1)$ rows and $(n+1)$ Columns. X is partitioned as bordered matrix, and X can be written as

$$X = \begin{bmatrix} A & u \\ u^T & a \end{bmatrix} \qquad 3.25$$

Where A is $x \times n$ matrix, u is a $n \times 1$ column vector, a is a scalar quaintly, then the inverse matrix of X is

$$X^{-1} = \begin{bmatrix} B & q \\ q^T & A \end{bmatrix} \qquad 3.26$$

Where B= $A^{-1} + tA^{-1}uu^TA^{-1}, q = -tA^Tu$ and $t = 1(a - u^TA^{-1}u)$.

The sequential incremental algorithms aim to efficiently update $\boldsymbol{y}_{N+1}$ from $\boldsymbol{y}_N$ whenever a new data pair is added. The mathematical proof for calculating the augmented matrix is brief [167]. Equation (3.19) becomes [168]:

$$\begin{bmatrix} 0 & z^T & Z_{new} \\ z & \Omega + \gamma^{-1}I & P \\ Z_{new} & P^T & \beta \end{bmatrix} \begin{bmatrix} b \\ \bar{\alpha} \\ \alpha_{new} \end{bmatrix} \begin{bmatrix} 0 \\ 1_{n\times1} \\ I \end{bmatrix} \qquad 3.27$$

Where P is x×n matrix, u is a n×1 column vector,P_i=$Z_iZ_{new}K(y_i, y_{new})$ and $\beta = z_{new}^2K(y_{new}, y_{new}) + \gamma^{-1}$. It is clear that the square matrix on the left side now becomes a $(n+2) \times (n+2)$ matrix, as there is a new Lagrange multiplier α_{new} for the new training vector y_{new}. This square matrix can be partitioned into a bordered matrix. The bordered matrix reduces the amount of calculation, leads to fast computation, and permits an effective inversion of a triangular matrix [168].

Bordered matrix of Eq.(3.27) becomes as follows :

$$\begin{bmatrix} 0 & z^T & Z_{new} \\ z & \Omega + \gamma^{-1}I & P \\ Z_{new} & P^T & \beta \end{bmatrix} = \begin{bmatrix} A_{old} & s \\ s^T & \beta \end{bmatrix} \qquad 3.28$$

Where s^T =[$Z_{new} \quad P^T$]

By applying Eq.(3.26), the inverse matrix of the new square matrix A_{new} can be illustrated in terms of the inverse of the old square matrix and column vector s, as follows:

$$A_{new}^{-1} = \begin{bmatrix} B & q \\ q^T & t \end{bmatrix} \quad 3.29$$

Where

$$B = A_{old}^{-1} + tA_{old}^{-1}ss^TA_{old}^{-1} \quad 3.30$$

$$q = \ -tA_{old}^{-1}s \quad 3.31$$

$$t = \frac{1}{\beta - s^TA_{old}^{-1}s} \quad 3.32$$

Form the result above, the new matrix A_{new} can be obtained without any matrix inversion. The new optimum values of $b, \bar{\alpha}$ and a_{new} can now be calculated as[168]:

$$\begin{bmatrix} b \\ \bar{\alpha} \\ \alpha_{new} \end{bmatrix} = A_{new}^{-1}\begin{bmatrix} 0 \\ 1_{n\times1} \\ I \end{bmatrix} = \begin{bmatrix} B & q \\ q^T & t \end{bmatrix}\begin{bmatrix} 0 \\ 1_{n\times1} \\ I \end{bmatrix} \quad 3.33$$

3.4 Residual Evolution based on Adaptive Threshold

This section describes an automated dynamic threshold scheme for real-time residual evolution based on data-driven approaches. An adaptive threshold is calculated based on main factors that influence residual error. The residual generation is compared to this dynamic threshold to address incipient fault for detection problem in the presence of modeling error and various unknown (unmeasured) disturbance. In previous research, model-based approaches are utilized to generate the residual error under the unavoidable modeling uncertainty [117, 141, 142, and 143,144]. However, the residual evaluation is basically a classification problem where the following methods can be used for the task: fuzzy logic, expert system, and pattern recognition [124, 145].

The adaptive threshold has the goal of enhancing the model-based framework of residual evaluation proposed in S. X. Ding and P. M. Frank (20002) [146], by including a data-driven approach for parameter estimation. The extension provides real time monitoring that is helpful for setting adequate corrective maintenance, and we use the LS-SVM as adaptive threshold for the identification of a slow developing incipient fault type.

The first step in the process is to define the system state space. For a system with measurable input vector u(t) ∈ $R^m,$ and measurable output vector y(t) ∈ $R^p,$ the linear state space model without fault, and both noise and disturbance, is

$$\begin{aligned} x(t) &= Ax(t) + Bu(t) \\ y(t) &= \quad Cx(t) + Du(t) \end{aligned} \qquad 3.34$$

Where, $X_N \in R^N$ is the unmeasurable state variable,t is the time index, and state space matrix A,B,C, and D determine the state space model.

The second step in the process is to define state space with modeling uncertainty. A general framework for unavoidable modeling uncertainty is provided by a dynamic system, and it can be described as.

$$\begin{aligned} x(t) &= Ax(t) + Bu(t) + Ev(t) + Lf_a(t) \\ y(t) &= \quad Cx(t) + Du(t) + Mf_s\,(t) \end{aligned} \qquad 3.35$$

The next step in the process is to generate the residual error. For the purpose the residual generator design based on analytical redundancy for fault detection has the following form:

$$\begin{aligned} \hat{x}\,(t) &= A\hat{x}(t) + Bu(t) + \Upsilon[y(t) - \hat{y}\,(t)] \\ \hat{y}\,(t) &= \quad C\hat{x}(t) + Du(t) \end{aligned} \qquad 3.36$$

Where, $y(t) - \hat{y}\,(t)$is the residual vector; $\hat{x} \in R^n$ is the state estimate vector; and $\hat{y}(t) \in R^n$ is the output estimation vector [123]. Error can be defined as $e(t) = \hat{y}\,(t)$– $y(t)$.

By implementing Laplace transforms for (3.35 &3.36) the result is

$$\varepsilon(t) = C[sI_s \text{ –A-HC}]^{-1}e(t_0) + C[sI_s \text{ – A-HC}]^{-1}Ed(s) + C[sI_s \text{ –A-HC}]^{-1}\text{G}f_a\text{(s)} - C\ [sI_s \text{ – A-HC}]^{-1}\Upsilon\text{Q}f_a\text{(s)} + \text{Q}f_a\text{(s)} \qquad 3.37$$

Finally, Eq. (3.37) can be used to define the threshold. Where adaptive threshold is designed based on the factors affecting the residual error and system state in Eq.(3.37) which include system input, parameter variation ,modeling error, and disturbance[169]. Fault cause leads to a mismatch between the behavior of the system and the free-fault model. Abrupt or incipient faults occur at some unknown time; the time-profile of the fault is given by

$$\|\varepsilon(t)\| = \|\hat{y}\,(t) - y(t)\| = \begin{cases} \leq \ \ \lambda \ if\ no\ fault\ occurs \\ > \lambda\, if\ fault\ has\ occurred \end{cases} \qquad 3.38$$

Where λ denotes the level of threshold, and the $\varepsilon(t)$ denote the unknown fault evolution behavior. The incipient fault refers to small values of slowly developing symptoms [169]. In this research, the estimation error $\hat{y}\,(t)$ is based on on-line nonlinear LS-SVM approximation models with adjustable parameters. From Eq.(3.37), the adaptive threshold considers the important vector related to the fault: let ε(s) be the threshold, γ(s) be system input, and d represent the disturbance, Δ(s) represent the vector composed of drifting parameter, $L_s(s)$ be the transfer function of the actual system , $L_m(s)$ be the transfer function of the model, $[\gamma(d)](s)$ system input subject to disturbance. Therefore, the threshold can be calculated to be

$$\varepsilon\text{(s)= [[}L_s\ (d, \Delta\text{(s))](s)- } L_m(s)] \times [[\gamma(d)](s)] \qquad 3.39$$

The equation (3, 39) can be re-formulated as

$$\varepsilon(s)= \delta\ [d,\Delta(s), [\gamma(d)](s)\ Y(s)] \qquad 3.40$$

Where, δ is the mapping function of threshold, which takes a variety of influencing factors in frequency-domain. Uncertainties due to usage, life cycle, environment, installation, disturbance and parameter drifting are difficult to measure in practice; they are usually ignored when calculating the threshold, and the adaptive threshold model is then defined as

$$\varepsilon(\delta)= \delta\ [R(s),Y(s)] \qquad 3.41$$

Where R(s) is system input, Y(s) is the model output under disturbance, and δ is the mapping function of the threshold. If we know the system input, model output and the residual error, then LS-SVR can be used to calculate the adaptive threshold. [169].

The result is interpreted based on the dynamic limits concepts of residual signal when the residual signal bounds the adaptive threshold in fault condition; otherwise, it is normal condition [124, 149]. .

3.5 Case Study: Capacity fault Diagnosis of Lithium-ion Battery

For the case study, we consider the problem discussed in section 1.3, where the objective of the case study is to detect and identify the capacity loss (fade) in the lithium-ion battery (see Figure 15) in on-line learning. Developing a model-based diagnosis technique for Li-ion battery requires a non-linear physical model. The physical model is derived from electrochemical analysis in order to capture the capacity fade under different operating conditions. As opposed to that, capacity loss in this research can be detected by substantial variation of Working Temperature (WT) and Energy Efficiency (EE) parameters based on data-driven approaches. For successful fault diagnosis, the online learning is necessary to capture the non-stationary and small fault at an early stage of the operation.

Figure 15: Lithium ion Cylindrical Battery[NASA Research Center].

Unlike most data-driven approaches, LS-SVM is easy to implement as on online learning algorithm. Further, LS-SVM can be used to monitor the Li-ion battery behavior entirely through observation data due to its simpler formulation, and capture an abnormal condition without high computational demand. Here, the fault diagnosis is based on residual error that represents a deviation from operation condition. Threshold is applied to the residual error for final fault diagnosis decision. A Fixed threshold method is commonly used to decide whether and where a fault has occurred, for instance, if the residual is greater than the threshold, the system is considered to be in fault condition. Otherwise, the system is considered to be in an operational condition. However, residual is derived without considering existing noise, modeling error, and disturbance. These unexpected influences will lead to non-zero residuals, resulting in false alarms. This means, if the level of fixed threshold is too large, it minimizes faults detection. On the other hand, if the level of the fixed threshold is too small, it maximizes false alarms. To tackle this problem, the fixed threshold has to cope with problem of the effects of noisy measurement, modeling error, and different operation condition in order to avoid false alarms, and incorrect fault diagnosis [141,144].

It has been shown [141,144,123] the advantage of adaptive threshold that is adapted to fault process. The adaptive threshold improves the monitoring performance with increased detection sensitivity and fewer false alarms, and tackles the drawback of the big error when using fixed threshold if fault diagnosis [19].

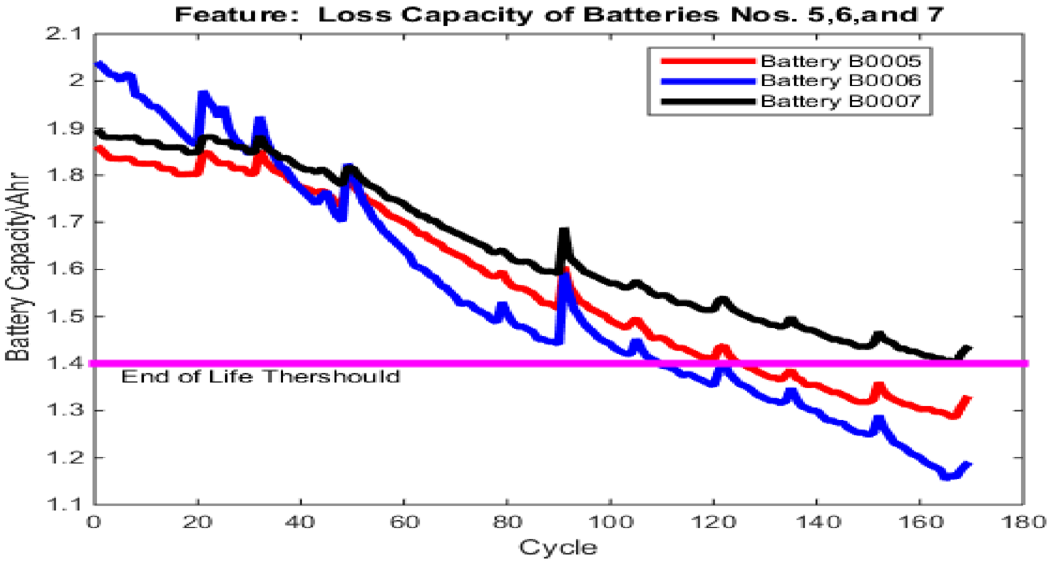


Figure 16: Capacity Loss (Fade) Feature for Batteries No. 5, 6, and 7.

The decision on the occurrence of faults in lithium-ion battery is based on residuals values, where the residual is larger or smaller than the level of adaptive threshold; the larger defines a fault state, and smaller defines an operational condition. Technically, Schneider and P.M. Frank [124] have shown that the effects of molding errors and random disturbance depend on the operating conditions of the process

Figure 16 shows the schematic fault behavior of the capacity loss for the three different batteries at the end of the experiments. It can be observed that one type of fault, the monotonically incipient fault, occurred for the three batteries. Incipient faults occur in the batteries at initial stages of operation. Therefore, if the maintenance correction is not applied early, the fault will propagate into end of life or degradation reach 1.4 Ahr. The curves in Figure 16 illustrate that the faults in the three batteries start at the initial stage of operation, and evolve slowly over a number of cycles. For the purpose of verifying the effectiveness of the adaptive threshold based on online LS-SVM for fault diagnosis, the performance of fault diagnosis in lithium-ion battery dataset will be compared with the classical fixed threshold based on off line LS-SVM.

5.1.1 Fixed threshold based on offline LS-SVM

The residuals are generated in the measurement space based on off-line LS-SVM, and the fixed thresholds are used to decide whether the faults have occurred or not. Figure 17 shows the result of fault detection for Battery Nos. 5, 6, and 7, respectively. The experiment is run for a total of 180 cycles from initial operation (first 20 cycles) to End of Life (EoL) which is 30 % fade in rated capacity. The total experiment is divided into seven equal time periods that occur consecutively. The fixed threshold (pink sold line) is chosen at 0.02 Ahr as mean of the residual deviation over a long period for fault alarm. The last 70 cycles (110-180) represent the end of life of three the lithium-ion batteries. For the fixed threshold test, the process data are divided into two situations, one situation containing the process capacity from 0 to 20 cycles, and other situation containing the capacity process from 20 cycles to the end of life. A first situation is illustrated in Figure 17 where the residuals for Batteries No. 5, 6, and 7 (red, blue, and black sold line) are almost zero in the first 20 cycles, which indicates a healthy operation condition. However, it is shown in Figure 16 that the degradation for the three batteries occurs immediately after the Li-ion batteries have been operated. The reason behind this phenomenon is that the trivial slow incipient faults do not appear quickly on the residual and cannot be detected by the fixed threshold. All the test results are based on Battery No. 5, 6, and 7 fault detection rates, and false alarm rates are shown in Figure 17 and Table 3. The second situation of capacity test in Figure 17, which starts at 20 cycles and continues until the end of life, will be reviewed for each battery.

Table 4: Offline- LS-SVM Residual based fixed Threshold for Fault detection Rate based on Lithium-Ion batteries No. 5, 6, and 7

Battery Number	Fault detection rates	False alarm rate
Battery No.5	20%	80%
Battery No.6	50%	40%
Battery No.7	8%	72%

The first experiment presents fault detection for battery No.5. As seen in Figure 17 (A), the residual signal is non-zero most of time after cycle 20; however, residual usually doesn't exceed the fixed threshold during the fault condition (the time between the residual deviations from zero start at 20 cycles, and end of life at 180 cycles). Residual signal is bounded by fixed threshold at the 48th, 90th, 120th, and 150th cycles respectively, as fault condition, Notice from Table 3 that fault detection has lower rates and greater false alarm. Moreover, Figure 17(A) shows the responses of the fixed threshold to the effects of unknown inputs including modeling error, and random disturbance during operation of the battery. The fixed threshold result is poor at detecting fault, and it is not able to cope with the change in residual behavior due to modeling error and random disturbance. Hence, this technique doesn't eliminate false alarm when the capacity behaves in non-stationary way. An example is presented in Figure 17(A): the residual is not bounded by fixed threshold in fault case at cycles 60, 80, and 140. In addition, the effect of incipient faults becomes visible only after the large magnitude of fault is increased above the fixed threshold. This can cause some hazardous conditions in the system.

The second experiment includes Battery No.6. As shown in Figure 17(B) the residual signal is bounded by fixed threshold after cycle 20, and approximately at cycles 21, 30, 4, 49, 61, 105, 121, 139, and 150. Also, it can be seen from Table 3 that the false alarm rates of experiment based on Battery No.6 using the fixed threshold are obviously lower than experiments based on Battery Nos.5 and 7. In fact, the capacity of Battery No. 6 degrades slowly until reaching the end of life. Furthermore, residual error of Batteries Nos. 5 and 6 increases rapidly and goes to maximum at 48th and 90th cycles resulting from self-charging during the rest period. Through the comparison of fault detection result was based on Battery No.6 and actual capacity performance of the battery, we can see that the fixed threshold has not coped with problem of effect of unknown inputs and is not adapted to different operation conditions of the batteries. As an example, the residual error is not bounded by the fixed threshold when fault has occurred at cycles 60, 80, 101, 115, 120, 140, and 16. Also this example highlights that the fixed threshold is not adaptable to different operation conditions at 48th and 90th cycles due to self-charging conditions of the battery.

The third experiment focuses on the Battery No.7. As shown in Figure 17(C), the residual signal is non-zero after cycles 20. However, the residual signal remains below its corresponding fixed threshold, and the residual is bounded by its threshold only at cycles 30, 50, and 150. Comparing the result with actual performance of battery No.7 illustrates that the fixed threshold scheme is not only misdetection the fault most of time, but also producing a false alarm rate see Table 3. Also the result show that the fixed threshold is not capable of detecting non-stationary faults in the Lithium-Ion battery. It has been found that fixed threshold has to be decreased for small fade capacity in order to prevent false alarm for the three batteries.

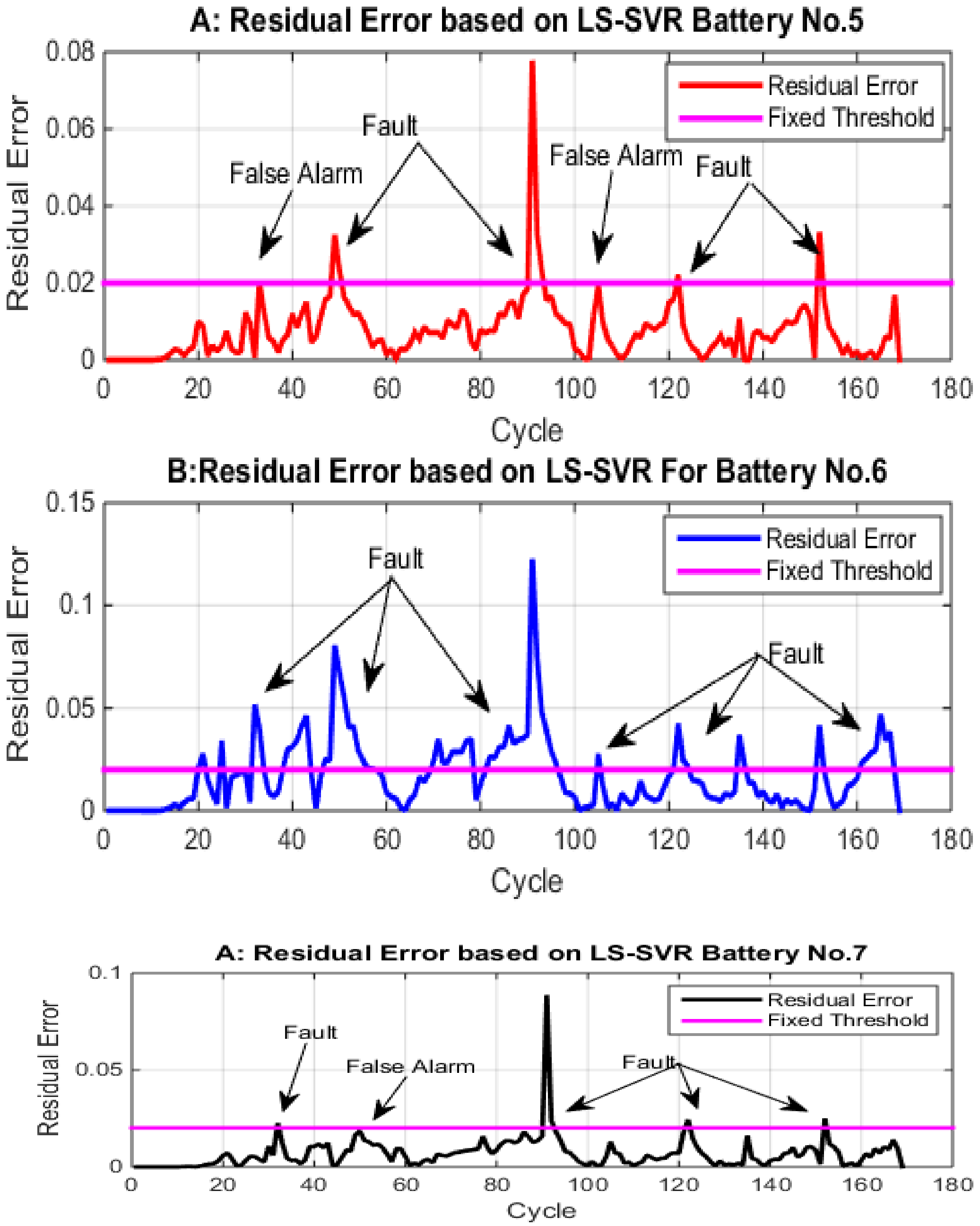


Figure 17: Residual Error based on LS-SVM for Capacity Degradation of Lithium-Ion Battery Nos. 7, 5, and 6.

To sum up, the residual evaluation based on the fixed threshold not only fails to detect the incipient fault accurately, but is also unable to learn the characteristics of the faults. Further, the fixed threshold is incapable of handling the influences of modeling uncertainty, tracks non-stationary fault processes, and tends to diminish the effect of small incipient faults on the residual [122].

The potential solution is to consider a robust tracking technique based on the online moving windows (MW) with an adaptive threshold to track the lithium-ion battery parameters and their change due to capacity fade.

5.1.2 Adaptive threshold based on online LS-SVM

The residual error is generated in the measurement space based on moving windows Least Square Support Vector Machine (MWLS-SVM for short). MWLS-SVM method does not only incorporate new data, but it is also efficient for tracking non-stationary change of lithium-ion battery parameters with less computational effort. Moreover, the existing noise, modeling error, and different operation conditions are tackled by adaptive threshold to prevent a false alarm, and manage an unexpected uncertainty. The adaptive threshold has the ability to identify the battery's abnormal condition.

The suitability of the MWLS-SVM scheme based on adaptive thresholds technique is illustrated through its application to the condition monitoring of three nonlinear cases of lithium-ion batteries, and its performance is compared with a fixed threshold method simultaneously.

Figures18-20 presents the fault diagnosis results for Battery Nos. 5, 6, and 7, respectively. It can be observed that the online adaptive threshold (black sold line) reflects the change of residual error (red dashed line), and characteristics of influences of modeling error, and different operation conditions of the lithium-ion battery. Generally speaking, each fault in the lithium-ion battery will have a unique curve threshold to identify fault and normal states. Further, Figure18 -20 compare the fault detection capabilities of the adaptive threshold approach resulting from the fixed threshold (pink sold line).

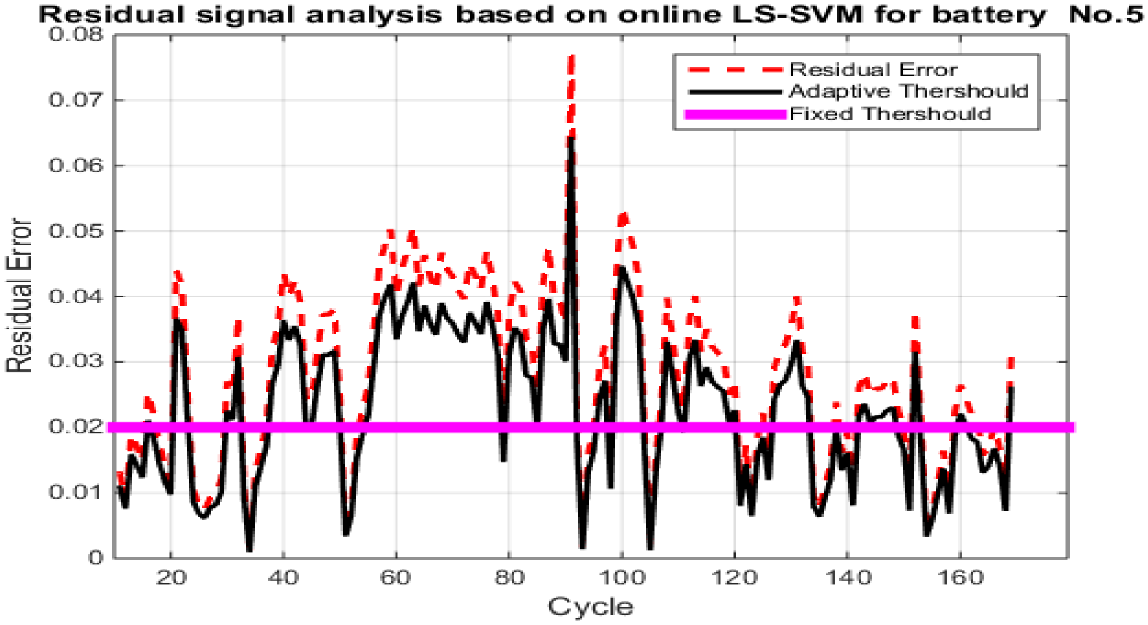


Figure 18: Online LS-SVM Residual Signal (red dashed) based Adaptive Threshold (black solid line) for Capacity Degradation of Lithium-Ion No. 5, Test 1.

It is observed from Figure18-20 that the adaptive threshold detects fault very quickly: the trivial incipient fault is detected at initial cycles (0-2 cycles) for all three cases. Whereas these faults are not detected based on the fixed threshold for battery number 5, and battery number 7. However, the level of the fixed threshold is too small. As mentioned before, the three batteries (see Figure16) experience incipient faults only and the severity of faults increases over the number of cycles. The three plots prove that the MW-LSSVM method is not only reflecting the change of the capacity fault, but also it is detecting the fault correctly at each time window. For example, Figure 18 shows the residual error for battery No.5. The result shows the adaptive threshold and fixed threshold based on MWLS-SVM perform differently. The adaptive threshold is much more significant, in terms of fault detection accuracy and reduction of false alarms, than the fixed threshold.

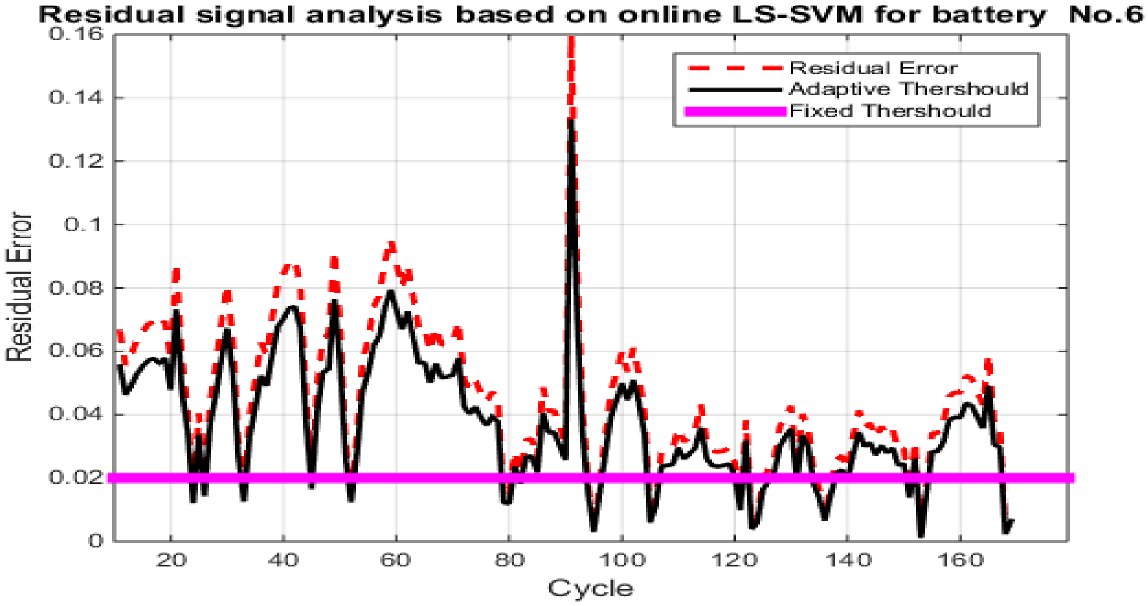


Figure 19: Online LS-SVM Residual Signal (red dashed) based Adaptive Threshold (black solid line) for Capacity Degradation of Lithium-Ion No. 6, Test 2.

To further explore the effectiveness of the MWLS-SVM based on adaptive threshold, we consider Battery No.7. The capacity fade always decreases monotonically, and it doesn’t experience a fast incipient degradation until end of life; it can be seen from Figure 20 that the proposed scheme is able to track the incipient fault error within 0.01 levels, and it prevents false alarms. On the other hand, the fixed threshold fails to reflect the change of residual error, and produces many false alarms during the operation. Moreover, Battery No.6 experienced a fast incipient fault as shown in Figure19. The fixed threshold produces false alarms, while the online proposal fault technique based on adaptive threshold produces no false alarms, and identifies the fault quickly. Also the curve of the adaptive threshold reflects the change of residual compared to the fixed threshold.

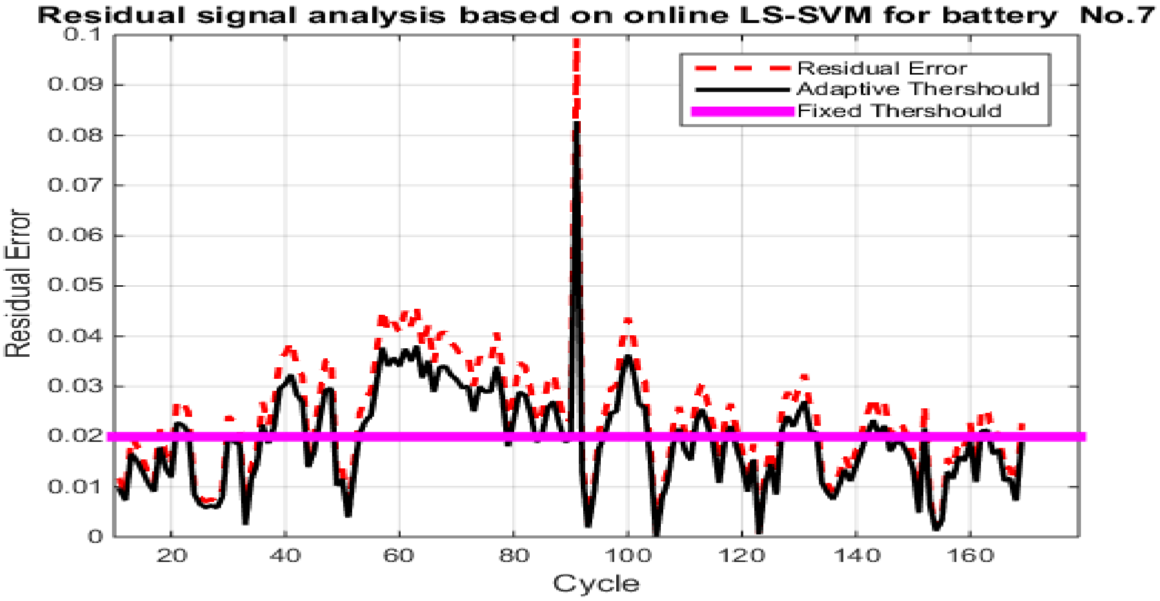


Figure 20: Online LS-SVM Residual Signal (red dashed) based Adaptive Threshold (black solid line) for Capacity Degradation of Lithium-Ion No. 7, Test 3.

It is observed from Figure18 -20 that the residual fault associated with the batteries No.5, 6, and 7 always remained above its fixed threshold after some time, while these residuals don't exceed their fixed thresholds in the case of the off line LS-SVM (see Fig.17). The reason behind that is the offline LS-SVM generally failed with ambiguous information, e.g. tracking a different operation. The online LS-SVM, however, is capable of tracking the variations of the battery parameters and it reflects the time-variance of the system parameters.

The proposed test shows the MWLS-SVM based on adaptive threshold not only detects the fault quickly, but also reflects the change of the residual. LS-SVM can be implemented easily without the need to understand the underlying physical mechanisms of the system; it has low computation cost, and it has the ability to train small samples. The results from the Lithium-ion battery dataset prove the effectiveness of the proposed method for fault diagnosis compared with offline fixed threshold.

3.6 Parameter Turning Using Particle Swarm Optimization (PSO)

LS-SVM is gaining popularity among industry and the academic community. In fact, LS-SVM has many attractive features: it computes the global solution, models non-linear problem, is highly generalized, and its performance is not dependent on dimensionality of the input space.. In spite of the promising results, LS-SVM has a drawback that limits the use of LS-SVM on many application and academic fields: there are free parameters (the regularization parameters(γ), and bandwidth of RBF Kernel(σ^2)). Where the quality of LS-SVM depends on a proper setting of value of the free parameters to ensure a good generalization performance [150, 151, 152]. For this reason, parameters need to be defined by the user. Many LS-SVM studies are performed by an expert who has a good understanding of SVM application [56,58,63]. The biggest challenge encountered in setting up the LS-SVM model is how to choose appropriate parameter values in order to obtain a good generalization of classification. The model selection of seeking an optimal parameter value is very difficult to select by trial and error. There are various optimization techniques; including genetic algorithms (GA), simulated annealing algorithms (SA), immune algorithms (IA), and particle swarm optimization (PSO) have been applied to tackle the model selection problem [155,158,159,151,160]. As suggested in [154], an optimized SVM by Particle Swarm Optimization (PSO) has many merits such as simple concept, higher accuracy, and faster convergence rate than other techniques including Genetic Algorithms (GA), Simulated Annealing (SA), and 10- field cross validation. Moreover, it has been successfully applied in many fields [156,157].

Particle swarm optimization (PSO) (Kennedy & Eberhart, 1995) is a probabilistic heuristic evolutionary algorithm which was originally developed to solve non-linear optimization problems. PSO is derived from simulation of social behaviors, such as birds flocking or fish schooling in finding foods. PSO performs searches based on a population (called a "swarm") of individuals (called "particles"), and each particle has a current position vector and a velocity vector. At each iteration, all the particles move in the multidimensional space in order to find the global optima [153, 155, 145].

In order to validate the performance of the PSO-LSSVM, PSO is used Mean Absolute Percentage Error (MAPE) to assess the quality of a solution. The performance metrics are shown by the following equation:

$$\text{MAPE(i)}=\frac{1}{L}\sum_{i=1}^{L} \left| \frac{100\, e(i)}{r(i)} \right| \qquad 3.42$$

Where e (i) Error defines deviation from desired output, L is the number of training data, r(i) is the length of the turning data. Eq. (3.42) serves as the estimation error index for identifying suitable parameters. MAPE is relatively easy to calculate, and the results are easily interpreted.

The optimization process base on PSO is characterized in the Figure. 21 for Battery No.5. The curve of fitness with 50 iterations is used to reach the convergence with an optimal parameter value of λ=5000, and σ^2= 0.01 and the minimum MAPE error = 0.0083. It is observed from Figure 21 that PSO converges after a little iteration (3 or 4). Figures 21 show that regularization of parameters has greater growth in the early iterations of the evolution.

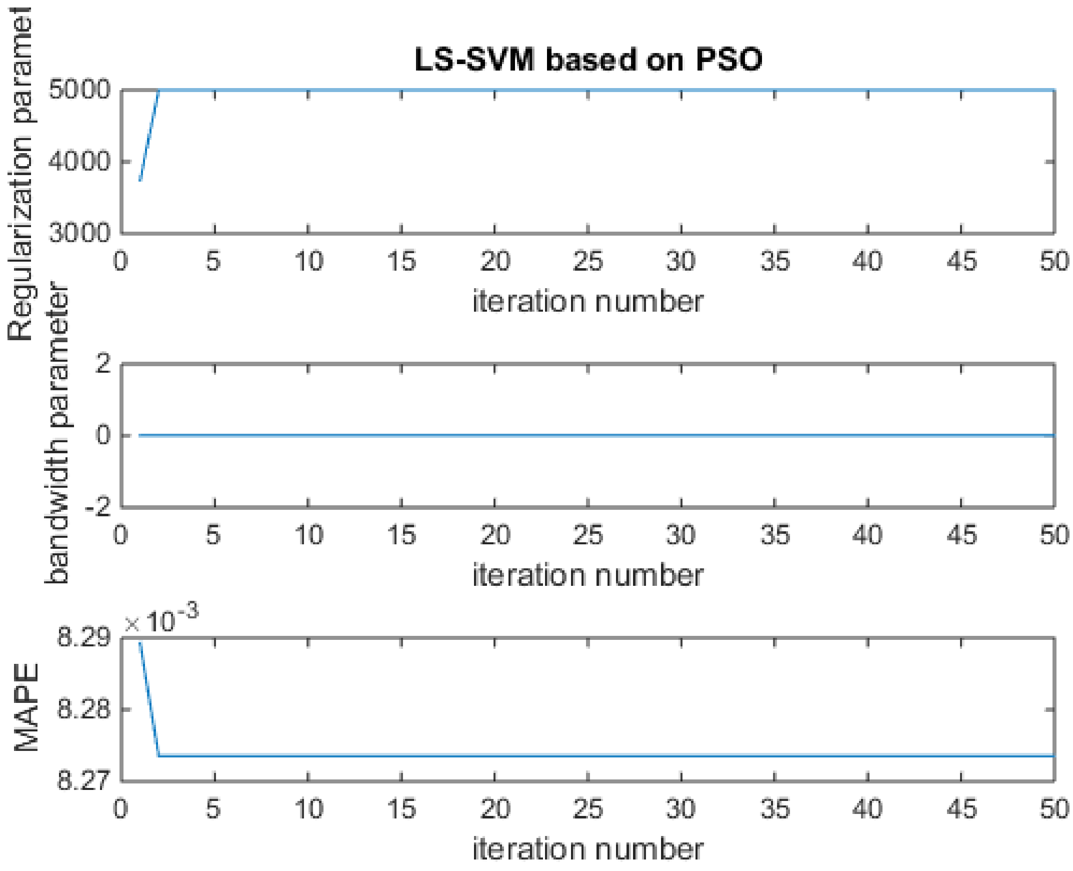


Figure 21: Regularization Parameters, Kernel Bandwidth Parameters Optimization for Battery No. 5 using PSO. The Fitness Function Validation based on MAPE Convergence in about 2 to 3 Iterations with an Optimal $\boldsymbol{\lambda = 5000}$, and $\boldsymbol{\sigma^2 = 0.01}$.

Table 2 shows the result of running the PSOLS-SVM algorithms in terms of statistical results and reports the solution found for regularization parameters λ, bandwidth parameter σ of RBF kernel, time consumption, and fitness function value of 3 independent batteries data set.

Table 5: The Accuracy and Computational Time of PSO LS-SVM Performance Results for the Battery No. 5, 6, and 7 dataset.

Optimization Analysis		**Model**			
		Parameters		**Performances**	
		γ	σ^2	**Time**	**MAPE**
Battery #5	**PSO**	5000	0.0100	42.268107	0.83
Battery#6	**PSO**	5000	0.0100	55.473687	3.95
Battery#7	**PSO**	5000	0.0100	41.641562	7.02

Several observations can be made based on the result in Table 2. We can notice that all solution provided for battery No 5, 6, and 7 based on PSO LS-SVR algorithms present higher accuracy, e.g., Minimum Mean Absolute Percentage Error (MAPE), and better performance after 50 iterations. From the Table2, it can be inferred that PSO has fast convergence rate with small training error. Therefore, PSO LS-SVRM is most suitable for the fault diagnosis and failure prognosis for Lithium ion Battery.

Chapter 4

Memory Particle Filter Based Least Square Support Vector Machine Frame Work for Failure Prognostic

We combine Memory-Particle Filter and LS-SVM, addressing the "supervised learning" scenario, in which the physical failure model is not available. This extension of Particle Filter (PF) framework provides a way of handling the absence of stochastic physical failure model. At the same time, LS-SVM can approximate the posterior density effectively in cases of a small or large training sample. This chapter organization is as follows. The extension of the particle filter in literature is provided in section 4.2.1 Review of the Sequential Monte Carlo Methods "Standard particle filter" approach is presented in section 4.2.2. The introduction of Memory-Particle Filter is provided in section 4.3. The methodology of the prognostic scheme of the proposed Memory particle filter for real time long –term prognostic is presented in section 4.4. Finally, the case study is presented where growing capacity fade on Li-ion batteries is predicted using the proposed algorithm in sections 4.5 and 4.6.

4.1 Introduction

Prognostics is the key motivation in developing next generation maintenance because it indicates in advance whether the system can attain its expected life. Prognostic is an emerging technique in both the research and the industrial communities, and is based on information collected via a sensor for estimating the Reminding Useful life (RUL). Prognostic has been applied to the field of maintenance in order to change the system logistic from state mentoring to state management. Accurate RUL allows optimized maintenance scheduling, and eliminates unnecessary maintenance. Moreover, accurate failure prognosis improves the logistics of spare parts to "Just in time," reduces life cost, and increases the system safety and reliability. [85, 82, 93, 170, 171, 24]. As implemented in this thesis, Memory-Particle Filter based on LSSVM framework has the following distinct advantages:

1-The framework eliminates the need for a physical failure model, which leads the particle filter to be more obtainable in the failure prognosis field.

2-The framework is robust when encountering a small training sample. Standard Particle Filter (PF) needs large sample set to properly approximate posterior density of the state evolution.

3-The framework is an on-line learning algorithm, which can be updated when a new data sample is received, thus avoiding batch training.

4- The framework increases the speed of the prognostic algorithms. The LS-SVM can help PFs identify and approximate the important sample more accurately and increase the computation of the weight of posterior.

5- The framework can improve the robustness of likelihood function estimation; in practice the likelihood function is influenced by the noisy observation. LS-SVM is very effective in the presence of noise, and does not suffer from local minima. As result, LS-SVM improves the accuracy of posterior approximation.

4.2 Related Work

4.2.1 Extension of Particle Filter

So far, a number of refinements to particle filters have been introduced in other domains, based on large training sample of past data in absence of physical dynamic model. For example, Mikami et al.[172,173] formulated a novel memory-based particle filter for tackling complex phenomena. The target application is face pose estimation, abrupt change, and recovery from tracking failure caused by occlusions; this method can handle nonlinear, time-variant and non-Markov dynamics. Panagaden and Taluder [174] proposed data centric approach as an extension of the particle filter to track the eye location of tropical cyclones by using historical data when accurate state prediction cannot specify; an explicit status update is not required in this technique since the prior distribution is prediction using historical trends. H. Chen, and H. A. Rahka [175] used the same formulation to develop a particle filter for travel time prediction by using real time and historical data. Moreover, K. Otsuka, et al.[176] extends the memory-based approach for predicting weather image pattern problems without the underlying dynamic models. The prediction is accomplished based on a similar sequence in historical dataset, and it retrieved based on the dissimilarity measure between current data and historical data in the future space.

While in the previous applications [172,173,176,175] the memory-based particle filter is expected to make only short-term predictions, the proposed failure prognostic is based on long-term prediction for Reminding Useful Life (RUL). The main goal of the proposed framework is to prevent a potential failure, and improves the maintenance schedules by knowing how much time is left. Particle filter framework is developed for Lithium-ion battery prognostics that rely on an available historical data only instead of an accurate stochastic physical failure model.

4.2.2 The Sequential Important sampling (SIS) algorithm or PF-based Prognosis

When considering the problem of state tracking in the failure prognosis, the propagation of the physical failure model and the complex system update using observation data is given by

$$x_k = f_k(x_{k-1}) + w_k \qquad 4.1$$

$$z_k = h_k(x_k) + v_k \qquad 4.2$$

Where

f_k : $R^{n_x} \times R^{n_\omega} \to R^{n_x}$ is the state transition function, and a possibly nonlinear vector function.

w_k ,$k \in N$ is independent identical distribution (i.i.d) states noise vector of known distribution, is a possibly non-Gaussian noise.

h_k : $R^{n_x} \times R^{n_v} \to R^{n_z}$ is the measurement function, and a possibly nonlinear vector function.

v_k ,$k \in N$ is independent identical distribution (i.i.d) measurement noise vector of known distribution, is a possibly non-Gaussian noise.

Bayesian learning represents a general probabilistic solution to the problem of dynamic state estimation x_k , given the measurement data related to equipment degradation up to current time k. This probabilistic approach is used to estimate the posterior probability density function (pdf) $p(x_k\,|z_{0:k})$ of the unobservable (target) state variable x_k ,which is inferred based on a sequence of the noise measurements $z_{1:k} = \{z_1, \dots \dots . z_k\}$. The initial distribution of the system state $p(x_0)$ is assumed to be known, and probability transition density is represented by $p(x_k\,|z_{k-1})$ [177,178,103,101,8]. The Chapman-Kolmogorov equation is used to infer the prior probability distribution of the system state x_k at time k (marginal filtering density); it is also called prediction step. The prior density of state at time k can be estimated by transition probability distribution $p(x_k\,|x_{k-1})$, which is given by time update process of equation (4.1)

$$\begin{aligned} p(x_k\,|z_{1:k-1}) &= \int p(x_k\,|x_{k-1}, z_{1:k-1})\, p|x_{k-1}|z_{1:k-1})dx_{k-1} \\ &= \int p(x_k\,|x_{k-1})\, p|x_{k-1}|z_{1:k-1})dx_{k-1} \end{aligned} \quad 4.3$$

At time K, a new measurement z_k is used for updating the prior distribution (update stage) via Bayes theorem, so the posterior distribution $p(x_k\,|z_{1:k})$ of current state x_k is obtained as [179].

$$p(x_k\,|z_{1:k}) = \frac{p(z_k\,|x_k)p(x_k\,|z_{1:k-1})}{p(z_k\,|z_{1:k-1})} \quad 4.4$$

Where $p(z_k\,|x_k)$ is a likelihood function defined by measurement update process of equation (4.2). Normalizing the constant is determined by

$$p(z_k\,|z_{1:k-1}) = \int p(x_k\,|z_{1:k-1})\, p(z_k\,|x_k\,)dx_k \quad 4.5$$

In the sequential Bayesian filtering framework, the recurrent relation of (4.3) and (4.4) give the exact Bayesian solution [177,112]. However, the analytical solution of $p(x_k\,|z_{1:k})$ is difficult to calculate directly, since it requires the evaluation of a complex high-dimensional integral. [178,170,225,].

One way to overcome this challenge is to resort to Sequential Important Sampling (SIS) algorithm or Particle Filter (PF) method. PF is the pioneering contribution of Gordon (1993) over the past two decades [111] for implementing the recursive Bayesian filtering via Monte Carlo simulation. The SIS is called also bootstrap filtering, the condensation algorithm, interaction particle approximations, and survival of the fittest [177,180]. The goal of PF is to estimate the posterior density function by a set of random samples (Particles) for $x_{0:k}^{i}\;\; i = 1,2 \ldots\ldots, k,$ is a set of support points, and their associated with weights $w\,_{k}^{i}\;\; i = 1,2 \ldots\ldots, k,$ where k is the number of particles. Let $\{x_{0:k}^{i}\,, w_{k}^{i}\}_{N}^{i}$ represent a collection of random measurements that characterizes the posterior probability $p(x_k\,|z_{1:k})$. The weight is normalized such that $\sum_i w_k^i$ =1. Then, the joint posterior density at time k is approximate as [180,112]

$$p(x_k\,|z_{1:k}) \approx \sum_{i=1}^{N} w_k^i\, \delta(\,x_k - x_k^i) \;\;,\;\; \sum_i w_k^i \;= 1 \qquad 4.6$$

The weight w_k^i can be recursively updated using the principle importance sampling with important density [225,235]. Let $x_k^i\; \sim \pi(x), i = 1 \ldots\ldots k$ samples are easy to draw from a proposal $\pi(.)$ called an importance density, then a weight approximate to the density $p(x)$ is given by:

$$p(x_k\,) \approx \sum_{i=1}^{N} w_k^i\, \delta(\,x - x^i) \qquad 4.7$$

Where $w^i \;\propto \frac{p(x^i)}{\pi(x^i)}$

Where w^{i} is the normalized weight of the i^{th}particle. Of course $\pi(x)$ is not achievable since we won‘t know the true distribution p(x)[111]. If the samples x_t^i were drawn from an importance density $\pi(x_k^i\,|z_k)$, the weight are given by [100].

$$w_k^i \propto \frac{p(x_k^i|z_k)}{\pi(x_k^i\,|z_k)}$$

The measurement z_k at time instant $p(x_{1:k}z_{1:k})$ is approximated with a new set of samples. Given the set of weights at instant $w_{k-1}^{1:N}$, the weights at instant k may be computed recursively using the weight update equation derived from the principle of importance sampling as[103]:

$$w_{k-1}^i \propto \frac{p\big(z_k\big|x_k^i\,\big)p(x_k^i\,|x_{k-1}^i\,)}{\pi(x_k^i\,|x_{k-1}^i\,, z_k)} \qquad 4.8$$

Where $\pi(x_k^i\,|x_{k-1}^i\,, z_k)$ is important density, which is choosing to generate the particle. When the important density is choosing to be the same as the prior pdf $p(x_k^i\,|x_{k-1}^i\,)$, then the weight update in equation (4.8) becomes as [111,177].

$$w_k^i \propto w_{k-1}^i . p\big(z_k\big|x_k^i\,\big)$$

When the new measurement is collected, weights are updated considering the importance of corresponding particles. According to the calculation of likelihood $p\big(z_k\big|x_k^i\big)$, the smaller error between a prediction and observation data results in large weight. See Table.6 for a quick review of the Particle Filter process. The weight can be updated using equation (4.9).

$$w_k^i = \frac{p\big(z_k\big|x_k^i\,\big)}{\sum_{j=1}^N p\big(z_k\big|x_k^j\big)} \qquad 4.9$$

One common problem during weight updating with SIS particle filter is degeneracy phenomenon [233,231]: after number iterations, the variance of important distribution weight increases over time and becomes progressively skewed, which results in all but one particle having negligible weight whose contribution is not significant to the target. Therefore, the approximation of posterior $p(x_k\,|z_{1:k})$ becomes very poor [112,177]. Although it is impossible to avoid the degeneracy issue [231], literature introduces a resampling technique that can reduce the effects of the degeneracy phenomenon [112]. The basic idea of resampling is to increase the number of particles or increase the efficiency of selecting prior density. Resampling replicates important weight and eliminates sample with low weight in every single step to obtain equal weight.

PF has been applied to realize prediction for the next p steps in the absence of future observation. Assuming the current set of particles and weights are a good approximation of the system state at time t_k, then the prediction of p-step ahead at time t_{k+1}can be approximated by using the law of total probability, by

$$\hat{p}\ (\hat{x}_{k+p}|\hat{p}_{k:k+p-1}) \approx \sum_{i=1}^{N} w_{k+p-1}^{i}\, \hat{p}(\hat{x}_{k+p}^{i}|\hat{x}_{k+p-1}^{i}) \qquad 4.10$$

The particles $\hat{x}_{k+p}^{i}$ are propagated in time using the physical model, while the current particle weights are propagated in time without any change [103].

Table 5: Sampling Important Resampling Particle Filter Algorithm

```
$\{x_k^i, w_k^i\}_{i=1}^N$= SIR [$\{x_k^i, w_k^i, Z_k$]
Initialize particles: generate sample from set $\{x_0^i\}_{i=1}^N$form the initial distribution.
 Prediction steps: Draw predicted sample $x_k^i \sim p(x_k|x_{k-1}^i), i = 1, \dots., N$
Update stage: After measurement $Z_k$ is obtained, the weight for each simple $\breve{w}_k^i = x_{k-1}^i p(x_k|x_{k-1}^i)$, and normalize $w_k^i = \frac{\breve{x}_k^i}{\sum_{i=1}^N \breve{w}_k^i}, i = 1, \dots., N$.
Resampling: initialize the Cumulative density function: $c_1 = 0$
For $i = 2: N$
Construct CDF: $c_1 = c_i + w_k^i$
End For
```

```
Start of the CDF:$i = 1$
Draw a starting point : $u_1 \sim U[0 + N^{-1}]$
For $j = 1: N$
CDF: $u_1 + U + N^{-1}(j - 1)$
While $u_j > c_i$
$i = i + 1$
End while
Assign sample $x_k^j = x_k^i$
```

4.3 Memory Particle Filter based on Least Square Support Vector Machine (M-PF SLSVM) Failure Prognosis Framework

In our research the traditional particle filter cannot be directly used for failure prognosis, since the physical failure model is not available or it is impossible to derive an accurate state model. However, historical failure data of degradation of a system under consideration may be available, and it can be used to solve this challenge. The main goal of M-PF is to use the prior distribution of the degradation state in the future time by weight most similar in reference sequence to the recent length of degradation [172,172,176]. The similarity between real-time degradation and reference degradation patterns can be determined based on degree of closeness of their paths.

This thesis presents a framework called Memory Particle Filter based on Least Square Support Vector Machine (M-PF SLSVM) with the aim of estimating RUL for the degrading lithium-ion battery. A real-time LS-SVM and a bank of historical failure data is used in the failure prognosis scheme, where **N** is the number of nonlinear incipient or abrupt failures described as trends of degradation. The historical data of the failure for the equipment under consideration are used here to provide a pool of information that represents an entire state sequence including trends of degradation sequence and replacements of physical failure model [174,173, 175].

The M-PF framework is analogous to human behavior for forecasting some events by recalling past patterns that are similar to the present pattern, and helping to make the decision [76,172,173]. This research covers incipient faults (slowly developing) and abrupt failures because incipient faults are very important in maintenance activities [122] and abrupt failures are often an indication of imminent breakdown of the system.

The proposed M-PF LSSVM framework consists of the post-failure bank and real time LS-SVM; the LS-SVM here is the instantaneous fault detection of incipient or abrupt fault. Under healthy operation conditions, only the LS-SVM fault detection is active to monitor the process for any fault. Once the fault is identified and detected, then M-PF scheme will be activated.

4.4 Methodology

The architecture of proposed M-PF LSSVM failure prognosis consists four stages: bank of historical failure tracks, online real-time fault detection based on LS-SVM, retrieval Process, and failure prediction. The flowchart shown in Figure 22 explains the framework of M-PF LSSVM. At first, the fault will be diagnosed at time t_k and the length of failure prediction steps is P.

Remark 1: It is assumed that the bank of N historical nonlinear incipient and abrupt failures scenarios are available (hereafter called reference degradation), representative of state of evolution of the faults process during a different historical failure time. The reference degradation is listed all the failure, i.e. From the start of the fault point until the degradation reaches predefined failure threshold. This is subject to assumptions (1, 2, and 3 in the chapter I) which enable the data to use in the following steps.

Remark 2: Collect condition monitoring data: LSSVM learning algorithms are used based on sliding time windows size L in order to continuously to monitor the system. Each time windows of observation data are treated as an independent time series. Under normal operating conditions (without faults), the proposed M-PFLSSVM failure prognosis is the only monitoring mechanism for the system. After fault detection and identification, the M-PF LSSVM begins sampling historical bank data.

When a fault is detected, the states-transition and measurement updates defined in equation 4.1 and 4.2 are used to propose particle filter, and both of them are nonlinear in our application. Here $z_{1:k}$ denotes the real-time degradation from the past until the current time k $z_{1:k} =\{z_{t-L+1}, z_{t-L+2}, \dots \dots z\}$. In the memory bank, matching reference degradation is denoted as $y_{1:k} =\{y_{T-L+1}, y_{T-L+2}, \dots \dots y\}$. Note that the length of prior degradation is not equal due to the different life service of the system, where L denotes the length of the data sequence. $d(x, y)$ Represents the distance between two degradation patterns x, y. The distance is defined as distance between real-time degradation and reference degradation that is calculated by Eq. (4.11). The result is represented as the degree of dissimilarity between these two degradation trajectories.

$$d(x, y) = z_k - y_k^{(i)} \qquad 4.11$$

Where $y_k^{(i)}$ subset holds retrieval candidate degradation similar to real time degradation that represents the degradation process until time k in the historical data, z_k represents the real time degradation until time k in measurement. Thus, dissimilarity between real time degradation and each of the candidate's reference degradations can be used to choose the best matching trajectory [174,176].

The degradation reference at time k is defined as state variablex_k, where each particle corresponds to one of the complete state tracks in the reference database. The state variable x_k is approximated by a set of particles $\{x_k^i\}_{i=1}^N$ and each x_k^i corresponds to reference historical degradation time sequence $[\Phi_{ref}(v_k^{(i)} j_k^{(i)} - L + 1)]$,$[\Phi_{ref}\ (v_k^{(i)} j_k^{(i)} - L + 2)], \dots, [\Phi_{ref}(v_k^{(i)} j_k^{(i)} - L + 1)]$, represented by the tail $y_k^{(i)}(v_k^{(i)}, j_k^{(i)})$ where $v_k^{(i)}$ denotes the index of the historical degradation (in our case it will be the temperature 24 $\circ\, C$), and $j_k^{(i)}$ is the index time associated with $v_k^{(i)}$. So each x_k^i is associated with degradation reference pattern $y_k^{(i)}$, which is used to match with real-time degradation sequence and calculate the weight of each particle x_k based on dissimilarity between real time and the historical degradation [175,174].

When the incipient or abrupt fault is detected by online LSSVM; the output of LSSVM is used as real time degradation from the past to the current time. The update of measurement degradation data form z_{k-1} to z_k is conducted by shifting the data, moving windows one step ahead. In this way, the particle or the reference degradation patterns update from x_{k-1} to x_k by shifting data sequence, one step ahead along the data reference of historical incipient failure scenarios. For each particle, the corresponding degradation pattern $y_k^{(i)}$ can be derived according to the relationship with $x_k^{(i)}$ represented by $y_k = h_k(\ x_k)$. The associated weight $w_k^{(i)}$ is calculated as the likelihood $(z_k|x_k^{(i)}\ )$, which is computed by comparing dissimilarity between real time and reference degradation $p(z_k - y_k^{(i)})$. In our research, we assume the likelihood function to be a normal distribution $N(0,1)$. Therefore, the distribution of reference degradation on the next time interval $k+1$ can be derived according to the relationship represented as $\{y_{k+1}^{(i)}, w_k^{(i)}\}_{i=1}^{N}$, and the associated weight $w_k^{(i)}$ can be calculated as the likelihood $p(z_k\ \text{I} x_t^i)$. For multi-step ahead with prediction horizon $k+p$, the propagation along reference degradation time sequence on dataset can be iteratively conducted, but the same weight updated by the current measurement is maintained for each particle. So the estimating the Remaining Useful life (RUL) is predicted as $\{y_{k+p}^{(i)}, w_k^{(i)}\}_{i=1}^{N}[111]$.

Figure 22 and Table 4 show schematics of computation of the proposed framework of Memory-Particle Filter based Least Square Support Vector Machine in general case. The procedure for this is as follows:

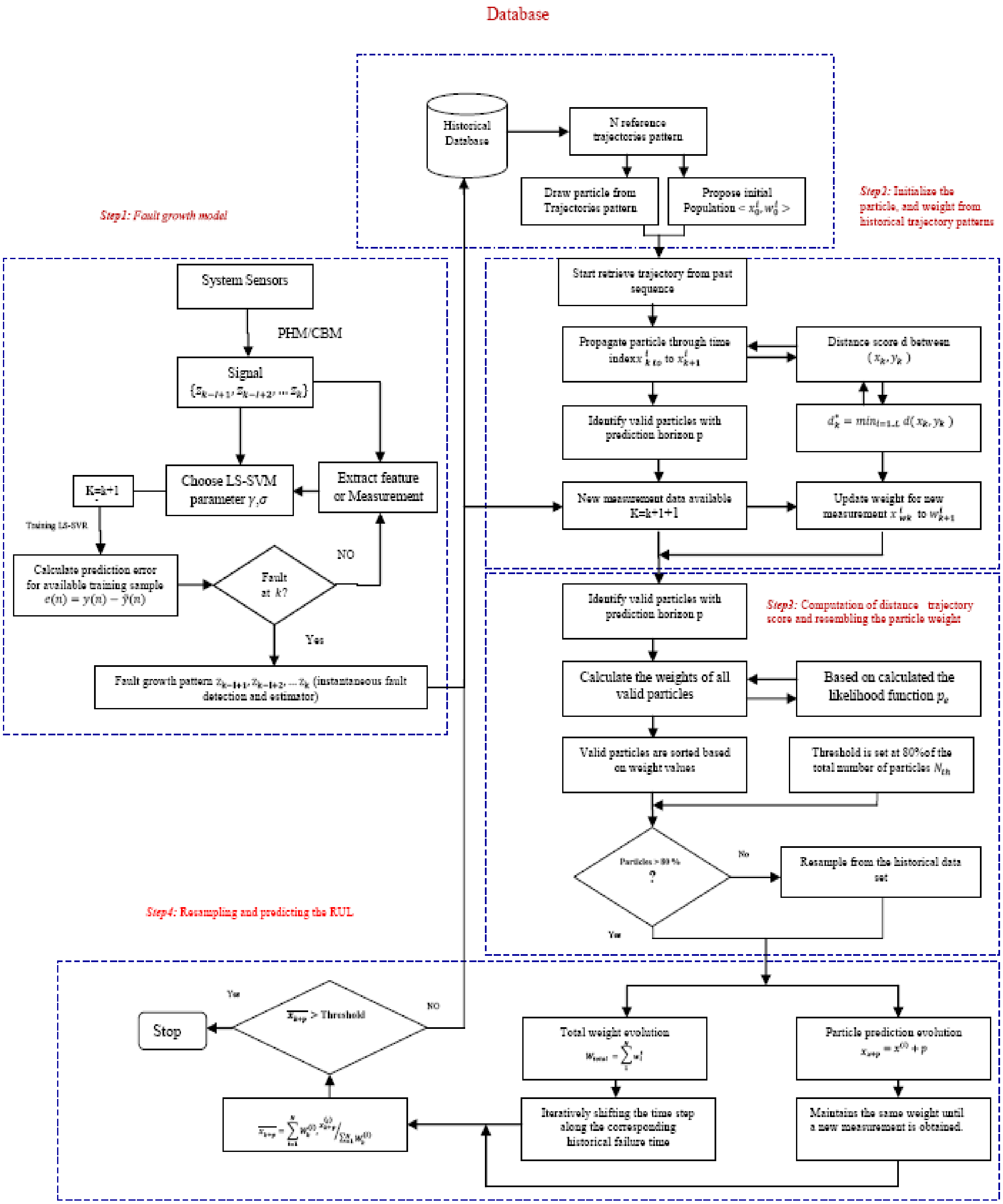

Database
Historical Database
N reference trajectories pattern
Draw particle from Trajectories pattern
Propose initial Population < x_0^i, w_0^i >
Step1: Fault growth model
Step2: Initialize the particle, and weight from historical trajectory patterns
System Sensors
PHM/CBM
Signal {$z_{k-l+1}, z_{k-l+2}, \dots z_k$}
Choose LS-SVM parameter γ, σ
Extract feature or Measurement
K=k+1
Training LS-SVR
Calculate prediction error for available training sample $e(n) = y(n) - \hat{y}(n)$
Fault at k?
NO
Yes
Fault growth pattern $z_{k-l+1}, z_{k-l+2}, \dots z_k$ (instantaneous fault detection and estimator)
Start retrieve trajectory from past sequence
Propagate particle through time indexx $x_{k\,to}^i$ to x_{k+1}^i
Distance score d between (x_k, y_k)
Identify valid particles with prediction horizon p
$d_k^* = min_{i=1..L}\, d(x_k, y_k)$
New measurement data available K=k+1+1
Update weight for new measurement x_{wk}^i to w_{k+1}^i
Identify valid particles with prediction horizon p
Step3: Computation of distance trajectory score and resembling the particle weight
Calculate the weights of all valid particles
Based on calculated the likelihood function p_e
Valid particles are sorted based on weight values
Threshold is set at 80%of the total number of particles N_{th}
Particles > 80 % ?
No
Resample from the historical data set
Yes
Step4: Resampling and predicting the RUL
Yes
Stop
$\overline{x_{k+p}}$ > Threshold
NO
Total weight evolution $W_{total} = \sum_{1}^{N} w_k^i$
Particle prediction evolution $x_{k+p} = x^{(i)} + p$
$\overline{x_{k+p}} = \sum_{i=1}^{N} W_k^{(i)} \cdot x_{k+p}^{(i)} / \sum_{i=1}^{N} W_k^{(i)}$
Iteratively shifting the time step along the corresponding historical failure time
Maintains the same weight until a new measurement is obtained.

Figure 22: The Flow Chart of the Memory Particle Filter based Data-Driven Approach.

Step 1: Initialization

After the incipient or abrupt fault type is detected during the real-time operation condition, the M-PF prognostic framework becomes active. Each particle $x_0^{(i)}$ represents complete historical failure scenarios in the database. The process is initialized by randomly selecting particle from failure in historical database index $\upsilon_0^{(i)}$ temperature $24°C$ (for Lithium-ion battery case study) and the corresponding $j_0^{(i)}$ time in the historical data set. The real –time degradation sequence of system and reference degradation in the database are Φ_{ref} $(\upsilon_k^{(i)} j_k^{(i)})$, and $\Phi_{Real}(\upsilon_k^{(i)} j_k^{(i)})$.

Step 2: Production Process

Unlike Sequential Important Sampling (SIS), where each particle is updated by a sampling from the prior distribution estimation [177,178,177], in the proposed framework, each particle corresponds to one of the complete failure scenarios in the reference database, and each particle is propagated along historical failure reference by extending the time index $j_t^{(i)}$ one step ahead. At every time step L, it is followed by recognizing the valid particle. The valid particle provides a sufficient time interval buffer considering the prediction horizontal p. whereas the invalid particles reference cannot provide sufficient prediction horizontal p, or distance between particle and real time degradation measurement is too far [172]. The valid particle with ability to predict P-step horizon is noted as $\Psi_k$for for k^{th} time interval [174]. Overview and explanation for the steps of M-PF LS-SVM algorithms is presented in Table.4.

Step 3: Measurement Updating Process

Compared to the conventional SIS particle filter, the most recent measurements are used to update the weight. The new updating weight is used to resample the particles [177,178]. In the proposed algorithm, the measurement update process is used to calculate the weights of all valid particles inΨ_k . For each valid particlex_i, the weight is calculated by the likelihood function with input of corresponding degradation pattern

$y_k^{(i)}$ and real time degradation z_k. Likelihood function relies on $Eq.(4.11)$ on the computing the distance between real-time degradation and reference degradation [175,174]. Afterward, all particles are sorted in descending order corresponding to their weight, and particles are divided into two categories. The first category includes the top N particles, representing the sequence that is more similar to real time degradation or particle with more weight; the second category or the result $(N - N_{th})$ includes the invalid particle with negligible weight, or particles Which cannot provide sufficient prediction horizon[174,172].

Step 4: Resampling Process

The SIS particle filters have inherent limitation in the degeneracy phenomenon or less diversity caused by sampling from discrete proposal distribution. Resampling is required in the SIS algorithms to tackle this limitation [178,181,177]. However, the presented algorithm will suffer from the degeneracy problem, too. In general, this issue can be overcome by increasing the number of particles, or by efficiently selecting important samples such as Sampling importance Resampling (SIR). SIR is employed to eliminate a sample with small weight and multiple samples with high weight [178]. Traditional threshold-based resampling algorithms include stratified, residual, branching correcting, and systematic resampling. They result in nondeterministic threshold, are time consuming, and increase complexity [182]. We propose deterministic resampling to reduce complexity and save computation for real- time. We refer to partial resampling instead of the traditional resampling technique (PR)[182].

In the resampling process, the rest $(N - N_{th})$ particles in the second category with negligible weight will be re-selected from reference historical degradation database. During the resampling process, only the historical data that have degradation pattern similar to real time degradation are selected to increase the efficiency of the failure propagation. Consequently, the historical degradation pattern with small dissimilarity to real time degradation has a large chance of being selected in resampling process

[175,182].

Step 5: Predicting RUL Scheme

To generate long-term prediction of the system state, the set of particles and weight $\{ \mathcal{X}_k^i ,\mathcal{w}_k^i\}_{i=1}^N$ which define the current posterior degradation states estimation are used as initial condition. Each particle is individually propagated into the future by recursively applying state transition model Eq.(4.1), until the value of each particle is bounded by the predefined failure threshold. Standard PF propagation models have been discussed in [103,100]. The memory method is used for short-term forecasting because the short-term prediction tends to be similar to the state in the recent past. To tackle the challenge of long-term prediction, we will use many historical failure trajectories. Moreover, the partial resampling process will be used to evaluate and select the best matching of historical failure sequence to real-time degradation [175,172]. The Long-term forecasting of the RUL is conducted by iteratively shifting the time index along the corresponding reference degradation; at the same time, each particle maintains the same weight until no new measurement is obtained. Consequently, aggregating the result from all particles can estimate the Remaining Useful Life (RUL) distribution. The expected average point of the failure is the point where extrapolation particles are intersected with the failure threshold, or the end of its serviceable life. Therefore, the RUL prediction is calculated as weight average of failure time for each particle [175,182].

Table 4: Real –Time Reliability Prediction for Complex System Based on Memory-Particle and LS-SVM.

$[\{x_k^{(i)}\}_{k=1}^N = MPF[\{x_{k=1}^{(i)}\}_{i=1}^N , z_k, \Phi]$

Where z_k is real-ime Φ Set of degradation related to the equipment monitor that is used as reference.

Initialize the particle x_k:{ $x_0^i|x_0^{(i)}$=$\Phi_{ref}(v_0^{(i)}, j_0^{(i)}, i \in [1:N]\}$

$For\ i = 1: N$

$v_0^{(i)} = random\ select\ based\ classifiction\ from[1,2, \dots, v]$

```
$j_0^{(i)} = random\ select\ a\ time\ index\ based\ \upsilon_0^{(i)}, i \in [1\colon N]$
Step 1: Time update
Identify valid particle with respect to prediction horizon p
$\delta_k = \{i | j_0^{(i)} \leq \Psi_{\upsilon_k^{(i)}} - p, i \in [1\colon N]$
Propagate the particle by shifting time windows$x_k^{(i)} \sim p(x_k^{(i)} | x_{k-1}^{(i)})$
$\upsilon_k^{(i)} = \upsilon_{k-1}^{(i)}, j_{k-1}^{(i)} + 1, i \in [1\colon N]$
Step 2: Measurement update
$W_k^{(i)} \propto p\left(z_k \middle| x_k^{(i)}\right) = p_{ek}\left(z_k - y_k^{(i)}\right), i \in \delta_k$
Select $Nth$ number of particles with least weight values
For $j = 1\colon Nth$
    $x_k^{(j)} = x_k^{(i)}, w_k^{(j)} = w_k^{(i)}, when\ i = \arg max_{i \in \delta_k} W_k^{(i)}, \delta_k = \delta_k - \{i\}$
End for
Step 3: Resampling
For $j = Nth + 1\colon N$
    Calculate the probability of selecting each historical selection$\gamma_k^n$
    $\gamma_k^n = p_{ek}(z_k - \Phi_{Real}(n, \psi_k^n)$, when $\psi_k^n$ = arg $max_{F \in [L, \Psi_n, \mathrm{p}]}$ $(z_k - \Phi_{Real}(n, F), n \in [1\colon \upsilon]$
    $\upsilon_k^{(j)} = randmly\ select\ residual\ from$ [ 1,2, ... $\upsilon$] acceding to the probability [$\gamma_k^1, \gamma_k^2$, .... $\gamma_k^\upsilon$]
    $x_k^{(i)} = \Phi_{ref}(\upsilon_k^{(j)}, \psi_k^{\upsilon_k^{(j)}})$, $w_k^{(j)} = \gamma_t^{\upsilon_k^{(j)}}$
Step 4: Prediction
 Draw $x_{k+p}^{(i)} \sim p(x_{k+p}^{(i)} \middle| x_k^{(i)}\Big), i \in [1\colon N]$
$\overline{x_{k+p}} = \sum_{i=1}^{N} W_k^{(i)}, {x_{k+p}^{(i)}} \Big/ {\sum_{=1}^{N} W_k^{(i)}}$
```

4.5 Case Study: Remaining Useful life Estimation of Incipient Capacity Failure for Lithium-ion battery

Battery data set from NASA public data repository with a growing capacity fade is chosen as real-world test case (see section 3.5). The objective of the model is to predict the capacity fade of a lithium-ion battery during operation. As was previously mentioned, M-PF prognostic is based on an incipient failure scenario only to ensure the lithium-ion battery continues its function safely throughout its life operation.

The Li-ion Battery is an electrochemical system. Therefore, the battery will have different characteristics under different operation conditions; also the battery doesn’t undergo a complete charge process or a complete discharge process in actual operation, and the capacity of the battery tends to deteriorate after a certain time [10]. Technically, the degradation of the lithium-ion battery is impacted by many factors such as current, voltage, and temperature due to electrochemical structure and the nature of the lithium-ion mechanism. Unlike traditional empirical data-driven approaches for prediction RUL capacity fades, the framework in this research uses Energy Efficiency (EE) and Working Temperature (WT). These physical quantities are first formulated to reduce dimensions of input space and reduce computational complexity. Also EE and WT are used to extract incipient fault growth patterns from the experiment data of Lithium-ion battery.

Developing a Memory-Particle Filter (M-PF) based prognostic for Li-ion battery doesn’t require the physical failure model from electrochemical analysis real-time, multi-step failure prediction under different operation conditions. The presented method uses real-time and historical data to replace the physical failure model.

By analyzing the different groups of Li-ion batteries dataset, we have found that only the first groups have significant degradation features because this group is examined under normal conditions [1]. Figure.23 shows degradation trends of capacity fade for the first group that includes battery Nos. 5, 6, and 7. Also we can notice the capacity fade features evolve over a number of cycles, and capacity fade decreases monotonically. For completeness of the information and physical description of the Lithium-ion battery can be found [2, 3, 6, 8, 11, 14,18].

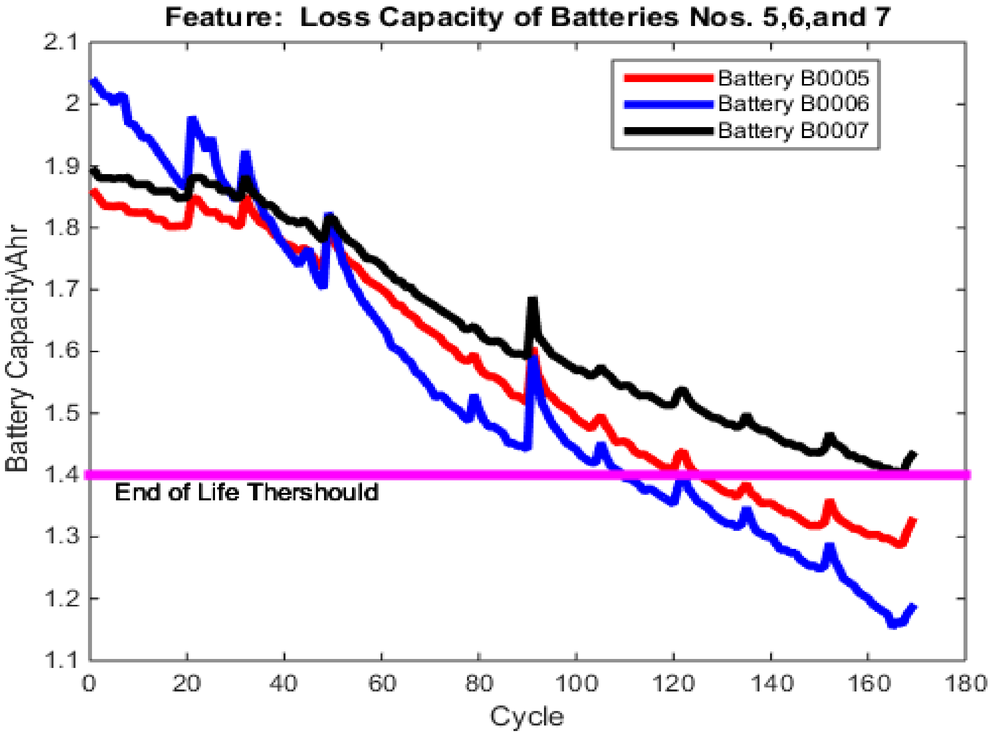


Figure 23: Capacity Degradation at Ambient Temperature 24 for Battery No. 5, 6, and 7. The Feature Correlates (monotonically) decreasing with Number of Cycles.

Figure 23 indicates that the degradation of the different battery increases rapidly during the 22th and 90th cycles, resulting from self-charging during the rest period. This effect is related to the increase in the available capacity of the battery; the explanation for this effect is related to the electrochemical structure and nature of the lithium-ion. During use of Li-ion batteries, some chemical reactions begin and these chemical products appear near the two electrodes; this retards the internal chemical reactions; therefore, the battery needs a short rest period to melt these chemical products [10]. The capacity will gradually decrease with age, heavy usage, and elevated self-charge and temperature. For safety reasons, the experiment was stopped when capacity was below End-of-Life (EoF) threshold; it is accepted that a 30 % of the rated capacity is EoF threshold or 1.4 A.h. It can be observed in Figure 23 that the degradation on battery No. 5 is similar to the one in battery No.6. To illustrate the feasibility and efficiency of M-PF for failure prognosis, the batteries No. 5, 6 and 7 are selected to evaluate our proposed method, and No. 6 will be used for comparison with another time series technique.

In this research, the prognostic is fulfilled at two different starting points respectively (i.e., cycle 55 and cycle 75 after the fault identification), which are selected to validate the prediction results. The implementation of the M-PF based on LS-SVM begins with determining the number of particles from historical data; the number of particles is chosen based on the degradation features of batteries and similar test condition with the temperature 24. Therefore, we select the Batteries No.05, No.06, No.07, No.18, No.54, No.46, No.47, and No.48 as 8 particles to be propagated to failure time.

The End of Life (EoL) of the battery No. 05 is about 124 cycles; the EoL of the battery No. 6 is 112 cycles, and EoL of the battery No.7 is 166 cycles as found in [1]. The results of the method are presented in Figures 23-25 in the form of three plots for battery Nos. 5, 6, and 7, respectively.

Fig.24 shows the testing result when M-PF and LS-SVM are combined using the proposed fusion prognostic framework. The framework shows prediction at 55 cycles (blue star). The prediction error is 4 cycles before the true failure time, and the estimation error is 3.23. Also Fig.24 shows the prediction result at 75 (red asterisk) with known capacity data. The prediction error is 2cycle and estimation error is 1.62. The reason for a higher prediction is that the capacity degradation for battery No. 5 is very similar to the No.6. Therefore, the estimation RUL is better not only in prediction accuracy but also in estimation of small errors.

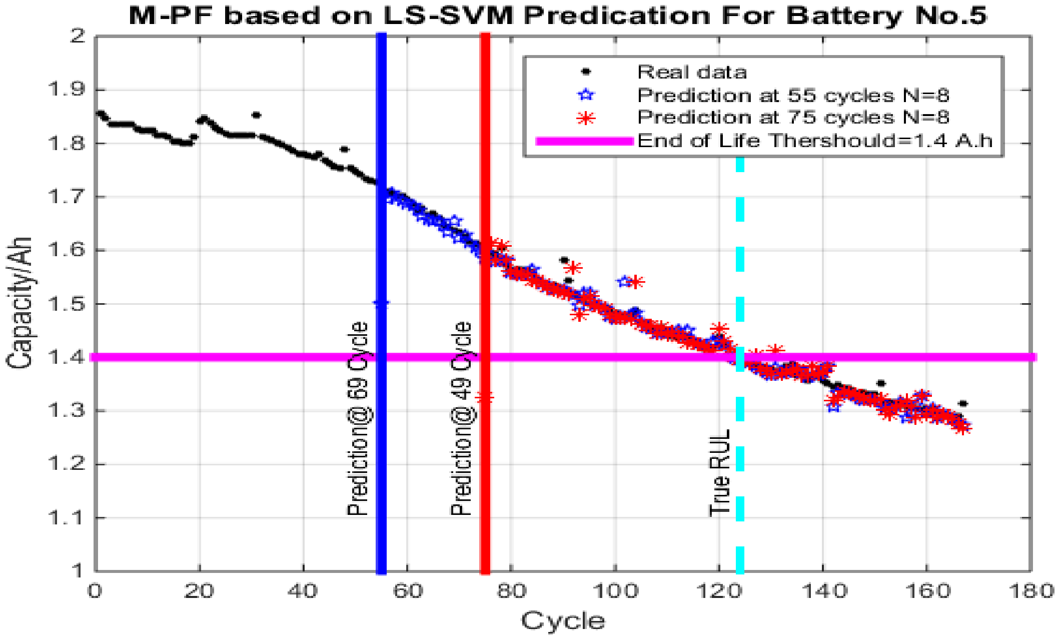


Figure 24: The RUL Prediction Result at 55 and 75 Cycles for the Battery No. 5 with M-PF Based on LS-SVM model, Exparment 1.

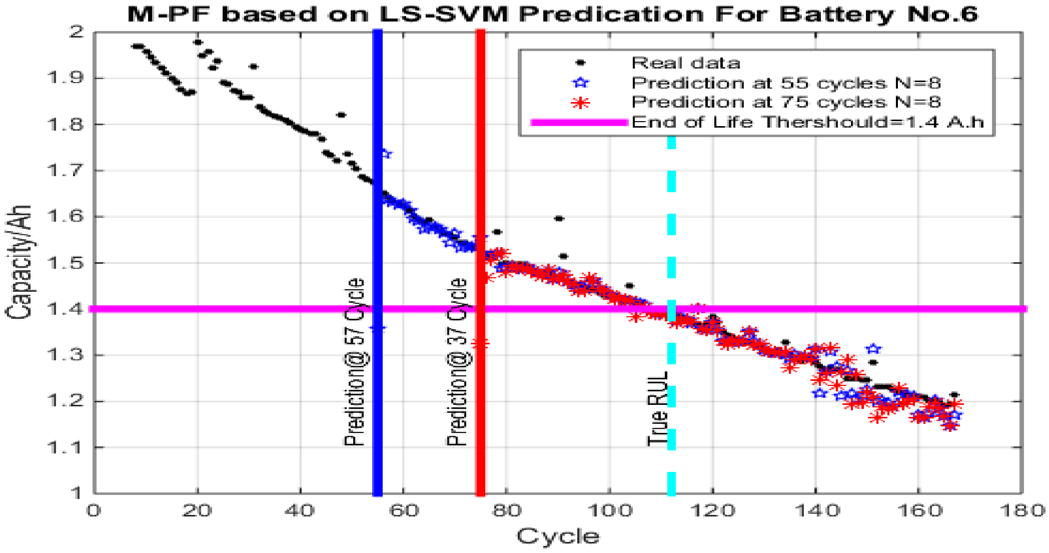


Figure 25: The RUL Prediction Result at 55 and 75 Cycles for the Battery No. 6 with M-PF Based on LS-SVM model, Exparment 2.

Figure 25 shows the remaining useful life estimation result (the battery No.6) at 55 and 75 cycles starting points, respectfully. The prediction results are much closer to the real Time of Failure(ToF)value than the output of the experiment based on battery No. 5, and 7. The prediction errors are 0.45 and 0   away from the true failure because the accuracy of the M-PF depends on the presence of similar failure track in the database. In fact, capacity degradation for battery No.6 is very similar to battery No. 5, and 7. As result, the accuracy of RUL estimation leverages close to real value and the estimate errors are significantly small (see Table 6).

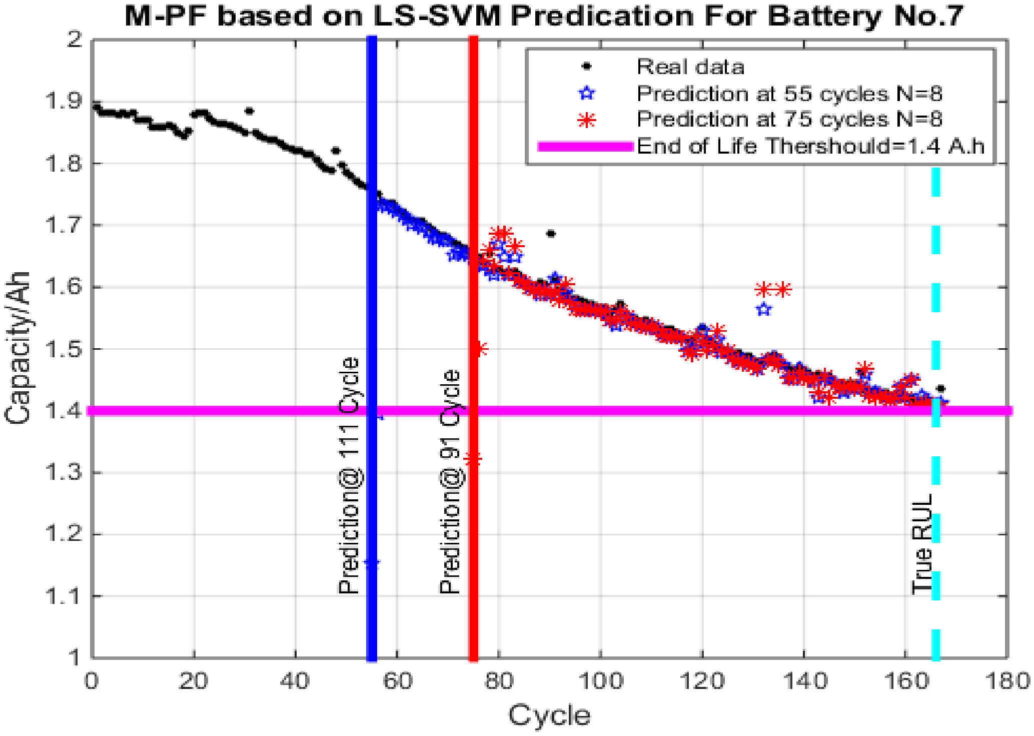


Figure 26: . The RUL Prediction Result at 55 and 75 Cycles for the Battery No. 7 with M-PF Based on LS-SVM model, Exparment 3.

Finally, battery No. 7 is considered, and the estimation results are shown in Figure. 26. One can observe From Figure. 26 and Table 6 that the estimation accuracy with proposal methods compares unfavorably with experiment based on Battery No. 5, and 6. For example, the prediction error is increased by 3.62 from the true failure at 55 cycles, and the prediction error is increased 2.4 from true value cycles at 75 cycles. The reason for this bad result is that capacity degradation for Battery No.7 shows a different degradation feature compared with Battery No. 6, No.5, and other particles available in the dataset.

Table 6: Summary of the Prediction Comparison for Battery No.5, No.6,and No.7

Battery No.		**RUL Comparison**			
		True RUL	**Predicted RUL**	**Error (cycle)**	**Estimation error (%)**
Battery #5	**At cycle 55**	124	120	4	3.23
	At cycle 75	124	122	2	1.62
Battery#6	**At cycle 55**	112	111.55	<1	0.45
	At cycle 75	112	112	0	0
Battery#7	**At cycle 55**	166	160	6	3.62
	At cycle 75	166	162	4	2.4

In conclusion, a data-driven prognostic framework based on Memory-Particle Filter inference framework and LS-SVM is presented. This framework is able to use real time observation and historical data without an accurate physical failure model to predict the RUL for a complex system. The proposal scheme is tested on growth of fade capacity of lithium-ion battery as a real-world test case. It is shown that the test scheme works well and is able to predict RUL. Validation of failure prognosis is an important issue and center stage in Condition Based Maintenance (CBM). In Chapter 5, the performance of the proposal prognostic scheme is compared with Recurrent Neural Networks (RNN), and Backpropagation (BB) Neural Network based prognosis approach.

4.6 Case Study: Remaining Useful life Estimation of Abrupt Capacity Failure for Lithium-ion battery

We have described an algorithm for predicting Remaining Useful Life (RUL) based on incipient fault. Incipient fault is represented as deviation that occurs slowly and develops over time. The importance of the incipient fault prognosis can be found in the reduction of the total cost of maintenance, which is realized by avoiding equipment failure and maximizing the service-life. In general situations, the particle filter can estimate RUL based on incipient fault as well, but the outcome of the PF will be erroneous if there is abrupt failure because it has a very short and fast life time. Abrupt failure is usually modeled as a step-like deviation in the parameter, and it is often an indication of imminent breakdown of the system, as shown in Figure 27. Therefore, fast and early predictions are the main objectives of the abrupt failures prognosis to avoid catastrophic consequences.

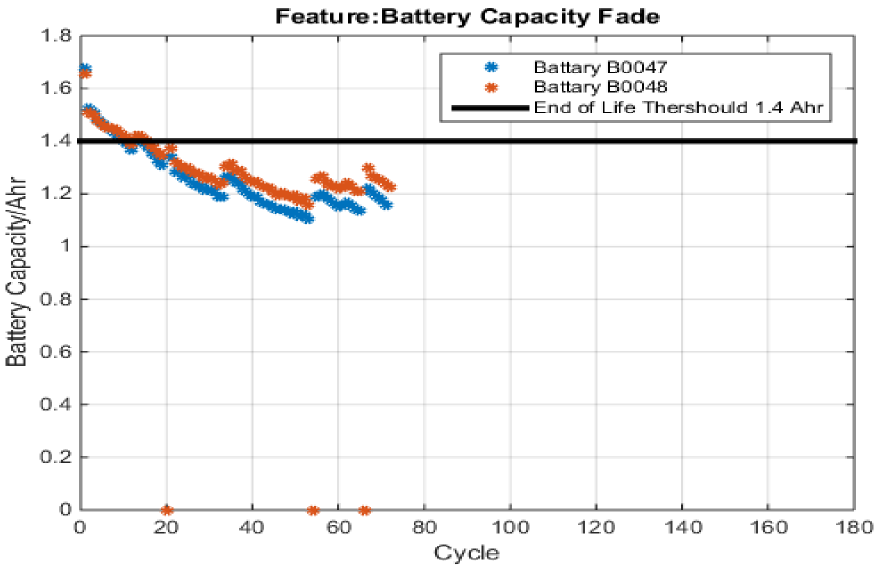


Figure 27: Abrupt Failure of Lithium-Ion Cylindrical Battery for Batteries 47 and 48 at Temperature 4°C.

The physical failure model cannot produces short-term prediction of RUL because of its limited resolution in term of observation and computation; also, traditional data driven approaches are severely limited by the complexity of the dynamic failure precipitation [1]. However, the M-PF relies on the assumption that the more similar current state sequence and historical state sequence degradation are, the more future degradation will continue to be similar. When there is a smaller minimum distance between real time and historical state, larger weight is given to particles. Therefore, the M-PF not only can predict the gradual monotonic fault, but also can predict the abrupt failure effectively.

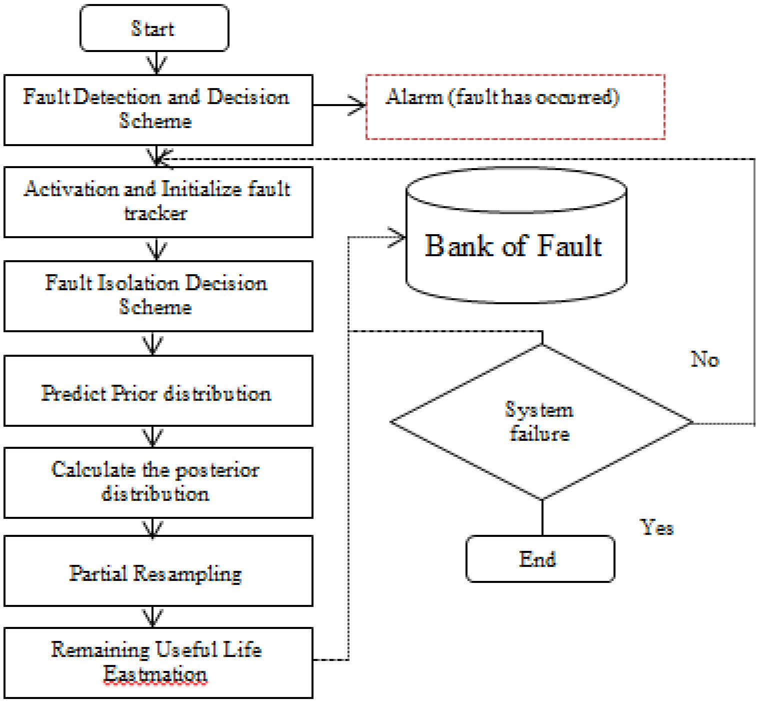

Start
Fault Detection and Decision Scheme
Alarm (fault has occurred)
Activation and Initialize fault tracker
Bank of Fault
Fault Isolation Decision Scheme
No
System failure
Predict Prior distribution
Calculate the posterior distribution
Yes
Partial Resampling
End
Remaining Useful Life Eastmation

Figure 28: Flow chart of Memory-Particle filter of abrupt failure prognosis

The capacity degradation in Figure 27 shows a feature of capacity fade time: capacity is mostly consumed during periods of accumulative damage (after the capacity reaches end of life), while the period of initial fades propagation is relatively short. Therefore, accurate failure prognosis is based on a fast fault tracking of the battery condition, and accurately measures the speed of a parameter's deviation from normal operation. Once the fault is detected, the fault diagnosis is used to isolate the abrupt fault. Each isolation estimator corresponds to one possible type of incipient or abrupt failure (there are only two different types of fault in the datasets in our case). Figure 28 shows the flow of our failure prognosis based on M-PF for failure prognsis.

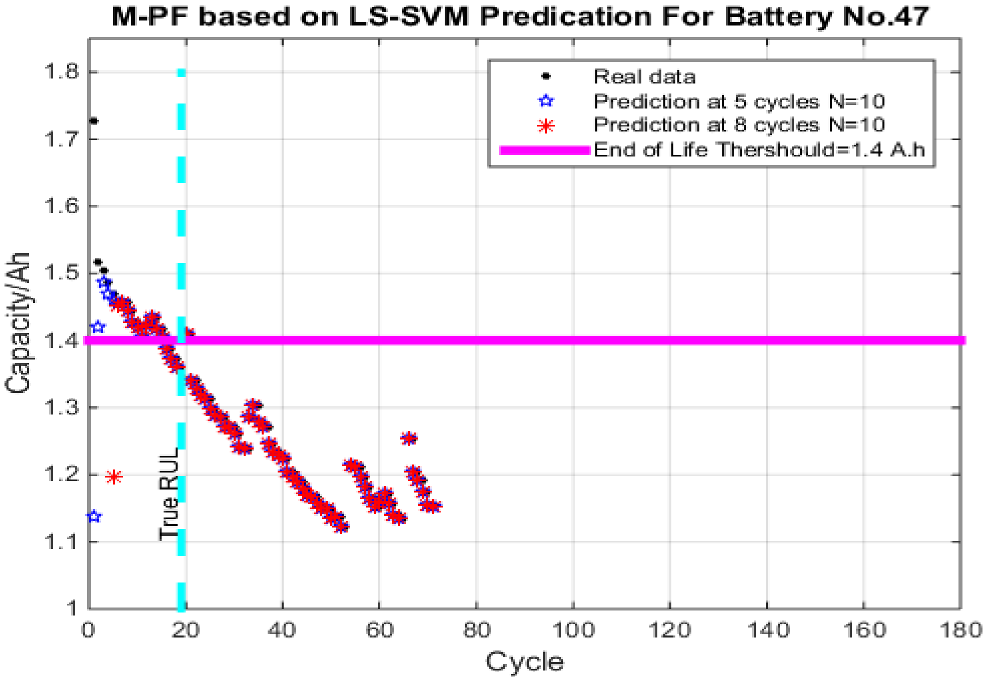


Figure 29: The RUL Prediction Result at 5 and 8 Cycles for the Battery No. 47 with M-PF Based on LS-SVM model,Test 1.

In this research, Memory-Particle filter (M-PF) is proposed to predict the abrupt failure if it is difficult to derive a physical model from the first principal; the proposed algorithm selects particles from a historical database bank and propagates particles using the historical data sequence. In the M-PF, we develop a Partial Resampling (PR) strategy to update particles and to replace invalid or low weight particles in order to provide a sequence similar to real time degradation. RUL process for abrupt failure consists of five steps: initialization, time update, measurement update, partial resampling, and RUL prediction. The details for each step are described in section 4.4. However, the RUL for abrupt failure has two aspects: one is that predicting system state in the near-term future tends to be similar to the system state in the recent past. The other is that not only historical data should be similar, but also its trend in pattern change should be close [176].

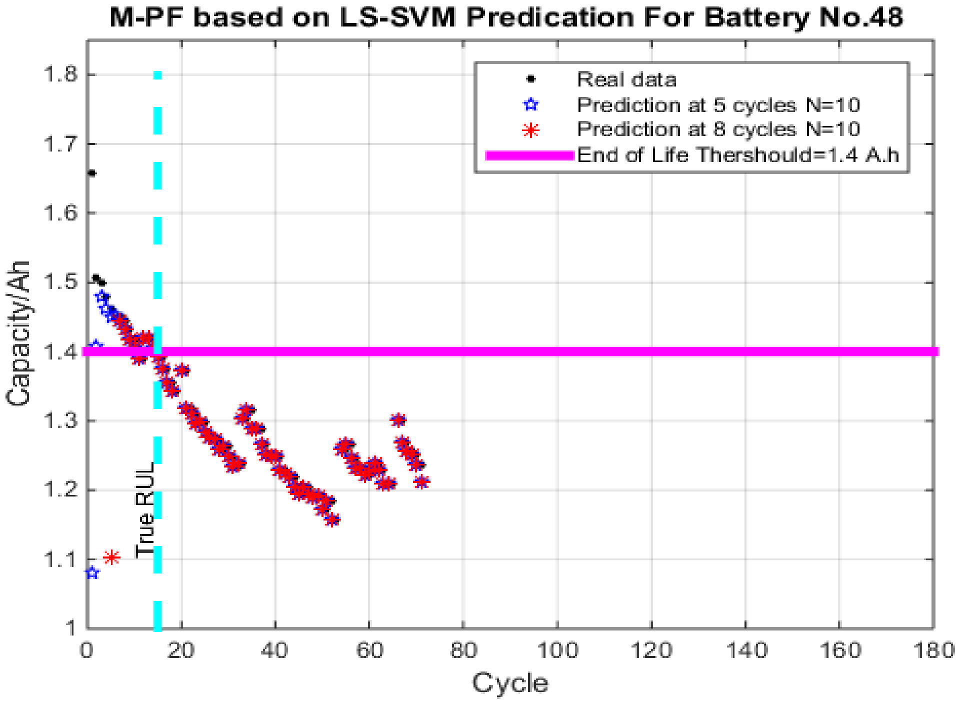


Figure 30: The RUL Prediction Result at 5 and 8 Cycles for the Battery No. 48 with M-PF Based on LS-SVM model, Test 2.

The case study considered is the same as section 1.5 concerning failure data in the capacity fade for Lithium-ion battery. Since the capacity of Batteries Nos. 47 and 48 (see Figure 27) decreases suddenly, and their lifetimes are short (abrupt failure), the data for both batteries are used for training the Memory-Particle Filter based on Least Square

Support Vector Machine. The prognostic is fulfilled at two different starting points respectively (i.e., cycle 5 and cycle 8 after the fault identification), and were arbitrarily selected to validate the prediction results. The implementation of the M-PF begins with determining the number of particles from historical data; the number of particles is chosen based on the degradation features of batteries. Based on this criterion, we selected the Batteries No.44, No.45, No.46, No.47 No.48, No.49, No.50, No.51, No.52, and No.53 as 10 particles to be propagated to failure time. The results are shown in Figure 29 and 30 and Table 7.

Table 7: Summary of the Prediction Comparison for Battery No47, and No.48 for Abrupt Failure

Battery No.		RUL Comparison			
		True RUL	Predicted RUL	Error (cycle)	Estimation error (%)
Battery #47	**At cycle 5**	19	17	2	10.6
	At cycle 8	19	18	1	5.3
Battery#48	**At cycle 5**	16	4	4	25
	At cycle 8	16	16	0	0

Figure 29 shows the M-PFLSSVM predictions based on Battery No.47 at 5 cycles (blue star), and at 8 cycles (red asterisk). The prediction error is 2 cycles away from the true failure. Also the figure shows the prediction result at 8 cycles. The prediction error is 1 cycle away from the true failure. Although the RUL estimates are not too far off the real value, the RUL is not significant because the capacity of battery degrades sharply and has a short life time. Therefore, quick action is needed to avoid unexpected consequences. The M-PF LSSVM based Battery No.48 performs less accurately in tracking the battery capacity than No. 47 at cycles 5, but The M-PF LSSVM gives better result at cycles 8 as shown in Figure 30 and Table 7. Finally, the experiment results shows that Memory-Particle Filter based on Least Square Support Vector Machine can effectively predict not only incipient fault with multiple-step RUL, but also abrupt failure based on current states to determine the end of life for Lithium-ion battery.

Chapter 5

Performance Metrics and Prognostic

While diagnosis evaluation has gained extensive research over the last century; there is no standard definition of acceptable standard evaluation for failure prognosis. In fact, many prognostic research projects have designed algorithms rather than evaluating their performance. Since the goal of this research is to reduce of the occurrence of unscheduled maintenance, it is necessary to justify accepting and deploying the proposed predictor. This research relies on A Saxena et al., [185,186] to evaluate the proposal result; these references present a set of metrics to assess the different prognostics algorithms, where modification may be needed to make them more appropriate when they are used for long-term failure prediction. Four types of errors are presented in section 5.1: Absolute Error (AE), Mean Squared Error (MSE) Mean Absolute Percentage Error (MAPE), and Standard Deviation (SD). To measure the effectiveness of the presented method, the framework is validated and compared with Backpropagation Neural Network (BNN), and Recurrent Neural Network (RNN) on the Lithium-ion battery data set in section 5.2. The effect of number of particles is discussed in section 5.3. Summary of research contribution is highlighted in section 5.4. Finally, future research and the conclusion are outlined in chapter 6 and 6.1 respectively.

5.1 Basic Prognostic Metrics

5.1.1 Accuracy of the Model

In order to measure the effectiveness of the proposed method, three accuracy performance measures are employed for failure prognostics: Absolute Error (AE),Mean Squared Error (MSE), and Mean Absolute Percentage Error (MAPE).

Absolute Error (AE) is defined as the basic measure of how close predicted failure times are to actual failure time. AE Is limited by the fact that existing variability and outliers don’t reflect in the performance.

$$AE = |\hat{y} - y | \qquad 5.1$$

$\hat{y}$ is the of predict failure time, and y actual failure.

An alternative to Absolute Error is Mean Squared Error (MSE). MSE is the average square difference between the predicted failure time and actual Time to Failure (ToF) Eq. (5.2).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y} - y)^2 \qquad 5.2$$

Finally, Mean Absolute Percentage Error (MAPE) is the average absolute percentage error between the failure predicted and actual ToF, MAPE takes into account the difference between the errors observed close to End of Life (EoL), and errors are found far from EoL (Eq. 5.3)

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} |\frac{\hat{y} - y}{{}^{y}/_{y_t}} | \qquad 5.3$$

y_t is the total number of predict failure time.

5.1.2 Precision Based Metrics

An alternative to the previous metrics that determine the exact deviation of the prediction from the actual failure, precision metrics focus on measuring the uncertainty associated with prediction and measure the narrowness of the interval in which the prediction lies.

Standard deviation is a measure that is used to quantify the amount of variability between the actual EoL and the predicted value Eq. (4.15).

$$S = \sqrt{\frac{\sum_{i=1}^{n}(\mid\hat{y} - y \mid m)^2}{n-1}} \qquad 5.4$$

Where m is the sample mean of error, and n number of values (the population)

5.2 Performance Comparison

The proposal Memory-Particle Filter based on LS-SVM prognosis scheme is compared with various popular non-linear data-driven approaches such as Backpropagation Neural Network (BNN) and Recurrent Neural Network (RNN).

Artificial Neural Networks (NNs) have been extensively implemented in prognostic problems. The common advantage of applying ANNs is that they do not require prior knowledge to capture complex phenomena, and they are able to identify and model non-linearity problems, [176,187,179].

5.2.1 Backpropagation Neural Network (BP)

BP is the most popular algorithm to train neural networks. A typical BP always has an input layer, an output layer, and at least one hidden layer. Strictly speaking, there is no limit on the number of hidden layers. However, one or two Layers are sufficient to handle most problems. BP is a method for gradient descent algorithms to minimize the mean square error between the output of a multilayer neural network with respect to its weight and actual output. Sigmoid function is commonly used as a continuous differentiable nonlinear function (Eq.5.5).

$$f(t) = (1 + e^{-t})^{-1} \qquad 5.5$$

If the BP gives a wrong answer, then the weights are corrected so that the error is lessened, so future responses of the BP are more likely to be correct [106]. last mothed for comparison is Recurrent Neural network (RNN).

5.2.2 Recurrent Neural Network (RNN)

RNN has been widely used in previous studies for real-time failure prognosis. There are several methods for structure training of RNN. We propose RNN for estimation of the Remaining Useful Life (RUL). Here, the structure of RNN includes each hidden unit connected to all input units, and output unit is connected to each hidden unit. RNN structure is similar to feedforward algorithms, except that each layer has a recurrent connection that allows the RNN to have a dynamic response to time series. its dynamic characteristics rely on gradient-descent method which computes a real time to complex nonlinear system[106,30].

The end of life predictors are trained based on Energy Efficiency (EE), and Work Temperature (WT). Battery No. 6 is selected to evaluate our proposed method of estimating Reminding Useful Life (RUL), where the true of End of Life (EoL) of No.6 is 112 cycles. Further, the Framework M-PF LSSVM performs system failure prognosis via 8 particles that are employed for P-steps prediction (see section 4.5). Each algorithm is used to predict the RUL at 55 and 75 which require approximately 57 to 40 cycles to learn the degradation trend; algorithms extrapolate the RUL until the end-of-life is reached, i.e. capacity fade prediction reaches predefined failure threshold which is 30 % fade in rated capacity, or 1.4 ampere hour.

The result shown in Figures 31-33 are obtained when Backpropagation (BP),M-PF LSSVM , and Recurrent Neural Networks (RNN) are triggered at 55 and 75 cycles. Figure 31 shows the testing result when Memory-Particle Filter and LSSVM are fused at 55 and 75 cycles. It can be seen from the Figure 31 and Table 8 that the proposed framework provides a more accurate prediction for the End of Life(EoL) than both BP and RNN (see figure 32 and 33).

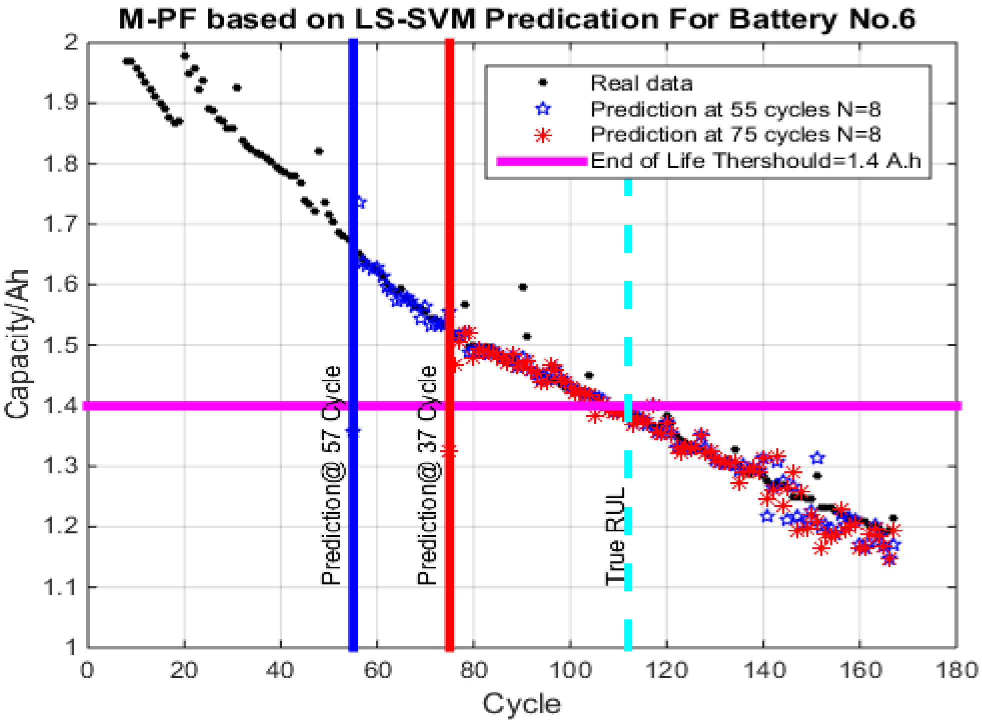


Figure 31: The Remaining Useful Life (RUL)Prediction Result at 55 and 75 Cycles for the Battery No. 7 with M-PF Based on LS-SVM model.

The Memory-Particle Filter based on Least Square Support Vector machine (M-PF LSSVM) decreases RUL error from 0.6 to 0 cycles, which is smaller than the error associated with the BP predictor (3 to 2 cycles early) and the RNN predictor (12 to 1 cycles early from real value). RUL estimation at cycles 55 provides the worst performance for the three methods, and the RNN produces significant deviation in the prediction accuracy as 12 cycles early from true value, because there is significant self-charge regeneration at 22, and 50 cycles. In fact, the iterative RUL prediction is based on the precise one-step forecasting. Technically, the error of one-step RUL forecasting will accumulate with the iterative process and the error accumulation will lead to a sharp accuracy decline in the final prediction results [21]. Therefore, a dynamic training model such as Bayesian filter is required to tackle this problem. However, this issue is less valid as the prediction horizon increases with an increase in training data. Forecasting for BP and RNN restarts at 75 cycles. It can be observed from Figures 32-33 that RUL can achieve better predictions at 75 cycles (2, and 1 early cycles) then prediction at 55 cycles.

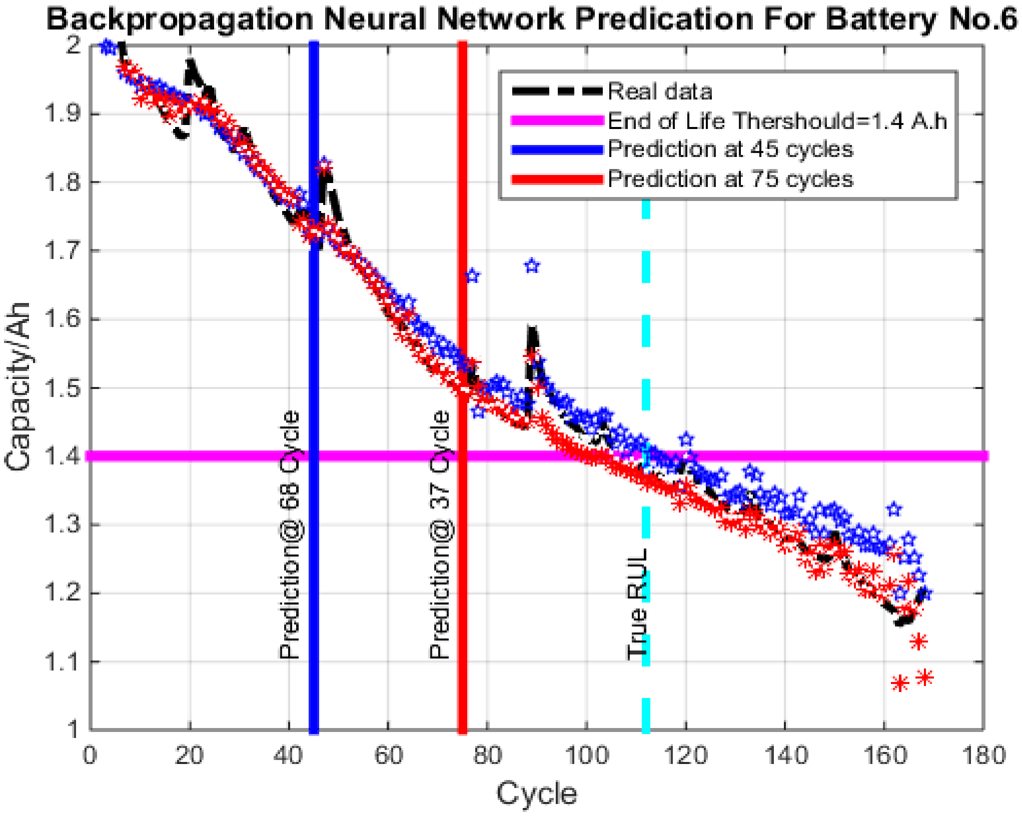


Figure 32: The RUL Prediction Result at 55 and 75 Cycles for the Battery No. 6 with Backpropagation Neural Network model (BP).

In order to evaluate the prediction at 55 and 75 cycles of classical BP, RNN and M-PLLSSVM predictors, the result is evaluated based on four metrics. The Absolute Error (AE), Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), and standard deviation (SD) are produced by using the three methods summarized in Table 8.

As demonstrated in the table, the least forecasting error is produced by the proposed M-PLLSSVM methods. Such result demonstrates that mean squared error value deceases from 0.000380 to 0.000270 (28.95% decrease), standard deviation decreases from 0.017883 to 0.016471 (7.9 % decrease), when prediction increases from 55 to 75 cycles. The AR, MAPE, SD, and MSE associated with three methods demonstrate that RNN produces higher errors at 55 cycles compared to both M-PLLSSVM and BP methods. However, when the prediction horizon increases from 55 to 75, the results shows that RNN captures the change of degradation and MSE decreases from 0.002181 to 0.000752 (65.6 %) as other metrics.

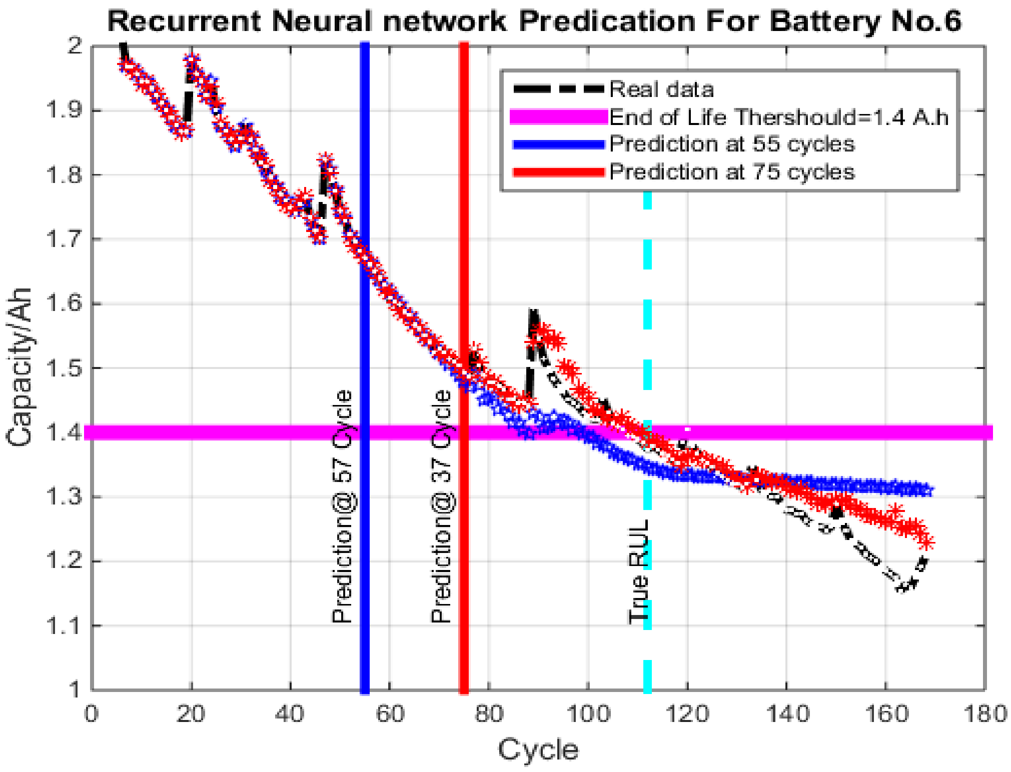


Figure 33: The RUL Prediction Result at 55 and 75 Cycles for the Battery No. 6 with Recurrent Neural network model (RNN)

PB produces slightly higher prediction errors then RNN at cycles 75; for instance, MSE is 0.000819 at cycle 75, which is 8.2 % higher error than RNN when the prediction horizon increases from 55 to 75 cycles. Finally, to investigate the maximum deviation between prediction and true End of Life (EoF), the Absolute Percentage Error (MAPE) and Mean Squared Error (MSE. Further, to evaluate the uncertainty of prediction and measure the narrowness of the forecasting, Standard deviation(S) is used by three algorithms. The results in Table 8 clearly demonstrate that M-PLLSSVM still produces the best performance. The M-PLLSSVM not only provides good accuracy to the RUL, but can predict the failure time distribution more precisely.

Table 8: Performance Evaluation for All Three Test Algorithms for Predictions Made Within Prediction Horizon at 55 and 75 Cycles.

Battery No.6	**At cycle**	**RUL Error criteria**					
		MAPE	**MSE**	**S**	**Real Cycles**	**Predictive**	**Error**
Backpropagation (BP)	**55**	3.106341	0.001567	0.032326	112	109	0.022988
	75	1.862969	0.000819	0.025783	112	110	0.007645
Recurrent Neural network	**55**	3.800592	0.002181	0.051006	112	100	0.063527
	75	1.764026	0.000752	0.025454	112	111	0.003525
M-PF LSSVM	**55**	1.248762	0.000380	0.017883	112	11.4	0.000011
	75	1.173662	0.000270	0.016471	112	112	0.000000

5.3 Effect of number particles (Sensitivity Analysis)

Technically, the accuracy of a particle filter depends on the number of the particles that are used to propagate the tracking states. The number of particles represents a tradeoff between computational cost and performance of failure prognosis[178]. In this research, historical failure data for a complex system are used to provide particles that represent an entire state sequence. The basic idea is that similar data sequences usually have similar trends [176,173]. Figures.34, 35, and 36 compare the RUL prediction error based on battery No.5 with different numbers of particles. Figure.34 shows the Memory-Particle Filter based on Least Square Support Vector machine (M-PF LSSVM) prediction at 55cycles (blue star) and at 75 cycles (red asterisk) with 23 particles. In this case, we repeat some particles that closely track to battery No.5 from the database. From Fig. 34 and Table 9, the estimation of RUL improves in both accuracy and precision with the inclusion of more particles. As more failure data become available, the particles begin to converge to this failure track in the database that is similar to real-time.

We examine the M-PF LSSVM using 3 and 15 particles; note that while the 3 particles used in the Figure 35 are similar to current degradation in Battery No. 5 (incipient failure), the 15 particles used in the Figure 36 are different (abrupt failure). From Figures. 35 and 36 and Table 9, one can see that the estimation accuracy with M-PF LSSVM is improved, as more of the similar particles to current failure track become available. For example, in Figure 36 the RUL prediction is not as good as the M-PF LSSVM approach based on 23 and 3 particles. The prediction error is 17 cycles at 55 and 15 cycles at 75, and the estimation of RUL is too far off from the actual failure. As shown in Figures above, the RUL improves in both accuracy and precision with the inclusion of particles most similar to observed faults.

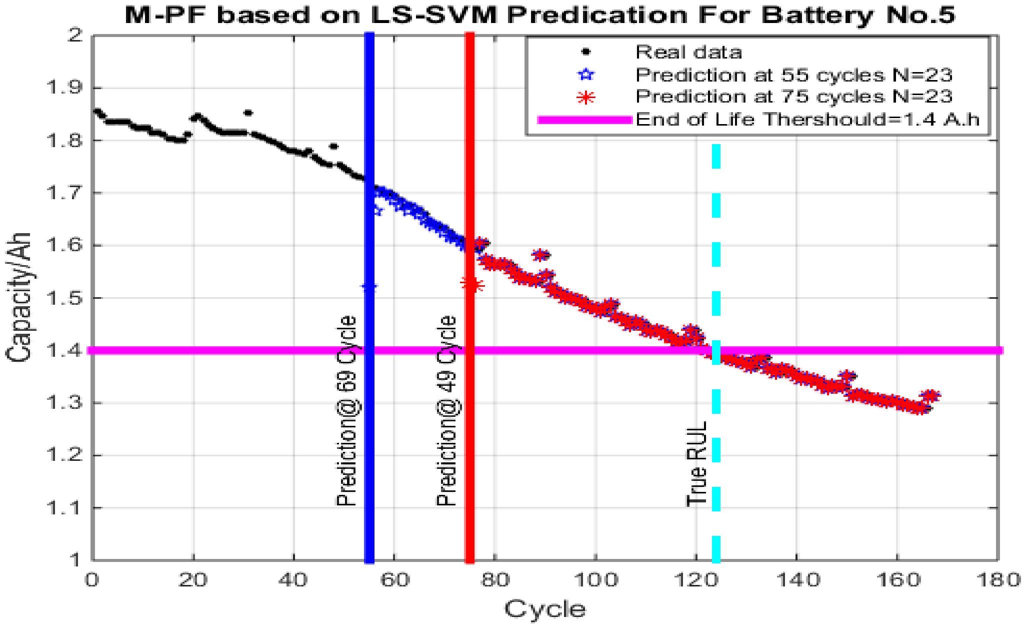


Figure 34: The RUL Prediction Result at 55 and 75 Cycles for the Battery No. 5 with M-PF Based on LS-SVM model with (23 particles),Test 1.

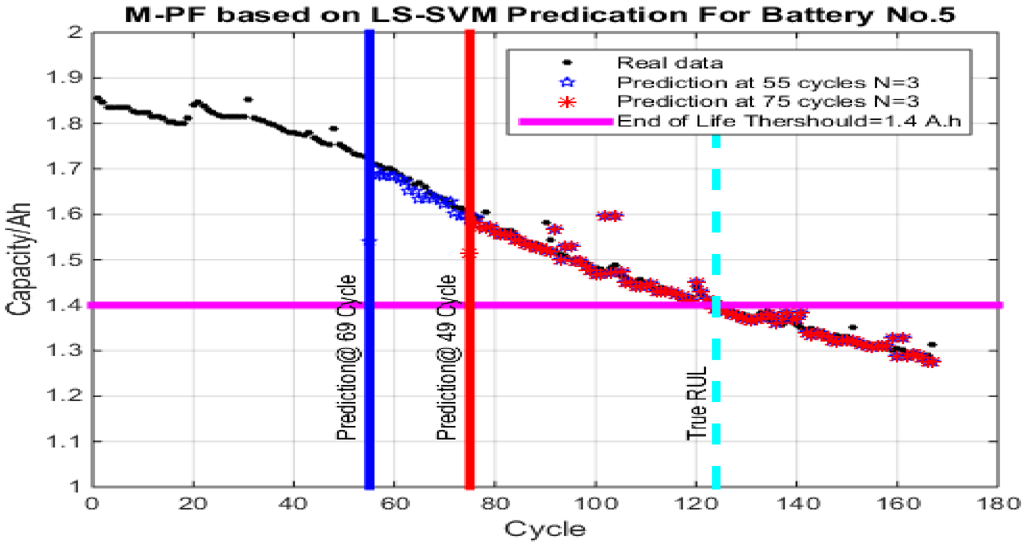


Figure 35: The RUL Prediction Result at 55 and 75 Cycles for the Battery No. 5 with M-PF Based on LS-SVM model with (3 particles),Test 2.

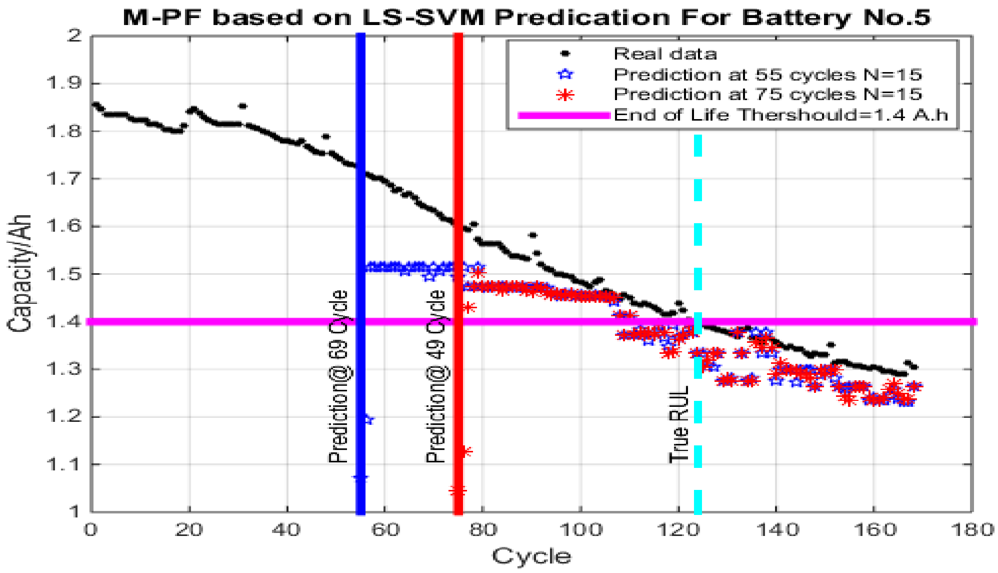


Figure 36: The RUL Prediction Result at 55 and 75 Cycles for the Battery No. 5 with M-PF Based on LS-SVM model with (15 particles),Test 3.

Table 9: Summary of the Prediction RUL Comparison based on Different Number of Particles.

Battery No.		**RUL Comparison**			
		True RUL	**Predicted RUL**	**Error (cycle)**	**Estimation error (%)**
Battery #5 with N=23	**At cycle 55**	124	123.3	0.7	0.57
	At cycle 75	124	123	1	0.80
Battery#5 with N=15	**At cycle 55**	124	107	17	13.8
	At cycle 75	124	109	15	12.09
Battery#5 with N=3	**At cycle 55**	124	119	5	4.03
	At cycle 75	124	119	4	3.23

Chapter 6

Conclusions and Recommendations

This thesis has produced a number of contributions to the Prognostic Health Management (PHM) field.

6.1 Summary of finding and Research Contribution

6.1.1 Methodologies to reduce multidimensional physical characteristics of lithium-ion batteries

- These methodologies include a step by step procedure to provide an estimation of battery capacity based on Energy Efficiency (EE) and battery Working Temperature (WT). These methodologies reduce multidimensional physical quantities that include current, voltage, usage duration, battery temperature, and ambient temperature.
- Energy Efficiency (EE) and battery Working Temperature (WT) don’t only preserve all the physical quantities (current, time, temperature and voltage) of the battery, but also reduce dimensions of data to two variables, and consequently reduce computational complexity.

6.1.2 An efficient data-driven for fault diagnosis

- An efficient incipient fault framework based on online Least Square Support Vector Machine (LS-SVM) is presented. The efficiency of incipient fault detection schemes is based on adaptive threshold, because the threshold can be updated by meaning of incremental and decremental algorithms whenever a new sample becomes available. Moreover, the adaptive threshold is robust for tracking slow small faults and prevents false alarms, and manages an unexpected uncertainty due to modeling error, and different operating conditions. The adaptive threshold technique is proposed to overcome the drawback of fixed threshold, and decide whether and where a fault has occurred.

- The functionality of the algorithms has been illustrated through the Li-ion battery data set that is obtained from NASA public data repository. The result confirms that the proposed algorithms have demonstrated better accuracy compared to existing fault diagnosis algorithms/techniques with fixed threshold.

6.1.3 A Novel data-driven Real-time Prognostic scheme for estimation Remaining Useful Life (RUL) using particle filtering with a non-existing physical failure model

- Memory Particle Filter (M-PF) based on Least Square Support Vector Machine (LS-SVM) for failure prognosis is presented. This algorithm is proposed to predict incipient failure for lithium-ion batteries system with the goal of supporting maintenance engineer’s decision making. In our framework, the particle filter is applicable when the physical failure model is not available or cannot be specified; M-PF is based on real-time and historical data, where the historical database replaces the physical failure model.
- The framework is entirely data-driven. First, LS-SVM is used to approximate the posterior density, and avoid loss diversity among the samples. Partial Resampling (PR) technique replaced the traditional resampling technique in the standard Particle Filter. PR is used to address the degeneracy challenge by eliminating particle with small weight and multiple samples with high weight. PR deterministic resampling reduces complexity and enables real-time computation. The experiment results demonstrate that M-PF LSSVM framework can increase the efficiency for failure prognosis and can maintain higher prediction accuracy. At the same time, the proposed approach provides the fast computation time and real time to ensure that the method can be used in real-time applications.

6.2 Recommendations for Further research

The data-driven framework has been successfully tested and partially validated by applying it to the case of diagnostic and prognostic technique in the Lithium-ion battery. However, there is still room for improvement in many directions:

1- A novel data-model fusion based on Memory Particle Filter and standard Particle Filter may develop to improve the accuracy of the system state forecasting. This framework strategically integrates the strengths of the data-driven Memory particle Filter (M-PF) method and model based standard Particle Filter in failure prognostics for complex systems. Moreover, this framework can overcome the limitations of both data-driven M-PF Filter method and model based PF such as quality of history data, and updating parameter during the failure prediction. As a result, the prognostic model will become more precise, and transparent.

2- Since the failure prognosis projects the current amount of degradation in the future using a prediction model in the absence of the future measurement, the Remaining Useful Life (RUL) estimation is associated with large-grain propagated uncertainty. M-PF is built based on Bayesian theory, which provides a way to model current amount of degradation as random variable, so the level of uncertainty can be properly managed and utilized through the framework. To improve the accuracy and precision of the failure prognosis based on M-PF. In future work, we will consider the uncertainty representation ability of the proposed failure prediction, and other important attributes, such as confident intervals, can be computed to provide probabilistic decision function.

References

[1] Saha, B.; Goebel, K. *Battery Data Set*; National Aeronautics and Space Administration (NASA) Ames Prognostics Data Repository: Moffett Field, CA, USA, 2007.

[2] B. Saha and K. Goebel, "Uncertainty management for diagnostics and prognostics of batteries using Bayesian techniques," In Proceedings of the IEEE aerospace conference, Big Sky, MT, pp. I-S, 2005.

[3] Datong Liu, et al. “Data-driven Prognostics for Lithium-ion Battery Based on Gaussian Process Regression,” 2012 Prognostics & System Health Management Conference ,PHM Beijing 2012 .

[4] Bing Long et al. “An improved autoregressive model by particle swarm optimization for prognostics of lithium-ion batteries ,” Microelectronics Reliability 53 ,PP. 821–831, 2013.

[5] Bhaskar Saha et al. “Comparison of Prognostic Algorithms for Estimating Remaining Useful Life of Batteries,” Microelectronics Reliability. Volume 53, Issue 6, Pages 805–810, June 2013.

[6] Amardeep Sidhu et al, “Adaptive Nonlinear Model-Based Fault Diagnosis of Li-Ion Batteries,” IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 62, NO. 2, FEBRUARY, 2015.

[7] Adnan” Nuhic et al. Health diagnosis and remaining useful life prognostics of lithium-ion batteries using data-driven methods,” Journal of Power Sources 239,pp 680-688,2013.

[8] Qiang Miao et al. “Remaining useful life prediction of lithium-ion battery with unscented particle filter technique,” Microelectronics Reliability 53 ,pp 805–810,2013.

[9] Juan Carlos et al. “Support Vector Machines Used to Estimate the Battery State of Charge,” IEEE TRANSACTIONS ON POWER ELECTRONICS, VOL. 28, NO. 12, DECEMBER 2013.

[10] Shuai Wang et al. “Prognostics of Lithium-Ion Batteries Based on BatteryPerformance Analysis and Flexible Support Vector Regression,” Energies, 7, 6492-6508, 2014.

[11] L.Hernandez and N.Gebraeel ,“Electronics Prognostics--.Driving Just-In-Time Maintenance”,IEEE,Anaheim,CA,2006,pp.187-194, Sept2006.

[12] Y. Liu and w. Chen, “Just-in-time Kernel Classifier for Online Process Diagnosis,” Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE),2013.

[13]K.Swearingen et al., “An Open System Architecture for Condition Based

Maintenance Overview," Big Sky, MT, Aerospace Conference, 2007 IEEE, 3-10 March 2007,pp.1-8.

[14] E.JJ. Amaya and A. J. Alvares, "SIMPREBAL: An Expert System for Real-Time Fault Diagnosis of Hydrogenerators Machinery," Bilbao, Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on, 13-16 Sept. 2010,pp.1-8.

[15] Guo Su etal., "Sensor Fault Detection with Online Sparse Least Squares Support Vector Machine," Control Conference (CCC), 2013 32nd Chinese, pp. 6220 – 6224, 26-28 July 2013.

[16] Z.WANG, et al.,"*A Novel Aircraft Engine Fault Diagnostic Prognostic System based on SVM",* Bali,2012 IEEE International Conference on Condition Monitoring and Diagnosis,pp.723-728, Sept.2012.

[17] J.Z. Sikorska et al. "prognostic modeling options for remaining useful life estimation by industry," Mechanical Systems and Signal Processing, Vol 25 ,pp. 1803–1836,2011.

[18] A. K. S. Jardine, D. Lin, and D. Banjevic. A review on machinery diagnostics andprognostics implementing condition-based maintenance. Mechanical Systems andSignal Processing, 20(7):1483-1510, 2006.

[19] Z. Shi, F. Gu, t al., "The development of an adaptive threshold for model-based fault detection of a nonlinear electro-hydraulic system," Control Engineering Practice 13,pp.1357–1367,2005

[20] E.JJ. Amaya and A. J. Alvares, "SIMPREBAL: An Expert System for Real-Time Fault Diagnosis of Hydrogenerators Machinery," Bilbao, Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on, 13-16,pp.1-8, Sept. 2010.

[21] Jianbao Zhou etal., "Remaining Useful Life Estimation with Dynamic Grey Relevance Vector Machine for Lithium-ion Battery," International Journal of Advancements in Computing Technology(IJACT). Volume5, Number6,March 2013

[22] N.Tandon ,and A. Pary "Condition Monitoring and Control for Intelligent Manufacturing, " Springer Series in Advanced Manufacturing, Condition Monitoring and Control for Intelligent Manufacturing,pp.109-136,2006.

[23] Xu Lishuang etal., "Sensor Fault Diagnosis Based on Least Squares Support Vector Machine Online Prediction," Robotics, Automation and Mechatronics (RAM), 2011 IEEE Conference on, pp. 275 - 279, 17-19 Sept. 2011

[24] L. Zhang, et al. ,"Review of Remaining Useful Life Prediction Using Support Vector Machine for Engineering Assets,"Chengde, IEEE,2013 International Conference on Quality, Reliability, Risk, aintenance, and Safety Engineering (QR2MSE), pp.1793-1799,2013.

[25]L. Wang et al., “Advanced Maintenance Strategy for Power Plants—Introducing Intelligent Maintenance System,IEEE,”Dallan, China, ,pp.7444-7448, June 2006.

[26] C. Ly et al., “Fault Diagnosis and Failure Prognosis for Engineering Systems:A Global Perspective,” Bangalore, India, 5th Annual IEEE Conference on Automation Science and Engineering, August 22-25, 2009,pp.108-115.

[27]F. Lewiset al., “Intelligent Fault Diagnosis and Prognosis for Engineering Systems. ,”A. Hess and B. Wu John Wiley & Sons,2006.

[28] A.K.S. “a review on machinery diagnosis and prognosis implementing condition -based maintenance,” mechanical system processing,20,pp.1483-1510,2006

[29] M.A.Taie et al.,“Remote prognosis, diagnosis and maintenance for automotive architecture based on least squares support vector machine and multiple classifiers,” IEEE, IV International Congress on Ultra Modern Telecommunications and Control Systems,pp.128-134, 2012.

[30] J. Liu et al. “An Adaptive Recurrent Neural Network for Remaining Useful Life Prediction of Lithium-ion Batteries.” Annual Conference of the Prognostics and Health Management Society, 2010.

[31] A.Starr et.al, “E-maintenance,” Springer London, Springer-Verlag London, Maintenance Today and Future Trends, Maintenance Today and Future Trends,”chap:2, pp 5-37, 2010.

[32] D. Liu et al., “On-line Adaptive Data-Driven Fault Prognostics of Complex Systems,” IEEE Vol. 978,pp.166-173, 2011.

[33] V.Venkatasubramanian et al., “A review of process fault detection and diagnosis Part I: Quantitative model-based methods,” Computers and Chemical Engineering, Vol.27, pp.293-311.2003.

[34] SShane Butler, Prognostic Algorithms for ConditionMonitoring and Remaining Useful Life Estimation . PhD thesis, National University of Ireland, Maynooth, 2012.

[35] Portia A. Cerny, “Data mining and Neural Networks from a Commercial Perspective,” [Online] Available:
ttp://citeseerx.ist.psu.edu/viewdoc/versions?doi=10.1.1.117.1820.
[36] Steven X., “Model-Based Fault Diagnosis Techniques Design Schemes, Algorithms, and Tools,” Design Schemes, Algorithms and Tools Series: Advances in Industrial Control, Springer-Verlag Berlin Heidelberg,.2008.

[37] V.Venkatasubramanian, et al.,“A review of process fault detection and diagnosis Part I: Quantitative model-based methods,” Computers and Chemical Engineering Vol.27 , pp.1483 -1510.2006.

[38] V.Venkatasubramanian, et al., "A review of process fault detection and diagnosis Part II: Quantitative model-based methods," Computers and Chemical Engineering ,Vol.27 , pp.293-311,2003.

[39] Steven X., "Model-Based Fault Diagnosis Techniques Design Schemes, Algorithms, and Tools," Design Schemes, Algorithms and Tools Series: Advances in Industrial Control, Springer-Verlag Berlin Heidelberg,.2008.]

[40]Inman, D. J., C. R. Farrar, V. L. Junior and V. S. Junior, Eds. (2005). Damage Prognosis : For Aerospace, chapter 1, pp. 4-5), Civil and Mechanical Systems, John Wiley and Sons 2005.

[41] A. Jardine et al., "A review on machinery diagnostics and prognostics implementing condition-based maintenance," Mechanical Systems and Signal Processing, Vol.20, pp.1483–1510, 2006.

[42] M. H. Hassoun, "Fundamentals of Artificial Neural Networks,"Asco Trade Typesctting Ltd, Hong Kong, Chp. 1, pp. January 1, 2003.

[43]Jack V. Tu, "Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes," Journal of Clinical Epidemiology, Volume 49, Issue 11, Pages 1225–1231, November 1996.

[43] M.Ge at el., "Fault diagnosis using support vector machine with an application in sheet metal stamping operations," Mechanical Systems and Signal Processing, Vol.18,pp. 143–159, 2004.

[44] J.Zhang et al., "Support vector regression for on-line health monitoring of large-scale structures," Structural Safety,Vol.28,pp.392-406,2006.

[45] VAPNIK, V., and A. LERNER, 1963. Pattern recognition using generalized portrait method. *Automation and Remote Control*, 24, 774–780

[46] E. E. Osuna et al. "Support Vector Machines: Training and Applications," MASSACHUSETTS INSTITUTE OF TECHNOLOGY ARTIFICIAL INTELLIGENCE LABORATORY, A.I. Memo No. 1602 March, 1997C.B.C.L Paper No. 144

[47] Vladimir Vapnik, The Nature of Statistical Learning Theory ,Spring Velage,NY,1995

[48] L.xiang, and C. Xiang, "An incremental algorithm of support vector machine based on distance ratio and k nearest neighbor,",IEEE, June 2011,pp.18-20.

[49] A.Widodo and B.Yang, "Review Support vector machine in machine condition monitoring and fault diagnosis," Mechanical Systems and Signal Processing, Vol. 21, pp. 2560–2574, Aug.2007.

[50] E. Byvatov, “Comparison of Support Vector Machine and Artificial Neural Network Systems for Drug/Nondrug Classification,” American Chemical Society, Vol. 43, pp1882-1889,2003.

[51] J. Wu et al., “Least Square Support Vector Machine Ensemblefor Daily Rainfall Forecasting Based on Linear and Nonlinear Regression,” Advances in Neural Network Research and Applications, Springer Berlin Heidelberg,Vol.67,l pp 55-64, 2010.

[52] H.S. Tang and T.Sato, “ Sequential LS-SVM for structural system identification,” hird European Workshop on Structural Health Monitoring, July 5-7, 2006.

[53] E. Byvatov, “Comparison of Support Vector Machine and Artificial Neural Network Systems for Drug/Nondrug Classification,” American Chemical Society, Vol. 43, pp1882-1889,2003.

[54] B. Roman, and L. Ekaterina, “Support vector machine regression (SVR/LS-SVM)—an alternative to neural networks (ANN) for analytical chemistry? Comparison of nonlinear methods on near infrared (NIR) spectroscopy data,” The Analyst, vol. 136, issue 8, p. 1703,2011.

[55] W.Yan and H. Shao,“Application of Support Vector Machine Nonlinear Classifier to Fault Diagnoses,” shanghai, China, Proceedings of the 4th World Congress on Intelligent Control and Automation, pp.2697-2700, 2002.

[56] J.A.K. Suykens , “Weighted least squares support vector machines: robustness and sparse approximation,” Neurocomputing,Vol. 48,pp.85–10,2002.

[57] Yaoyong Li et al., “Adapting SVM for Data Sparseness and Imbalance: A Case Study on Information Extraction,” Natural Language Engineering 1 (1): 1–32. 2009

[58] J.A.K. Suykens and J.Vandewalle, “ Least Squares Support Vector Machine Classifiers,” Neural Processing Latters, Vol.9,pp.293-300, 1999.

[59]F.Salfner et al., “A Survey of Online Failure Prediction Methods ,” ACM Computing Surveys, Vol. 42, No. 3, Article 10, Publication date: March 2010.

[60] C.-C. Chuang et al.; “Robust least squares-support vector machines for regression with outliers,” Fuzzy Systems, 2008. FUZZ-IEEE 2008. (IEEE World Congress on Computational Intelligence). IEEE International Conference on, Hong Kong,pp. 312 – 317, June 2008.

[61] J.MA et al., “Accurate On-line Support Vector Regression”, Neural Computation ,Vol.15 pp.2683–2703 (2003)

[62] Y. Fan, and Z.Song, “Dynamic Least Squares Support Vector Machine,” Dalian, China IEEE, Proceedings of the 6th World Congress on Intelligent Controland

Automation, Dalian, China, June 21 - 23, 2006.pp.4886-4889.

[63] J.A.K.SUYKENS and J.VANDEWALLE, “Least Squares Support Vector Machine Classifiers,” Neural Processing Letters, Vol.9,pp. 293–300, 1999.

[64]L. Jiao etal., “Fast Sparse Approximation for Least Squares Support Vector Machine,” IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 18, NO. 3, MAY 2007.

[65] B.Yange and A.Widodo, “Support Vector Machine for Machine fault Diagnosis and Prognosis,” Journal of System Design and Dynamics,Vol.2,No. 1, 2008.

[66] L. B Jack and A.K.Nandi,“ Fault detection using support vector machines and artificial neural networks, augmented by genetic algorithm,” Mechanical Systems and Signal Processing Vol.16(2–3),pp. 373–390,2002.

[67] W.Hu et al.“Robust Anomaly Detection Using Support Vector Machines,” In Proceedings of the International Conference on Machine Learning.

[68] X. Long et al. “Anomaly Detection of Spacecraft Based on Least Squares Support Vector Machine,” Shenzhen, IEEE,Prognostics & System Health Management Conference,IEEE, pp.1-6,2011.

[69] T.S.Khawaja et al., “An Efficient Novelty Detector for Online Fault Diagnosis based on Least Squares Support Vector Machines,” IEEE AUTOTESTCON,Salt lake city,UT,Setemper, pp.202-207, 2008.

[70] Dubey et ai. “A Novel Fault Classification Scheme Based on Least Square SVM ,” Engineering and Systems (SCES), 2012 Students Conference on, Allahabad, Uttar Pradesh,pp. 1 – 5,16-18 March 2012.

[71] L.Min et al., “Diagnostics of Incipient Faults in Analog Circuits,”Harbin, IEEE International Conference on Electronic Measurement & Instruments, 2013, pp.833-838.

[72] G.Yang et al., “Fault diagnosis using a probability least squares support vector classification machine,” Mining Science and Technology,Vol. 20 ,pp.0917–0921, 2010.

[73] H.B.Zheng et al.,“Fault diagnosis of power transformers using multi-class least square support vector machines classifiers with particle swarm optimization,” IET Electr. Power Appl, Vol. 5, Iss. 9, pp. 691–696, 2011.

[74] C.H.Wei et al.,“A Hybrid Least-square Support Vector Machine. Approach to Incipient Fault Detection for Oil-immersed Power Transformer,” Electric Power Components and Systems, Vol. 42 No. 5, Taylor & Francis Group, LLC, 2014.

[75] Y. Zhang et al, “Application of Least Squares Support Vector Machine in Fault Diagnosis”, Part II, C. Liu, J. Chang, and A. Yang (Eds.): ICICA, Springer-Verlag

Berlin Heidelberg , CCIS 244, 2011,pp. 192–200.

[76] B.Long at el., “Improved diagnostics forth incipient fault in analog circuits using LSSVM based on PSO algorithm with Mahalanobis distance,” Neurocomputing,Vol.133,pp.237–248, 2014.

[77] A. Hess, G. Calvello, P. Frith, S.J. Engel, and D. Hoitsma. Challenges, issues, and lessons learned chasing the "big p". real predictive prognostics. part 2. In Aerospace Conference, 2006 IEEE, pages 1-19, 2006.

[78] A. Hess, G. Calvello, P. Frith, S.J. Engel, and D. Hoitsma. Challenges, issues, and lessons learned chasing the "big p". real predictive prognostics. part 1. In Aerospace Conference, 2006 IEEE, pages 3610 - 3619, 2005.

[79] W.Yan and H. Shao,“Application of Support Vector Machine Nonlinear Classifier to Fault Diagnoses,” shanghai, China, Proceedings of the 4th World Congress on Intelligent Control and Automation,, pp.2697-2700, 2002.

[80] S.J.Engel et al. “Prognostics, the real issues involved with predicting life remaining,” Aerospace Conference Proceedings, 2000 IEEE, Big Sky, MT,pp. 457 - 469 vol.6, Mar 2000.

[81] J.Z. Sikorska et al. “prognostic modeling options for remaining useful life estimation by industry,” Mechanical Systems and Signal Processing, Vol 25 ,pp. 1803–1836,2011

[82] . L.Liao , and F. Kottig, “Review of hybrid prognostics approaches for remaining useful life prediction of engineered systems, and an application to battery life prediction,” Reliability, IEEE Transactions on, (Volume:63 , Issue: 1 ,pp191 – 207, March 2014.

[83] X.-Sheng Si et al. “Remaining useful life estimation – A review on the statistical datadriven approaches ,”European Journal of Operational Research,Vol 213 pp.1–14, 2011

[84] Jaw, L. and W. Wang (2004). A run-time test system for maturing intelligent system/vehicle capabilities - SIDAL. 2004 IEEE Aerospace Proceedings, 6-13 March 2004, Big Sky, MT, USA, IEEE,2004.

[85] Byington, C. S., M. J. Roemer and T. Galie (2002). Prognostic Enhancements to Diagnostic Systems for Improved Condition-Based Maintenance. IEEE Aerospace Conference Proceedings, Big Sky, MT, USA, IEEE,2002.

[86] J.Lee et al. “ Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications,” Mechanical Systems and Signal Processing, Volume 42, Issues 1–2, Pages 314–334, January 2014.

[87] Jaw, L. and W. Wang (2004). A run-time test system for maturing intelligent system/vehicle capabilities - SIDAL. 2004 IEEE Aerospace Proceedings, 6-13 March

2004, Big Sky, MT, USA, IEEE,2004.

[88] E. Zio. “Prognostics and Health Management of Industrial Equipment,” Author manuscript, published in "Diagnostics and Prognostics of Engineering Systems: Methods and Techniques, Seifedine,2012.

[89] L.Wang et al “ Advanced Maintenance Strategy for Power Plants - Introducing Intelligent Maintenance System.” Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on, Dalian,pp. 7444 – 7448, 2006.

[90] J.Lee et al. “ Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications,” Mechanical Systems and Signal Processing, Volume 42, Issues 1–2, Pages 314–334, January 2014.

[91] Jianhui Luo et al. “Model-based prognostic techniques [maintenance applications],” AUTOTESTCON 2003. IEEE Systems Readiness Technology Conference. Proceedings, pp.330 – 340, 2003.

[92] A.Kabir et al. “A Review of Data–Driven Prognostics in Power Electronics,” Electronics Technology (ISSE), 2012 35th International Spring Seminar on, Bad Aussee,pp. 189 – 192, May 2012.

[93] A. Heng, S. Zhang, A. C. C. Tan, and J. Mathew. Rotating machinery prognostics:State of the art, challenges and opportunities. Mechanical Systems and Signal Processing, 23(3):724{739, 2009.

[94]Z. Guo and B.Guangchen,“Application of Least Squares Support Vector Machine for Regression to Reliability Analysis,” Chinese Journal of Aeronautics,Vol. 22, pp.160-166,2009.

[95] J.Qu and M.J. Zuo, “An LSSVR-based algorithm for online system condition prognostics,” Expert Systems with Applications,Vol 39 , pp.6089–6102, 2012.

[96] X. Li, et al. “Life and Reliability Forecasting of the CSADT using Support VectorMachines,”San Jose, CA, IEEE, Jan.2010,pp.1-6,2010.

[97]. Jihong Yan et al. “A Hybrid Method for On-line Performance Assessment and Life Prediction in Drilling Operations, “A Hybrid Method for On-line Performance Assessment and Life Prediction in Drilling Operations,” Automation and Logistics, 2007 IEEE International Conference on, Jinan, 2500 – 2505, Aug. 2007.

[98] Leto Pee. “Data driven prognostics using a Kalman filter ensemble of neural network models,” Prognostics and Health Management, 2008. PHM 2008. International Conference on. Denver, CO,pp. 1 – 6,2008.

[99] R. Huang at et . “Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods,” Mechanical Systems and Signal

Processing, Volume 21, Issue 1, Pages 193–207, January 2007.

[100] M. Orchard et al. “Advances in Uncertainty Representation and Management for Particle Filtering Applied to Prognostics,” Springer Netherlands, Applications of Intelligent Control to Engineering Systems ,pp 23-35, 2009.

[101] Q.Miao “Remaining useful life production of lithium -ion battery unscented particle filter technique,”Mircroelectronics Reliability, Vol.53,pp.805-810,2013.

[102] C. M.Z and Z. D.H “Cparticle filter based fault prediction of nonlinear system,”In:IFACSystem proceedings of safe Process, Washington(2003)

[103] M. E. Orchard. A Particle Filtering based Framework for On-Line Fault Diagnosis and Fault Prognosis. PhD thesis, Georgia Institue of Technology, 2007.

[104] E. Zio and Giovanni“ Particle filtering prognostic estimation of the remaining useful life of nonlinear components,” Reliability Engineering & System SafetyVolume 96, Issue 3, March, Pages 403–409, 2011.

[105] Leto Pee. “Data driven prognostics using a Kalman filter ensemble of neural network models,” Prognostics and Health Management, 2008. PHM 2008. International Conference on. Denver, CO,pp. 1 – 6,2008.

[106]. N. Sivanandam, S. N Deepa (2006) Introduction to Neural Networks Using Matlab 6.0. McGraw-Hill Education. Computer Engineering series.

[107] B. Saha et al. “Prognostics Methods for Battery Health Monitoring Using a Bayesian Framework,” Instrumentation and Measurement, Vol.21 pp.291 - 296 Feb. 2009

[108] L. Yu et al; “A particle filter and SVM integration framework for fault-proneness prediction in robot dead reckoning system”, WSEAS TRANSACTIONS on SYSTEMS ,Volume 10 Issue 11, November 2011.

[109]. L. Chengliang et al, “A novel method based on LS-SVR combing with Strong Tracking Particle Filter for machinery condition prognosis,”Machanical Engineering Science, Vol 228(6),pp 1048-1062,2014.

[110] C. Xiongzi et al. “A Novel PF-LSSVR-based Framework for Failure Prognosis of Nonlinear Systems with Time-varying Parameters”, Chinese Journal of Aeronautics, Volume 25, Issue 5, Pages 715–724, October 2012.

[111]. O. Cappe et al. “An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo,” Proceedings of the IEEE (Volume:95 , Issue: 5),pp 899 – 924, May 2007.

[112] A. Doucet et al. “On sequential Monte Carlo sampling methods for Bayesian

filtering ,", Volume 10, Issue 3, pp 197-208,2000.

[113] G. Zhu et al ; "Improving particle filter with support vector regression for efficient visual tracking," Image Processing, 2005. ICIP 2005. IEEE International Conference on, Volume:2 , - pp. 422-5, 2005.

[114] B. Saha et al. "An integrated approach to battery health monitoring using bayesian regression and state estimation," Autotestcon, 2007 IEEE, Baltimore, MD,pp. 646 – 653, 2007.

[115] Markus Hürzeler and Hans R. "Künsch,Monte Carlo Approximations for General State-Space Models,"Journal of Computational and Graphical StatisticsVol. 7, No. 2 ,pp. 175-193, Jun., 1998),

[116] Jihong Yan et al. "A Hybrid Method for On-line Performance Assessment and Life Prediction in Drilling Operations, "A Hybrid Method for On-line Performance Assessment and Life Prediction in Drilling Operations," Automation and Logistics, 2007 IEEE International Conference on, Jinan, 2500 – 2505, Aug. 2007.

[117] Ron J. Patton, and J. "Chen ADVANCES IN FAULT DIAGNOSIS USING ANALYTICAL REDUNDANCY," Plant Optimisation for Profit (Integrated Operations Management and Control), IEE Colloquium, London, 28 Jan 1993.

[118]R. J. Patton, P. M. Frank, and R. N. Clark, Eds., Fault Diagnosis inDynamic Systems: Theory and Applications. Upper Saddle River, NJ:Prentice-Hall, 1989.

[119] Chen and R. J. Patton, Robust Model-Based Fault Diagnosis for Dynamic Systems. Boston, MA: Kluwer, 1999.

[120] E.A.Simeon and A.J.Alvares, "A DATA-DRIVEN FRAMEWORK FOR PREDICTING THE REMAINING USEFUL LIFE OF HYDROELECTRIC EQUIPMENTS," ABCM Symposium Series in Mechatronics - Vol. 5, Emerging Technologies and AI Applications Page 583.2009.

[121] M. S. Lebold, K. M. Reichard, D. Ferullo, and D. Boylan. "Open systems architecture for condition-based maintenance: Overview and training manual. Technicalreport, Penn State University/Applied Research Laboratory, 2003.

[122] M. A. Demetriou and M. M. Polycarpou "Incipient Fault Diagnosis of DynamicalSystems Using Online Approximators," Automatic Control, IEEE Transactions on (Volume:43 , Issue: 11),pp. 1612 – 1617, 06 August 2002.

[123] Ke Le et al. "Adaptive Thresholding - A Robust Fault Detection Approach," Proceedings of the 36th Conference on Decision & Control San Diego, California USA December 1997.

[124] H. Schneider and P.M. Frank , "Fuzzy Logic Based Threshold Adaption for Fault

Detection in Robots," Control Applications, 1994., Proceedings of the Third IEEE Conference on, 1127 - 1132 vol.2, Glasgow, 1127 - 1132 vol.2, 24-26 Aug 1994.

[125] Y. Ma and P. Guttorp, "Estimating daily mean temperature from synopticclimate observations," Int. J. Climatol., vol. 33, no. 5, pp. 1264–1269,Apr. 2013.

[126] Deepshikha Agarwal, and Nand Kishor, "A Fuzzy Inference-Based Fault Detection Scheme Using Adaptive Thresholds for HealthMonitoring of Offshore Wind-Farms," IEEE SENSORS JOURNAL, VOL. 14, NO. 11, NOVEMBER 2014.

[127] P. M. Frank and X. Ding,"Frequency domain approach to optimally robust residual generation and evaluation for model-based fault diagnosis," Automatica, Volume 30, Issue 5, pp. 789–804 May 1994.

[128] R.Fujimaki, "Anomaly Detection Support Vector Machine and Its Application to Fault Diagnosis," Washington ,DC,Eighth IEEE International Conference on Data Mining,2008,pp.797-802,2008.

[129] M. Davy et al., "An online support vector machine for abnormal events detection," Signal Processing, Elsevier B.V, Vol.86, pp.2009-2025, 2005.

[130] M.Davy and S.Godsill,"Detection of abrupt spectral changes using support vector machines an application to audio signal segmentation," Orlando,FL,USA,IEEE,May.2002,Vol.2,pp.1313-1316,2002

[131] A.Shigeo, "Support Vector Machines for Pattern Classification," Advances in Computer Vision and Pattern Recognition, Springer London Dordrecht Heidelberg New York, 2010

[132]]Burges, C.J.C., "A tutorial on support vector machinesfor pattern recognition, in Data Mining and KnowledgeDiscovery". 1998. p. 121–167,1998

[133] Z.Zhang et al., "Web Prediction Using Online Support Vector Machine,"Hong Kong,Nov.2005,pp.456,2005

[134] D. Jin and S. Lin, "Advances in Mechanical and Electronic Engineering, A Novel Lagrangian Support Vector Machine and Application in the Crane Gear Fault Diagnosis System,Lecture Notes in Electrical Engineering Volume 176, pp 369-373, 2012

[135] L. Hu etal., "Fault Diagnosis of Hydraulic Actuator Based on Least Squares Support Vector Machines," Automation and Logistics, 2007 IEEE International Conference on, Jinan,pp. 985 – 989, 18-21 Aug. 2007.

[136] L.Zhao and K.Yang, "Machinery Fault Diagnosis Using Least Squares Support Vector Machine," Part III, Springer-Verlag Berlin Heidelberg, LNCS 4493,2007, pp. 342–349.

[137] L.Zhao and K.Yang, “Machinery Fault Diagnosis Using Least SquaresSupport Vector Machine,” Part III, Springer-Verlag Berlin Heidelberg, LNCS 4493,2007, pp. 342–349.

[138] F. Zhao et al. , “LEAST SQUARES SUPPORT VECTOR MACHINE BASED CONDITION PREDICTION FOR BEARING HEALTH,”Daejeon,Korea, 15th International Congress on Sound and Vibration, 6-10 July 2008,Vol.1-4.

[139] M. Espinoza et al. “ Least Squares Support Vector Machines and Primal Space Estimation”Proceedings of the 42nd IEEE, Conference on Decision and Control , Maui, Hawaii USA, December 2003

[140] Jo.K Suykens ; Eng Huay ; K.K Phua ; T. Van Gestel ; J. De Brabanter ; “Least Squares Support Vector Machines,”B. De Moor ; Joos Vandewalle, United Kingdom: World Scientific Pub Co Pte 2002.

[141] Xiaodong Zhang et al. “Abrupt and Incipient Fault Isolation of Nonlinear Uncertain Systems,” Proceedings of the American Control Conference Chicago, Illinois ,June 2000.

[142] Yixin Diao and Kevin M. “Stable Fault-Tolerant Adaptive Fuzzy/Neural Control for a Turbine Engine,” IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, VOL. 9, NO. 3, MAY 2001.

[143] Xidong Tang et al., “ Adaptive actuator failure compensation for nonlinearMIMOsystems with an aircraft control application,” Automatica , Vol.43 ,pp.1869 – 1883,2007.

[144] Xiaodong Zhang et al., “A Robust Detection and Isolation Scheme for Abrupt and Incipient Faults in Nonlinear Systems,” IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. 47, NO. 4, APRIL 2002.

[145] Marios M. Polycarpou et al. “Automated Fault Detection and Accommodation: A Learning Systems Approach,” IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, VOL. 25, NO. 11, NOVEMBER 1995.

[146] S. X. Ding and P. M. Frank “An Approach to the Detection of Multiplicative Faults in Uncertain Dynamic Systems,” Yruweding, uf Ihe 41rl IEEE Conference un Decision and ConlrulLa,Veins, December 2002.

[147] Juntong Qi et al. “Fault Detection Design for RUAV with an Adaptive Threshold Neural-Network Scheme,” 2007 IEEE International Conference on Control and Automation WeD5-5 Guangzhou, CHINA - May 30 to June 1, 2007

[148]Y. Cui et al. “An adaptive fault detection threshold hardware circuit design to reduce False Alarms,” Safety and Reliability: Methodology and Applications – Nowakowski et al. Taylor & Francis Group, London,2014

[149] P. M.Frank "Residual evaluation for fault diagnosis based on adaptive fuzzy thresholds," Qualitative and Quantitative Modelling Methods for Fault Diagnosis, IEE Colloquium on, London pp 401 – 411, 24 Apr 1995.

[150] V.Cherkassky, Y. Ma "Practical selection of SVM parameters and noise estimation for SVM regression," Neural Networks 17 ,pp 113–126, 2004.

[151] A. ÁLVAREZ MEZA et al. "PARAMETER SELECTION IN LEAST SQUARES-SUPPORT VECTOR MACHINES REGRESSION ORIENTED, USING 39-GENERALIZED CROSS-VALIDATION," Dyna, Nro. 171, pp. 23-30. Medellin, february, 2012.

[152] Miandare, M.S. et al. "VoIP anomaly detection by combining OCSVM and PSO algorithm," Telecommunications (IST), 2012 Sixth International Symposium on, Tehran, pp. 1038 – 1043, 6-8 Nov. 2012.

[153] Isis Didier Lins et at "A Particle Swarm-optimized Support Vector Machine for Reliability Prediction," Qual. Reliab. Engng. Int. ,28 ,141—158, 2012.

[154] Shih-Wei Lin et al. "Particle swarm optimization for parameter determination andfeature selection of support vector machines," Expert Systems with Applications 35 ,1817–1824,2008.

[155] Sheng Ding and Li Chen " Intelligent Optimization Methods for High-Dimensional Data Classification for Support Vector Machines," Intelligent Information Management, 2010

[156] Bishal Gurung; "Fitting Nonlinear Time-series Model Using Swarm Optimization Technique, Advance in Electronic and Electric Engineering, Volume 4, Number 6, pp. 537-540,2014.

[157] M. ASHHAR&A. R. MOTAMED "Stock Market Index Prediction via Hybrid. Inertia Factor PSO and Constriction Coefficient PSO," International Journal of Academic Research in Accounting, Finance and Management Sciences Vol. 4, No.3,pp. 144–154, July 2014.

[158] X. Gao et al. "Classification of Hyperspectral Image Based on SVM Optimized by A New Particle Swarm Optimization," Remote Sensing, Environment and Transportation Engineering (RSETE), 2012 2nd International Conference on, Nanjing, pp. 1 – 41, 3 June 2012.

[159] Chunli Xie et al. "Parameters Optimization of Least Squares Support Vector Machines and Its Application.", JOURNAL OF COMPUTERS, VOL. 6, NO. 9, SEPTEMBER 2011.

[160] Yuan Ren and Guangchen Bai "Determination of Optimal SVM Parameters by

Using GA/PSO.” JOURNAL OF COMPUTERS, VOL. 5, NO. 8, AUGUST 2010.

[161] Sheng-Fa Yuana,b and Fu-Lei “Chu Fault diagnostics based on particle swarm optimization and support vector machines,” Mechanical Systems and Signal Processing 21 , 1787–1798,2007.

[162] P. Laskov et al., “Incremental Support Vector Learning:Analysis, Implementation and Applications,” Journal of Machine Learning Research,Vol. 7,pp.1909–1936,2006.

[163]H. Tanga, et al.,“Online weighted LS-SVM for hysteretic structural system identification”, Engineering Structures, Vol.28,pp. 1728–1735,2006.

[164] F. Chen et al. “Detecting Bots via Incremental LS-SVM Learning with Dynamic Feature Adaptation,” Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, pp.386-394, , August 21–24, 2011.

[165] Liu, X.; Zhang,G.; Zhan,Y.; Zhu,E.; “An Incremental Feature Learning Algorithm Based on Least Square Support Vector Machine,” Springer Berlin Heidelberg, Frontiers in Algorithmics, pp 330-338, 2008.

[166] A. Shiltonet al., “Incremental Training of Support Vector Machine,” Neural Network ,IEEE Transaction,Vol.16,pp.114-131,2005.

[167] C. C. Cowen, P. A. Ferguson, D. K. Jackman, E. A. Sexauer, C. Vogt, and H. J. Woolf. Finding norms of Hadamard multipliers. Linear Algebra Appl., 247:217-235, 1996.

[168] H. M.MINGCHI and O.K. ERSOY,“Recursive Update Algorithm for Least Squares Support Vector Machines,” Neural Processing Letters, Vol.17,pp. 165–173, 2003.

[169] H. Liu et al.; “An adaptive threshold based on support vector machine for fault diagnosis,” Reliability, Maintainability and Safety, 2009. ICRMS 2009. 8th International Conference on, Chengdu,pp. 907 – 911, July 2009.

[170] S. Cheng et al, “A fusion prognostics method for remaining useful life prediction of electronic products,” Automation Science and Engineering,” 2009. CASE 2009. IEEE International Conference on, Bangalore, pp. 102 – 107, 22-25 Aug. 2009.

[171] B. Sun et al. “Benefits analysis of prognostics in systems,” Prognostics and Health Management Conference, 2010. PHM '10., Macao,pp. 1 – 8, 2010

[172] D.Mikami, et al., " Memory-based Particle Filter for Face Pose Tracking Robust under Complex Dynamics," in Computer Vision and Pattern Recognition, 2009.

[173] Dan Mikami, etal . "Memory-based Particle Filter for Tracking 3 Objects with Large Variation in Pose and Appearance," in The 11th European Conference on 4

Computer Vision, 2010, pp. 215-228.

[174] A. Panangadan and A.Talukder, "A Variant of Particle Filtering Using Historic Datasets for 41 Tracking Complex Geospatial Phenomena," presented at the 18th SIGSPATIAL International.

[175] H. Chen , and H. A. Rahka . "Data-Driven Particle Filter for Multi-step Look-Ahead Travel Time Prediction," Tranporttion Research Part C, C43,pp.112-126,2014.

[176] K. Otsuka, et al. "Memory-Based Forecasting for Weather Image Patterns," in Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, pp. 330-336.2000.

[177] M. Sanjeev et al. "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," Signal Processing, IEEE Transactions on ,Volume:50 , Issue: 2 ,pp. 174 – 188,2002.

[178] Branko Ristic, and Sanjeev Arulampalam, Beyond the Kalman Filter: Particle Filters for Tracking23 Applications. Boston, MA, 2004.

[179] Burges, C.J.C., "A tutorial on support vector machinesfor pattern recognition, in Data Mining and KnowledgeDiscovery". 1998. p. 121–167,1998.

[180] C. ANDRIEU et al. "An Introduction to MCMC for Machine Learning," Machine Learning, Kluwer Academic Publishers. Manufactured in The Netherland 50, 5–43,. 2003

[181] A. Banerjee et al . "Efficient Particle Filtering via Sparse Kernel Density Estimation," Image Processing, IEEE Transactions on ,Volume:19 , Issue: 9 , pp.2480 – 2490, 2010.

[182] M. Bolic et al. "Resampling Algorithms for Particle Filters: A Computational Complexity Perspective," EURASIP Journal on Advances in Signal Processing Volume 2004, 1 January 2004.

[183] G. CHAUHANet al., "Change Processes towards Flexible Lean Manufacturing: A Framework," International Journal of Performability Engineering, Vol.6, No. 4, pp.363-372,© RAMS Consultants, July 2010.

[184] M. Bengtsson . On condition based maintenance and its implementation in industrial settings, PhD thesis, Mälardalen University, Sweden, 2007.

[185] 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. 1 No. 1, 2010.

[186] Abhinav Saxena, Jose Celaya, Bhaskar Saha, Sankalita Saha, Kai Goebel, "Metrics

for evaluating performance of prognostic techniques,” Prognostics and Health Management, 2008. PHM 2008. International Conference on, Denver, CO,pp. 1 – 17, 6-9 Oct. 2008

[187] R.O. Duda, P.E. Hart, and D.G. Stork, Pattern Classification, New York: John Wiley & Sons, 2001.

[188] Haifeng Wang. &, Shanghai Jiao “Comparison of SVM and LS-SVM for Regression” Neural Networks and Brain, 2005. ICNN&B '05. International Conference on. Beijing,PP. 279 – 283, 13-15 Oct. 2005.

Vita

Mohammed Ali Lskaafi was born in Awamia, Saudi Arabia. He holds a B.Sc. Agriculture degree in Food Science and Technology from the King Saud University Riyadh, Saudi Arabia. In August 2009, he moved to Erie–Pennsylvanian where he attended Gannon University. There, he graduated with a M.S in Engineering Management. He moved to Knoxville, Tennessee, to study in the college of Engineering at the University of Tennessee, Knoxville, in August 2011. He is currently working towards his PhD at the University of Tennessee, Knoxville. His PhD thesis is based on exploring Artificial Intelligence techniques for data-driven Fault diagnosis and failure prognosis in Condition Based Maintenance. His research is focused on anomaly detection based on online Least Square Support Vector Machine, and real time prognosis via Memory-Particle Filter. Upon the completion of his studies, Mohammed intends to work in the industry as well as keep his affiliations with academia.

.