Zhenyou Zhang

Data Mining Approaches for Intelligent Condition-based Maintenance

A Framework of Intelligent Fault Diagnosis and Prognosis System (IFDPS)

Thesis for the degree of Philosophiae Doctor

Trondheim, May 2014

Norwegian University of Science and Technology
Faculty of Engineering Science and Technology
Department of Production and Quality Engineering

NTNU – Trondheim
Norwegian University of
Science and Technology

Acknowledge

It gives me an immense pleasure to present the thesis report in its completed form. First of all, I would like to thank my main supervisor Prof. Kesheng Wang for his guidance, but more importantly for his moral support through this research work. Without his timely advice and through knowledge in Data Mining and Condition-based Maintenance, the research would not have been accomplished such a great success. I am extremely thankful for his support.

I thank my co-supervisor Odd Myklebust for his support and help during these years.

I thank my colleagues PhD Candidate Quan Yu, Associate Prof. Yiliu Liu, Dr. Lijuan Dai, Dr. Rhythm S. Wadhwa and all of whom helped and supported me during these years.

I thank Prof. Lilan Liu who came to NTNU as a visit scholar from Shanghai University. I was very pleasure to work together with her during her half year visiting.

I thank Dr. Guijuan Lin who came to NTNU as an exchange PhD student from Tongji University. I was very pleased to work with her during three months.

I also thank two Master students, Deborah Cruciani and Roberta Cusanno, from University of Bologna to work together in fault diagnosis and maintenance scheduling optimization.

Last, but the most important, I am extremely grateful to my parents and my wife for being there for me all these years, for their patience, support and encouragement.

Trondheim, May 2014

Zhenyou Zhang

Abstract

Condition-based Maintenance (CBM) is a maintenance policy that take maintenance action just when need arises with real-time condition monitoring. Intelligent CBM means a CBM system is capable of understanding and making maintenance decisions without human intervention. To achieve this objective, it is needed to detect current conditions of mechanical and electrical systems and predict the fault of the systems accurately. What's more, the maintenance scheduling need to be optimized to reduce the maintenance cost and improve the reliability, availability and safety based on the results of fault detection and prediction.

Data mining is a computational process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems. The goal of the data mining is to extract useful information from a data set and transform it into an understandable structure for further use.

This thesis develops framework of Intelligent Fault Diagnosis and Prognosis System (IFDPS) for CBM based on Data Mining Techniques. It mainly includes two tasks: the one is to detect and predict the condition of the equipment and the other is to optimize maintenance scheduling accordingly. It contains several phases: sensor selection and its placement optimization, signal processing and feature extraction, fault diagnosis, fault prognosis and predictive maintenance scheduling optimization based on results of fault diagnosis and prognosis. This thesis applies different data mining techniques containing Artificial Neural Network such as Supervised Back-Propagation (SBP) and Self-Organizing Map (SOM), Swarm Intelligence such as Particle Swarm Optimization (PSO), Bee Colony Algorithm (BCA) and Ant Colony Optimization (ACO), and Association Rule (AI) in most of these phases.

The outcomes of the thesis can be applied in mechanical and electrical system in industries of manufacturing, wind and hydro power plants.

IV

Table of Contents

Table of Contents

List of Figures

List of Figures

List of Figures

XIII

List of Figures

List of Tables

List of Tables

Abbreviations

CBM   Condition-based Maintenance

IFDPS   Intelligent Fault Diagnosis and Prognosis System

DM   Data Mining

RUL   Remaining Useful Life

CM   Corrective Maintenance

TPM   Time-based Preventive Maintenance

RCM   Reliability Centered-Maintenance

EPRI   Electric Power Research Institute

FMEA Failure Mode and Effect Analysis

OM   Opportunity Maintenance

DOM   Design out Maintenance

PM   Predictive Maintenance

AM   Anticipatory Maintenance

RFID   Radio-frequency Identification

GA   Genetic Algorithm

SVM   Support vector machine

HMM   Hidden Markov Model

RBF   Radial-basis-Function

FLS   Fuzzy Logic System

ANN   Artificial Neural Network

HCI   Hybrid Computational Intelligence

FFT   Fast Fourier Transform

PSO   Particle Swarm Optimization

BCA   Bee Colony Algorithm

ACO   Ant Colony Optimization

STFT   Short Time Fourier Transform

FEA   Finite Element Analysis

SOM   Self-organizing Mapping

BP   Back-Propagation

SCADA   Supervisory Control and Data Acquisition system

Abbreviations

CI   Computational Intelligence

AR   Association Rules

DT   Decision Trees

SI   Swarm Intelligence

CFT   Continues Fourier Transform

DFT   Discrete Fourier Transform

WVD   Wigner-ville Distribution

TSA   Time Synchronous Averaging

WT   Wavelet Transform

WP   Wavelet Packet

SBP   Supervised Back-Propagation

SSL   Semi-supervised Learning

RKHS   Reproducing Kernel Hilbert Spaces

TID   Ttransaction Identifier

HBCA   Honey Bee Colony Algorithm

MEMS   Microelectromechanical System

RTD   Resistance Temperature Detector

WSN   Wireless Sensor Network

GPS   Global Positioning System

ARR   Analytical Redundancy Relation

PCA   Principal Component Analysis

BTA   Boosting Tree Algorithm

CWT   Continuous Wavelet Transform

DWT   Discrete Wavelet Transform

WPD   Wavelet Packet Decomposition

CBR   Case-based Reasoning

SDWPC   Standard Deviation of Wavelet Packet Coefficients

PDF   Probability Density Function

GMS   Generating Unit Maintenance Scheduling

IPSO   Improved PSO

RSOM   Routing and Scheduling Optimization of Maintenance

OWT   Offshore Wind Turbines

1 Introduction

1.1 Motivation of Present Work

With the rapid development of manufacturing, automobile, aeronautics and aerospace industries, the equipment of those become more and more complex and integrated, and thus an unanticipated breakdown of the equipment can cause more losses in economy and human sources. To avoid the unanticipated failure of equipment, the maintenance action should be performed before the machine becoming failure. The number of maintenance actions should not exceed its necessary, or it may increase the cost of maintenance and reduce the product life. Therefore, the maintenance action should be performed just before the machine failure. To reach this objective, condition monitoring has to be performed in equipment and processes of manufacturing and operations to support the maintenance decision. Therefore, the motivation for present work can be described as following paragraphs.

Because of the complex, integration and associativity of the equipment, the right maintenance policy of equipment has to be researched for reducing the loss and increasing the life cycle of products. As mentioned above, for the key components of equipment, the maintenance action should be performed just when it is necessary before failure. Condition-based Maintenance (CBM) policy is based on the condition of equipment and tries to maintain the correct equipment at right time. To implement CBM policy, the healthy condition of equipment needs to be assessed according to the real-time information from the sensors mounted on the equipment. The present work establishes a framework called IFDPS for CBM to reducing the maintenance cost and increase the life cycle of products.

To carry out CBM policy, equipment health must be assessed based on the condition information of the equipment. Fault diagnosis, which means detecting, isolating, and identifying an impending or incipient failure condition in which affected component is still operational even though at a degraded mode, is a very important technique to obtain the information. Normally, when a machine goes down, most of downtime is used to identify the causes of the failure while only a small part of that is used to repair or maintain the machine. Diagnostics can answer this question: why the performance of the observed process, or equipment is degrading, or in other word, what is the cause of the observed process or machinery degradation [Djurdjanovic et al., 2003]. It is to see that diagnostics' function is to identify the components or causes of the failure happening or about to happen. Therefore, the present work introduces many Data Mining (DM) methods to carry out diagnostics to tell the staff which components should be repaired or maintained.

Meanwhile, to carry out CBM policy, the fault prognosis is very important as well to support maintenance decision. The prognostics can answer the question: when the observed process, or equipment is going to fail, or degrade to the point when its performance becomes unacceptable [Djurdjanovic et al., 2003]. CBM policy can make predictive maintenance scheduling based on the condition of machine.

Condition, the state of a machine, is related to the Remaining Useful Life (RUL). In the industrial and manufacturing arenas, fault prognosis can be used to estimate the remaining useful life of a machine or a component once an impending failure condition is detected, isolated and identified. It is obviously seen that fault prognosis with fault diagnosis is a basis of predictive maintenance scheduling. Therefore, the present work also proposes methods for fault prognosis to support CBM policy.

To carry out CBM, condition monitoring is very important and obtaining parameter information of machine is the bases for all the processes include diagnostics, prognostics and predictive maintenance decision. Normally, sensors are used to collect information of machines. There are two issues need to be considered for sensors. The one is what kind of sensors should be chosen to collect the information. The other is where the sensors should be set up on the machine to get the information continuous or periodically. Actually, the present work focuses on the second issue, i.e. the sensor placement optimization.

Data Mining (DM) techniques could be very useful for maintenance scheduling, prognostics, diagnostics and sensor placement selection. Many companies, such as BMW, ABB, Boeing and Statoil, have lots of history data. But the data has not been used effectively in current time. DM techniques can be used to extract useful information from the history data to support all process mentioned above.

Therefore, during the three years of PhD work, Intelligent Fault Diagnosis and Prognosis System (IFDPS) for Condition-based Maintenance in Manufacturing systems and processes is established. The framework IFDPS includes almost all processes of sensor selection, sensor placement optimization, fault diagnosis, fault diagnosis and prognosis, and maintenance scheduling optimization. It is hoped that IFDPS can help the companies to carry out near-zero breakdown manufacturing and further to carry out zero-defect manufacturing.

1.2   Literature Review

As mentioned above, the present work mainly based on the maintenance policy, methods of diagnostics and prognostics, signal process and sensor strategy. In this section, the state-of-the-arts for these topics are reviewed briefly.

1.2.1   Review of Maintenance Strategies

Maintenance is defined [EN 13306: 2001, 2001] as the combination of all technical, administrative and managerial actions during the life cycle of an item intended to retain it in, or restore it to, a state in which it can perform the required function (a function or a combination of functions of an item which are considered necessary to provide a given service). It is a set of organized activities that are carried out in order to keep an item in its best operational condition with minimum cost acquired. The maintenance actions could be either repair or replacement activities, which are necessary for an item to reach its acceptable productivity condition, and these

activities should be carried out with a minimum possible cost. In the period of pre-World War II, people saw maintenance as an added cost to the plant which did not increase the value of finished product, and thus, the maintenance at that era was restricted to fixing the unit when it breaks because it was the cheapest option. During and after World War II at the time when the advances of engineering and scientific technology developed, people developed other types of maintenance, which were much cheaper such as preventive maintenance and in addition, people in this era classified maintenance as a function of the production system. Nowadays, increased awareness of such issues as environment safety, quality of product and services makes maintenance one of the most important functions that contribute to the success of the industry and world-class companies are in continuous need of a very well organized maintenance plan to compete world-wide. The brief history of maintenance mentioned above can be seen in Fig. 1.1 [Shenoy & Bhadbury, 1998].
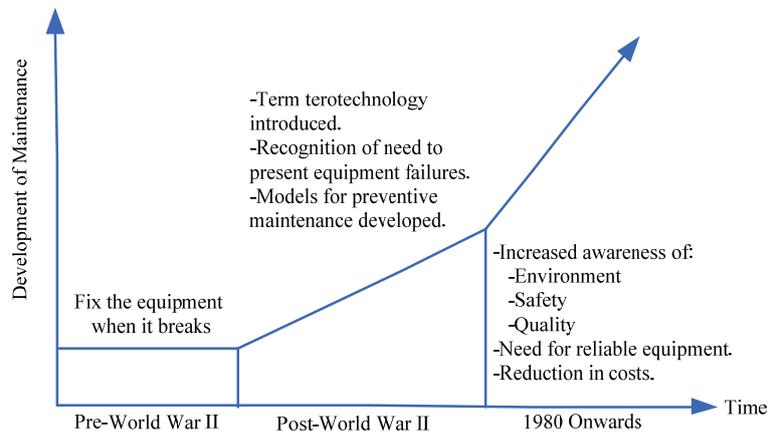
Fig. 1.1 Maintenance History

It is very important for a manufacturing company to choose a right maintenance policy because the affections of maintenance are not only on economy, reliability and availability but also on personnel safety. The range of maintenance cost is from 15% for manufacturing companies and 40% for iron and steel industry of the whole cost of manufactured parts and machines in 1990s [Keith Mobley, 2002] and even more nowadays. The corresponding cost in United Stated is more than 200 billion dollars every year [Chu et al., 1998]. This shows the significance of maintenance in the viewpoint of economy. Unexpected failure causes tremendous losses in economy and production, and may also cause hazard of staff and equipment in manufacturing plant. Therefore, the maintenance actions are performed before the failure is very important which is referring to the preventive maintenance and predictive maintenance.

Maintenance objectives should be consistent with and subordinate to production goals. The relation between maintenance objectives and production goals is

reflected in the action of keeping production machines and facilities in the best possible condition. Typically, the objectives of maintenance can be classified into three groups [Boucly, 2001; Marquez, 2007; Wireman, 1990]:

- Technical objectives. These objectives are the operational imperatives from the business sector of a company or plant. In general, operational imperatives are linked to a satisfactory level of equipment availability and people safety. A generally accepted method to measure the fulfillment of this goal is the Overall Equipment Effectiveness (OEE), as described in TPM method [Nakajima, 1988].
- Legal objectives/Mandatory regulations. Normally it is a maintenance objective to fulfill all these existing regulations for electrical devices, pressure equipment, vehicles, protection means, etc.
- Financial objectives. To satisfy the technical objective at the minimum cost. From a long term perspective global equipment life cycle cost should be a suitable measure for this.

Generally, the objectives can be list as bellowing:

1) Maximizing production or increasing facilities availability at the lowest cost and at the highest quality and safety standards.
2) Reducing breakdowns and emergency shutdowns.
3) Optimizing resources utilization.
4) Reducing downtime.
5) Improving spares stock control.
6) Improving equipment efficiency and reducing scrap rate.
7) Minimizing energy usage.
8) Optimizing the useful life of equipment.
9) Providing reliable cost and budgetary control.
10) Identifying and implementing cost reductions.

A maintenance action may include a set of maintenance activities: inspection, monitoring, routine maintenance, overhaul, rebuilding and repair. Inspection can be performed by measuring, observing, testing or gauging the relevant features of an item before, during or after other maintenance activity. Monitoring is a kind of activities performed manually or automatically, continuously or periodically intended to obtain the actual state of the equipment which can be used to evaluate parameters changes of the equipment when the equipment is in operating state. Routine maintenance is a kind of regular elementary maintenance activities, such as cleaning, tightening of connections and checking lubrication, which usually do not need special qualification authorization or tools. Overhaul is a comprehensive set of examinations and actions performed at prescribed intervals of time or a number of operations in order to maintain the required level of reliability, availability and safety, and sometimes may require partial or complete dismantling of the items. Rebuilding is performed when the equipment or components are approaching their useful life or should be regularly replaced in order to provide the equipment with a useful life that may be greater than the lifespan of the original equipment. Repairing is a physical action to restore the required functions of faulty equipment [Marquez, 2007]. A maintenance action could include some of one or

more above activities. The maintenance may also need fault diagnosis and prognosis for monitored equipment.

With a long history development, maintenance has been made great progress. At the beginning, maintenance action is performed when the equipment become failure. However, this kind of maintenance policy cannot meet the requirement of the industry and many other types of maintenance are emerged during the several decades as seen in Fig. 1.2 [EN 13306: 2001, 2001]. In many literatures, Condition-based Maintenance (CBM) is also called predictive maintenance. This section mainly reviews corrective maintenance and preventive maintenance briefly, and review predictive maintenance in detail.
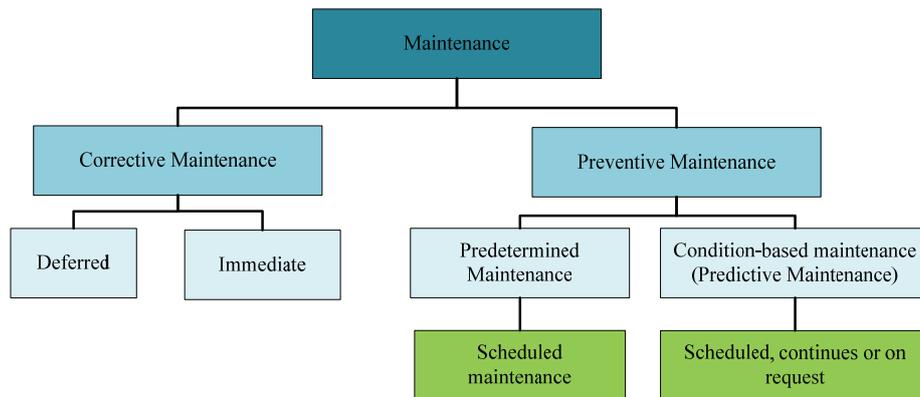
Fig. 1.2 Maintenance Types

1.2.1.1   Corrective Maintenance (CM)

Corrective Maintenance is similar to repair work, which is undertaken after a breakdown or when obvious failure has been located. That is why it is also called run-to failure maintenance, maintenance-on-failure or breakdown maintenance. In CM, the plant item is allowed to failure before maintenance is performed and thus that it is only suitable if the consequences of failures are small, such as light bulb. It is only appropriate to apply CM policy if it does not matter whether the machine fails, or how long the repair will take or how much it will cost. Sometimes a failure is not predictable using any instrument or analysis, and only checking for failure will detect the fault. Unfortunately the strategy is widely used in inappropriate situations. At failure, the task of the repair team is to restore the machine to a state in which it can perform the required function as quickly as possible [Holmberg et al., 2010]. Therefore, CM at its best should be utilized only in non-critical areas where capital costs are small, consequences of failure are slight, no safety risks are immediate, and quick failure identification and rapid failure repair are possible.

Corrective maintenance is maintenance carried out after fault recognition and intended to put the equipment into a state in which it can perform a required function. It could be immediate or deferred [Marquez, 2007]. Immediate maintenance means the maintenance is carried out without delay after a fault has been detected to avoid unacceptable consequences, while deferred maintenance

means the maintenance is not immediately carried out after fault detection but is delayed according to given maintenance rules.

The CM policy has its advantages. Its planning is very simple because the maintenance action is needed only when the failure happens and the plan is only to consider the failure rate. The maintenance work is not scheduled until it is really needed. However, it has major disadvantages [Holmberg et al., 2010]:

- Failure can, and probably will, occur at an inconvenient time, *e.g.*, when the plant is at full load, or while it is starting.
- A component fault may go unnoticed, leading to expensive consequential damage, *e.g.*, bearing seizure causes damage to a shaft.
- Dangerous and/or expensive failure consequences should be expected.
- No data are available regarding the past, present and possible future state of the machine.
- A large breakdown crew may need to be available on standby. All the required expertise should be either within the plant or easily accessed from external resources, which is almost always costly, or a longer waiting time should be expected.
- A large spares inventory is necessary to ensure quick repair.
- Failures exceeding the capacity of the repair team lead to "fire-fighting".

1.2.1.2   Preventive Maintenance

Preventive Maintenance can be defined as maintenance carried out at predetermined intervals or according to prescribed criteria and intended to reduce the probability of failure or the degradation of the functioning of an item [EN 13306: 2001, 2001]. The preventive maintenance action can be condition-based or predetermined maintenance. Predetermined maintenance carried out in accordance with established intervals of time or number of units of use but without previous condition investigation. Condition based maintenance is preventive maintenance based on performance and/or parameter monitoring and the subsequent actions which is also known as predictive maintenance which is reviewed in section 1.2.1.3.

Predetermined maintenance means that the maintenance is scheduled without the occurrence of any monitoring activities [Niu et al., 2010; Zhang & Wang, 2013]. The scheduling can be based on the number of hours in use, the number of times an item has been used the number of kilometers the items has been used, according to prescribed dates.

While predetermined maintenance is not the optimum maintenance program, it does have several advantages over that of a purely corrective program as follows [Sullivan et al., 2010]:

- Cost effective in many capital-intensive processes.
- Flexibility allows for the adjustment of maintenance periodicity.
- Increased component life cycle.
- Energy savings.
- Reduced equipment or process failure.

- Estimated 12% to 18% cost savings over corrective maintenance program.

However, there are still some disadvantages [Sullivan et al., 2010]:

- Catastrophic failures still likely to occur.
- Labor intensive.
- Includes performance of unneeded maintenance.
- Potential for incidental damage to components in conducting unneeded maintenance.

1.2.1.3   Predictive Maintenance (PM)

Condition based maintenance is also known as predictive maintenance which means a set of activities that detect changes in the physical condition of equipment (signs of failure) in order to carry out the appropriate maintenance work for maximizing the service life of equipment without increasing the risk of failure. It depends on continuous or periodic condition monitoring equipment to detect the signs of failure.

Condition-based Maintenance (CBM) is most popular policy in modern industries [Carnero Moya, 2004; Dieulle et al., 2001; Han & Song, 2003]. CBM is maintenance when need arises which is performed after one or more indicators show that equipment is going to fail or that equipment performance is deteriorating. It was introduced in 1975 in order to maximize the effectiveness of PM decision making. CBM is a maintenance program that recommends maintenance actions (decisions) based on the information collected through condition monitoring process [Jardine et al., 2006]. In CBM, the lifetime (age) of the equipment is monitored through its operating condition, which can be measured based on various monitoring parameters, such as vibration, temperature, lubricating oil, contaminants, and noise levels. The motivation of CBM is that 99 percent of equipment failures are preceded by certain signs, conditions, or indications that a failure is going to occur [Bloch & Geitner, 2012]. Therefore, CBM is needed for better equipment health management, lower life cycle cost, catastrophic failure avoidance etc. [Ahmad & Kamaruddin, 2012].

The heart of CBM is the condition monitoring process, where signals are continuously monitored using certain types of sensor or other appropriate indicators [Campos, 2009]. Thus, maintenance activities (e.g., repairs or replacements) are performed only 'when needed' or just before failure [Andersen & Rasmussen, 1999]. In general, the main goal of CBM is to perform a real-time assessment of equipment conditions in order to make maintenance decisions, consequently reducing unnecessary maintenance and related costs [Gupta & Lawsirirat, 2006].

Monitoring is defined as: 'An activity which is intended to observe the actual state of an item' [SS-EN 13306, 2001]. In other words, condition monitoring is a tool used to indicate the condition of equipment in a system [Hameed et al., 2009]. In general, the purpose of the condition monitoring process is twofold. First, it collects the condition data (information) of the equipment. Second, it increases

knowledge of the failure causes and effects and the deterioration patterns of equipment [Ahmad & Kamaruddin, 2012].

The condition monitoring process can be carried out into two ways: on-line and off-line. On-line processing is carried out during the running state of the equipment (operating state), while off-line processing is performed when the equipment is not running. In addition, condition monitoring can be performed either periodically or continuously. Typically, periodical monitoring is carried out at certain intervals, such as every hour or every working shift end, with the aid of portable indicators, such as hand-held meters, acoustic emission units, and vibration pens. The condition monitoring process also includes evaluations based on human senses to measure or evaluate equipment conditions, such as degree of dirtiness and abnormal color. As for continuous monitoring, as its name suggests, monitoring is performed continuously and automatically based on special measurement devices, such as vibration and acoustic sensors.

There are two main limitation of continuous monitoring exist: it is expensive because many special devices are required and inaccurate information may be obtained because the continuous flow of data creates increased noise. In contrast, the main limitation of periodic monitoring is the possibility of missing some important information of equipment failure between monitoring intervals [Jardine et al., 2006]. Most equipment failures are preceded by certain signs, conditions, or indications that such a failure was going to occur and many condition monitoring techniques can be used to monitor equipment conditions [Bloch & Geitner, 2012].

PM has some advantages over other maintenance policies: 1) Improving availability and reliability by reducing downtime; 2) Enhancing equipment life by reducing wear from frequent rebuilding, minimizing potential for problems in disassembly and reassembly and detecting problems as they occur; 3) Saving maintenance costs by reducing repair costs, reducing overtime and reducing parts inventory requirements; 4) Decreasing number of maintenance operations causes decreasing of human error influence. However, there are still some challenges of PM: 1) Initiating PM is costly because the cost of sufficient instruments could be quite large especially if the goal is to monitor already installed equipment; 2) The goal of PM is accurate maintenance, but it is difficult to achieve for the complexity of equipment and environment; 3) Introducing PM will invoke a major change in how maintenance is performed, and potentially to the whole maintenance organization in a company. Organizational changes are in general difficult.

There are many kinds of techniques, such as sensors techniques, signal process techniques, fault diagnosis techniques, fault prognosis techniques and maintenance optimization techniques, can be used to support maintenance decision making. All these techniques will be reviewed.

1.2.2 Review of Sensor System and Sensor Placement Optimization

1.2.2.1 Sensor Classification

There are many kinds of data could be collected from the system because of the complex and integrated manufacturing system and process [Vachtsevanos et al., 2006]. Therefore, the selection of the suitable sensors is the key for effective condition monitoring. A variety of sensors exist to effectively monitor and control various process parameters (Fig. 1.3):

- Mechanical sensors such as acceleration, position, displacement, speed sensors and strain gauges etc.;
- Performance sensors such as pressure, fluid and thermodynamic sensors etc.;
- Electrical measurement sensors such as eddy-Current proximity probes and micro-electromechanical system sensors etc.;
- Fibre-optic sensors.

Mechanical Sensor Systems
- Accelerometers (Vibration Measurements)
- Strain gauges
- Ultrasonic Sensor System
- Position, speed, acceleration, torque, strain
- ......

Performance Sensors
- Temperature Sensors / Thermography
- Pressure, Fluid and thermodynamic
- Optical properties and biochemical elements
- ....

Electrical Measurement
- Eddy-Current Proximity Probes
- Microelectromechanical System (MEMS) Sensors

Fiberoptic Sensors

Fig. 1.3 The Classification of Sensors.

Mechanical sensor systems have been studied extensively, and a large number of such devices are currently in use to monitor system performance for operational state assessment and tracking of fault indicators. A number of mechanical quantities - position, speed, acceleration, torque, strain, temperature, etc. - are commonly employed in dynamic systems. Most of devices for measuring these quantities are available commercially, and their operation has been amply described in textbooks and publications [Silva, 1989; Stuart & Allocca, 1984]. However, the most useful Mechanical sensors for condition monitoring are accelerometers and strain gauge.

System performance and operational data are monitored routinely in all industrial establishments, utility operations, transportation systems, etc. for process control, performance evaluation, quality assurance, fault diagnosis and prognosis, and

maintenance decision support purposes. A large number of sensor systems have been developed and employed over the years. The list includes devices that are intended to measure such critical properties as temperature; pressure; fluid, thermodynamic, and optical properties; and biochemical elements, among many others. Sensors based on classic measuring elements—inductive, capacitive, ultrasound-have found extensive applications. More recently, biochemical sensors have begun taking central stage, and their detection principles and requirements are described in the technical literature [Guardia, 1995]. Characteristic chemical-sensor properties with potential application to structural fault diagnosis include liquid and solid electrolytic sensors, photochemical sensors, humidity sensors, and field-effect and mass-sensitive devices. As an example of their principles of operation, consider conductivity sensors. In these devices, the interaction of the gas with the solid (semiconducting metal oxide or organic semiconductor) causes a change in conductivity. A change in resistance also can be caused by a change in the temperature of the sensor material [Vachtsevanos et al., 2006].

Electromechanical, electrical, and electronic systems constitute a major component of industrial engine. They are the dominant element in such areas as transportation systems, utilities, biomedical instrumentation, communications, computing, etc. A number of sensor systems have been developed and applied in the recent past in an attempt to interrogate critical components and systems for fault diagnosis and prognosis. Transducing principles based on eddy-current response characteristics, optical and infrared signal mentoring, microwaves, and others have been investigated [Vachtsevanos et al., 2006; Zou et al., 2000].

Fiber optics has penetrated the telecommunications and other high technology sectors in recent years. They find utility in the sensor field because of their compact and flexible geometry, potential for fabrication into arrays of devices, batch fabrication, etc. Fiberoptic sensors have been designed to measure strain, temperature, displacement, chemical concentration, and acceleration, among other material and environmental properties. Their main advantages include small size, light weight, immunity to electromagnetic and radio frequency interference, high- and low-temperature endurance, fast response, high sensitivity, and low cost. The basic physics leads to a very stable, accurate, and linear temperature sensor over a large temperature range. These sensors are also quite small and therefore ideal for applications where restricted space or minimal measurement interference is a consideration. The size also leads to a very small time response as compared with other temperature measurement techniques [Ansari, 1998; Lienhart & Brunner, 2003; Vachtsevanos et al., 2006].

Normally, the output of sensor is electrical signal whatever the physical signals is. The electrical signals need to be transferred to a database for analysis. The signals can be transferred with cables or wireless network. Recently, a fast development technology RFID can be used to transfer the signals. Transferring signals with cables is a kind of traditional methods and is very effective. However, for some companies such as wind generator plant, the monitoring equipment may be far away from the plant and thus wireless network and RFID could be used to solve this problem.

1.2.2.2   Wireless Sensor Networks (WSNs)

Wireless sensor networks (WSNs) are important in applications where wires cannot be run owing to cost, weight, or accessibility. Properly designed WSNs can be installed and calibrated quickly and can be up and running in a very short time frame [Lewis, 2004]. Typically, WSNs generally consist of a data-acquisition network, a data-distribution network monitored and controlled by a management center as shown in Fig. 1.4 [Lewis, 2004]. Too many of available technologies make even the selection of components difficult, let alone the design of a consistent, reliable, robust overall system.

The basic issue in WSNs is the transmission of messages to achieve a prescribed message throughput and quality of service which can be specified in terms of message delay, message due dates, bit error rates, packet loss, economic cost of transmission, transmission power, etc. Depending on quality of service, the installation environment, economic considerations, and the application, one of several basic network topologies may be used. A communication network is composed of nodes, each of which has computing power and can transmit and receive messages over communication links, wireless or cabled. The basic network includes fully connected, mesh, star, ring, tree, bus as shown in Fig. 1.5 [Lewis, 2004]. A single network may consist of several interconnected subnets of different topologies. Networks are further classified as Local Area Networks (LAN), e.g. inside one building, or Wide Area Networks (WAN), e.g. between buildings.
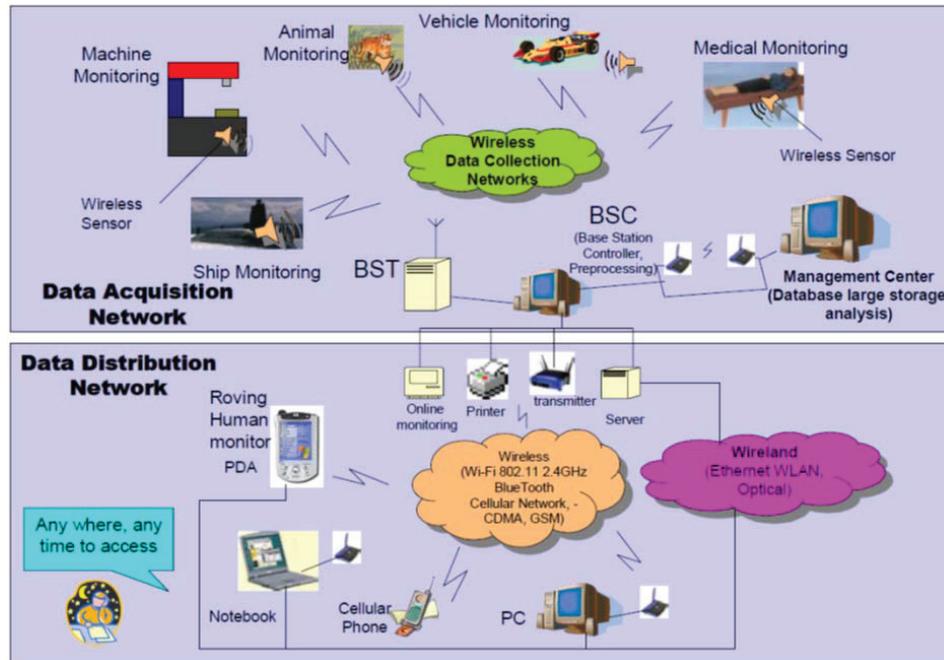
Fig. 1.4 Wireless Sensor Networks

11
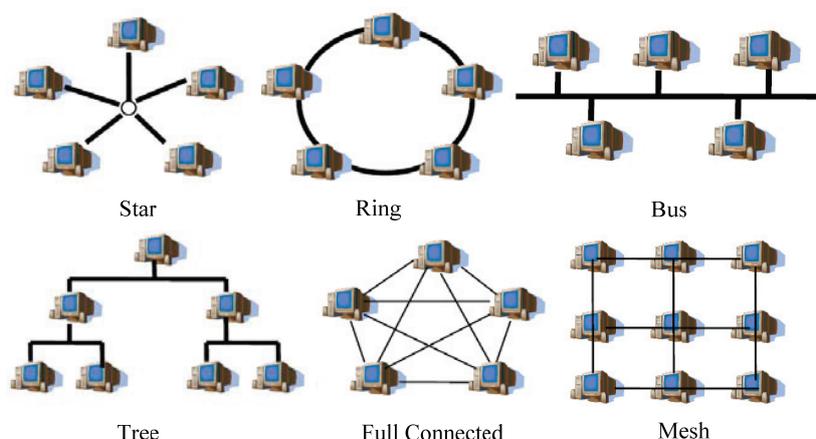
Fig. 1.5 Basic Network Topologies

Fully connected networks suffer from problems of NP-complexity [Garey & Johnson, 1979]; as additional nodes are added, the number of links increases exponentially. Therefore, for large networks, the routing problem is computationally intractable even with the availability of large amounts of computing power. Mesh networks are regularly distributed networks that generally allow transmission only to a node's nearest neighbors. The nodes in these networks are generally identical, so that mesh nets are also referred to as peer-to-peer nets. Mesh nets can be good models for large-scale networks of wireless sensors that are distributed over a geographic region. Since there are generally multiple routing paths between nodes, these nets are robust to failure of individual nodes or links. An advantage of mesh nets is that, although all nodes may be identical and have the same computing and transmission capabilities, certain nodes can be designated as 'group leaders' that take on additional functions. If a group leader is disabled, another node can then take over these duties [Lewis, 2004]. Star topology means that all nodes are connected to a single hub node. The hub requires greater message handling, routing, and decision-making capabilities than the other nodes. If a communication link is cut, it only affects one node. However, if the hub is incapacitated the network is destroyed. Ring topology means all nodes perform the same function and there is no leader node. Messages generally travel around the ring in a single direction. However, if the ring is cut, all communication is lost. In the bus topology, messages are broadcast on the bus to all nodes. Each node checks the destination address in the message header, and processes the messages addressed to it. The bus topology is passive in that each node simply listens for messages and is not responsible for retransmitting any messages [Lewis, 2004].

1.2.2.3 Radio-frequency Identification (RFID)

Radio-frequency identification (RFID) is one of numerous technologies grouped under the term of Automatic Identification (Auto ID), such as bar code, magnetic inks, optical character recognition, voice recognition, touch memory, smart cards, biometrics etc. Auto ID technologies are a new way of controlling information and

material flow, especially suitable for large production networks [Ilie-zudor et al., 2006]. RFID is the use of a wireless non-contact radio system to transfer data from a tag attached to an object, for the purposes of identification and tracking [http://en.wikipedia.org/wiki/Radio-frequency_identification]. In general terms, it is a means of identifying a person or object using a radio frequency transmission. The technology can be used to identify, track, sort or detect a wide variety of objects [Lewis, 2004]. Recently, RFID become more and more interesting technology in many fields such as agriculture, manufacturing and supply chain management.

The history of RFID technology can be tracked back to the radio-based identification system used by allied bombers during World War II [Garfinkel & Holtzman, 2005]. Early identification Friend or For (IFF) systems were used to distinguish Allied fighter and bomber by identifying the correct signals sent by Allied aircrafts, from aircrafts sent by enemy at night. After the war, Harry Stockman realized that it is possible to power a mobile transmitter completely from the strength of a received radio signal, and then he introduced the concept of passive RFID systems [Stockman, 1948]. In 1972, a patent application for "inductively coupled transmitter-responder arrangement" was filed which is used separate coils for receiving power and transmitting the return signal [Kriofsky & Kaplan, 1975]. In 1979, a patent application for "identification device" (two antennas was combined) was filed which is seen as a RFID landmark because it emphasized the potentially small size of RFID device [Beigel, 1982]. The 1980s became the decade for full implementation of RFID technology, though interests developed somewhat differently in various parts of the world. The greatest interests in the United States were for transportation, personnel access, and to a lesser extent, for animals. In Europe, the greatest interests were for short-range systems for animals, industrial and business applications, though toll roads in Italy, France, Spain, Portugal, and Norway were equipped with RFID. The 1990s was a significant decade for RFID since it saw the wide scale deployment of electronic toll collection in the United States. The world's first open highway electronic tolling system opened in Oklahoma in 1991 and then extended to the whole world. Interest was also keen for RFID applications in Europe during the 1990s. Both Microwave and inductive technologies were finding use for toll collection, access control and a wide variety of other applications in commerce [Landt, 2001]. The 21st century opens with the smallest microwave tags built using, at a minimum, two components: a single custom CMOS integrated circuit and an antenna. Tags could now be built as sticky labels, easily attached to windshields and objects to be managed [Landt, 2005]. It seems that there are still a great many developments of RFID to look forward to as the history continues to teach that and RFID will be presented in our daily life.

As mentioned above, most of applications of RFID are in logistics or Auto ID area. However, from its principle, it is possible to apply this technology in signal transmission in condition monitoring (vibration measuring). However, there are no matured products of the RFID sensor for measuring vibration in production so far. Generally, there are two kinds of RFID vibration sensor could be developed. The one is combining the RFID tag and vibration sensor together to compose a new

RFID vibration sensing tag. The other is to connect vibration sensor to RFID tag and the RFID system only used to transmit vibration data to RFID reader and further to host computer. This application can make the measuring vibration become very flexible and effective [Wang & Zhang, 2012].

1.2.2.4   Sensor Placement Optimization

The basic problem for condition monitoring is to deduce the existence of a defect in a structure from measurements taken at sensors distributed on the structure. The correctness of defect diagnosis depends on the method of pattern recognition for fault and effectiveness of signals from the sensors mounted on the machines. While carrying out on-site condition monitoring for a machine, the inappropriate distribution of sensors might result in weak incentives of certain order or modal, and affect the accuracy of fault identification. The aim of optimizing the placement of sensors is to obtain as much as possible of machine structural information with as few as possible sensors, which benefit the company in the economy viewpoint. Because of constraints of machine structure and environment, and consideration of economy, only a small number of sensors are installed when a condition monitoring system is established. It is very important to design the optimal position of the sensor to mount in order to ensure the accuracy and correctness of monitoring and fault judgment.

There are many literatures in optimal placement optimization of sensors in machine level. The spatial controllability was used to find the optimal placement of collocated actuator-sensor pairs for effective average vibration reduction over the entire structure, and the maintaining modal controllability and observability were used to select vibration modes for a thin plate [Halim & Moheimani, 2003]. Recently, intelligent optimization algorithm has developed well which is a method to simulate the biological and physical process which can be used in sensor placement optimization. Many researchers focus on Genetic Algorithm (GA) application in sensor placement optimization and make up for a lot of shortage of the traditional optimization algorithm [Li et al., 2000; Liu et al., 2008; Sun et al., 2008]. But GA has to adopt binary coding and has complex operation process such as mutation, genetic and crossover. PSO adopts real number coding to avoid the complex operation, which is simple and easy to realize. So it is easy to apply in sensor placement optimization. PSO and finite element analysis were combined together to search the sensors optimal placement of a gearbox [Pan et al., 2010]. Binary PSO and Analytical redundancy Relations (ARRs) were combined to optimize the sensor placement for fault diagnosis [Du et al., 2011]. The sensor placement optimization is a very important aspect for many applications such as modal test and parameter identification [Cherng, 2003; Papadimitriou, 2004; Pennacchi & Vania, 2008], fault diagnosis [Bhushan & Rengaswamy, 2000; Staszewski, 2002; Worden & Burrows, 2001] and process monitoring [Wang et al., 2002]. The PhD work tries to apply Swarm Intelligence (SI) such as Particle Swarm Optimization (PSO) and Bee Colony Algorithm (BCA), and finite element analysis in sensor placement optimization in order to get enough information of machine structure using a small number of sensors and ensure the accuracy and correctness of condition monitoring.

1.2.3   Review of Fault Diagnosis and Prognosis

During a system failure, only a small fraction of the downtime is spent to maintain or repair the components that cause the fault. Up to 80% of that is spent to locate the source of the fault [Kegg, 1984]. In case of complex installation such as automotive manufacturing plant, one minute downtime may cause as high as $20,000 cost [Spiewak et al., 2000]. Early fault diagnosis is crucial for avoiding major malfunction and massive loss in economy and productivity. In diagnosing rotating machinery, sound emissions or vibration signals are used to monitor the performance of the machine and could be used to judge whether the machine is failure or degrading. Many useful techniques for signal analysis have been applied. These techniques can be classified into three types: time domain [Chen et al., 2008; Wang et al., 2010], frequency domain such as Fast Fourier Transform [Corinthios, 1971; Liu et al., 2010; Rai & Mohanty, 2007] and time-frequency domain such as the Short Time Fourier Transform [Portnoff, 1980], Hilbert-Huang Transform [Yu et al., 2007], Wigner-ville distribution [Andria et al., 1994; Staszewski et al., 1997; Wang et al., 2008] and Wavelet Transform [Dongyan Chen & Trivedi, 2005; Lin & Qu, 2000; Prabhakar et al., 2002; Seker & Ayaz, 2003; Tse et al., 2004; Wu & Chen, 2006; Wu & Kuo, 2009; Wu & Liu, 2009; Zheng et al., 2002]. Autoregressive model method can also be used to extract features of a machine or component for fault diagnosis and prognosis [Li et al., 2009]. Wavelet transform is the best of these tools because short time Fourier transform only provides a constant time-frequency resolution, and Wigner-ville distribution produced interface terms on the time-frequency domain in a critical condition [Wu & Chen, 2006]. It has particular advantages for characterizing signals at different localization levels in time as well as signal processing, image processing, pattern recognition, seismology and machine fault diagnosis.

After processing vibration signals and extracting the features, the more important thing is identifying the fault and predicting the remaining useful life. There are many methods could be used in this area. Support vector machine (SVM) learning is a popular machine learning application due to its high accuracy and good generalization capabilities [Saravanan et al., 2008]. Li et al. [Li et al., 2005] proposed a hidden Markov model (HMM)-based fault diagnosis in speed-up and speed-down process for rotary machinery. In the implementation of the system, one PC was used for data sampling and another PC was used for data storage and analysis. Wu and Chow [Wu & Chow, 2004] presented a self-organizing map (SOM) based radial-basis-function (RBF) neural network method for induction machine fault detection. The system was implemented by utilizing a PC and additional data acquisition equipment. Many methods based on ANN have been developed for online surveillance with knowledge discovery, novelty detection and learning abilities [Kasabov, 2001; Markou & Singh, 2003; Marzi, 2004]. ANN, Fuzzy Logic System (FLS), Genetic Algorithms (GA) and Hybrid Computational Intelligence (HCI) systems were applied in fault diagnosis and a case of centrifugal pump was utilized to show how the methods work [Wang, 2002]. Decision tree method was used to identify fault in of mean shifts in bivariate processes in real time [He et al., 2011]. Probability based Bayesian network methods was used to identify vehicle fault which can be used to diagnose single-fault and multi-fault

[Huang et al., 2008]. Lee, et al. [Lee et al., 2006] developed an intelligent prognostics and e-maintenance system named "Watchdog Agent" with the method of Statistical matching, and performance signature and Support Vector Machine (SVM) based diagnostic tool.

There exist some literatures integrating these techniques for fault diagnosis and prognosis. Momoh and Button integrated FFT and ANN to analyze and identify the fault of aerospace DC arcing [Momoh & Button, 2003]. Fourier transform and wavelet transform were integrated to detect and identify the fault of induction motor using stator current information [Lee, 2011]. Wavelet analysis techniques and ANN were integrated for fault diagnosis in induce motors [Lee, 2011], automotive generator [Wu & Kuo, 2009] and gear box [Saravanan & Ramachandran, 2010] and the results were pretty good. In the PhD work, some techniques are integrated together to classify and predict fault and further to predict the remaining useful life. These results can be used to support the maintenance decision making and optimizing the scheduling.

1.2.4 Review of Maintenance Scheduling Optimization

As mentioned above, PM is a dynamic schedule according to the state of equipment from continuous and/or periodic inspection. It utilizes the product degradation information extracted and identified from on-line sensing techniques to minimize the system downtime by balancing the risk of failure and achievable profits. Mathematically, the maintenance scheduling problem is a multiple-constraint, non-linear and stochastic optimization problem. This kind of problem has been studied for several decades and many kinds of different methods have been applied to solve it. Two methods for PM optimization had been developed during 1980s. The first method [Perla, 1984; Walker, 1987] performs cost/benefit analysis of each analyzed piece of manufacturing equipment. It is based on identifying important equipment firstly, and then predicting its future performance with and without changes in the regularly scheduled maintenance program. The second approach is the Reliability-Centered Maintenance (RCM) [Crellin, 1986; Hook et al., 1987; Vasudevan, 1985]. This methodology was adopted from the commercial air transport industry. It is based on a series of orderly steps, including identification of system/subsystem functions and failure modes, prioritization of failures and failure modes (using a decision logic tree), and finally selection of PM tasks that are both applicable (i.e. have the potential of reducing failure rate) and effective (i.e. economically worth doing). In the last two decades, many kinds of intelligent computational methods, such as the artificial neural network method, simulated annealing method, expert system, fuzzy systems and evolutionary optimization, have been applied to solve the maintenance scheduling problem and obtained many very exciting results [Huang, 1998; Miranda et al., 1998; Satoh & Nara, 1991; Sutoh et al., 1994; Yoshimoto et al., 1993]. And also, with the rapid development of the evolutionary theory, genetic algorithms (GAs) had become a very powerful optimization tool and obtained wide application in this area [Arroyo & Conejo, 2002; Back et al., 1997; Huang et al., 1992; Lai, 1998; Lee & Yang, 1998; Wang & Handschin, 2000]. In recently years, several new intelligent

computational methods such as Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) have been applied in preventive maintenance scheduling [Benbouzid-Sitayeb et al., 2008; Pereira et al., 2010; Yare & Venayagamoorthy, 2010].

All the above methods of maintenance scheduling are based on the specified time periods other than based on the condition of the equipment or facilities. PM is a good strategy which could be used to improve reliability and increase useful life of the equipment and reduce the cost of maintenance according to the condition of machine. When the condition of a system, such as its degradation level, can be continuously monitored, PM policy can be implemented, according to which the decision of maintaining the system is taken dynamically on the basis of the observed condition of the system. Recently, genetic algorithms, Monte Carlo method, Markov and semi-Markov methods are applied in PM [Amari et al., 2006; Barata et al., 2001, 2002; Be′ renguer et al., 2000; Grall et al., 2008; Marseguerra et al., 2002]. Normally, to make a dynamic PM scheduling, there are main three tasks as following and will be discussed in Chapter 8.

- Establishing a predictive maintenance model mathematically;
- Finding a suitable optimization method to optimize the predictive maintenance model;
- Making a dynamic maintenance decision based on the predictive model and optimization method.

1.3   Contributions

This section provides the overview of scientific contributions to the topic of Condition-based Maintenance (CBM) especially in data mining approaches. CBM is a technique that has not yet been implemented on a large scale in industry. Many companies have installed various sensors on their equipment and used this gathered information to determine the current health of the system. However, the gathered information has seldom been used effectively for fault diagnosis and prognosis. This PhD work tries to find an easy way to implement CBM technique.

The main contributions of this thesis are located in data mining approaches other than the other techniques such as model based or statistical methods. The framework contains many relevant aspects, i.e. sensor placement optimization, signal processing and feature extraction, fault diagnosis and prognosis, and maintenance scheduling optimization. The contributions of this thesis are described as follows.

During the PhD work, the framework called Intelligent Fault Diagnosis and Prognosis System (IFDPS) for Condition-based Maintenance (CBM) is established. The thesis tried to apply data mining techniques in most of the aspects in this framework.

Sensor placement optimization is a nontrivial problem of fault diagnosis and prognosis for equipment. Optimal positions of sensors mounted on equipment can

improve the effectiveness and reliability of condition monitoring and improve the quality of data collection. This thesis proposes a method for sensor placement optimization in machine level by combining Finite Element Analysis (FEA) and Swarm Intelligence, i.e. Particle Swarm Optimization (PSO) and Bee Colony Algorithm (BCA). The method can find the optimal positions of a number of sensors.

Techniques of signal processing and feature extraction are crucial for obtaining key performance information so that the system can diagnose and prognose effectively. The thesis analyze the vibration signal through traditional methods such as fast Fourier transform (FFT), short-time Fourier transform (STFT) and some modern signal analysis techniques such as wavelet transform, etc. These techniques together feature extraction method such as Principal Component Analysis (PCA)

For fault diagnosis, the thesis combines the methods of signal processing and feature extraction mentioned above, and some data mining techniques such as Artificial Neural Network (ANN) and Self-organizing Map (SOM). These methods can detect and diagnose the fault effectively.

For fault prognosis, the thesis proposes a methodology to predict the indicator of component fault based on the collected information by sensors and ANN other than based on the traditional statistics methods. This methodology has already applied to wind turbine fault prognosis and it works effectively. The method establishes ANN model for the indicator in normal condition of wind turbine using the history SCADA which is collected by wind farm operator but not use effectively. Then the thresholds of different conditions can be set by using the history data with different extent fault. Finally ANN model can be applied online to monitor the wind turbines and gives staff earn warning of fault so that they can schedule the maintenance actions in advance to reduce downtime, production loss and maintenance cost.

For different purpose, the different maintenance models are established. Based on these models, the maintenance schedule can be optimized by Swarm Intelligence (SI), i.e. Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO) and Bee Colony Algorithm (BCA).

The algorithms of PSO, ACO and BCA are improved or modified in order to be applied to maintenance scheduling optimization.

1.4   List of Scientific Articles

➢ **Zhang, Z**., Wang, Y. and Wang, K., (2013). Fault Diagnosis and Prognosis using Wavelet Packet Decomposition, Fourier Transform and Artificial Neural Network. *Journal of Intelligent Manufacturing*, vol. 24 (6), pp. 1213-1227 (doi: 10.1007/s10845-012-0657-2).

➢ **Zhang, Z.,** Wang, Y. and Wang, K., (2013). Intelligent Fault Diagnosis and Prognosis Approach for Rotating Machinery integrating Wavelet Transform, Principal Component Analysis, and Artificial Neural Networks. *International Journal of Advanced Manufacturing Technology*, vol. 68 (1-4), pp. 763-773 (doi: 10.1007/s00170-013-4797-0).

Chapter 1: Introduction

- **Zhang, Z**., and Wang, K., (2014). Wind turbine fault detection based on SCADA data analysis using ANN. *Advances in Manufacturing*, 2(1), pp. 70-78 (doi: 10.1007/s40436-014-0061-6).
- Liu, Y., **Zhang, Z.** and Liu, Z., (2011). Customized Configuration for Hierarchical Products: Component Clustering and Optimization with PSO. The *International Journal of Advanced Manufacturing Technology*, 57. pp. 9-12.
- **Zhang, Z.** and Wang, K., (2013). Wind Turbine Fault Detection Based on SCADA Data Analysis Using ANN. *International Workshop of Advanced Manufacturing and Automation (IWAMA2013),* Nov. 27, pp. 323-335.
- Wang, K., Sharma, V. and **Zhang, Z.** (2013). SCADA Data Interpretation for Condition-based Monitoring of Wind Turbines. *International Workshop of Advanced Manufacturing and Automation (IWAMA2013),* Nov. 27, pp. 307-321.
- **Zhang, Z.** and Wang, K., (2012). Sensors Placement Optimization for Condition Monitoring. *Proceedings of International Workshop of Advanced Manufacturing and Automation (IWAMA2012),* June 20-21, pp. 69-76.
- **Zhang, Z.** and Wang, K., (2012). IFDPS-Intelligent Fault Diagnosis and Prognosis System for Condition-based Maintenance. *International Workshop of Advanced Manufacturing and Automation (IWAMA2012)*, June 20-21, pp.77-84.
- **Zhang, Z.** and Wang, K. (2010). Application of Improved Discrete Particle Swarm Optimization (IDPSO) in Generating Unit Maintenance Scheduling. *International Workshop of Advanced Manufacturing and Automation (IWAMA2010),* pp. 79-86.
- **Zhang, Z.** and Wang, K., (2012). Dynamic Condition-Based Maintenance Scheduling Using Bee Colony Algorithm. *Proceedings of International Asia Conference on Industrial Engineering and Management Innovation (IEMI2012)*, Oct. 27-29, pp.1607-1618.
- **Zhang, Z.** and Wang, K., (2011). Fault isolation using self-organizing map (SOM) ANNs. *IET International Conference of Wireless Mobile & Computing*, Nov. 14-16, pp. 425-431.
- Wang, K. and **Zhang, Z.,** (2012). *Application of Radio Frequency Identification (RFID) to Manufacturing*. SFI-Norman, SINTEF (ISBN 978-82-14-05388-3), pp. 1-24.
- Wang, K. **and Zhang, Z.** (2011). *Intelligent Fault Diagnosis and Prognosis System (IFDAPS) for Condition-based Maintenance*. Trondheim: SINTEF A17147 (ISBN 978-82-14-05057-8) pp. 1-21.
- Wang, K., Sharma, V. and **Zhang, Z.,** (2013). *SCADA Data Mining for wind turbine fault diagnosis and failure prognosis: Principles, Trends, Applications and Research Areas*, Trondheim: SINTEF (ISBN 978-82-14-05496-5), pp. 1-20.
- **Zhang, Z.** and Wang, K. (2012). Fault Diagnosis using Association Rules. In Wang, K. and Wang, Y. edit: *Data Mining for Zero-Defect Manufacturing*, pp. 53-75.
- Cusanno, R., **Zhang, Z**. Regattieri, A. and Wang, K., (2012). Apply Particle Swarm Optimization for Condition-based Maintenance Scheduling. In Wang, K. and Wang, Y. edit: *Data Mining for Zero-Defect Manufacturing*, pp. 117-131.
- Crucian, D., **Zhang, Z**. and Wang, K., (2012). Fault Diagnosis and Prognosis Using Self-organizing Map. In Wang, K. and Wang, Y. edit: *Data Mining for Zero-Defect Manufacturing*, pp. 101-115.

1.5 Outline of Thesis

The present thesis is structured in 9 Chapters. Chapter 2 describes the general structure of framework of IFDPS very briefly. Chapter 3 introduces some Computational Intelligence (CI) techniques, such as Artificial Neural Network (ANN), Association Rules (AR), Decision Trees (DT), Particle Swarm Optimization, Bee Colony Algorithm and Semi-Supervised Learning, which are applied in the phases of the framework of Intelligent Fault Diagnosis and Prognosis (IFDPS) for Condition-based Maintenance. Chapter 4 introduces the sensor strategies very briefly and proposes the data mining methods application in sensor placement optimization. Chapter 5 describes the techniques of signal processing and feature extraction. Chapter 6 presents the data mining technology applications in fault diagnosis. Chapter 7 proposes a fault prognosis method for fault prognosis based on fault indicator prediction by data mining techniques. Chapter 8 presents how to apply the Computational Intelligence methods in maintenance scheduling optimization both for CBM and Preventive maintenance. Finally the conclusions and future research directions are presented in Chapter 9. The problem of fault diagnosis and prognosis and CBM are the very hot research points recently. The present work is focus on the data mining methods for phases of IFDPS.

2 Framework of Intelligent Fault Diagnosis and Prognosis Systems (IFDPS) for CBM

2.1 Introduction

Any operation or process done on machine or its components to enhance the efficiency of machine before or after the breakdown is called maintenance [Deshpande & Modak, 2002]. It contains all technical and administrative activities, including management activities, which have the objective to sustain or recover equipment state and thus enable it to perform at a required level. The maintenance cost ranges between 15% (for manufacturing companies) and 40% (for iron and steel industry) of the cost of the manufactured goods [Mobley, 1990]. In the United States, this corresponds to more than 200 billion dollars every year. This shows the importance of maintenance from an economical point of view. Usually, three different types of maintenance are considered [Chu et al., 1998], that is: corrective maintenance, preventive maintenance and predictive maintenance. Corrective maintenance consists in repairing a system only after a breakdown occurred which contains all maintenance performed in order to repair a failure [Wilson, 2002]. Corrective maintenance is probably the most commonly used approach, but it is easy to see its limitations. When equipment fails, it often leads to downtime in production and therefore this approach is often expensive. Preventive maintenance consists in maintaining the system periodically to prevent breakdown. Statistics of failures are used to define the period such as every 100 working hours, or every 10 000 km for a car. Because of uncertainty of products, the lifecycle can vary much even if they are products in the same class. The periodical maintenance may carry out far before or after the time of the product become failure. Therefore, the predictive maintenance (also called Condition-based Maintenance) became a much better alternative maintenance strategy, which consists in starting a maintenance operation only when required by the state of the system, i.e. when a potential failure is detected. With the predictive maintenance strategy, the maintenance action can be done just before the product become failure, and thus it can prolong the product life without breakdown.

Condition-based Maintenance (CBM) is the use of machinery run-time data to determine the machinery condition and hence its current fault/failure condition, which can be used to schedule, required repair and maintenance prior to breakdown [Vachtsevanos et al., 2006]. To support CBM policy, a framework called Intelligent Fault Diagnosis and Prognosis System (IFDPS) is established in KDL for manufacturing systems and processes. This Chapter mainly introduces the general idea of IFDPS and its functions very briefly.

2.2 Objectives and Benefits

The main objective of IFDPS is to establish a framework to show how to use the signals, databases, analysis tools and maintenance decision-making techniques for

reaching near zero-breakdown in sustainable manufacturing. It is a part of a big project called SFI Norman (NORMAN - Center for Research-based Innovation). The final aim of IFDPS is to reach zero-defect manufacturing in which the first step is to reach zero-breakdown manufacturing. Therefore, there are several benefits of the framework of IFDPS.

- It can monitor plant floor assets, link the production and maintenance operation system, acquire data, collect feedback from remote customer site, and integrate it into upper level enterprise applications, discovery and generate maintenance knowledge.
- It can monitor the degradation of manufacturing machine and process, and predict the condition (remaining useful life) of the equipment.
- It can make predictive maintenance decision to prevent occurrence and development of failures effectively, ensure the safety of equipment and personnel, and reduce economic lost caused by failure.
- It can use fault diagnosis, performance assessment of level of degrading, fault prognosis models to reach zero-breakdown performance and further to reach zero-defect manufacturing, and improve the productivity of a company.

2.3 Structure of IFDPS

Fig. 2.1 shows the general structure of IFDPS which presents from the machine degrading, sensors, signal processing, fault diagnosis and prognosis, and maintenance scheduling optimization. The main tasks performed by IFDPS are the following:

- Continuous collection of data from different sensors mounted on the machine includes the information of machine and environment.
- Continuous processing the data collected from sensors in order to get useful information to evaluate the off-line and on-line health condition of the machine and also to detect if some symptoms of degradation or anomalies are present or could become present.
- According to the useful information mentioned above, the condition or the fault can be identified. If there are any degradation become unaccepted, the system can tell staff which components or machines and when should be maintained or repaired.
- According to the condition of the component or machine, the remaining useful life can be predicted.
- According to the condition of equipment and predicted remaining useful life, the maintenance action plan can be scheduled by some intelligent computational optimization algorithm.

The techniques of subtasks are presented in the following sections.

Chapter 2: Framework of Intelligent Fault Diagnosis and Prognosis Systems (IFDPS) for CBM
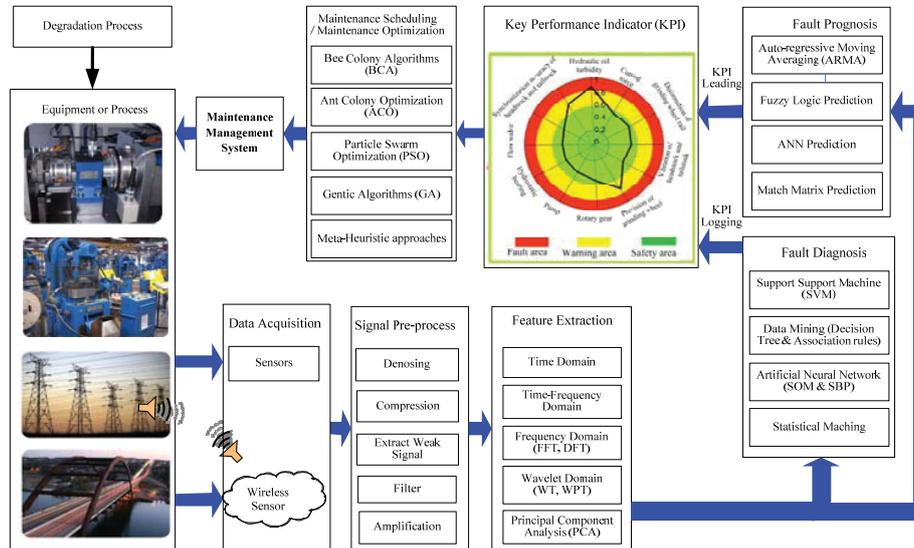
Fig. 2.1 Framework of IFDPS

2.3.1   Data Acquisition

Data acquisition is first phase of the IFDPS and is a basis of fault diagnosis and prognosis and hence is foundation of intelligent Condition-based Maintenance scheduling. The tasks of this phase are selecting a suitable sensors and optimal sensor strategy. Sensors and sensing strategies constitute the foundational basis for fault diagnosis and prognosis. Strategic issues arising with sensor suites employed to collect data that eventually will lead to online realization of diagnostic and prognostic algorithms are associated with the type, number, and location of sensors; their size, weight, cost, dynamic range, and other characteristic properties; whether they are of the wired or wireless variety; etc. [Vachtsevanos et al., 2006]. Data collected by transducing devices rarely are useful in their raw form. Such data must be processed appropriately so that useful information may be extracted that is a reduced version of the original data but preserves as much as possible those characteristic features or fault indicators that are indicative of the fault events we are seeking to detect, isolate, and predict the time evolution of. Thus such data must be preprocessed, that is, filtered, compressed, correlated, etc., in order to remove artifacts and reduce noise levels and the volume of data to be processed subsequently. Furthermore, the sensor providing the data must be validated; that is, the sensors themselves are not subjected to fault conditions. Once the preprocessing module confirms that the sensor data are "clean" and formatted appropriately, features or signatures of normal or faulty conditions must be extracted. This is the most significant step in the IFDPS architecture whose output will set the stage for accurate and timely diagnosis of fault modes. The extracted-feature vector will serve as one of the essential inputs to fault diagnostic algorithms.

Following will introduce the two aspects: sensor category and placement optimization.

2.3.1.1   Classification of Sensors

There are many methods can be used to classify sensors such as measurends, detection mass used in sensors, materials and applications [White, 1987]. However, for fault diagnosis and prognosis, we only use the sensors can measure physical measurends. For condition monitoring, it is generally agreed that two classes of sensors are making significant inroads into system monitoring for fault diagnosis and prognosis. The first one refers to classic or traditional transducers aimed at monitoring mechanical, structural, performance and operational and electrical/electronic properties that relate to failure mechanisms of mechanical, structural, and electrical systems. In this category, there are many sensors that measure fluid and thermodynamic, thermal, and mechanical properties of a variety of systems or processes—gas turbines, ground vehicles, pumps, aerospace systems, etc. The second important category refers to sensor systems that are placed almost exclusively to interrogate and track system properties that are related directly to their failure mechanisms. The most useful sensors in fault diagnosis and prognosis are first category which is shown in Fig. 1.3. When the sensors need to be selected, many aspects such as position, accuracy, ease of fitting and cost, need to be considered. After the sensors are chosen, the most important this is where to install the sensors.

2.3.1.2   Sensor Placement Optimization

Researches of sensor placement become very important issues for obtaining as much as possible information of machines or components using as few as possible sensors considering efficiency, effectiveness and economic issues. Traditionally, the sensor are placed to meet control and performance monitoring objectives [Al-Shehabi & Newman, 2001; Chen & Li, 2002; Faulds & King, 2000; Giraud & Jouvencel, 1995]. It is instructive to take advantage of such sensors in a fault diagnosis monitoring scheme because they can provide useful information relating to fault behaviors of critical system variables. More recently, research on sensor placement has focused on two different levels: the component level and the system level. At the component level, attempts have been reported regarding placement at the component's range, for example, a bearing or an object in three dimensional views [Faulds & King, 2000; Naimimohasses et al., 1995]. For complex, large-scale systems consisting of multiple components/subsystems, a fault may propagate through several components. With a large number of possible sensor locations, selection of an optimal location, as well as the number and types of sensors, poses a challenging problem that must be addressed at the system level.

IFDPS optimize the distribution of sensor placement in both component level and system using some Swarm Intelligence (SI) such as Particle Swarm Optimization (PSO) and Bee Colony Algorithm (BCA). For component level, the structure is analyzed using Finite Element Method (FEM) which information can be used to optimize the sensor placement using SI. For system level, the information transmit

flaw is analyzed which information can also be used to optimize the sensor placement.

2.3.2 Signal Preprocessing and Feature Extraction

Generally, there are two steps to deal with the signals from sensors. The one is signal preprocessing which is intended to enhance the signal characteristics that eventually may facilitate the efficient extraction of useful information that is the indictors of the condition of monitoring component or subsystem. The tools in this category include filtering, amplification, data compression, data validation, and de-noising which generally aim at improving the signal-to-noise ratio. And the other is extracting features from preprocessed signals that are characteristic of an incipient failure or fault. Generally, the features can be extracted from three domains: time domain, frequency domain and time-frequency domain. All possible signal preprocessing and feature extraction methods are shown in Table 2.1 and which features could be selected depend on the real machines or system. All these kinds of methods are selectable in IFDPS and which methods are applied can be decided by real machine or system analysis. What's more, in order to express the enough information to express the condition, the methods in the table can be combined together to be indicators of the condition.

Table 2.1 The Methods of Signal Pre-process and Signal Process.

Signal Preprocessing	Signal process		
	Time Domain	Frequency Domain	Wavelet Domain
Filter, Amplification, Signal Conditioning, Extracting Weak Signals, De-noising Vibration Signal Compression and Time Synchronous Averaging (TSA)	Mean, RMS, Shape factor, Skewness, Kurtosis, Crest factor, Entropy Error, Entropy estimation, Histogram Lower and Histogram upper	Continues Fourier Transform (CFT), Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT), Wigner-ville Distribution (WVD) and Short Time Fourier Transformation (STFT)	Wavelet Transform (WT) and Wavelet Packet (WP)

2.3.3 Fault Diagnosis and Identification

Fault diagnosis strategies have been developed in recent years and have found extensive utility in a variety of application domains. The enabling technologies typically fall into two major categories: model based and data-driven, as shown in Fig. 2.2. Model-based techniques rely on an accurate dynamic model of the system and are capable of detecting even unanticipated faults. They take advantage of the actual system and model outputs to generate a ''discrepancy'' or residual, as it is known, between the two outputs that is indicative of a potential fault condition. On

the other hand, data-driven techniques often address only anticipated fault conditions, where a fault ''model'' now is a construct or a collection of constructs such as neural networks, expert systems, etc. that must be trained first with known prototype fault patterns (data) and then employed online to detect and determine the faulty component's identity.

```
    ┌──────────┐                      ┌──────────┐
    │ Sensors  │                      │ Sensors  │
    │ Readings │                      │ Readings │
    └────┬─────┘                      └────┬─────┘
         ▼                                 ▼
┌───────┐ ┌──────────┐         ┌──────────┐ ┌──────────┐
│ Model ├▶│ Residuals │        │Knowledge ├▶│Inference │
└───────┘ │    or     │        │& training│ │          │
          │innovations│        └──────────┘ └────┬─────┘
          └────┬──────┘                          ▼
               ▼                            ┌──────────┐
          ┌──────────┐                      │ Decision │
          │ Decision │                      │  logic   │
          │  logic   │                      └──────────┘
          └──────────┘
        Model-based                      Data-driven
     (Unanticipated faults          (Anticipated faults only
       Model required)                  No Model needed)
```
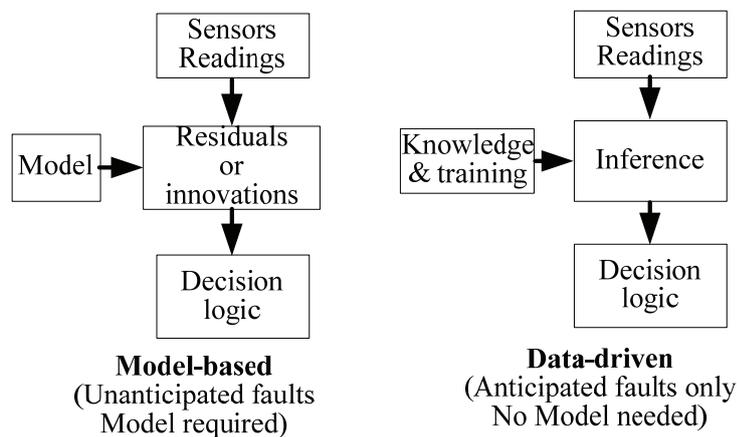
Fig. 2.2 Model-based and Data-driven Fault Diagnosis Techniques

IFDPS focus on the data-driven techniques and hybrid techniques. If the historical data can be obtained easily, the data-driven is very good to identify the fault and evaluate the condition. When only part of historical can be obtained, the hybrid techniques which combine the data-driven techniques and model-based techniques can be used to evaluate the condition of machine effectively. The semi-supervised learning method also can be used to evaluate condition and identify fault when only part of historical data is available and it is very effective. All these techniques are selectable according to the real manufacturing system analysis.

2.3.4   Fault Prognosis and Remaining Useful Life Evaluation

Generally speaking, current prognostic approaches can be classified into three basic groups: physical model prognostics, stochastic model prognostics and data-driven model prognostics.

Since physical models usually are based on a physical mechanism (failure mechanism) or process (failure process), they are valid for all problems where the process/mechanism leads to a failure. Sometimes, they are restricted to specific types of components. Typical for physical models is that the input parameters have a clear meaning and represent real (and often measurable) quantities or natural, physical or material constants. Thus, the models provide a clear understanding about the model input and the output, resulting in a so-called white-box model. Therefore, physical models are appealing for those who wish to get better understanding of the mechanisms and processes leading to failure [Loucks et al., 2005]. Physical models are in particular useful for design improvement.

Chapter 2: Framework of Intelligent Fault Diagnosis and Prognosis Systems (IFDPS) for CBM

In a first step, a physical model must be established if a good model is not available. This can be a challenging work and requires good knowledge about the problem that is modelled. However, once a good model is available, it can be applied to all comparable problems where good estimates or measurements of the model input parameters are available. Since the processes in real world may be quite complicated and may be affected by many mechanisms and effects, one usually has not the possibility to take all of them into account. Thus, a physical model may be restricted to include the main mechanisms and main effects only.

Physical models are often empirical, which means they are based on observation or experiments. Physical models can basically be used for all kinds of predictions, both long-term and short-term, depending on what they are designed for.

Most stochastic can be applied for many different problems. An advantage of stochastic models applied to lifetime models are of general nature and prediction is that both an estimate of the mean lifetime and various estimates of uncertainty can be established, such as variance of the lifetime, confidence intervals for parameters and predictions, etc. Parameter estimation in stochastic modelling is based on the observation of the model output. Thus, observations of the model output, such as observations of lifetime or degradation, are usually collected as basis for parameter estimation. When possible, one should fit different stochastic models to the data and choose the model that gives the best prediction. Many techniques exist to choose the best model and to check the goodness of fit (e.g. p-value, confidence intervals, comparison of maximum likelihood values and various graphical methods such as probability plots). As alternative to data collection, expert judgement can be used for parameter estimation. There exist different techniques for expert judgement, e.g. [Cooke, 1992]. Stochastic models can basically be used for both short-term and long-term predictions. However, for lifetime prediction, they are mostly used to make medium and long-term predictions. Furthermore, they are often used in system modelling or as input in other models (such as maintenance and optimization models) where the main interest is in long- term averages (such as failure rates). They can also successfully be applied for comparing and explaining the lifetime influence of different designs or other factors either by looking on the results from different samples or by incorporating explanatory variables in the model.

However, in the absence of a reliable and accurate system model, and statistical data, another approach to determine the remaining useful life is to monitor the trajectory of a developing fault and predict the amount of time until the developing fault reaches a predetermined level requiring action, which is the so called data-driven prognostic method. Data-driven techniques utilize monitored operational data related to system health. They can be beneficial when understanding of first principles of system operation is not straightforward or when the system is so complex that developing an accurate alternative model is prohibitively expensive. An added value of data-driven techniques is their ability to transform high-dimensional noisy data into lower dimensional information useful for decision-making [Dragomir et al., 2007]. Furthermore, recent advances in sensor technology and refined simulation capabilities enable us to continuously monitor the health of operating components and manage the related large amount of reference data.

Chapter 2: Framework of Intelligent Fault Diagnosis and Prognosis Systems (IFDPS) for CBM

Many data-driven models can be classified as black-box models because the relation of input and output variables and the model parameters is unclear in such types of models. Parameter estimation in black-box models is often based on learning and training. Thus, the models require data, and often data covering a time period where a failure was observed, in order to make a prediction of the lifetime. Learning can be based on data from a situation identified as normal. Then, all situations that are different from the normal situation may be defined as abnormal and (potentially) erroneous. Such an approach is appealing for diagnostic applications, because the observation of failures is not required. However, this approach is not sufficient for making predictions of the remaining lifetime.

Data-driven models are mostly suitable for making short-term predictions when the component reaches the end of life and when a potential failure becomes apparent in monitoring data. Since there are many models and methods in the field of AI, that in addition often are quite different, it is difficult to make general statements about models properties and the ways of parameter estimation. Many models can be considered as black-box models. Some others however, as for example expert systems, are white-box models where the internal model logic is based on expert knowledge.

IFDPS evaluate the remaining useful life by data-driven techniques because of physical or mathematical model absence. Traditional prognostic techniques are to find the relationship between the remaining useful life and time of the machine or component has been used. IFDPS try to find the relations between the remaining useful life and the condition of machine or component. Fig. 2.3 shows an example of this relationship. When the condition is identified, the remaining useful life can be predicted with a standard deviation.
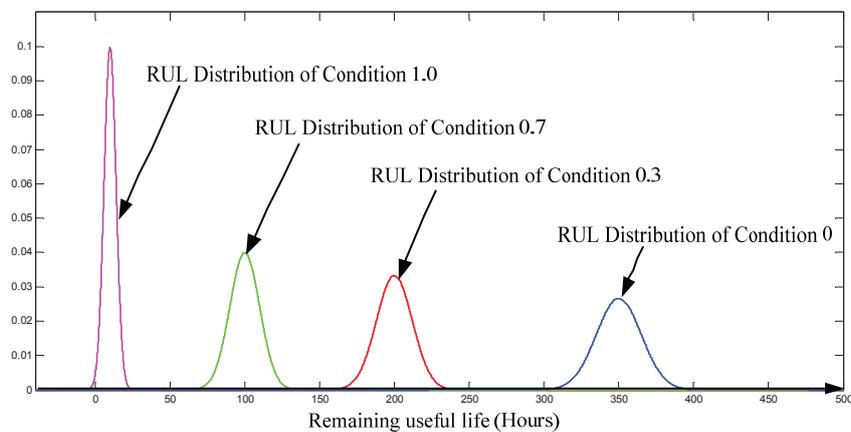
Fig. 2.3 Remaining Useful Life Distribution for Each Condition

After the condition of the component is determined, the remaining useful life can be evaluated according to the condition. Most current RUL estimation methods are based on the event data or condition monitoring data which want to find the relationship between RUL and time the component used or RUL and feature values

[Lee et al., 2006; Si et al., 2011]. The method of Fig. 2.3 tries to find the relationship between the RUL and the condition of a component that is evaluating RUL by the condition and RUL distribution for each condition. The distributions of RUL are obtained by the statistical methods. For example, if the condition of a component is 0, the remaining useful is 350h with a certain standard deviation. When the condition is 1.0, the RUL is much closed to 0 which means the component has to be maintained or repaired. From Fig. 2.3 the RUL distribution become narrow that means the RUL evaluation is more accuracy when the condition closed to failure. Therefore the confidence value of RUL increases with the condition deterioration.

IFDPS also propose another method to predict the RUL by establishing ANN model for machines in normal condition and set thresholds in several different levels. This method has applied in real industries such as wind power industry which is described in Chapter 7.

2.3.5 Maintenance Scheduling Optimization

The functions of the maintenance are finding out fault status of maintenance object and maintenance effect, and selecting right maintenance policy to achieve expected maintenance effect. The purpose of it is making maintenance decision based on current information to prevent occurrence and development of failure effectively, ensure the security of equipment and personnel, and reduce economic lost caused by failure. Maintenance scheduling optimization is a kind of NP problem and the SI algorithms could be a very good technique to solve this kind of problem. IFDPS apply Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO) and Bee Colony Algorithm (BCA) and try to find the optimal dynamic predictive maintenance scheduling. All these kind methods are selectable in IFDPS to solve maintenance scheduling optimization problems.

2.4 Summary

IFDPS (Intelligent Fault Diagnosis and Prognosis System) for Condition-based Maintenance is developed to monitor the manufacturing system and process, and to classify and predict faults and states, and to evaluate remaining useful life. Based on this system, the suitable maintenance actions can be made before any failure happens to ensure the security of equipment and personnel, and reduce economic lost caused by failure. In this framework, the suitable sensors should be selected to monitor the manufacturing system firstly, and then the collected data from the sensors is processed. For signal processing, the parameters of time domain, frequency domain and time-frequency domain can be used to process the signals and extract features as indicators of machines' condition. The condition of machines can be identified and predicted based on the extracted features from the real time signals. The remaining useful life can be evaluated based on the condition of machine and finally the maintenance decision can be made using some Swarm Intelligence algorithms.

Chapter 2: Framework of Intelligent Fault Diagnosis and Prognosis Systems (IFDPS) for CBM

3 Data Mining Techniques for IFDPS

3.1 Introduction

There are many aspects can be researched for the Framework of IFDPS. The volumes of data from sensors and processing procedure become tremendous filling the computers and networks. Sometimes, the data is too huge and too complicated to analyze effectively and thus how to get the useful information from these data becomes very significant point. This PhD work mainly focus on the application of Data Mining (DM) techniques in all processes of IFDPS such as sensor placement optimization, fault diagnosis, fault prognosis and maintenance scheduling optimization. There are already some researches in these areas but most of these researches focus on one process. DM technology has recently become a hot topic for decision-makers because it provides valuable, hidden business and scientific "Intelligence" from a large amount of historical data. It is a kind of methods to extract information and knowledge from recorded data. This Chapter describes some DM techniques used in the PhD work.

Data mining can be defined as the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner [Hand et al., 2001]. It is the entire process of applying computer-based methodology, including new techniques for knowledge discovery, from data. It draws ideas and resources from multiple disciplines, including machine learning, statistics, database research, high performance computing and commerce. This explains the dynamic, multifaceted and rapidly evolving nature of the data mining discipline. Generally, there are two main goals of data mining: prediction and description. Prediction involves using some variables or fields in the dataset to predict unknown or future values of other variables of interest. Description focuses on finding patterns describing the data that can be interpreted by humans. Therefore, the data mining activities can be classified into two categories: predictive data mining which produces the model of the system described by the given dataset, and the descriptive data mining which produces new, nontrivial information based on the available dataset. The main tasks of DM techniques are [Kantardzic, 2003]:

- *Classification* – discovery of a predictive learning function that classifies a data item into one of several predefined classes.
- *Regression* – discovery of predictive learning function, which maps a data item to a real-value prediction variable.
- *Clustering* – a common descriptive task in which one seeks to identify a finite set of categories or clusters to describe the data.
- *Summarization* – an additional descriptive task that involves methods for finding a compact description for a set of data.
- *Dependency Modeling* – finding a local model that describes significant dependencies between variables or between the values of a feature in a data set or in a part of a data set.

- *Change and Deviation Detection* – discovering the most significant changes in the data set.

To carry out these tasks, many DM techniques are available so far and more techniques will appear in the future. This Chapter will introduce some DM techniques used in the IFDPS framework.

3.2   Artificial Neural Networks (ANN)

The pattern classification theory has become a key factor in fault diagnosis and prognosis. Some classification methods for equipment performance monitoring use the relationship between the type of fault and a set of patterns which is extract from the collected signals without establishing explicit models. Currently, ANN is one of the most popular methods in this domain. ANN is a model that emulates a biological neural network [Wang, 2005]. The origin of ANN can be traced back to a seminar paper by McCulloch and Pitts [McCulloch & Pitts, 1943] that demonstrated a collection of connected processors, loosely modeled on the organization of brain, could theoretically perform any logical or arithmetic operation. Then, the development of ANN techniques is very fast which is extensive to many categories containing Back-propagation (BP), Self-organization Mapping (SOM) and Radial Basis Function (RBF), etc. The application of artificial neural network models lies in the fact that they can be used to infer a function from observations. This is particularly useful in applications where the complexity of the data or task makes the design of such a function by hand impractical. This attribution is very nontrivial in diagnostic problems. BP neural network is a main type of ANN used to solve fault diagnosis and prognosis problems.

ANN can deal with complex non-linear problem without sophisticated and specialized knowledge of the real systems. It is an effective classification techniques and low operational response times needed after training. The relationship between the condition of component and the features is not linear but non-linear. BP neural network does not need to know the exact form of analytical function on which the model should be built. This means neither the functional type nor the number and position of the parameters in the model-function need to know. It can deal with multi-input, multi-output, quantitative or qualitative, complex system with very good abilities of data fusion, self-adaptation and parallel processing. Therefore, it is very suitable to be selected as a method of fault diagnosis and prognosis. There are many papers dealing with the use of ANN and most of their contributions are ANN training efficiency and strategies for ANN itself. ANN in IFDPS will be used to detect and predict the condition of machines with other techniques such as wavelet analysis and Fourier transform. Two ANN techniques of Supervised Back-Propagation (SBP) and Self-Organizing Map are introduced in this subsection.

3.2.1   Supervised Learning ANNs

BP neural network which is the most widely used neural network model currently was proposed by Rumelhart and McCelland in 1986 [Rumelhart et al., 1986]. It is a

multilayer feed-forward network usually containing the input layer, hidden layer, and output layer (Fig. 3.1), which trained by an error back propagation algorithm. The biggest advantage of ANNs trained by back propagation is that there isn't need to know the exact form of analytical function on which model should be built. So it's not necessary have neither the function type not even the number and position of the parameters in the model function. Moreover, BP network can learn and store a lot of input-output model mapping without mathematical equations which describing this mapping. The learning method of BP is the steepest descent method which is adjusting the weights and thresholds of the network to minimize the sum of squared errors. The general procedure of BP network training can be summarized as follows [Wang, 2005].

1) Initialize the weights to small random vales (-1, 1);
2) Select a training vector pair (input and the corresponding desired output) from the training set and present the input vector to the inputs layer of the ANN;
3) Calculate the actual outputs (forward phase);
4) Adjust the weights to reduce the difference according to the error between actual output and target (backward phase);
5) Return to step 2 and repeat for each pattern p until the total error has reached an acceptable level;
6) Stop.

The backward phase and forward phase are described separately in this section.
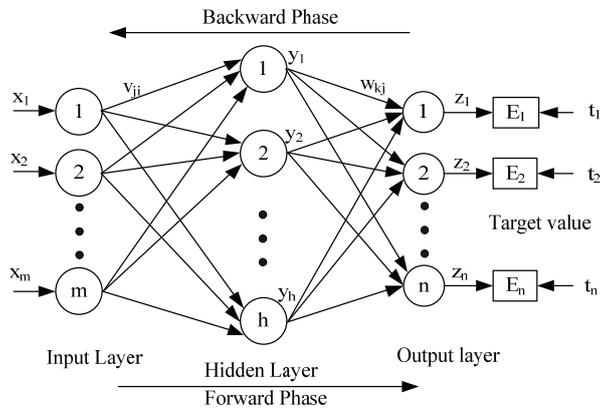
Fig. 3.1 A BP Neural Network with Single Hidden Layer

3.2.1.1 Forward Phase

Fig. 3.1 shows a BP network with signal hidden layer which is used to explain how the BP network works. In the figure, there are m nodes of input layer, h nodes of hidden layer and n nodes of output layer. Weight connections between input layer $i^{th}(i=1,2,\cdots,m)$ node and hidden layer $j^{th}(i=1,2,\cdots,h)$ node are denoted as v_{ji}, while Weight connections between hidden layer $j^{th}(i=1,2,\cdots,h)$ node and output

layer $k^{th} (i = 1, 2, \cdots, n)$ are denoted as w_{kj}. x_i represents the ith input value, y_j represents the output value of jth node of hidden layer, z_k represents the kth output, and t_k represents the target value of the kth output. The following terms are now defined:

$$H_j = \sum_{i=1}^{m} v_{ji} x_i \quad i = 1, 2 \cdots, m; j = 1, 2, \cdots h \tag{3.1}$$

$$I_k = \sum_{j=1}^{h} w_{kj} y_j \quad k = 1, 2 \cdots, n; j = 1, 2, \cdots h \tag{3.2}$$

where H_j is the combined or net input to hidden layer node j, while I_k is the net input to the node k of the output layer. The output for each node of hidden layer and output layer can be obtained as following respectively:

$$y_i = f(H) \quad j = 1, 2, \cdots, h \tag{3.3}$$

$$z_k = f(I_k) \tag{3.4}$$

where f is an activation function. Finally, the output of node k of output layer can be rewritten as:

$$z_k = f\left[\sum_{j=1}^{h} w_{kj} f(\sum_{i=1}^{m} v_{ji} x_i) \right] \tag{3.5}$$

3.2.1.2   Backward Phase

After calculating outputs of all nodes of output layer, the backward phase can be started to calculate according to Gradient Decent Learning [Wang, 2005]. The update rule for output layer node can be obtained:

$$\Delta w_{kj} = \eta \frac{\partial E}{\partial w_{kj}} = \eta \delta_k y_i = \eta (t_k - z_k) f'(I_k) y_i \tag{3.6}$$

where η is a constant often called the learning rate. Then the update rule for hidden layer nodes can be obtained as:

$$\Delta v_{ji} = \eta \delta_j x_i = \eta x_i f'(H_j) \sum_{k=1}^{n} \delta_k w_{kj} = \eta x_i f'(H_j) \sum_{k=1}^{n} [(t_k - z_k) f'(I_k) w_{kj}] \tag{3.7}$$

Then, all weights w_{kj} and v_{ji} can be updated according to Eq. (3.6) and Eq. (3.7) respectively as following:

$$w_{kj}^{new} = w_{kj}^{old} + \Delta w_{kj} = w_{kj}^{old} + \eta y_j (t_k - z_k) f'(I_k) \tag{3.8}$$

$$v_{ji}^{new} = v_{ji}^{old} + \Delta v_{ji} = v_{ji}^{old} + \eta x_i f'(H_j) \sum_{k=1}^{n} [(t_k - z_k) f'(I_k) w_{kj}] \tag{3.9}$$

This is the process of forward phase and backward phase in BP network training. Afterward, the whole training processes can be done according to the steps described above. The objective of ANN training is to obtain suitable weights to meet the inputs and the targets of training data. After the training of BP network, for each set of test data or query data, there is a set of output calculated by the final updated weights. BP network is a very useful model in real application especially when the real physical model and mathematic model are unavailable. It acts as a black box, which allows no physical interpretation of its internal parameters and functions. This propriety is very important to apply BP network in condition monitoring because most real mathematic models are unavailable. For a specific application in fault diagnosis and prognosis, after training by features extracted from processed historic data, the BP network can classify the fault and predict the states of the monitored components or machine units.

3.2.2   Self-Organizing Map (SOM)

Machine learning is an approach of using data to synthesize programs. In a case when the data are input/output pairs, it is called supervised learning as BP learning mentioned above. In a case, where there are no output values and the learning task is to gain some understanding of the process that generated the data, this type of learning is said to be unsupervised [Kankar et al., 2011]. The concept of SOM was introduced by Teuvi Kohonen in 1982 [Kohonen, 1982], and numerous versions, generalizations, accelerated learning scheme, and application of SOM have been developed since then. It is a type of Artificial Neural Network that is trained using unsupervised learning mode to produce a low-dimensional, discretized representation using the input datasets of the training samples. SOM is the closest of all Artificial Neural Networks architectures and learning schemes to the biological neuron network. Its network is composed by only one layer of neurons arranged in two-dimensional plane with a well-defined topology.

The most important unsupervised ANNs learning algorithm is the Kohonen competitive learning algorithm, and Fig. 3.2 shows a typically example of Kohonen map. The neurons on the output layer (also called competitive layer) can find the organization of relationship among input patterns. The output of each neuron isn't connected to all of the other neurons in the plane, but only to a small number that are topologically close to it. The network map shows the natural relationship between the patterns, that is, each input neuron is connected to every neuron on the competitive layer which is organized as two-dimensional grids. The network is presented by a set of training input patterns without target output patterns. At the beginning one of the patterns is chosen randomly, and then each neuron in the input layer of the SOM takes on the value of the corresponding entry in the input pattern. In the competitive learning, only one neuron in the output layer is selected after input occurs, regardless of how close the other neurons are from the best one. This is so-called "Winner takes it all" method.
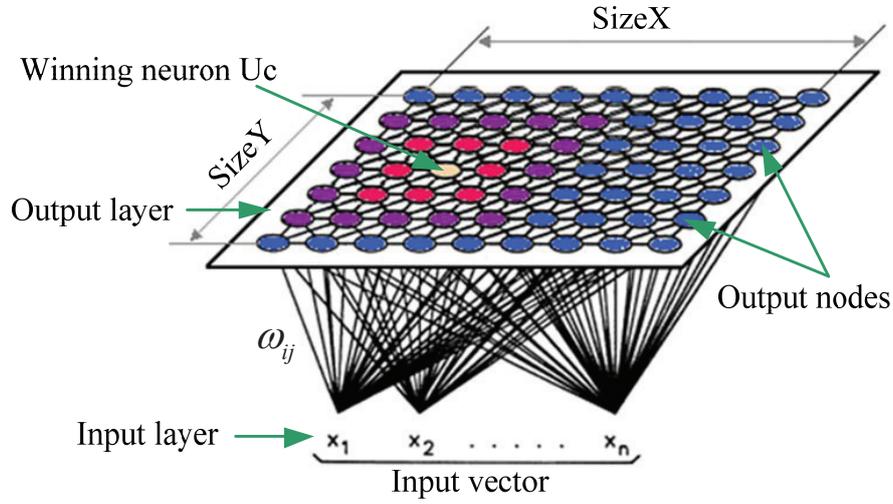
Fig. 3.2 Kohonen Model of SOM

In generally, the learning process of SOM network can be several steps [Wang, 2005].

1) Initializing the weight vector randomly ω_{ij}, the learning rate η and other relative training parameters.

2) For each input vector, the responses of all neurons in the output layer are calculated and the winning node U_c is selected. The winning node means its weight ω_{ij} best matches the input vector that is the Euclidean Distance is the smallest among all nodes.

3) After the winning node is selected, identifying the neighborhood around U_c, that is the set of competitive units close to the winning node. Fig. 3.3 shows the two examples of a neighborhood around wining node: the one is rectangular lattice and the other is the hexagonal lattice. The size of the neighborhood begins with a large enough size and then decreases with the number of iterations of the network.

4) Updating the weight vectors of node U_c and all nodes in the neighborhood around it by the following functions:

$$\omega_j(t+1) = \begin{cases} \omega_j(t)+\eta(t)\cdot f(d_c-d_i)\cdot(x-\omega_j(t)) & j \in H(t) \\ \omega_j(t) & otherwise \end{cases} \tag{3.10}$$

$$H(t) = H_0(1-\frac{t}{T}) \tag{3.11}$$

$$\eta(t) = (\eta_{max}-\eta_{min})\frac{T-t}{T-1}+\eta_{min} \tag{3.12}$$

where:

t : the current learning epoch;

x : input vector;

T : the total number of learning epoch;

H_0 : the initial neighborhood size;

$d_c - d_i$: the topological distance between the central neuron c and the current neuron i ;

f : topology dependent function.

$H(t)$: the actual neighborhood size in t^{th} epoch;

$\omega_j(t)$: the weigh vectors of U_c and its neighborhood in t^{th} epoch;

$\eta(t)$: the learning rate in t^{th} epoch;

5) Updating the learning rate using Eq. (3.12).
6) Reduce the neighborhood function using Eq. (3.11).
7) Loop from 2) to 6) until no noticeable changes of the feature map.

a) Rectangular Lattice   b) Haxagnoal Lattice Lattice
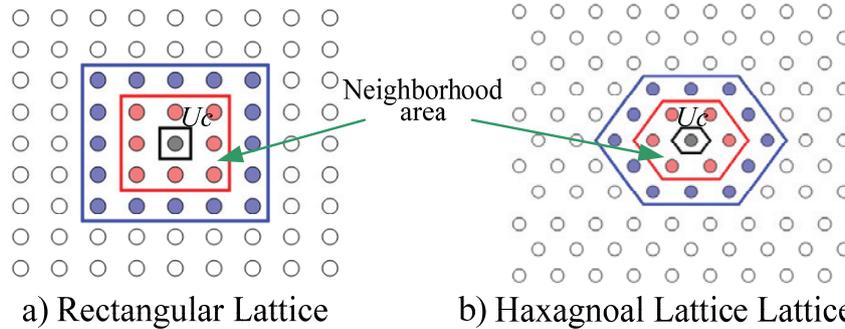
Fig. 3.3 Different Forms of the Neighborhood in SOM Network around U_c

SOM network has some advantages and some disadvantages. SOM permits to cluster data where there is no prior knowledge of the results or of the clustering. It is able to convert multi-dimensional data clusters into the form of a two-dimensional grid preserving the topological relationship of the data. It may be used where there is ample supply of "good normal" data containing some but little bad or usual data. That is engine monitoring or alarm monitoring. The SOM has very serious computational disadvantages, which affects the performance of large scale application running on parallel computers. In order to find which neuron is to be stimulated, the program has to check all of the neurons. This is a big restriction when large SOM network are to be trained. Sometimes grid size may need to be adjusting in response to number of clusters expected.

3.3   Semi-supervised Learning Methods (Manifold Regularization)

Semi-supervised learning (SSL) is halfway between supervised learning and unsupervised learning. In addition to unlabeled data, the algorithm is provided with

some supervision information – but not necessarily for all examples. In this case, the data set $X = (x_i)_{i \in [n]}$ can be divided into two parts: the points $X_l = (x_1, \cdots, x_l)$ for which labels $Y_l = (y_1, \cdots, y_l)$ are provided, and the points $X_u = (x_{l+1}, \cdots, x_{l+u})$, the labels of which are not known. SSL is very useful in real industry application especially when the history data are huge but only a small part of them are labeled. Semi-supervised learning methods fall into five categories: SSL with generative models, SSL with low density separation, graph-based methods, co-training methods, and self-training methods [Blum & Chawla, 2001; Yuan, 2012].

Recently graph-based methods with more applicable assumption have attracted considerable attention. Specifically, graph-based manifold regularization [Belkin et al., 2006] exploits the geometric structure of the marginal distribution of the CM data in the feature space. The incorporation of unlabeled data has demonstrated the potential for improved accuracy in time series prediction [Wei & Keogh, 2006], speech recognition [Jansen & Niyogi, 2005], calibration-effort reduction problem [Pan et al., 2001]. In this paper, we are looking for a general semi-supervised classification framework for fault detection. The manifold regularization based methods is a good option.

The Manifold regularization combines the ideas of spectral graph theory, manifold learning and kernel methods in a coherent and natural way to incorporate both the cluster assumption and the manifold assumption in Reproducing Kernel Hilbert Spaces (RKHS) regularization framework. In this section, we address the manifold regularization based SSL framework concisely following the description of [Belkin et al., 2006]. More details refer to [Sindhwani et al., 2005].

As mentioned above, consider a set of l labelled samples $\{(x_i, z_i)\}_{i=1}^{l}$ and a set of u unlabelled samples $\{x_j\}_{j=l+1}^{l+u}$, where $x_i, x_j \in R^d$ are the feature vectors collected from the input space $\mathcal{X}$ according to the marginal distribution $\mathcal{P}_{\mathcal{X}}$, and $z_i \in R$ is the classified label determined by the conditional distribution $\mathcal{P}(z \mid x)$. Manifold regularization introduces the regularized risk functional as an additional regularizer that serves to impose this assumption on the learning problem. The learning problem corresponds to solving the following optimization problem:

$$f^* = \arg\min_{f \in H_K} \frac{1}{l} \sum_{i=1}^{l} \mathcal{V}(x_i, z_i, f) + \gamma_A \|f\|_K^2 + \gamma_I \int_M \langle \nabla_M, \nabla_M \rangle \qquad (3.13)$$

which finds the optimal function f^* in the RKHS space H_K of functions $f : \mathcal{X} \to R$ corresponding to a Mercer kernel $K : \mathcal{X} \times \mathcal{X} \to R$, e.g. a linear or Gaussian kernel.

The first term of the regularized risk functional in Eq. (3.13) is defined on the loss function $\mathcal{V}$ measured the discrepancy between predicted value $f(x_i)$ and actual label z_i. The second term controls the complexity of f in terms of the RKHS norm, with γ_A being the RKHS norm regularization parameter. The third term is specific to manifold regularization and is based on the assumption that the support of $\mathcal{P}_{\mathcal{X}}$ forms a compact sub-manifold $\mathcal{M}$. It controls the complexity of f in the intrinsic geometry of $\mathcal{P}_{\mathcal{X}}$, with γ_I being the corresponding manifold regularization

parameter. The third term is approximated using the graph Laplacian defined on all $l+u$ labelled and unlabelled examples without using the label information. Hence the optimization problem can be reformulated as:

$$f^* = \arg\min_{f \in H_K} \frac{1}{l} \sum_{i=1}^{l} \mathcal{V}(x_i, z_i, f) + \gamma_A \|f\|_K^2 + \frac{\gamma_I}{(l+u)^2} \hat{f}^T L \hat{f} \qquad (3.14)$$

where $\hat{f} = (f(x_1), \ldots, f(x_{l+u}))'$ and L is the Laplacian matrix of a graph that models the underlying geometric structure.

From the extended Represener theorem [Belkin et al., 2006], the optimal function can be expressed in the following form:

$$f^*(x) = \sum_{i=1}^{l+u} \alpha_i K(x_i, x) \qquad (3.15)$$

When lose function $\mathcal{V}$ in Eq. (3.14) is adopt to be the squared loss function $\mathcal{V}(x_i, z_i, f) = (z_i - f(x))^2$, the Laplacian Regularized Least Squares (LapRLS) algorithm formulates the optimization problem:

$$f^* = \arg\min_{f \in H_K} \frac{1}{l} \sum_{i=1}^{l} (z_i - f(x))^2 + \gamma_A \|f\|_K^2 + \frac{\gamma_I}{(l+u)^2} \hat{f}^T L \hat{f} \qquad (3.16)$$

For the LapRLS, the optimal solution in Eq. (3.16) $\alpha^* = (\alpha_1^*, \ldots, \alpha_{l+u}^*)'$ is given by:

$$\alpha^* = (JK + \gamma_A lI + \frac{\gamma_I l}{(l+u)^2} LK)^{-1} Z \qquad (3.17)$$

where K is the $(l+u) \times (l+u)$ Gram matrix over all labelled and unlabelled samples, Z is an $(l+u)$-dimensional label vector given by $Z = (z_1, \ldots, z_l, 0, \ldots, 0)'$, and $J = (1, \ldots, 1, 0, \ldots, 0)$ is an $(l+u) \times (l+u)$ diagonal matrix with the first l diagonal entries being 1 and the rest being 0.

When lose function $\mathcal{V}$ in Eq. (3.14) is adopt to be the hinge loss function $\mathcal{V}(x_i, z_i, f) = (1 - z_i f(x))$, the algorithm formulates the Laplacian Support Vector Machines (LapSVM). Please refer to [Belkin et al., 2006] in details. The manifold regularization algorithms SSL can be summarized as following [Belkin et al., 2006]:

Input: l labelled examples $\{(x_i, y_i)\}_{i=1}^{l}$, u unlabelled examples $\{x_j\}_{j=l+1}^{l+u}$.

Output: Estimated function $f : \mathbb{R}^n \to \mathbb{R}$.

Step 1: Construct data adjacency graph with $(l+u)$ nodes using, for example, k-nearest neighbours or a graph kernel. Choose edge weights w_{ij}, for example, binary weights or heat kernel weights $w_{ij} = e^{\frac{-\|x_i - x_j\|^2}{4t}}$.

Step 2: Choose a kernel function $K(x, y)$. Compute the Gram matrix $K_{ij} = K(x_i, x_j)$.

Step 3: Compute graph Laplacian matrix: $L = D - W$ where D is a diagonal matrix given by $D_{ij} = \sum_{j=1}^{l+u} W_{ij}$.

Step 4: Choose γ_A and γ_I .

Step 5: Compute α^* using Eq. (3.17) for square loss (Laplacian RLS).

Step 6: Output function $f^*(x) = \sum_{i=1}^{l+u} \alpha_i^* K(x_i, x)$.

3.4   Association Rules

Association rules are one of the major techniques of data mining and it is perhaps the most common form of local-pattern discovery in unsupervised learning systems. Association rules mining retrieve all possible interesting associations (patterns, relationships or dependencies) in large sets of the data items which are stored in the form of transactions that can be generated by an external process, or extracted from relational database or data warehouse. Due to good scalability characteristics of the association rules algorithm and the ever-growing size of the accumulated data, association rules are an essential data mining tool for extracting useful knowledge from database. The most important thing in this case would be a rule that is interesting, that tells you something about your database that you have not already known and probably weren't able to explicitly articulate.

3.4.1   Market-basket Analysis

Market-basket analysis is one of the most intuitive applications of association rules which strive to analyze customer buying patterns by finding associations between items that customers put into their basket. For example, customers visit to a grocery store or an online purchase who may buy milk and bread together and even that some particular brands of milk are more often bought with certain brands of bread. That means for each customer, there is a typical transaction. Retails accumulate huge collections of transactions by recording business activities over time. Then, the transactions database is analyzed to find sets of items, or itemsets that appear together, such as bread and milk, in many transactions. These and other more knowledge can be used to maximize the profits by helping to design successful marketing campaigns and customizing store layout. A number of association rules can be generated from the market basket database as shown in Fig. 3.4.
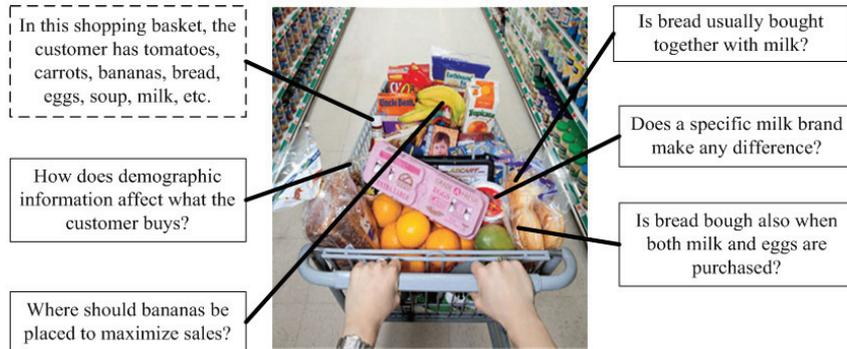
Fig. 3.4 Application of Association Rules in Market-basket Analysis

From the database of sales transactions, the important associations among items such that the presence of some items in a transaction will imply the presence of other items in the same transactions can be discovered. Let $I = \{i_1, i_2, \cdots, i_m\}$ be a set of literals which called items. Let D (database) be a set of transactions where each transaction T is a set of items such that $T \subseteq I$. Note that the quantities of the items bought in a transaction are not considered which means each item is a binary variable indicating whether an item was bought or not. Each transaction is associated with an identifier called a transaction identifier (TID). An example of the model for such transaction database is given in Table 3.1.

Table 3.1 A Model of a Simple Transaction Database

TID	Items
001	Apples, Celery, Diapers
002	Beer, Celery, Eggs
003	Apples, Beer, Celery, Eggs
004	Beer, Eggs

A transaction T is said to contain a set of items X if and only if $X \subseteq T$. A transaction rules implies the form $X \Rightarrow Y$ where $X \subseteq I$, $Y \subseteq I$, and $X \cap Y = \phi$. The rule $X \Rightarrow Y$ holds in the transaction set D with confidence c if $c\%$ of the transaction in D that contain X also contain Y. The rule $X \Rightarrow Y$ has support s in the transaction set D if $s\%$ of the transaction in D that contain $X \cup Y$. Here, two important concepts are defined bellowing:

—**Support**, which indicates the frequency (probability) of the entire rule with the respect to D. It is defined as a ratio of the number of transactions containing A and B to the total number of transactions (the probability of both A and B co-occurring in D):

$$support(A \Rightarrow B) = P(A \cup B) = \frac{\|\{T \in D | A \cup B \subseteq T\}\|}{\|D\|} \qquad (3.18)$$

—**Confidence,** which indicates the strength of implication in the rules. It is defined as ratio of the number of transactions containing A and B to the number of transaction s containing A (Conditional probability of B given A):

$$confidence(A \Rightarrow B) = P(A|B) = \frac{\|\{T \in D|A \cup B \subseteq T\}\|}{\|T \in D|A \subseteq T\|} \tag{3.19}$$

It is often desirable to pay attention to only those rules that may have a reasonably large support. Such rules with high confidence and strong support are referred to as strong rules. The task of mining association rules is essentially to discover strong association rules in large databases.

3.4.2 Mining Association Rules Steps

The following four steps are used to generate association rules:

1) Prepare input data in the transaction format;
2) Select items of interest, i.e. itemsets;
3) Calculate support counts to evaluate whether the selected itemsets are frequency which depend on whether the support s is above the predetermined minimum threshold;
4) Generate the association rules for the database that have confidence c above the predetermined minimum threshold using the large itemset.

The computational performance of an association-rule mining is determined by the second and the third step above. Then, the large itemsets are identified; the corresponding association rules can be derived in a straightforward manner. Efficient counting of large itemsets is thus the focus of most mining algorithm, and many efficient solutions have been designed to address previous criteria. Therefore, the following discussion concentrates on these two steps.

3.4.3 The Apriori Algorithm

The simplest way to calculate frequency itemsets is to consider all possible itemsets, compute their support, and check whether they are higher than the predetermined minimum threshold. The number of test of this method grows exponentially with the number of the items, and thus for large problems the computations would take an unacceptable long time. This reasoning resulted in the development of the Apriori algorithm.

The Apriori algorithm uses prior knowledge about an important property of frequent itemsets. The Apriori property of an itemset says that all nonempty subsets of a frequent itemset must also be frequent. In other words, if a given itemset is not frequent, any superset of this itemset will also be not frequent, because it cannot occur more frequently than the original itemset. The simplest superset of an itemset is the itemset with one more added item. The Apriori property is used to reduce the number of itemsets that must be searched to find frequent itemsets. The association-rule mining algorithm, the Apriori algorithm, performs the iterative search through itemsets, starting with 1-itemsets, through 2-itemsets, 3-itemsets, etc. In general, it finds and processes k-itemsets based on the

exploration of ($k-1$)-itemsets. Using the Apriori property, the Apriori algorithm is shown in Fig. 3.5.

Based on the Apriori property, in each iteration, k-itemsets that do not satisfy the minimum support are removed and only the remaining k-itemsets are used to generate itemsets for the next, $k+1$, iteration. This process substantially reduces the number of itemsets that have to be checked if they are frequent. The only unknown in implementing the Apriori algorithm is how to perform generation of C_k, which is a set of k-itemsets based on L_k-1. These k-itemsets are checked against the minimum support to derive L_k. The C_k is generated in two steps:

1) For each frequent itemset (FI) from L_k-1, find each item i that does not belong to FI, but belongs to some other frequent ($k-1$)-itemset in L_{k-1}. Add i to FI to create a k-itemset. Remove duplicate k-itemsets after all additions for all ($k-1$)-itemsets are finished.

2) If frequent ($k-1$)-itemsets from L_k-1 have ($k-2$)-items in common, then create a k-itemset by adding the two different items to ($k-2$) common items.
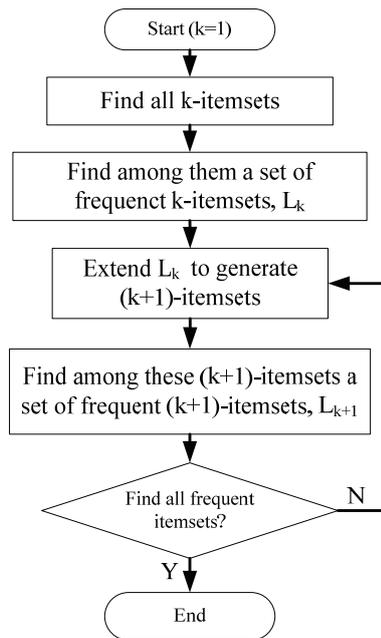
Fig. 3.5 The process of Aprioi Algorithm

3.4.4   Generating Association Rules from Frequent Itemset

The last step of the four that are used to generate single-dimensional association rules is to generate association rules from frequent itemsets. The association-rule mining algorithm requires the generation of strong rules, i.e., those that satisfy both minimum confidence and minimum support. The minimum support level is

43

guaranteed by using frequent itemsets, and thus we need only to generate the rules and prune those rules which do not satisfy the minimum confidence. The confidence can be defined based on the corresponding support values as follows:

$$confidence(A \Rightarrow B) = P(A|B) = \frac{support(A \cup B)}{support(A)} \qquad (3.20)$$

Where $support(A \cup B)$ is the number of transactions in D containing the itemset $A \cup B$, and $support(A)$ is the number of transactions in D containing the itemset A. Based on this formula, each frequent itemset FI is used to generate association rules in two steps:

a) Generate all nonempty subsets of items, Y, of FI;

b) For each Y, output the rules $Y \Rightarrow (FI - Y)$ if the value of $\dfrac{support\_count(FI)}{support\_count(Y)}$

   is larger than minimum confidence threshold.

To demonstrate the Apriori algorithm in action, we generate association rules from the transactional data given in Table 3.1. Fig. 3.6 shows the process of how to select the frequent itemsets. Finally, the association rules can be derived as bellowing from the generated frequency 3-itemset $\{B, C, E\}$ with support=50% which also have to satisfy the minimum confidence=60%.

B and C $\Rightarrow$ E with support = 50% and confidence = 2 / 2 = 100%

B and E $\Rightarrow$ C with support = 50% and confidence = 2 / 3 = 66.7%

C and E $\Rightarrow$ B with support = 50% and confidence = 2 / 2 = 100%

B $\Rightarrow$ C and E with support = 50% and confidence = 2 / 3 = 66.7%

C $\Rightarrow$ B and E with support = 50% and confidence = 2 / 3 = 66.7%

E $\Rightarrow$ B and C with support = 50% and confidence = 2 / 3 = 66.7%
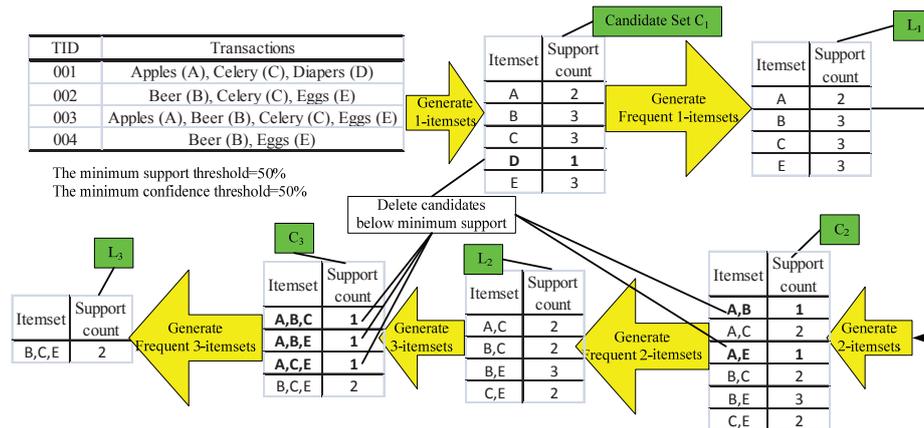
Fig. 3.6 Example Generation of Association Rules using Apriori Algorithm

3.4.5 Improving the Efficiency of the Apriori Algorithm

Since the amount of the processed data in mining frequent itemset tends to be huge, it is significant to devise efficient algorithms to mine such data. Our basic Apriori algorithm scans the database several times depending on the size of largest frequent itemsets. Several refinements have been proposed that focus on reducing the number of database scan, the number of candidate itemsets counted in each scan, or both.

3.4.5.1 Partition-based Apriori

Partition-based Apriori is an algorithm that requires only two scans of the transaction database. The database is divided into disjoint partitions, each small enough to fit into available memory. In the first scan, the algorithm reads each partition and computes local frequent itemsets on each partition. The frequent local itemsets may or may not be frequent in transaction database D, but any itemset that is potentially frequent in D must be frequent in at least one subset. Therefore, local frequent itemsets from all subsets become candidate itemsets for D. The collection of all local frequent itemsets is referred to as global itemsets with respect D. In the second scan, the algorithm counts the support of all global frequent itemsets toward the complete database D. Then comparing between the support values and minimum support threshold and find out which of the global candidate itemsetsare frequent itemsets. The process of partition-based Apriori algorithm to select frequent itemsets is shown in Fig. 3.7.
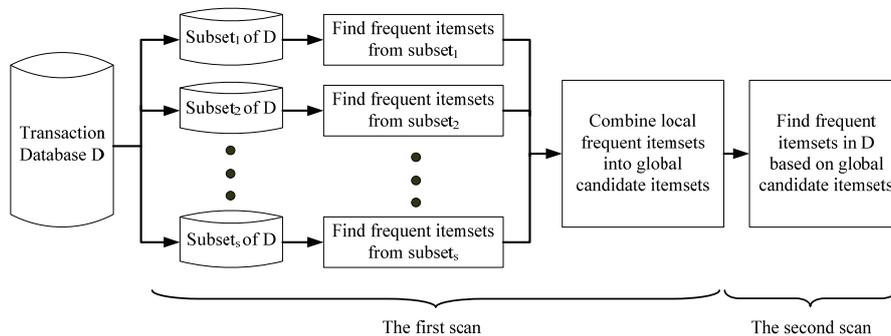
Fig. 3.7 Generation of Frequent Itemsets using Partition-based Apriori

3.4.5.2 Sampling

As the database size increases, sampling appears to be an attractive approach to data mining. Sampling generates association rules based on a sampled subset of transactions in D. In this case, a randomly selected subset S of D is used to search for the frequent itemsets. The generation of frequent itemsets from S is more efficient (faster), but some of the rules that would have been generated from D may be missing, and some rules generated from S may not be present in D, i.e., the "accuracy" of the rules may be lower. Usually the size of S is selected so that the transactions can fit into the main memory, and thus only one scan of the data is required (no paging). To reduce the possibility that we will miss some of the

frequent itemsets from D when generating frequent itemsets from S, we may use a lower support threshold for S as compared with the support threshold for D. This approach is especially valuable when the association rules are computed on a very frequent basis.

3.4.5.3 Hashing

Hashing is used to reduce the size of the candidate k-itemsets, i.e., itemsets generated from frequent itemsets from iteration $k-1$, C_k, for $k>1$. For instance, when scanning D to generate L1 from the candidate 1-itemsets in C_1, we can at the same time generate all 2-itemsets for each transaction, hash (map) them into different buckets of the hash table structure, and increase the corresponding bucket counts. A 2-itemset whose corresponding bucket count is below the support threshold cannot be frequent, and thus we can remove it from the candidate set C_2. In this way, we reduce the number of candidate 2-itemsets that must be examined to obtain L_2.

3.4.5.4 Transaction removal

Transaction removal removes transactions that do not contain frequent itemsets. In general, if a transaction does not contain any frequent k-itemsets, it cannot contain any frequent ($k+1$) itemsets, and thus it can be removed from the computation of any frequent t-itemsets, where $t>k$.

3.5 Swarm Intelligence

Swarm intelligence (SI), which is an Artificial Intelligence (AI) discipline, is concerned with the design of intelligent multi-agent systems by taking inspiration from the collective behavior of social insects such as ants, termites, bees, and wasps, as well as from other animal societies such as flocks of birds or schools of fish. Colonies of social insects have fascinated researchers for many years, and the mechanisms that govern their behavior remained unknown for a long time. Even though the single members of these colonies are non-sophisticated individuals, they are able to achieve complex tasks in cooperation. Coordinated colony behavior emerges from relatively simple actions or interactions between the colonies' individual members. Many aspects of the collaborative activities of social insects are self-organized and work without a central control. For example, leafcutter ants cut pieces from leaves, bring them back to their nest, and grow fungi used as food for their larvae. Weaver ant workers build chains with their bodies in order to cross gaps between two leaves. The edges of the two leaves are then pulled together, and successively connected by silk that is emitted by a mature larva held by a worker. Other examples include the capabilities of termites and wasps to build sophisticated nests, or the ability of bees and ants to orient themselves in their environment [Abraham et al., 2006]. The research scientists extract the term swarm intelligence according to these collaborative activities of social insects. The term swarm intelligence was first used by Beni in the context of cellular robotic systems

where simple agents organize themselves through nearest-neighbor interaction [Beni, 1988]. Meanwhile, the term swarm intelligence is used for a much broader research field [Bonabeau et al., 1999]. Swarm intelligence methods have been very successful in the area of optimization especially to find an optimal solution for NP problem, which is of great importance for industry and science.

The main algorithms of Swarm Intelligence currently are Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and Bee Colony Algorithm (BCA). ACO deals with artificial system that is inspired from the foraging behavior of real ants, which are used to solve discrete optimization problems [Dorigo et al., 1996]. The main idea is the indirect communication between the ants by means of chemical pheromone trials, which enables them to find short paths between their nest and food. PSO incorporates swarming behaviors observed in flocks of birds, schools of fish, or swarms of bees, and even human social behavior, from which the idea is emerged [Clerc & Kennedy, 2002; Kennedy & Eberhart, 2001; Parsopoulos & Vrahatis, 2004]. PSO is a population-based optimization tool, which could be implemented and applied easily to solve various functional optimization problems, or the problems that can be transformed to functional optimization problems. As an algorithm, the main strength of PSO is its fast convergence, which compares favorably with many global optimization algorithms like Genetic Algorithms (GA) [Goldberg, 1989], Simulated Annealing (SA) [Orosz & Jacobson, 2002; Triki et al., 2005] and other global optimization algorithms. For applying PSO successfully, one of the key issues is finding how to map the problem solution into the PSO particle, which directly affects its feasibility and performance. There are several advantages of Swarm Intelligence:

Flexibility - the swarm can quickly respond to internal perturbations and external challenges.

Adaptability - The swarm can adapt to a changing environment.

Robustness - even if one or more individuals in the swarm fail, the swarm can still complete its tasks.

Self-organization - Paths to solutions are emergent rather than predefined.

Decentralization - The swarm needs relatively little supervision or top-down control. In other words, there is no central control in the swarm.

Scalability - The control mechanisms used are not dependent on the number of agents in the swarm.

This section will mainly introduce the algorithms of ACO, PSO and BCA.

3.5.1   Ant Colony Optimization (ACO)

Ant colonies can accomplish complex tasks that far exceed the individual capabilities of a single ant [Dorigo & Stützle, 2004]. The ACO model is applied firstly to solve the Travelling Salesman Problem (TSP). The two main phases of the algorithm constitute the solution construction and the pheromone update. For TSP, m ants concurrently build a tour and select cities randomly at the beginning

of the tour construction. At each construction step, ant k decides which city to visit next according to a random proportional rule. The probability with which ant k, currently at city i, chooses to go to city j is:

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{ij}]^\alpha [\eta_{ij}]^\beta}, \quad if \quad j \in N_i^k \tag{3.21}$$

where τ_{ij} is the pheromone deposited on $arc(i,j)$, $\eta_{ij} = 1/d_{ij}$, which represents the visibility of city j towards city i which is inversely proportional to the distance d_{ij}, α and β are two parameters which determine the relative influence of the pheromone trail and the heuristic information, and N_i^k is the set of cities that ant k has not visited yet [Dorigo & Stützle, 2004].

The pheromone trails are updated after tours constructing by evaporating at a constant rate and accumulating with new deposits:

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k, \quad \forall (i,k) \in L \tag{3.22}$$

where $0 < \rho \leq 1$ is the pheromone evaporation rate and $\Delta\tau_{ij}^k$ is the amount of pheromone that ant k deposits on the arcs it has visited, defined as follows:

$$\Delta\tau_{ij}^k = \begin{cases} 1/C^k & \text{if arc(i, j) belongs to } T^k \\ 0 & otherwise \end{cases}$$

where C^k is the length of the tour T^k built by ant k. By using this rule, the probability increases that forthcoming ants will use this arc. A brief pseudo-code and the implementation steps of ACO can be written as following pseudo-code and Fig. 3.8.

Begin
    Initialization
While stopping criterion not satisfied **do**
    Deploy each ant in a starting city
      For each ant
        Repeat
          Calculate probability of remaining cities selected to be next city
          Choose next city according to probability using roulette wheel selection algorithm
        Until all cities are visited
          Update pheromone
      End for
    Update the best route (beat solution)
End while
    Record and output the beat route (solution)
End
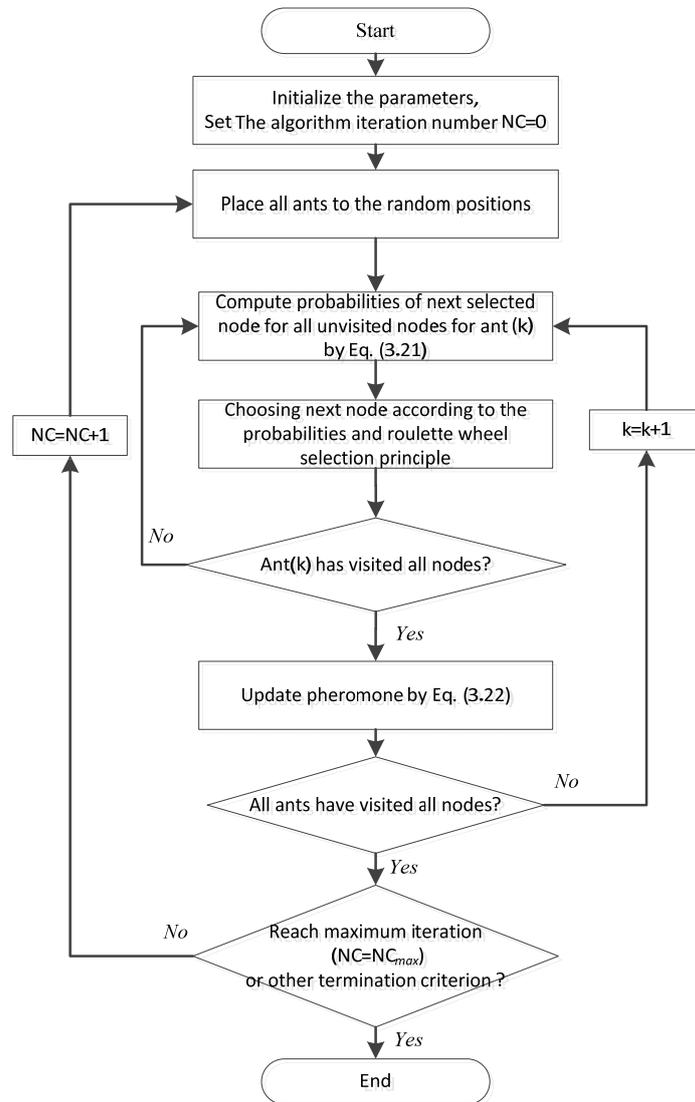
Fig. 3.8 The implementation steps of ACO

3.5.2   Particle Swarm Optimization

3.5.2.1   Biological Metaphor

We can imagine such a scenario that a group of birds were searching of food randomly. There is only one food in this region. All the birds don't know where the food is but know how far they away from the food. Then, what is the best strategy to find the food?

The easiest way is to search the region around of nearest bird from the food. And according to their own experience of flying to judge the position of food.

49

3.5.2.2   Basis Algorithm of PSO

The Particle Swarm Optimization (PSO) algorithm is a heuristic approach motivated by the observation of social behavior of composed organisms such as birds flocking (Fig. 3.9). A number of simple entities – the particles – are placed in the search space of some problem or function, and each evaluates the objective function at its current location. Each individual in the particle swarm is composed of D dimensional vectors, where D is the dimensionality of the search space.
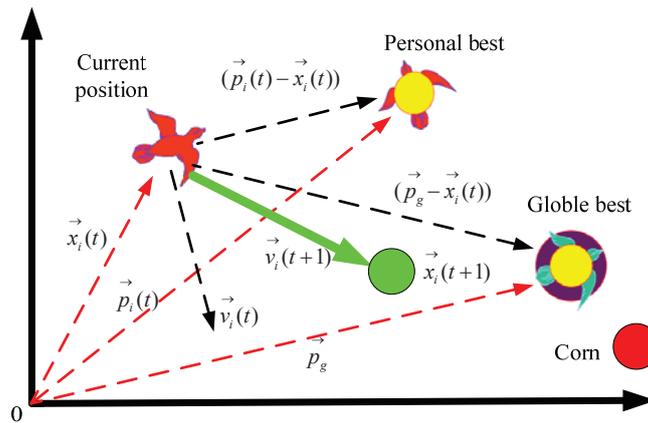
Fig. 3.9 Birds Flocking of PSO

The current position $\vec{x}_i$ can be considered as a set of coordinates describing a point in space. If the current position is better than any that has been found so far, then the coordinates are stored in the vector $\vec{p}_i$. The value of the best function result so far is stored in a variable that can be called $\vec{p}_g$. The objective, of course, is to keep finding better positions and updating $\vec{p}_i$ and $\vec{p}_g$. New points are chosen by adding $\vec{v}_i$ coordinates to $\vec{x}_i$, and the algorithm operates by adjusting $\vec{v}_i$, which can effectively be seen as a step size. The steps of implementing PSO were shown as follows:

Step 1: Initialize a population array of particles with random positions and velocities on D dimensions in the search space.

Step 2: Loop

Step 3: For each particle, evaluate the desired optimization fitness function in D variables.

Step 4: Compare particle's fitness evaluation with that of its $\vec{p}_i$. If current value is better than that of $\vec{p}_i$, then set $\vec{p}_i$ equal to the current coordinates.

Step 5: Identify the particle in the neighborhood with the best success so far, and assign it to the variable $\vec{p_g}$.

Step 6: Change the velocity and position of the particle according to the following equations:

$$\vec{v_i}(t+1) = \omega \cdot \vec{v_i}(t) + c_1 \cdot r_1(\vec{p_i} - \vec{v_i}(t)) + c_2 \cdot r_2(\vec{p_g} - \vec{v_i}(t)) \qquad (3.23)$$

$$\vec{x_i}(t+1) = \vec{x_i}(t) + \vec{v_i}(t+1) \qquad (3.24)$$

where ω is the inertia weighting; c_1 and c_2 are acceleration coefficients, positive constraint; r_1 and r_2 are the random numbers deferring uniform distribution on [0, 1]; i represents i^{th} iteration.

Step 7: If a criterion is met (usually a sufficiently good fitness or a maximum number of iterations), exit loop.

The flowchart of PSO can be seen as Fig. 3.10. In PSO, every particle remembers its own previous best value as well as the neighborhood best. therefore it has a more effective memory capability than the GA. PSO is also more efficient in maintaining the diversity of the swarm, since all the particles use some information related to the most successful particle in order to improve themselves, whereas in GA, the worse solutions at every generation are discarded and only the good ones are saved for next generation. Therefore in GA the population does the evolution around a set of best individuals in every generation. In addition, PSO is easier to implement and there are fewer parameters to adjust compared with GA [Valle et al., 2008].

3.5.2.3   The Parameters of PSO

The role of inertia weight ω in Eq. (3.23), is considered critical for the convergence behavior of PSO. The inertia weight is employed to control the impact of the previous history of velocities on the current one. Accordingly, the parameter ω regulates the trade-off between the global (wide-ranging) and local (nearby) exploration abilities of the swarm. A large inertia weight facilitates global exploration, i.e. searching new areas, while a small one tends to facilitate local exploration, i.e. fine-tuning the current search area. A suitable value for the inertia weight ω usually provides balance between global and local exploration abilities and consequently results in a reduction of the number of iterations required to locate the optimum solution. Initially, the inertia weight is set as a constant. However, some experiment results indicates that it is better to initially set the inertia to a large value, in order to promote global exploration of the search space, and gradually decrease it to get more refined solutions [Eberhart & Shi, 2000]. Thus, an initial value is set to maximum one ω_{max} (for example around 1.2) and gradually reducing towards the minimum one ω_{min} (for example around 0.6) can be considered as a good choice. A better method is to use some adaptive approaches (example: fuzzy controller), in which the parameters can be adaptively fine-tuned

according to the problems under consideration [Liu & Abraham, 2005; Shi & Eberhart, 2001].

```
┌─────────────────────────┐
│         Start           │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│ Initialize Parameters of PSO │
│      (c₁,c₂,w,x,v,N)      │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│ Evaluate the objective function │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│   Update pbest and gbest │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│ Update velocity and position of each │
│ particle with Eq. (3.21) and Eq. (3.22) │
└─────────────────────────┘
            │
            ▼
       Is termination
    condition satisfied ?  ── NO
            │ YES
            ▼
┌─────────────────────────┐
│          END            │
└─────────────────────────┘
```
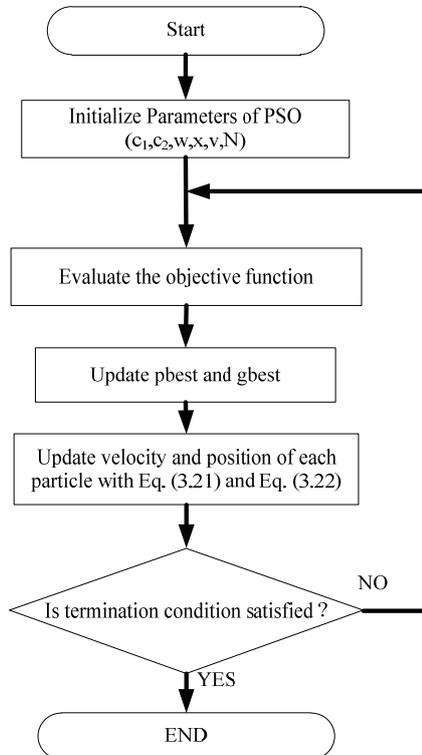
Fig. 3.10 The flowchart of PSO algorithm

The parameters c_1 and c_2, in Eq. (3.23), are not critical for the convergence of PSO. However, proper fine-tuning may result in faster convergence and alleviation of local minima. As default values, usually, $c_1 = c_2 = 2$ are used, but some experiment results indicate that $c_1 = c_2 = 1.49$ might provide even better results. From Eq.(3.23), it is better for local exploitation when $c_1 > c_2$, while it is better for global exploration when $c_1 < c_2$. Recent work reports that it might be even better to choose a larger cognitive parameter, c_1, than a social parameter, c_2, but with $c_1 + c_2 \leq 4$ [Clerc & Kennedy, 2002]. Therefore, the parameter c_1 can be changed from $c_{1\min}$ to $c_{1\max}$ and the parameter c_2 can be changed from $c_{2\max}$ to $c_{2\min}$ regularly in order to make the algorithm promote global exploration in the beginning and get more refined solutions (local exploitation) in the end.

3.5.2.4   Variants of PSO

There are many different variants of the PSO algorithm. Some of these variants have been proposed to incorporate either the capabilities of other evolutionary computation techniques, such as hybrid versions of PSO or the adaptation of PSO

parameters for a better performance (adaptive PSO). In other cases, the nature of the problem to be solved requires the PSO to work under complex environments as in the case of the multi-objective or constrained optimization problems or tracking dynamic systems. There are also some discrete variants of PSO and other variations to the original formulation that can be included to improve its performance. This section will present some of them.

A. Binary PSO

Kennedy and Eberhart proposed a discrete binary version of PSO for binary problems [Kennedy & Eberhart, 1997]. In their model a particle will decide on "yes" or "no", "true" or "false", "include" or "not to include" etc. also this binary values can be a representation of a real value in binary search space.

In the binary PSO, the particle's personal best and global best is updated as in continuous version. The major difference between binary PSO with continuous version is that velocities of the particles are rather defined in terms of probabilities that a bit will change to one. Using this definition a velocity must be restricted within the range [0, 1]. So a map is introduced to map all real valued numbers of velocity to the range [0, 1] [Kennedy & Eberhart, 1997]. The normalization function used here is a sigmoid function as:

$$Sig(v_{ij}(t)) = \frac{1}{1 + e^{-v_{ij}(t)}} \tag{3.25}$$

where $v_{ij}(t)$ means the jth component of vector $\vec{v}_i(t)$. The Eq. (3.23) is also used to update the velocity vector of the particle. And the new position of the particle is obtained using the following equation:

$$x_{ij}(t+1) = \begin{cases} 1 & r_{ij} < sig(v_{ij}(t+1)) \\ 0 & otherwise \end{cases} \tag{3.26}$$

where: r_{ij} is a uniform random number in the range [0, 1].

B. Hybrid PSO (FPSO)

A natural evolution of the particle swarm algorithm can be achieved by incorporating methods that have already been tested in other evolutionary computation techniques. Many authors have considered incorporating selection, mutation and crossover, as well as the differential evolution (DE), into the PSO algorithm. The main goal is to increase the diversity of the population by: 1) either preventing the particles to move too close to each other and collide [Blackwell & Bentley, 2002; Krink et al., 2002] or 2) to self-adapt parameters such as the constriction factor, acceleration constants [Miranda & Fonseca, 2002], or inertia weight [Lovbjerg & Krink, 2002]. As a result, hybrid versions of PSO have been created and tested in different applications. The most common ones include hybrid of genetic algorithm and PSO (GA-PSO), evolutionary PSO (EPSO) and differential evolution PSO (DEPSO and C-PSO). All these variants of PSO can be seen in paper [Valle et al., 2008] which described very detail.

There are also some other variants of PSO such as Fuzzy PSO [Shi & Eberhart, 2001; D. Tian & Li, 2009], Adaptive PSO [Valle et al., 2008], Gaussian PSO [Krohling, 2004, 2005; Secrest & Lamont, 2002], Dissipative PSO (DPSO) [Biskas et al., 2006; Xie et al., 2002], PSO With Passive Congregation (PSOPC) [He et al., 2004], Stretching PSO (SPSO) [Kannan et al., 2004; Parsopoulos & Vrahatis, 2002], Cooperative PSO (CPSO) [Baskar & Suganthan, 2004; Bergh & Engelbrecht, 2004; El-Abd & Kamel, 2006], and Comprehensive Learning PSO (CLPSO) [Liang et al., 2006]. Each variant of PSO mentioned above have improved its performance in one or more aspects. We can choose a suitable one when we need apply PSO or its variants to find optimal solution.

3.5.3 Bee Colony Algorithm

Bee Colony Algorithm (BCA) is based on the waggle dance which was discovered by the Austrian ethnologist and Nobel laureate Karl von Frisch in 1967 [Frisch, 1967]. In the recent years some people has used this knowledge to develop algorithms to solve real problems. There are several different algorithms inspired from the bee colony behaviors such as Artificial Bee Colony (ABC) [Karaboga & Basturk, 2007; Karaboga, 2005], Bees Algorithm (BA) [Pham et al., 2006], Honey Bee Colony Algorithm (HBCA) [Chong et al., 2006], and Bee Colony Optimization (BCO) [Teodorovic & Dell'Orco, 2005]. This section will introduce an algorithm called Bee Colony Algorithm (BCA) [Karaboga & Akay, 2009; Karaboga & Basturk, 2007].

3.5.3.1 Biological Metaphor

BCA is inspired from the behavior of bee colony during their forage and thus this biological behavior is introduced firstly before the BCA algorithm. A first explanation of the behaviour of the bees, was given by the biologist Karl von Frisch [Frisch, 1967]. The bees use a dance language inside the hive to communicate the location of the food sources. Sources which are located closer to the hive are solicited by a round dance. With this dance, the bees describe several circles with a changing orientation. Food sources with a further distance to the hive are communicated through a waggle dance which is the most important aspect for our purpose.

A waggle dance consists (Fig. 3.11) of one to 100 or more circuits, each of which has two phases: the waggle phase and the return phase. For this dance, the dancing bee starts to bounce with its abdomen. A worker bee's waggle dance involves running through a small figure-eight pattern: a waggle run (waggle phase) followed by a turn to the right to circle back to the starting point (return phase), another waggle run, followed by a turn and circle to the left, and so on in a regular alternation between right and left turns after waggle runs.

The meaning of the direction and duration of waggle runs is the direction and distance of the patch of flowers being advertised by the dancing bee. Flowers located directly in line with the sun are represented by waggle runs in an upward direction on the vertical combs, and any angle to the right or left of the sun is coded by a corresponding angle to the right or left of the upward direction. The distance

between hive and recruitment target is indicated by the duration of the waggle runs. The farther the target, the longer the waggle phase, with a rate of increase of about 75 milliseconds per 100 meters.
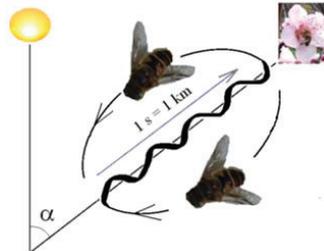
Fig. 3.11 The waggle dance

After unloading the collected food, a foraging bee returning to the beehive from a food source (employed bee) decides whether to abandon the food source or not. If the food source is abandoned, the bee observes the dances of other employed bees and follows one of the possible ways adverted for other bees as a follower bee or starts to search for an entirely new source as a scout bee. However, if the food source is not abandoned, the employed bee decides whether to dance for the source to recruit other bees or not and just keep on going to the same food source without adverting it. Fig. 3.12 shows the decision model of bees' behaviour.
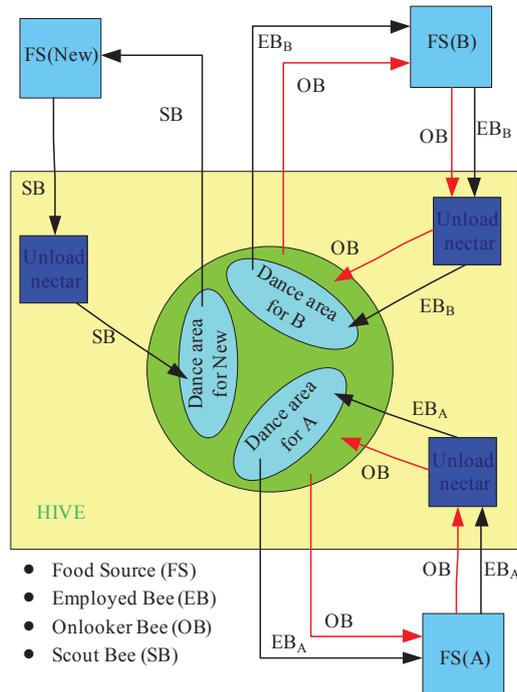
Fig. 3.12 The Behavior of the Bees

Some researchers have developed a model for foraging behaviour of a honeybee colony based on reaction–diffusion equations as Karaboga did. His model, that leads to the idea of collective intelligence of bee swarms consists of three essential components: food sources, employed foragers, and unemployed foragers, and defines two ways for the bee colony behaviour: recruitment to a food source and abandonment of a source [Karaboga & Akay, 2009]. The explanation of the main components of the model is:

- Food Sources: In order to select a food source, a forager bee evaluates several properties related with the food source such as its closeness to the hive, richness of the energy, taste of its nectar, and the easiness or difficulty of extracting this energy. To simplify, the quality of a food source can be represented by only one quantity although it depends on various parameters mentioned above.
- Employed Bees: An employed bee is employed at a specific food source which is currently exploiting, carrying information about this specific source and sharing it with other bees waiting in the hive. The information includes distance, direction and profitability of the food source.
- Unemployed Bees: A forager bee that looks for a food source to exploit is called unemployed. It can be either a scout who searches the environment randomly or an onlooker who tries to find a food source by means of the information given by the employed bee. The mean number of scouts is about 5–10%.

3.5.3.2 Algorithm of BCA

In BCA algorithm [Karaboga & Akay, 2009; Karaboga & Basturk, 2007], the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality of the associated solution. The number of the employed bees and the onlooker bees is equal to the number of solutions in the population. The process of the behavior of bees to search food can be described as following Fig. 3.12: the first phase is called employed bee phase. In this phase, every food source (FS) is visited by one employed bee (EB) who then take nectar to hive, and do the waggle dance in the dance area to express the quality of nectar. The second phase is onlooker bee (OB) phase. The onlooker bees will chose the food source to visit according to the waggle dance by the employed bees. The finally phase is scout bee (SB) phase. If the reaming food source is not good, the scout bees will be set out to find new food sources, take the nectar back to hive and dance in the dance area. The new food sources with the old ones will be combined together to be visited by onlooker bees and employed bees according to their qualities of nectar.

The algorithm of BCA can be described as following steps:

1) Initialize the positions of solutions $\vec{x}_i$, the colony size (NP), the maximum cycle number ($maxCycle$), the number of parameters (D), and the number of trials to improve a source ($limit$).
2) Evaluate the population using fitness function.

3) **Repeat** ($Cycle = 1$)

4) Produce new solutions $\vec{v}_i$ (food source positions) in the neighbourhood of $\vec{x}_i$ for the employed bees using Eq. (3.27) and evaluate these solutions using fitness function.

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \tag{3.27}$$

where

ϕ_{ij} : Random number between [-1, 1].

i : $\{1, 2, \cdots, C\}$ the ith food source.

j : the jth component of parameters.

k : $\{1, 2, \cdots, SN\}$ randomly chosen index of parameter (dimension) which is different from j.

5) Apply the greedy selection process for employed bees between $\vec{v}_i$ and $\vec{x}_i$.

6) Calculate the probability value p_i for the solutions $\vec{x}_i$ by means of their fitness values using Eq. (3.28).

$$p_i = fit_i / \sum_{n=1}^{SN} fit_n \tag{3.28}$$

7) Produce the new solutions $\vec{v}_i$ (new positions) for the onlookers from the solutions $\vec{x}_i$ using Eq. (3.27), which selected depending on p_i , and evaluate them.

8) Apply the greedy selection process for the onlooker bees between $\vec{v}_i$ and $\vec{x}_i$.

9) Determine the abandoned solution (source), if exists, replace it with a new randomly produced solution $\vec{x}_i$ for the scout using the following equation.

$$x_i^j = x_{i\min}^j + rand[0,1](x_{i\max}^j - x_{i\min}^j) \tag{3.29}$$

10) Memorize the best food source position (solution) achieved so far.

11) $Cycle = Cycle + 1$

12) **Exit** if $Cycle = maxCycle$ or other criterion is met.

In the process of BCA algorithm, step 5) and step 6) constitute the employed bee phase, step 7) and step 8) constitute the onlooker bee phase while step 9) is scout bee phase. The problem of dynamic CBM scheduling is a kind of *NP* problem and the BCA is a good method to find the optimal solution for this kind problem.

3.6   Summary

This Chapter introduced the basic concepts of Data Mining techniques and algorithms. ANN includes supervised learning and unsupervised learning is mainly applied in the case of the accurate physical model or mathematical model is unavailable, but the huge history data are available. When there are huge history

data but only a small part of them are labeled, the Semi-supervised learning can be very good method to build the model. Association rules are mainly used to find the relations between the features. Swarm Intelligence, such as particle Swarm Optimization and Bee Colony Algorithm, is mainly used to solve the optimization problems, find the optimal solution for NP problems.

There are too many methods and algorithm of Data Mining techniques. This Chapter only introduced some of them which will be applied in IFDPS system.

4 Sensor Classification and Sensor Placement Optimization

4.1 Introduction

A sensor is a converter that measures physical quantity and converts it into signal which can be read by an observer or by instruments. It is device that detects changes in the ambient conditions or in the state of another device or a system, and conveys or records this information in a certain manner. Sensors and sensing strategies constitute the foundational basis for fault diagnosis and prognosis systems. Most of sensors are well-developed in market and the customers just need to choose suitable sensors to collect data which can be used to monitor the condition of components or machines. When choosing sensors for diagnostics and prognostics, many parameters and features of the sensors must to be considered which are the type, number, and location of sensors; their size, weight, cost, dynamic range, and other characteristic properties; whether they are of the wired or wireless variety; etc. The raw data collected from the sensors are rarely useful because they may contain much noise or no explicit features. These data must be processed appropriately so that useful information may be extracted that is a reduced version of the original data but preserves as much as possible those characteristic features or fault indicators that are indicative of the fault events we are seeking to detect, isolate, and predict the time evolution of. Thus such data must be preprocessed, that is, filtered, compressed, correlated, etc., in order to remove artifacts and reduce noise levels and the volume of data to be processed subsequently. Furthermore, the sensor providing the data must be validated; that is, the sensors themselves are not subjected to fault conditions. Once the preprocessing module confirms that the sensor data are ''clean'' and formatted appropriately, features or signatures of normal or faulty conditions must be extracted. This is most important in the framework of IFDPS because it is the input of the processes of diagnostics and prognostics [Wachtsevanos et al. 2006]

Sensor suites are specific to the application domain, and they are intended to monitor such typical state awareness variables as temperature, pressure, speed, vibrations, etc. Some sensors are inserted specifically to measure quantities that are directly related to fault modes identified as candidates for diagnosis. Among them are strain gauges, ultrasonic sensors, proximity devices, acoustic emission sensors, electrochemical fatigue sensors, interferometers, etc., whereas others are of the multipurpose variety, such as temperature, speed, flow rate, etc., and are designed to monitor process variables for control and/or performance assessment in addition to diagnosis. More recently we have witnessed the introduction of wireless devices in the area of condition monitoring.

For a normal word 'sensor' device, it actual have two components: sensor and transducer. A sensor is defined as a device that is sensitive to light, temperature, electrical impedance, or radiation level and transmits a signal to a measuring or control device. On the other hand, a transducer is defined as a device that receives energy from one system and retransmits it, often in a different form, to another system. A measuring device passes through two stages while measuring a signal.

First, the measurand (a physical quantity such as acceleration, pressure, strain, temperature etc.) is sensed by the sensor. Then, the measured signal is transduced into a form that is particularly suitable for transmitting, signal conditioning, and processing. For this reason, output of the transducer stage is often an electrical signal that is then digitized. The sensor and transducer stages of a typical measuring device are represented schematically in Fig. 4.1.
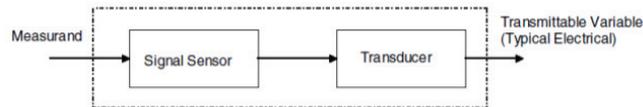
Fig. 4.1 Schematic Representation of a Measuring Device

The sensor strategies are mainly focused two issues: the one is which kind of sensors is suitable to measure the signals, and the other is which place the sensors should be set up. This Chapter will introduce these two issues.

4.2   Classification of Sensors

There are many kinds of sensors in the business market. White [White, 1987] presented out a sensor classification scheme for categorizing sensors which are recalled in the following tables. Table 4.1 shows most measurands for which sensors may be needed under the headings: acoustic, biological, chemical, electric, magnetic, mechanical, optical, radiation (particle), and thermal, etc. With a particular measurand, one is primarily interested in sensor characteristics such as sensitivity, selectivity, and speed of response which is shown in Table 4.2 called technological aspects of sensors. Table 4.3 shows the detection means used in sensors. Table 4.4 is intended to indicate the primary phenomena used to convert the measurand into a form suitable for producing the sensor output. The application fields are listed in Table 4.5. Most sensors contain a variety of materials (for example, almost all contain some metal). The entries in Table 4.6 should be understood to refer to the materials chiefly responsible for sensor operation.

Table 4.1 Measurands of Sensors

A. Measurands
  Al. Acoustic
    Al.1 Wave amplitude, phase, polarization, spectrum
    A1.2 Wave velocity
    A1.3 Other (specify)
  A2. Biological
    A2.1 Biomass (identities, concentrations, states)
    A2.2 Other (specify)
  A3. Chemical
    A3.1 Components (identities, concentrations, states)
    A3.2 Other (specify)
  A4. Electric
    A4.1 Charge, current

A4.2 Potential, potential difference
A4.3 Electric field (amplitude, phase, polarization, spectrum)
A4.4 Conductivity
A4.5 Permittivity
A4.6 Other (specify)
A5. Magnetic
A5.1 Magnetic field (amplitude, phase, polarization, spectrum)
A5.2 Magnetic flux
AS.3 Permeability
AS.4 Other (specify)
A6. Mechanical
A6.1 Position (linear, angular)
A6.2 Velocity
A6.3 Acceleration
A6.4 Force
A6.5 Stress, pressure
A6.6 Strain
A6.7 Mass, density
A6.8 Moment, torque
A6.9 Speed of flow, rate of mass transport
A6.10 Shape, roughness, orientation
A6.11 Stiffness, compliance
A6.12 Viscosity
A6.13 Crystallinity, structural integrity
A6.14 Other (specify)
A7. Optical
A7.1 Wave amplitude, phase, polarization, spectrum
A7.2 Wave velocity
A7.3 Other (specify)
A8. Radiation
A8.1 Type
A8.2 Energy
A8.3 Intensity
A8.4 Other (specify)
A9. Thermal
A9.1 Temperature
A9.2 Flux
A9.3 Specific heat
A9.4 Thermal conductivity
A9.5 Other (specify)
A10. Other (specify)

Table 4.2 Technological Aspects of Sensors

B. Technological Aspects of Sensors
B1 Sensitivity
B2 Measurand range
B3 Stability (short-term, long-term)
B4 Resolution
B5 Selectivity
B6 Speed of response
B7 Ambient conditions allowed
B8 Overload characteristics
B9 Operating life
B10 Output format
B11 Cost, size, weight

Table 4.3 Detection Means Used in Sensors

C. Detection Means Used in Sensors
C1 Biological
C2 Chemical
C3 Electric, Magnetic, or Electromagnetic Wave
C4 Heat, Temperature
C5 Mechanical Displacement or Wave
C6 Radioactivity, Radiation
C7 Other (specify)

Table 4.4 Sensor Conversion Phenomena

D. Sensor Conversion Phenomena
Dl. Biological
D1.1 Biochemical transformation
D1.2 Physical transformation
D1.3 Effect on test organism
Dl .4 Spectroscopy
D1.5 Other (specify)
D2. Chemical
D2.1 Chemical transformation
D2.2 Physical transformation
D2.3 Electrochemical process
D2.4 Spectroscopy
D2.5 Other (specify)
D3. Physical
D3.1 Thermoelectric
D3.2 Photoelectric
D3.3 Photomagnetic
D3.4 Magnetoelectric
D3.S Elastomagnetic
D3.6 Thermoelastic
D3.7 Elastoelectric
D3.8 Thermomagnetic
D3.9 Thermooptic
D3.10 Photoelastic
D3.11 Other (specify)

Table 4.5 Fields of Application

F. Fields of Application
F1 Agriculture
F2 Automotive
F3 Civil engineering, construction
F4 Distribution, commerce, finance
F5 Domestic appliances
F6 Energy, power
F7 Environment, meteorology, security
F8 Health, medicine
F9 Information, telecommunications
F10 Manufacturing
F11 Marine
F12 Military
F13 Scientific measurement
F14 Space
F15 Transportation (excluding automotive)
F16 Other (specify)

Table 4.6 Sensor Materials

E. Sensor Materials
El Inorganic
E2 Organic
E3 Conductor
E4 Insulator
E5 Semiconductor
E6 Liquid, gas or plasma
E7 Biological substance
E8 Other (specify)

The scheme shown in above tables can facilitate comparing sensors, communicating with other workers about sensors, and keeping track of sensor progress and availability. Categorizing might help one think about new sensing principles that could be explored, and Table 4.2 might serve as a checklist to consult when considering commercial sensors. Refer to the sensor application in Condition Monitoring, the first step is to determine which measurands need to be measured which is shown in Table 4.1, and then analyze the requirements of the system to decide the technological aspects of the selected sensors which are shown in Table 4.2. In this Chapter, only the sensors can be used to collect data for fault diagnosis and prognosis are considered. Fig. 4.2 shows the most kinds of sensors applied in condition monitoring for fault diagnosis and prognosis. Some of these kinds of sensors are described following.

Mechanical sensor systems have been studied extensively, and a large number of such devices are currently in use to monitor system performance for operational state assessment and tracking of fault indicators. A number of mechanical quantities—position, speed, acceleration, torque, strain, etc.—are commonly employed in dynamic systems. The most widely used sensors in condition monitoring for manufacturing machines are vibration sensors and strain gauges.

63

Mechanical Sensor Systems (A6)
- Accelerometers (Vibration Measurements)
- Strain gauges
- Ultrasonic Sensor System
- Position, speed, acceleration, torque, strain
- ......

Performance Sensors (A1, A3, A7 and A9)
- Temperature Sensors / Thermography
- Pressure, Fluid and thermodynamic
- Optical properties and biochemical elements
- ....

Electrical Measurement (A4)
- Eddy-Current Proximity Probes
- Microelectromechanical System (MEMS) Sensors

Fiber Optic Sensors
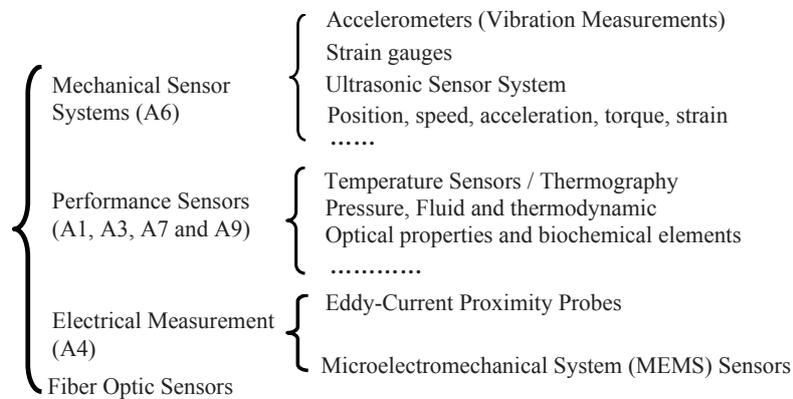
Fig. 4.2 The Classification of Sensors.

Recent years have seen an increased requirement for a greater understanding of the causes of vibration and the dynamic response of failing structures and machines to vibratory forces. An accurate, reliable, and robust vibration transducer therefore is required to monitor online such critical components and structures. Piezoelectric accelerometers offer a wide dynamic range and rank among the optimal choices for vibration-monitoring apparatus. They exhibit such desirable properties as [Wachtsevanos et al., 2006]:

- Usability over very wide frequency ranges;
- Excellent linearity over a very wide dynamic range;
- Electronically integrated acceleration signals to provide velocity and dis- placement data.
- Vibration measurements in a wide range of environmental conditions while still maintaining excellent accuracy
- Self-generating power supply
- No moving parts and hence extreme durability
- Extremely compact plus a high sensitivity-to-mass ratio

Piezoelectric accelerometers are used to measure all types of vibrations regardless of their nature or source in the time or frequency domain as long as the accelerometer has the correct frequency and dynamic ranges.

A strain-gauge sensor is based on a simple principle from basic electronics that the resistance of a conductor is directly proportional to its length and resistivity and inversely proportional to its cross-sectional area. Applied stress or strain causes the metal transduction element to vary in length and cross-sectional area, thus causing a change in resistance that can be measured as an electrical signal. Certain substances, such as semiconductors, exhibit the piezoresistive effect, in which application of strain greatly affects their resistivity. Strain gauges of this type have a sensitivity approximately two orders greater than the former type. The transducer usually is used within a Wheatstone bridge arrangement, with one, two, or all four of the bridge arms being individual strain gauges, so that the output voltage change

is an indication of measurand (the strain) change. The output of the strain-gauge is a simple voltage signal that can be connected to an oscilloscope to view the strain output or to the data-acquisition system to take strain-gauge data.

Ultrasonic sensor systems are being considered for monitoring the health of critical structures such as airplanes, bridges, and buildings. Ultrasonic methods are particularly suitable for structural health monitoring both because ultrasonic waves travel long distances and thus have the potential to monitor a large volume of material and because ultrasonic methods have proven useful for nondestructive inspection of such structures during maintenance. There are three main types of ultrasonic waves that are suitable for structural health monitoring: guided waves, bulk waves, and diffuse waves. Regardless of the type of wave, the strategy is to monitor changes and then detect, localize, and characterize damage based on the nature of the change. This strategy of looking for changes can enable detection sensitivity to be similar to that of nondestructive inspection despite the limitation of fixed sensors. The theory behind ultrasonic ranging is quite simple (as shown in Fig. 4.3). Typically a short ultrasonic burst is transmitted from the transmitter. When there is an object in the path of the ultrasonic pulse, some portion of the transmitted ultrasonic wave is reflected and the ultrasonic receiver can detect such echo. By measuring the elapsed time between the sending and the receiving of the signal along with the knowledge of the speed of sound in the medium, the distance between the receiver and the object can be calculated.
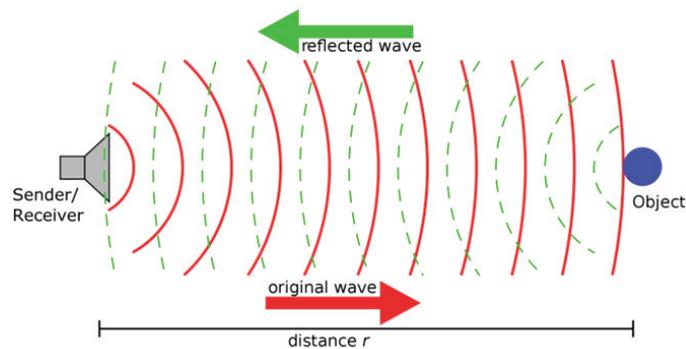
Fig. 4.3 Principle of Ultrasonic Sensors

System performance and operational data are monitored routinely in all industrial establishments, utility operations, transportation systems, etc. for process control, performance evaluation, quality assurance, and fault diagnosis purposes. A large number of sensor systems have been developed and employed over the years. The list includes devices that are intended to measure such critical properties as temperature; pressure; fluid, thermodynamic, and optical properties; and biochemical elements, among many others. Sensors based on classic measuring elements—inductive, capacitive, ultrasound— have found extensive applications.

Temperature variations in many mechanical, electrical, and electronic systems are excellent indicators of impending failure conditions. Temperatures in excess of control limits should be monitored and used in conjunction with other

measurements to detect and isolate faults. Temperature sensing has found numerous applications over the years in such areas as engineering, medicine, environmental monitoring, etc. Therefor the temperature sensors play a very important role in condition monitoring. A temperature sensor is a device that gathers data concerning the temperature from a source and converts it to a form that can be understood either by an observer or another device. Temperature sensors come in many different forms and are used for a wide variety of purposes, from simple home use to extremely accurate and precise scientific use. They play a very important role almost everywhere that they are applied. The best known example of a temperature sensor is the mercury-in-glass thermometer. Mercury expands and contracts based on changes in temperature; when these volume changes are quantified, temperature can be measured with a fair degree of accuracy. The outside temperature is the source of the temperature measurements and the position of the mercury in the glass tube is the observable quantification of temperature that can be understood by observers. Typically, mercury-in-glass thermometers are only used for non-scientific purposes because they are not extremely accurate. In some cases, they can be used in high school or college chemistry labs when a very accurate measurement of temperature is not important. The most common temperature sensors in scientific area are resistance temperature detectors (RTDs), whose principle of operation is variation of the resistance of a platinum wire or film as a function of temperature. Platinum usually is employed because of its stability with temperature and the fact that its resistance tends to be almost linear with temperature. Such temperature devices have higher accuracy than that of mercury-in-glass thermometer and thus are used widely in condition monitoring when the temperature of the machines or environment needs to be monitor accurately.

Electrical measurements are the methods, devices and calculations used to measure electrical quantities. Measurement of electrical quantities may be done to measure electrical parameters of a system. Using transducers, physical properties such as temperature, pressure, flow, force, and many others can be converted into electrical signals, which can then be conveniently measured and recorded. According to the principle, a number of sensor systems based on the Electrical measurements have been developed and applied in the recent past in an attempt to interrogate critical components and systems for fault diagnosis and prognosis. Transducing principles based on eddy-current response characteristics, optical and infrared signal mentoring, microwaves, and others have been investigated.

Response characteristics of induced eddy currents in conducting media are monitored for changes in their behavior owing to material anomalies, cracks, shaft or mating-part displacements, etc. Eddy-current proximity probes are a mature technology that has been used for protection and management of rotating machinery. They are employed commonly in high-speed turbo machinery to observe relative shaft motion directly, that is, inside bearing clearances of fluid-film interfaces. Zou et al. describe the application of eddy-current proximity sensing to the detection of a crack in a seal/rotor drive-shaft arrangement [Zou et al., 2000].

Of interest also are sensor systems that can be produced inexpensively, singly or in an array, while maintaining a high level of operational reliability. Microelectromechanical systems (MEMS) and sensors based on fiber-optic technologies are finding popularity because of their size, cost, and ability to integrate multiple transducers in a single device. Micro-machined MEMS devices in silicon or other materials are fabricated in a batch process with the potential for integration with electronics, thus facilitating on-board signal processing and other ''smart'' functions. A number of MEMS transducer and sensor systems have been manufactured in the laboratory or are available commercially, monitoring such critical parameters as temperature, pressure, acceleration, etc. [Wachtsevanos et al., 2006].

Fiber optics has penetrated the telecommunications and other high-technology sectors in recent years. They find utility in the sensor field because of their compact and flexible geometry, potential for fabrication into arrays of devices, batch fabrication, etc. Fiber optic sensors have been designed to measure strain, temperature, displacement, chemical concentration, and acceleration, among other material and environmental properties. Their main advantages include small size, light weight, immunity to electromagnetic and radio frequency interference (EMI/RFI), high- and low-temperature endurance, fast response, high sensitivity, and low cost. Fiber optic technologies are based on extrinsic Fabry-Perot interferometry (EFPI), chemical change in the fiber cladding, optical signal changes owing to fiber stress and deformation, etc.

There are also some other kinds of sensors available in the market and most of them are very good to meet the monitoring requirement. We only need choose suitable ones to collect data from the machines for monitoring.

4.3   Wireless Sensor Networks

A sensor network is a group of specialized sensors with a communications infrastructure intended to monitor and record conditions at diverse locations. Commonly monitored parameters are temperature, humidity, pressure, wind direction and speed, illumination intensity, vibration intensity, sound intensity, power-line voltage, chemical concentrations, pollutant levels and vital body functions.

Sensor networks may consist of many different types of sensors such as seismic, low sampling rate magnetic, thermal, visual, infrared, acoustic and radar, which are able to monitor a wide variety of ambient conditions that include the following [Estrin et al., 1999]:

- Temperature,
- Humidity,
- Vehicular movement,
- Lightning condition,
- Pressure,
- Soil makeup,

- Noise levels,
- The presence or absence of certain kinds of objects,
- Mechanical stress levels on attached objects, and
- The current characteristics such as speed, direction, and size of an object.

A sensor network consists of multiple detection stations called sensor nodes, each of which is small, lightweight and portable. Every sensor node is equipped with a transducer, microcomputer, transceiver and power source. The transducer generates electrical signals based on sensed physical effects and phenomena. The microcomputer processes and stores the sensor output. The transceiver, which can be hard-wired or wireless, receives commands from a central computer and transmits data to that computer. The power for each sensor node is derived from the electric utility or from a battery.

Sensor networks can be deployed in the following two ways [Intanagonwiwat et al., 2000]:

- Sensors can be positioned far from the actual phenomenon, i.e., something known by sense perception. In this approach, large sensors that use some complex techniques to distinguish the targets from environmental noise are required.
- Several sensors that perform only sensing can be deployed. The positions of the sensors and communications topology are carefully engineered (Fig. 1.5). They transmit time series of the sensed phenomenon to the central nodes where computations are performed and data are fused.

The above described features ensure a wide range of applications for sensor networks. Some of the application areas are health, military, and security. For example, the physiological data about a patient can be monitored remotely by a doctor. While this is more convenient for the patient, it also allows the doctor to better understand the patient's current condition. Sensor networks can also be used to detect foreign chemical agents in the air and the water. They can help to identify the type, concentration, and location of pollutants. In essence, sensor networks will provide the end user with intelligence and a better understanding of the environment [Akyildi et al., 2002]. Sensor networks can also be very helpful in condition monitoring for manufacturing machines, wind turbines, transporters and infrastructure because they may be distributed in different place. Potential applications of sensor networks may include:

- Condition monitoring for factory or infrastructure;
- Industrial automation;
- Automated and smart homes;
- Video surveillance;
- Traffic monitoring;
- Medical device monitoring;
- Monitoring of weather conditions;
- Air traffic control;
- Military applications;
- Robot control.

While many sensors connect to controllers and processing stations directly (e.g., using local area networks), an increasing number of sensors communicate the collected data wirelessly to a centralized processing station which are compose a Wireless Sensor Network (WSN). This is important since many network applications require hundreds or thousands of sensor nodes, often deployed in remote and inaccessible areas. Therefore, a wireless sensor has not only a sensing component, but also on-board processing, communication, and storage capabilities. With these enhancements, a sensor node is often not only responsible for data collection, but also for in-network analysis, correlation, and fusion of its own sensor data and data from other sensor nodes. When many sensors cooperatively monitor large physical environments, they form a WSN. Sensor nodes communicate not only with each other but also with a base station (BS which could be a gateway) using their wireless radios, allowing them to disseminate their sensor data to remote processing, visualization, analysis, and storage systems. For example, Fig. 4.4 shows two sensor fields monitoring two different geographic regions and connecting to the Internet using their base stations [Dargie and Poellabauer, 2010].
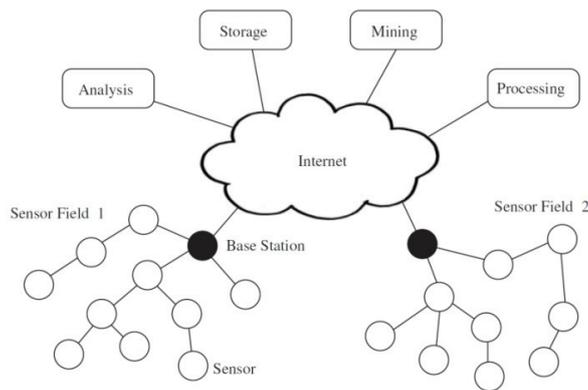
Fig. 4.4 Wireless Sensor Networks

The capabilities of sensor nodes in a WSN can vary widely, that is, simple sensor nodes may monitor a single physical phenomenon, while more complex devices may combine many different sensing techniques (e.g., acoustic, optical, magnetic). They can also differ in their communication capabilities, for example, using ultrasound, infrared, or radio frequency technologies with varying data rates and latencies. While simple sensors may only collect and communicate information about the observed environment, more powerful devices (i.e., devices with large processing, energy, and storage capacities) may also perform extensive processing and aggregation functions. Such devices often assume additional responsibilities in a WSN, for example, they may form communication backbones that can be used by other resource-constrained sensor devices to reach the base station. Finally, some devices may have access to additional supporting technologies, for example, Global Positioning System (GPS) receivers, allowing them to accurately determine their position. However, such systems often consume too much energy to be feasible for low-cost and low-power sensor nodes [Dargie and Poellabauer, 2010].

The well-known IEEE 802.11 family of standards was introduced in 1997 and is the most common wireless networking technology for mobile systems. It uses different frequency bands, for example, the 2.4-GHz band is used by IEEE 802.11b and IEEE 802.11g, while the IEEE 802.11a protocol uses the 5-GHz frequency band. IEEE 802.11 was frequently used in early wireless sensor networks and can still be found in current networks when bandwidth demands are high (e.g., for multimedia sensors). However, the high-energy overheads of IEEE 802.11-based networks make this standard unsuitable for low-power sensor networks. Typical data rate requirements in sensor networks are comparable to the bandwidths provided by dial-up modems, therefore the data rates provided by IEEE 802.11 are typically much higher than needed. This has led to the development of a variety of protocols that better satisfy the networks' need for low power consumption and low data rates. For example, the IEEE 802.15.4 protocol [Callaway et al., 2002] has been designed specifically for short- range communications in low-power sensor networks and is supported by most academic and commercial sensor nodes.

The network topologies can be seen as in Fig. 1.5 and the most widely used ones are topologies of star and mesh. When the transmission ranges of the radios of all sensor nodes are large enough and the sensors can transmit their data directly to the base station, they can form a star topology as shown on the left in Fig. 4.5. In this topology, each sensor node communicates directly with the base station using a single hop. However, sensor networks often cover large geographic areas and radio transmission power should be kept at a minimum in order to conserve energy; consequently, multi-hop communication is the more common case for sensor networks (shown on the right in Fig. 4.5). In this mesh topology, sensor nodes must not only capture and disseminate their own data, but also serve as relays for other sensor nodes, that is, they must collaborate to propagate sensor data towards the base station. This routing problem, that is, the task of finding a multi-hop path from a sensor node to the base station, is one of the most important challenges and has received immense attention from the research community. When a node serves as a relay for multiple routes, it often has the opportunity to analyze and pre-process sensor data in the network, which can lead to the elimination of redundant information or aggregation of data that may be smaller than the original data. The more detailed information about Wireless Sensor Networks can be found at the reference of [Dargie & Poellabauer, 2010].
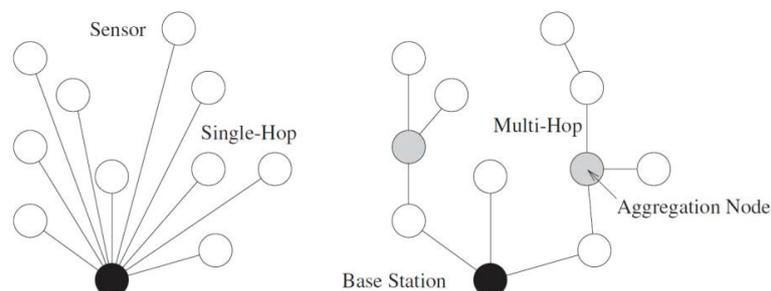
Fig. 4.5 Single-hop Versus Multi-hop Communication in Sensor Networks

4.4   RFID Sensor Networks

Radio Frequency IDentification (RFID) is one of numerous technologies grouped under the term of Automatic Identification (Auto ID), such as bar code, magnetic inks, optical character recognition, voice recognition, touch memory, smart cards, biometrics etc. Auto ID technologies are a new way of controlling information and material flow, especially suitable for large production networks [Ilie-zudor et al., 2006]. RFID is the use of a wireless non-contact radio system to transfer data from a tag attached to an object, for the purposes of identification and tracking. In general terms, it is a means of identifying a person or object using a radio frequency transmission. The technology can be used to identify, track, sort or detect a wide variety of objects [Lewis, 2004]. Recently, RFID become more and more interesting technology in many fields such as agriculture, manufacturing and supply chain management.

The history of RFID technology can be tracked back to the radio-based identification system used by allied bombers during World War II [Garfinkel & Holtzman, 2005]. Early identification Friend or For (IFF) systems were used to distinguish Allied fighter and bomber by identifying the correct signals sent by Allied aircrafts, from aircrafts sent by enemy at night. After the war, Harry Stockman realized that it is possible to power a mobile transmitter completely from the strength of a received radio signal, and then he introduced the concept of passive RFID systems [Stockman, 1948]. In 1972, a patent application for "inductively coupled transmitter-responder arrangement" was filed which is used separate coils for receiving power and transmitting the return signal [Kriofsky & Kaplan, 1975]. In 1979, a patent application for "identification device" (two antennas was combined) was filed which is seen as a RFID landmark because it emphasized the potentially small size of RFID device [Beigel, 1982]. The 1980s became the decade for full implementation of RFID technology, though interests developed somewhat differently in various parts of the world. The greatest interests in the United States were for transportation, personnel access, and to a lesser extent, for animals. In Europe, the greatest interests were for short-range systems for animals, industrial and business applications, though toll roads in Italy, France, Spain, Portugal, and Norway were equipped with RFID. The 1990s were a significant decade for RFID since it saw the wide scale deployment of electronic toll collection in the United States. The world's first open highway electronic tolling system opened in Oklahoma in 1991 and then extended to the whole world. Interest was also keen for RFID applications in Europe during the 1990s. Both Microwave and inductive technologies were finding use for toll collection, access control and a wide variety of other applications in commerce [Landt, 2001]. The 21st century opens with the smallest microwave tags built using, at a minimum, two components: a single custom CMOS integrated circuit and an antenna. Tags could now be built as sticky labels, easily attached to windshields and objects to be managed [Landt, 2005]. It seems that there are still a great many developments of RFID to look forward to as the history continues to teach that and RFID will be presented in our daily life.

4.4.1 RFID System

Typical RFID systems fundamentally consist of four elements: the RFID tags, the RFID readers, the antennas and choice of radio characteristics, and the computer network (if any) that is used to connect the readers (Fig. 4.6). Tags are attached to objects and each of them has a certain amount of internal memory (E2PROM) in which it stories information about the object, such as its unique ID number, or in some cases more details including manufacture data and product composition. When these tags pass through a field generated by a reader, they transmit this information back to the reader, thereby identifying the object. Until recently, the focus of RFID technology was mainly on tags and readers which were being used in systems where relatively low volumes of data are involved. This is now changing as RFID in the supply chain is expected to generate huge volumes of data, which will have to be filtered and routed to the backend IT systems. To solve this problem companies have developed special software packages (Middleware), which act as buffers between the RFID front end and the IT backend [Wang & Zhang, 2012].
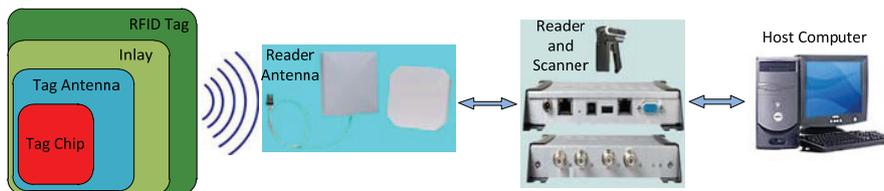
Fig. 4.6 Typical RFID System
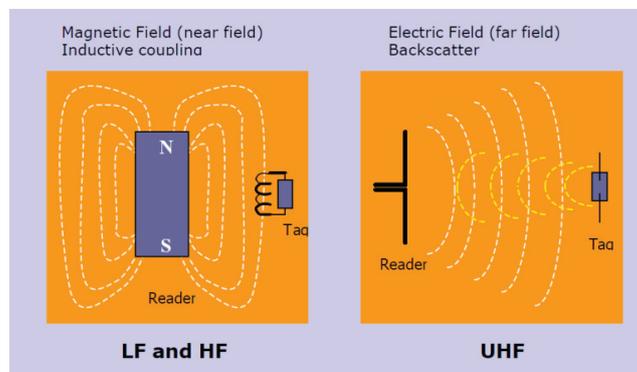
Fig. 4.7 RFID Tags Communication Methods

There are two main communication principles between RFID readers/antennas and RFID Tags: inductive coupling and backscatter reflection which are used in near field and far field respectively (Fig. 4.7). The principle of inductive coupling means transferring energy from one circuit to another through mutual inductance. Near field employs inductive coupling of the tag to the magnetic field circulating around the reader antenna (like a transformer). In RFID systems using inductive coupling, the reader antenna and the RFID tag antenna each have a coil which

together forms a magnetic field so that the tag draws energy from the field to change the electrical load on the tag antenna. The change is picked up by the reader and read as a unique serial number. Far field uses similar techniques to radar (Backscatter reflection) by coupling with the electric field. RFID tags using backscatter technology reflect radio waves at the same carrier frequency back to the tag reader, using modulation to transmit the data.

The communication process between the reader and tag is managed and controlled by one of several protocols, such as the ISO 15693 and ISO 18000-3 for HF or the ISO 18000-6, and EPC for UHF. Basically what happens is that when the reader is switched on, it starts emitting a signal at the selected frequency band (typically 860 - 915MHz for UHF or 13.56MHz for HF). Any corresponding tag in the vicinity of the reader will detect the signal and use the energy from it to wake up and supply operating power to its internal circuits. Once the Tag has decoded the signal as valid, it replies to the reader, and indicates its presence by modulating (affecting) the reader field.

The communication principle can be used to compose parts of wireless sensor network.

4.4.2   Embedded RFID Sensor Monitoring

RFID sensor enabled tags, which can be used in such fields as project tracking, environmental monitoring, automotive electronic system, telemedicine and manufacturing processes controlling, etc., are bred as the result. Without doubts, they will play important roles in more and more areas as the technology is progressively growing. Roughly, the primary sensors in use today can be classified according to their functions in many categories such as: temperature, pressure, acceleration, inclination, humidity, light, gas sensor and chemical sensors [Ruhanne et al., 2008] .

Fig. 4.8 shows the system architecture for a generic sensor tag and its interaction with RFID systems as it passes through various stages of the manufacturing, assembling and supply chain. The RFID tags can be combined to the sensor devices (many different sensors) and transfer the sensing data to the RFID reader and further to the database through radio waves. Typically for the supply, there are a number of RFID portals and at each of these passive RFID tag is interrogated. The data obtained could be used for improving the process and scheduling of supply chain and production process [Wang & Zhang, 2012]. For the manufacturing systems and processes, there are many sensors mounted on the machines which can be combined with RFID tags. The collected data can be transmitted to RFID reader and database, and the data with some processing techniques can be used to monitor the condition of the machine and improve the performance.
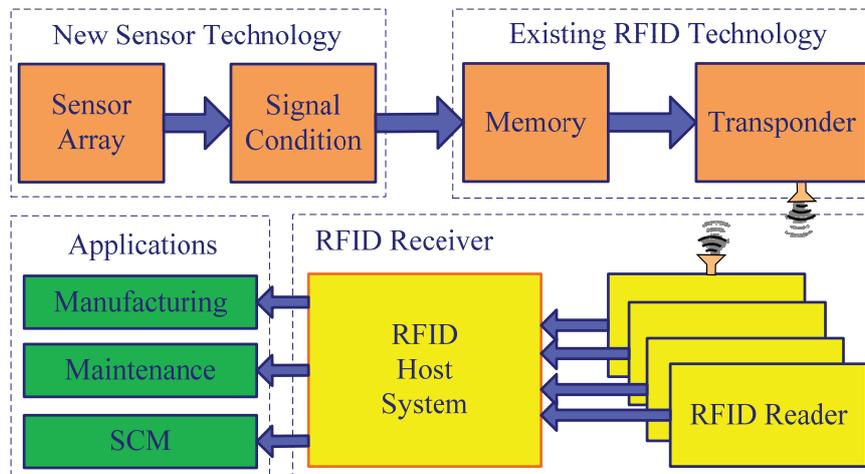
Fig. 4.8 General Structure of Embedded RFID Sensing System

4.5 General Sensor Networks

This section is a summary of sensor network techniques mentioned above. The wired network, Wi-Fi wireless networks, Bluetooth and RFID can be integrated together to collect data according to the requirements of real projects. The general structure of the integrated sensor networks are shown in Fig. 4.9. In the real application sites, suitable sensors are selected to collect data of the machines. The collected data can be transmitted to database through wired network, wireless network (Wi-Fi), RFID and Bluetooth. The customers may use one or more these kinds of methods to transfer the data according to the requirements and considering the cost of human resource, economy and so on.

4.6 Sensor Placement Optimization (SPO)

The basic problem for condition monitoring is to deduce the existence of a defect in a structure from measurements taken at sensors distributed on the structure. The correctness of defect diagnosis depends on the method of pattern recognition for fault and effectiveness of signals from the sensors mounted on the machines. While carrying out on-site condition monitoring for a machine, the inappropriate distribution of sensors might result in weak incentives of certain order or modal, and affect the accuracy of fault identification. The aim of optimizing the placement of sensors is to obtain as much as possible of machine structural information with as few as possible sensors, which benefit the company in the economy viewpoint. Because of constraints of machine structure and environment, and consideration of economy, only a small number of sensors are installed when a condition monitoring system is established. It is very important to design the optimal position of the sensor to mount in order to ensure the accuracy and correctness of monitoring and fault judgement.

Fig. 4.9 General Sensor Network Structure

There are many literatures in optimal placement optimization of sensors in machine level. The spatial controllability was used to find the optimal placement of collocated actuator-sensor pairs for effective average vibration reduction over the entire structure, and the maintaining modal controllability and observability were used to select vibration modes for a thin plate [Halim & Reza Moheimani, 2003]. Recently, intelligent optimization algorithm has developed well which is a method to simulate the biological and physical process which can be used in sensor placement optimization. Many researchers focus on Genetic Algorithm (GA) application in sensor placement optimization and make up for a lot of shortage of the traditional optimization algorithm [Li et al., 2000; Liu et al., 2008; Sun et al., 2008]. But GA has to adopt binary coding and has complex operation process such as mutation, genetic and crossover. PSO adopts real number coding to avoid the complex operation, which is simple and easy to realize. So it is easy to apply in sensor placement optimization. PSO and finite element analysis were combined together to search the sensors optimal placement of a gearbox [Pan et al., 2010]. Binary PSO and Analytical redundancy Relations (ARRs) were combined to optimize the sensor placement for fault diagnosis [Du et al., 2011]. The sensor placement optimization is a very important aspect for many applications such as modal test and parameter identification [Cheng 2003; Papadimitriou 2004; Pennacchi and Vania 2008], fault diagnosis [Bhushan and Rengaswamy, 2000; Molter et al., 2010; Staszewski, 2002; Worden and Burrows, 2001] and process monitoring [Wang et al., 2002]. This section tries to apply PSO and finite element analysis in sensor placement optimization in order to get enough information of

machine structure using a small number of sensors and ensure the accuracy and correctness of condition monitoring.

4.6.1 Problem Description

Modal analysis (finite element analysis) is a very important method for fault diagnosis and condition monitoring. Faults of a machine, such as crack, axis loosening and fatigue, usually accompany with the change of physical parameters, such as natural frequency, modal damping, vibration mode and frequency response function. The faults can be diagnosed according to these changes. The machine's vibration is supposed to be a n degree of freedom linear time-invariant system which differential function can be written as [Wei and Pan 2010]:

$$M \ddot{x}(t) + C \dot{x}(t) + Kx(t) = f(t) \tag{4.1}$$

where: M, C and K are the system mass, damping and stiffness matrix respectively which are $n \times n$ matrix. $x(t)$, $\dot{x}(t)$ and $\ddot{x}(t)$ are n order response vectors of system displacement, velocity and acceleration respectively. $f(t)$ represents n order excitation force vector. Then the frequency displacement response function can be obtained by Fourier transform and set $x(t) = xe^{j\omega t}$ as:

$$x(\omega) = H(\omega)F(\omega) \tag{4.2}$$

where $H(\omega)$ means the frequency displacement response function which is a matrix. If the actuation is charged in i point of the machine, the frequency response function of j point can be written as:

$$H_{ij}(\omega) = \sum_{r=1}^{n} \frac{\phi_{jr}\phi_{ir}}{-\omega^2 M_r + j\omega C_r + K_r} \tag{4.3}$$

where M_r, C_r, K_r and ϕ_r represent modal mass, modal damping, modal stiffness and each order vibration mode vector. Eq. (4.3) shows the relationship between transfer function and the modal parameters and for a certain machine the value of $(-\omega^2 M_r + j\omega C_r + K_r)$ is always the same because it only depends on the frequency and damping ratio. Therefore, the value of frequency response function depends on vibration mode vector of i and j points.

Let $\phi = [\phi_1, \phi_2, \cdots, \phi_n]$ be a displacement mode in which ϕ_i is a N dimension vector where N means the freedom degree of the machine structure. Let m be the number of sensors (or number of measurement points) mounted on the machine while $o = N - m$ be a non-measurement points. The fitness function can be as:

$$f = \sum_{i=1}^{n} \sum_{j=1}^{n} \left| \sum_{r \in o} \phi_{ri} \phi_{rj} \right| \tag{4.4}$$

where ϕ_{ri} means the rth component of jth vibration mode and $r \in o$ means all calculation vectors are of non-measurement points. Compared Eq. (4.3) and Eq.(4.4), it is only task to find the minimum value of Eq. (4.4) for the optimal distribution of sensors. Therefore, it is chosen to be fitness function to find optimal placement of sensors.

4.6.2 Application of PSO in Sensor Placement Optimization

4.6.2.1 The Process of PSO Application in Sensor Placement Optimization

The principle of PSO has been introduced in Section 3.5. This section induced how to apply PSO to solve the Sensor placement optimization problems. Sensor placement can be solved by simple mathematical calculation, but it will be time-consuming. For example, if there are n possible measuring points and m sensors are available to set up there will be $n!/(m!(n-m)!)$ times need to be calculated. PSO is a good optimization algorithm which can easily solve this problem. Fig. 4.10 shows the structure to apply PSO in sensor placement optimization. First of all, the machine structure is analyzed using finite element analysis, and at the same, according to the shape and the application, all possible measurement points can be determined. From result of above step, all vibration displacement modes can be calculated. Then, input all of these data to PSO to find the optimal sensor placement which can be sent to design and management center. According to the result, the staff can improve the structure design, or make it is easy to monitor the machine with high accuracy and correctness.
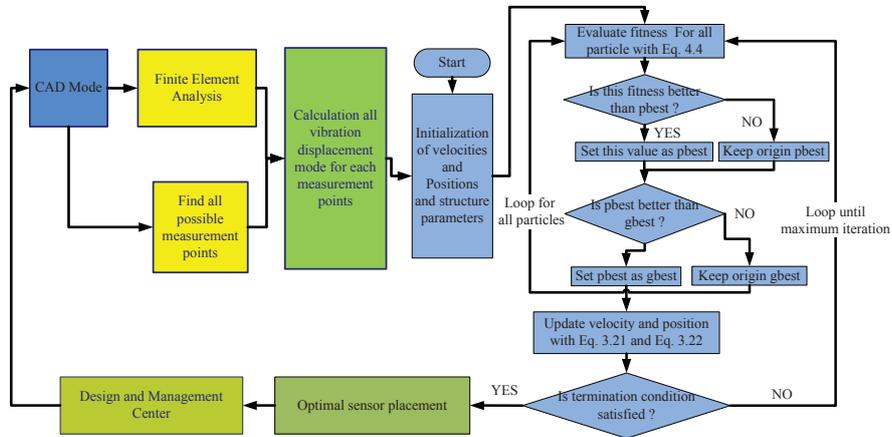
Fig. 4.10 Structure of PSO Application in Sensor Placement Optimization

4.6.2.2 Case Study and Its Results

In order to validate the effectiveness of the proposed method, a blower is chosen to analyze. The 3D model is bolted based in the practical installation and possible 10 measurement points are chosen to analyze (Fig. 4.11). When it is analyzed, the elastic modulus is set to E 210 Gpa, mass density to 7800 kg/m3 and Poisson ratio

to 0.3. The 3D solid model of the blower is built using the three-dimensional software Solidworks and then import to ANSYS 13.0 to carry out finite element analysis calculation and modal analysis. The blower is bolted to the floor in real installation, and thus the boundary condition of baseboard of blower is set to fixed constraint. This study calculates total 10 order natural frequency (Table 4.7) and its 10 vibration mode shapes of the blower are obtained. The finite element model and its first four vibration modes are shown in Fig. 4.12. Fig. 4.12(a) to Fig. 4.12 (d) shows from first to forth order of vibration shape mode respectively. In these figures, the arrows mean the movement directions of that mode. The natural frequency results (displacement) in total is shown in Table 4.8, and in three different directions (X, Y and Z) are shown in Table 4.9-Table 4.11.

Table 4.7 Main Natural Frequencies of Blower

Order	Frequency	Order	Frequency
1	58.282	6	185.62
2	116.77	7	214.41
3	121.98	8	229.1
4	164.62	9	243.49
5	165.52	10	250.3
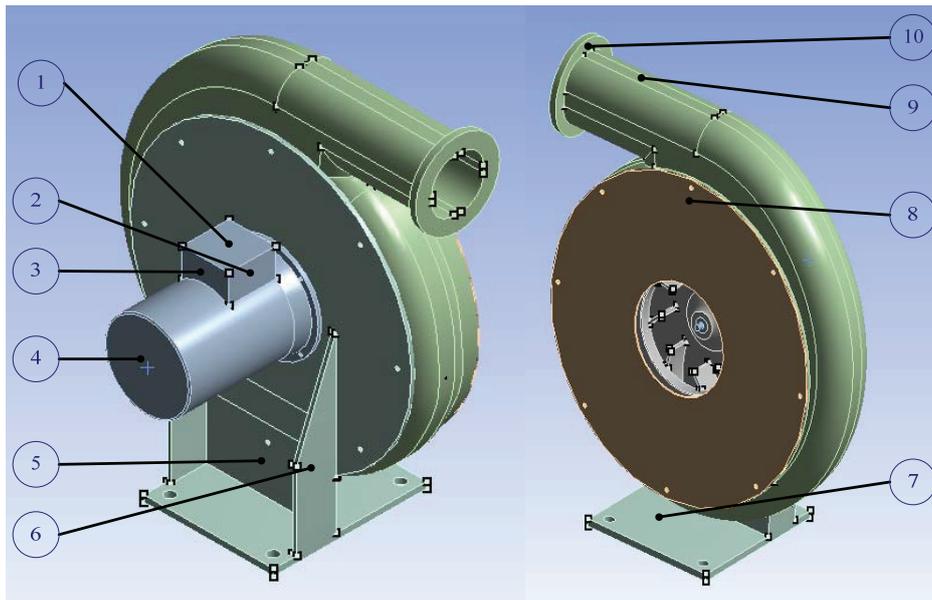
Fig. 4.11 Initial Placement of Measuring Points on the Blower
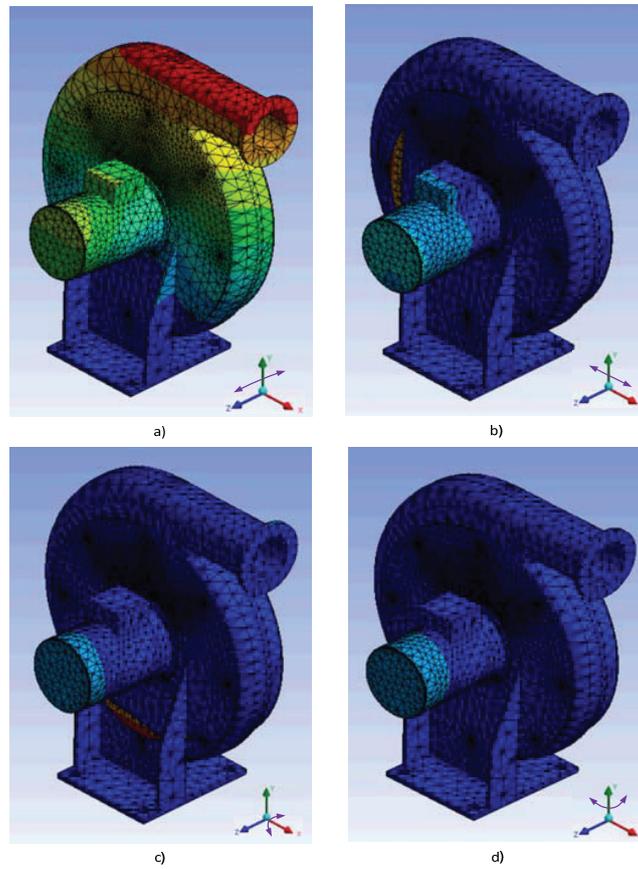
Fig. 4.12 The Finite Element Model of Blower and Its First Four Modes

Table 4.8 Total Displacement Mode for Each Point Order

Measuring Point	1st order	2nd order	3rd order	4th order	5th order	6th order	7th order	8th order	9th other	10th order
1	0.17836	9.060e-2	7.433e-2	6.391e-2	4.649e-2	0.2028	0.32562	0.10588	4.435e-2	3.509e-2
2	0.16318	9.666e-2	6.106e-2	6.268e-2	4.406e-2	0.18447	0.30796	9.893e-2	4.048e-2	4.081e-2
3	0.17333	0.13224	8.327e-2	7.811e-2	4.548e-2	0.259	0.31935	0.11036	4.282e-2	2.617e-2
4	0.17537	0.21849	0.12898	0.10694	4.355e-2	0.38771	0.30514	0.1153	4.059e-2	3.384e-2
5	5.862e-3	1.840e-3	6.917e-4	1.681e-3	2.018e-3	1.222e-2	2.505e-2	9.816e-3	6.142e-4	4.418e-3
6	2.433e-2	2.246e-2	9.974e-3	1.343e-2	6.225e-3	5.518e-2	3.595e-2	3.119e-2	3.028e-3	3.148e-2
7	5.9661e-10	2.6120e-10	2.7633e-10	7.6294e-11	2.009e-11	1.951e-10	4.300e-10	5.0898e-11	2.5647e-11	8.9125e-11
8	0.2517	9.544e-3	8.621e-2	3.165e-2	3.646e-2	0.10458	0.14308	0.19168	9.056e-2	0.12752
9	0.31278	4.151e-2	9.815e-2	5.240e-2	4.757e-2	0.11371	0.1602	0.18728	4.534e-2	0.12261
10	0.3195	6.082e-2	0.10333	6.415e-2	5.369e-2	0.20737	0.15862	0.2448	4.327e-2	0.16902

Table 4.9 X Directional Displacement Mode for Each Point Order

Measuring Point	1st order	2nd order	3rd order	4th order	5th order	6th order	7th order	8th order	9th other	10th order
1	-5.07e-3	9.585e-2	-4.30e-3	6.389e-2	3.341e-3	0.19631	-5.48e-2	0.10362	-7.03e-4	2.618e-2
2	-4.91e-3	8.861e-2	-4.54e-3	6.260e-2	3.204e-3	0.18686	-5.21e-2	9.645e-2	-6.69e-4	2.406e-2
3	-6.96e-3	0.1349	-5.64e-3	7.755e-2	4.552e-3	0.25141	-6.62e-2	0.10779	-1.02e-3	1.254e-2
4	-1.23e-2	0.21812	-8.48e-3	0.10665	7.873e-3	0.38268	-9.18e-2	0.10778	-1.79e-3	-2.65e-2
5	4.178e-5	9.337e-4	-1.40e-4	1.545e-3	6.528e-5	3.761e-3	-1.82e-3	2.943e-3	-3.67e-5	1.914e-3
6	-7.02e-3	6.974e-3	7.819e-4	7.756e-3	1.026e-3	3.040e-2	7.178e-3	-1.79e-4	2.035e-3	-3.20e-3
7	6.717e-12	-1.3584e-10	1.3694e-11	-5.702e-11	-9.6312e-12	-3.1304e-10	6.3539e-11	1.2883e-11	1.4565e-12	3.8828e-11
8	-9.822e-4	8.667e-3	2.214e-4	3.020e-2	-4.575e-3	3.122e-2	-3.36e-2	0.19303	3.795e-4	0.12611
9	2.46e-3	3.002e-2	-2.68e-3	4.980e-2	-2.146e-3	9.763e-2	-4.31e-2	0.18305	1.835e-5	0.11062
10	-2.6e-3	3.381e-2	-3.37e-3	5.157e-2	-2.602e-3	0.10477	-4.23e-2	0.18105	-4.48e-4	0.10807

Table 4.10 Y Directional Displacement Mode for Each Point Order

Measuring Point	1st order	2nd order	3rd order	4th order	5th order	6th order	7th order	8th order	9th other	10th order
1	-5.21e-2	-4.59e-3	4.28e-2	-3.77e-4	1.085e-2	-1.19e-2	-6.76e-2	-3.50e-3	-9.81e-3	-3.84e-3
2	-4.95e-2	-7.17e-3	4.459e-2	-4.24e-3	1.072e-2	-1.89e-2	-6.46e-2	-1.18e-2	-9.65e-3	-8.59e-3
3	-8.09e-2	-6.84e-3	6.938e-2	-1.05e-3	1.763e-2	-1.77e-2	-0.1073	-5.09e-3	-1.67e-2	-6.50e-3
4	-0.1542	-1.28e-2	0.12835	-1.65e-3	3.355e-2	-3.35e-2	-0.2039	-8.85e-3	-3.13e-2	-1.23e-2
5	8.573e-4	2.268e-4	2.018e-4	1.432e-4	-2.52e-4	5.930e-4	-7.87e-4	1.417e-3	-3.39e-6	2.517e-4
6	-4.25e-3	2.351e-3	-2.38e-3	-2.21e-3	1.412e-3	-3.25e-4	-7.73e-3	-1.11e-2	-8.34e-4	-8.52e-3
7	-9.490e-11	3.1173e-11	-6.281e-12	5.5385e-12	7.3163e-13	2.3211e-11	-3.944e-11	2.3103e-12	-0.605e-13	5.8361e-13
8	6.86e-2	5.901e-4	2.12e-2	-3.92e-4	-1.79e-2	-2.00e-3	-3.3e-2	-1.30e-3	-2.1e-2	-1.94e-3
9	5.224e-2	-6.94e-2	1.575e-2	-1.43e-2	-4.16e-3	-3.77e-2	-1.39e-2	-2.21e-2	-3.56e-3	-1.24e-2
10	3.905e-2	-1.87e-2	1.841e-2	-2.56e-2	-3.92e-2	-6.34e-2	-1.13e-2	-6.66e-2	-4.11e-4	-3.80e-2

Table 4.11 Z Directional Displacement Mode for Each Point Order

Measuring Point	1st order	2nd order	3rd order	4th order	5th order	6th order	7th order	8th order	9th other	10th order
1	0.16931	1.879e-2	-5.70e-2	8.68e-3	-4.53e-2	6.517e-2	0.31279	2.239e-2	4.327e-2	2.347e-2
2	0.15121	-2.33e-2	-4.03e-2	-5.27e-3	-4.25e-2	-2.36e-3	0.29841	1.322e-2	3.957e-2	3.204e-2
3	0.15458	1.774e-2	-4.40e-2	4.932e-3	-4.21e-2	5.924e-2	0.29066	2.198e-2	4.027e-2	2.328e-2
4	8.796e-2	1.260e-2	1.308e-2	8.660e-3	-2.73e-2	4.474e-2	0.1988	1.825e-2	2.641e-2	1.603e-2
5	-6.27e-3	1.490e-3	-9.35e-4	1.195e-3	1.903e-3	7.195e-3	2.383e-2	-6.82e-3	8.225e-4	-6.28e-3
6	1.839e-2	-2.14e-2	9.795e-3	-1.11e-2	-5.78e-3	-4.79e-2	3.553e-2	2.910e-2	2.075e-3	3.007e-2
7	2.2494e-10	8.1034e-11	6.7551e-11	-4.605e-11	-1.717e-11	-3.0784e-10	-6.1781e-10	-4.387e-11	-2.7575e-11	-7.837e-11
8	0.23909	3.971e-3	8.188e-2	1.070e-4	-7.81e-3	-8.09e-3	-0.1400	-2.76e-2	3.71e-2	-1.72e-2
9	0.31009	-1.83e-2	9.733e-2	-1.60e-2	-4.64e-2	-7.03e-2	-0.1563	2.965e-2	-4.47e-2	5.089e-2
10	0.3156	-4.82e-2	0.1005	-3.22e-2	-5.38e-2	-0.1671	-0.1538	0.1491	-4.33e-2	0.12277

All parameters are presented in above figures and tables. Accordingly, the process of PSO application in sensor placement optimization in Fig. 4.10 and the fitness function Eq. (4.4), the optimal sensor placement of the blower can be obtained using PSO. For the PSO algorithm, the number of particles is initialized as 10 and n $(1 \sim 10)$ sensors are assumed to place on blower measuring points. The weight ω is set to 1.2-0.8 with decreasing linearly, and the acceleration coefficient c_1 and c_2 is set as 1.2. The vibration mode parameters in Table 4.8-Table 4.11 are input to the PSO respectively which can be used to calculate the fitness value.

Table 4.12 shows the smallest fitness value and the corresponding sensor placement for the different number of measuring points using the total displacement mode for each measuring point (Table 4.8). From this table, the amount of information on the blower increases with the increasing of measuring points, because the fitness become smaller and smaller. The smallest fitness is very big (8.824) when there only one sensor place on the blower while it became very small even equal 0 when the number of sensors increasing to 8, 9 to 10. From this table, the importance of measuring points can be obvious observed. The point 4 is the most important while the point 7 is the least important. The amount information also can be calculated from this table. Just take measuring point 6 as example, its amount information can be calculated as fitness value in point 5 minus in point 6 (2.422-1.213=1.209).

Table 4.13, Table 4.14 and Table 4.15 present the smallest fitness values and the corresponding sensor placement for the different number of measuring points using displacement modes for each measuring point in X direction, Y direction and Z

direction respectively. With these tables, the same conclusions can be obtained as the Table 4.12, and what's more, when the same number of sensors is planned to installed to the blower, the optimal places may different using different displacement modes. When optimal sensor placement is applied in real machine, it is very significant to know which direction is important for deformation referring to failure of machine.

Fig. 4.13 to Fig. 4.16 show fitness values changes with the changes of iteration PSO ($n=5$) for total, X direction, Y direction and Z direction respectively. From these figures, the optimal sensor placement can be obtained within 20 iterations of PSO for using all kinds of displacement mode. Combining all these figures and tables, PSO can successfully solve the optimal sensor placement problem.

As PSO has it important advantages in solving the optimization and NP problems, it is employed to solve sensor placement optimization problem for improving product design and fault diagnosis. Fitness is established for PSO application in sensor placement optimization based on the analysis on placement guidelines of vibration sensors. Generally, the proposed method combined the structure finite element modeling and its modal analysis, and PSO the carry out the optimal sensor placement distribution. The proposed method combining PSO and FEM analysis can be applied in machine level and component level but not system level because it need finite element mode and modal analysis of the structure. Therefore, the future research will be on the method for optimal sensor distribution in system level.

Table 4.12 Optimal Sensor Placement for Different Number of Measuring Point using Total Displacement Mode

Measuring Point No.	Sensor place position	Fitness	Measuring Point No.	Sensor place position	Fitness
1	4	8.824	6	1 2 3 4 9 10	1.213
2	4 10	6.793	7	1 2 3 4 8 9 10	0.059
3	3 4 10	5.183	8	1 2 3 4 6 8 9 10	0.004
4	3 4 9 10	3.786	9	1 2 3 4 5 6 8 9 10	4.1E-18
5	1 3 4 9 10	2.422	10	1 2 3 4 5 6 7 8 9 10	0

Table 4.13 Optimal Sensor Placement for Different Number of Measuring Point using X Direction Displacement Mode

Measuring Point No.	Sensor place position	Fitness	Measuring Point No.	Sensor place position	Fitness
1	4	1.7706	6	1 2 3 4 9 10	0.1886
2	3 4	1.3236	7	1 2 3 4 8 9 10	0.0046
3	1 3 4	1.0166	8	1 2 3 4 6 8 9 10	0.0002
4	1 3 4 10	0.7351	9	1 2 3 4 5 6 8 9 10	4.26E-19
5	1 2 3 4 10	0.4605	10	1 2 3 4 5 6 7 8 9 10	0

Table 4.14 Optimal Sensor Placement for Different Number of Measuring Point using Y Direction Displacement Mode

Measuring Point No.	Sensor place position	Fitness	Measuring Point No.	Sensor place position	Fitness
1	4	0.3971	6	1 2 3 4 9 10	0.0299
2	3 4	0.2888	7	1 2 3 4 8 9 10	0.0017
3	3 4 10	0.1860	8	1 2 3 4 6 8 9 10	2.240e-5
4	3 4 9 10	0.1257	9	1 2 3 4 5 6 8 9 10	4.17E-20
5	2 3 4 9 10	0.0729	10	1 2 3 4 5 6 7 8 9 10	0

Table 4.15 Optimal Sensor Placement for Different Number of Measuring Point using Z Direction Displacement Mode

Measuring Point No.	Sensor place position	Fitness	Measuring Point No.	Sensor place position	Fitness
1	10	2.7715	6	1 2 3 8 9 10	0.2538
2	9 10	2.0660	7	1 2 3 4 8 9 10	0.0478
3	1 9 10	1.4790	8	1 2 3 4 6 8 9 10	0.0032
4	1 3 9 10	0.9906	9	1 2 3 4 5 6 8 9 10	2.287E-18
5	1 2 3 9 10	0.5705	10	1 2 3 4 5 6 7 8 9 10	0
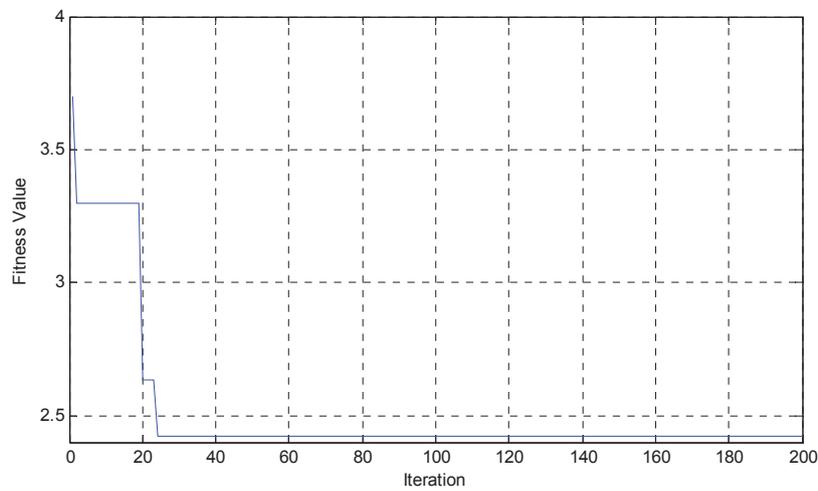
Fig. 4.13 Fitness Changes with Change of Iteration PSO ($n=5$) for Total Displacement Mode
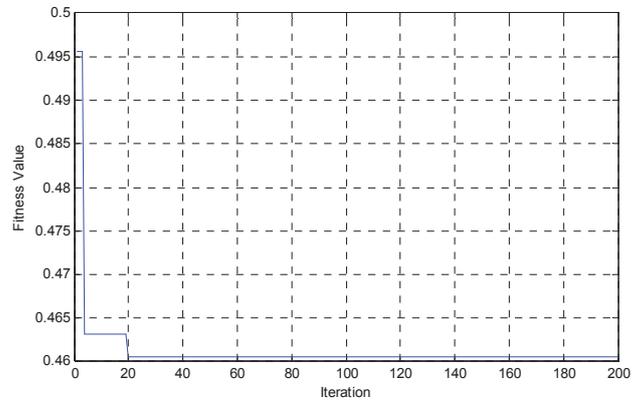
Fig. 4.14 Fitness Changes with Change of Iteration PSO ($n = 5$) for X Direction Displacement Mode
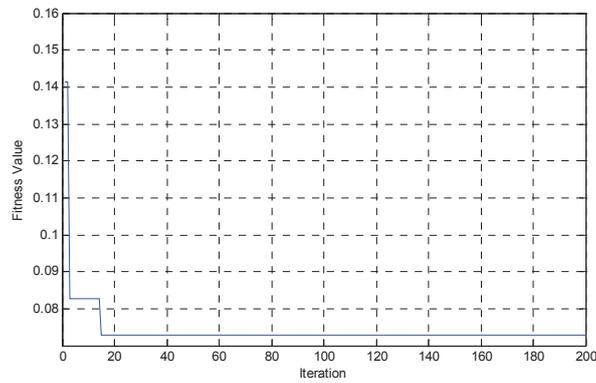
Fig. 4.15 Fitness Changes with Change of Iteration PSO ($n = 5$) for Y Direction Displacement Mode
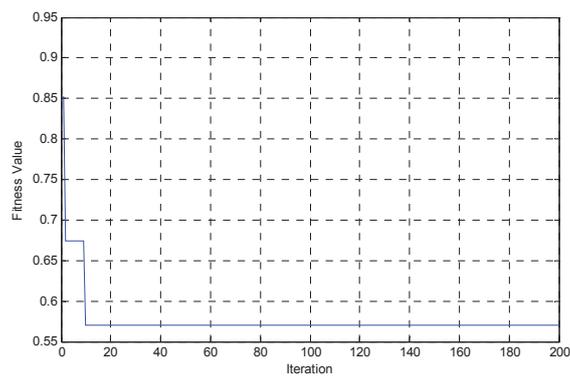
Fig. 4.16 Fitness Changes with Change of Iteration PSO ($n = 5$) for Z Direction Displacement Mode

84

4.6.3   Application of BCA in Sensor Placement Optimization

4.6.3.1   The Process of Application of BCA in Sensor Placement Optimization

The principle of BCA was introduced in Section 3.5.2 and this part will present how to apply BCA to solve the problem of sensor placement optimization. Fig. 4.17 shows the process of application of BCA to find the optimal sensor placement for manufacturing machines and other equipment. The necessity of using intelligent algorithm was described in Section 4.6.2.1. The 3D model of a machine can be established using Solidworks and transfer it into FEM software ANSYS to calculate the vibration displacement mode for all measuring point. The parameters obtained from vibration displacement modes are input to BCA to find the optimal sensor placement to get as much as information using as less as sensors. The results can be used to improve the machine design, management and operations.
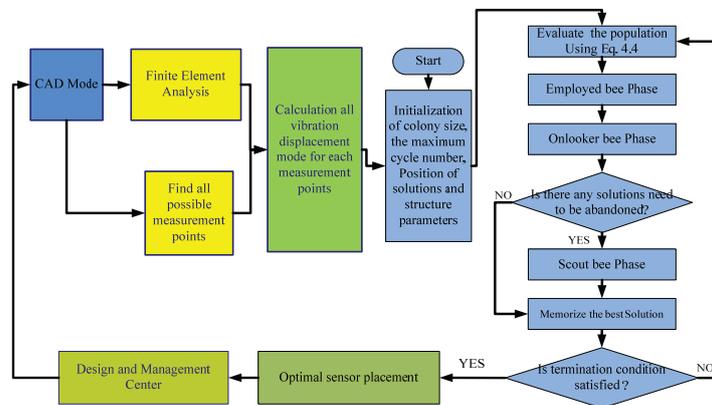
Fig. 4.17 Structure of BCA Application in Sensor Placement Optimization

4.6.3.2   Case Study and Its Results

The object of this case study is the same as Section 4.6.2.2 and the finite element model analysis is the same as well. Therefore, the 3D model and the vibration mode shapes of blower are shown in Fig. 4.11 and Fig. 4.12, and the parameters of the blower are the same as shown in Table 4.7 to Table 4.11. All these parameters are input to the BCA to find the optimal sensor placement. For the BCA algorithm, the colony size is set to 10, the maximum cycle number is set to 50, and the number of trial to improve a solution is set to 20.

The final results are the same as PSO application in Section 4.6.2.2 in Table 4.12 to Table 4.15, and the explanations are also the same as that. In order to compare to the PSO method, Fig. 4.18 to Fig. 4.21 show the the fitness value changing of iteration of BCA in total, X direction, Y direction and Z direction displacement mode for 5 sensors ($n=5$) installing on the blower. From these four figures, the optimal can be found within 10 iterations of BCA and the convergence is faster than PSO compared to Fig. 4.13 to Fig. 4.16.

Fig. 4.18 Fitness Changes with Change of Iteration BCA ($n = 5$) for Total Displacement Mode
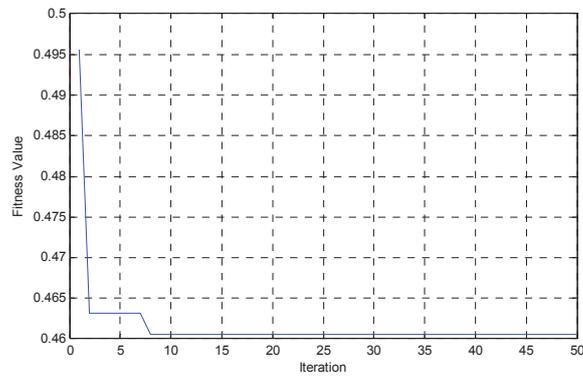
Fig. 4.19 Fitness Changes with Change of Iteration BCA ($n = 5$) for X Direction Displacement Mode
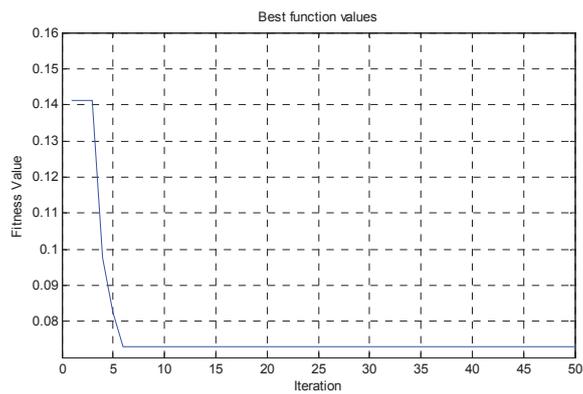
Fig. 4.20 Fitness Changes with Change of Iteration BCA ($n = 5$) for Y Direction Displacement Mode

Fig. 4.21 Fitness Changes with Change of Iteration BCA ($n = 5$) for Z Direction
Displacement Mode

4.7 Summary

This Chapter introduced sensor classification scheme for categorizing and list some criteria to categorize sensors. Most of sensors are very mature in the market. When a machine needs to be monitored, the properties of signals and the parameters of sensors can be firstly determined, and then the suitable sensors can be found from the market. The more important thing in this Chapter is to define a sensor placement optimization problem which is a NP problem, and introduce two Swarm Intelligence algorithms: PSO and BCA to solve this problem. The Swarm Intelligence algorithms are very good at solving the NP problems, and thus, they are suitable to solve the sensor placement optimization problems. Finally, a case study is described in this Chapter which shows that both BCA and PSO can find the optimal sensor placement accurately and fast.

When a machine needs to be monitored, one always wants to use as few as possible sensors to obtain as much as possible information of the machine. To find the optimal sensor placement could be a basis of condition monitoring of manufacturing machines which can reduce the number of sensors used and thus reduce the cost.

Chapter 4: Sensor Classification and Sensor Placement Optimization

5　Signal Preprocessing and Feature Extraction

5.1　Introduction

The main challenge of Condition-based Maintenance is that how to find the relations between the collected data/signals and the conditions of machines. For a complex machine, there could be many sensor are installed for monitoring its condition and thus too many signals and information are collected. However, the collected data cannot indicate the machine condition automatically, and sometimes it is very difficult to get the real machine condition because the mass of signals. Data are rarely useful or usable in their row form, because they may contain too much noise or too weak, and sometimes only because they are too large. Consider, for example, vibration data sampled at 100 kHz. Such large amounts of data are unmanageable unless they are processed and reduced to a form that can be manipulated easily by fault diagnostic and prognostic algorithms. The objective in processing the raw sensor data is to reflect the true and correct information of machine from the signals.

Generally, there are three steps of the raw sensor signal processing: signal preprocessing, feature extraction and feature selection. The aim of signal preprocessing is to improve the general quality of the signal, or in other words, improving the signal-to-noise ratio, for more accurate analysis and measurement, which eventually may facilitate the efficient extraction of useful information, that is, the indicators of the condition of a failing component or subsystem. The tools of preprocessing include filtering, amplification, data compression, data validation, and de-noising. The aim of feature extraction is to extract features or indicators from the preprocessed data that are characteristic of an incipient failure or fault. The main aim of feature selection is to determine a minimal feature subset from a problem domain while retaining a suitably high accuracy in representing the original features. Table 5.1 shows the techniques of these three phases. This Chapter will introduce some of these techniques which are used in IFDPS.

Table 5.1 Vjg'Ogyjqfu'qh'Ukipcn'Rtgrtqeguu."Hgcwtg'Gzvtcevkqp"cpf "Hgcwtg'Ugngevkqp

Signal Preprocessing	Feature Extraction			Feature Selection
	Time Domain	Frequency Domain	Time-Frequency Domain	
Filter, Amplification, Signal Conditioning, Extracting Weak Signals, De-noising, Vibration Signal Compression, etc.	Mean, RMS, Shape factor, Skewness, Kurtosis, Crest factor, Entropy Error, Entropy estimation, etc.	Continues Fourier Transform (CFD), Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT), etc.	Short Time Fourier Transformation (STFT), Wavelet Transform (WT), Wavelet Packet (WP), etc.	Principal Component analysis, Support Vector Machine, Boosting Tree Algorithm, etc.

5.2   Signal Preprocessing

There are many methods for signal preprocessing as shown in Table 5.1. As mentioned above, the main aims of signal pre-processing are to improve the signal-to noise ratio, enhance the signal characteristics, and facilitate the efficient extraction of useful information from the signals. The electrical signals generated by sensors are often not adequate for useful information extraction because they may be very nosy, of low amplitude, biased and dependent on secondary parameters such as temperature and humidity. What's more, the quantities of interested parameters may be not able to be measured directly but can only measure their related quantities. Therefore, signal conditioning is required which can be performed with hardware and/or software which can include: amplification, filtering, converting, range matching, isolation and any other processes required to make sensor output suitable for processing after conditioning [Gutierrez-Osuna et al., 2003]. Denoising techniques aim at eliminating noise from measured data while trying to preserve the important signal features (such as texture and edges) as much as possible[Ramani et al., 2008]. It is very important step to enhancing data reliability and improving the accuracy of signal analysis methods. Wavelet based denoising methods have been successfully applied for signal analysis to improve the signal-to-noise ratio[Benouaret et al., 2012; Patil & Chavan, 2012]. Soft-thresholding [Donoho, 1995] and wavelet-shrinkage denoising [Zheng et al., 2000] are two popular denoising methods. There are still some other denoising techniques: adaptive threshold denoising for fault detection in power systems [Yang & Liao, 2001], acoustic emission signal denoising for fatigue cracks detection in rotor heads [Menon et al., 2000], denoising using modulus maxima algorithm for structure fault detection in fighter aircraft [Hu et al., 2000], signal decomposition technique (wavelets, wavelet packets and matching pursuit method) based denoising methods for improving signal-to-noise ratio of knee-joint vibration signals [Krishnan & Rangayyan, 2000], and reducing background noise level using The second order displaced power spectral density (SDPSD) function for localized defects in roller bearings [Piñeyro et al., 2000]. The amount of data collected from industrial systems tends to be voluminous and, in most cases, difficult to manage because the increasing of sensors and sample rates. Therefore, data compression is very important for condition monitoring system especially for those implemented online or Internet-based systems. Transient analysis is mostly used to compress data because it can significantly improve the performance of sensor arrays with careful instrument design and sampling procedures which are: improving selectivity, reducing acquisition time and increasing sensor lifetime. There are main three classes of transient analysis methods: Sub-sampling method [Gutierrez-Osuna et al., 1999; Kermani et al., 1998; Roussel et al., 1998; White et al., 1996], parameter-extraction method [Eklöv et al., 1997; Gibson et al., 1997; Llobet et al., 1997; D. M. Wilson & DeWeerth, 1995], and system-identification method [Eklöv et al., 1997; Gutierrez-Osuna et al., 1999; Nakamoto et al., 2000]. The signal preprocessing techniques for condition monitoring are mature, and more techniques and more detail can be referred in the literatures [Gutierrez-Osuna et al., 2003; Marwala, 2012; Vachtsevanos et al., 2006].

5.3 Feature Extraction

Feature and condition indicator extraction and selection play curial roles in condition monitoring especially for accuracy and reliability of fault diagnosis and prognosis. The function of condition monitoring mainly depends on a set of features extracted from sensor data that can distinguish between fault categories of interest, and detect and isolate a specific fault at early initiation stages. These features should be fairly insensitive to noise and within fault class variations. It should beware that not losing useful information in feature extraction stage. For time series signals, such as vibration signals, voltage signals and current signals, the features can be extracted from four domains: time domain, frequency domain, time-frequency domain and wavelet domain.

5.3.1 Feature Extraction in Time Domain

Features in time domain is very traditional methods for extraction features, but is very widely used in fault diagnosis and prognosis which mainly computer the statistical parameters from signals. The following features are some of these statistical parameters [Vachtsevanos et al., 2006; Wang & Zhang, 2010]:

Peak value,

$$P_v = \frac{1}{2}\left[\max\left(x_i\right) - \min\left(x_i\right)\right] \tag{5.1}$$

where x_i $(i = 1, 2\cdots, N)$ is the amplitude at sampling point i and N is the number of sampling points.

RMS value,

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_i\right)^2} \tag{5.2}$$

Standard deviation,

$$SD = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_i - \bar{x}\right)^2} \tag{5.3}$$

Kurtosis value,

$$K_v = \frac{\frac{1}{N}\sum_{i=1}^{N}\left(x_i - \bar{x}\right)^4}{\left(RMS\,Value\right)^4} \tag{5.4}$$

Crest factor,

$$Crf = \frac{Peak\,Value}{RMS\,Value} \tag{5.5}$$

91

Clearance factor,

$$Clf = \frac{Peak\,Value}{\left(\frac{1}{N}\sum_{i=1}^{N}\sqrt{|x_i|}\right)^2} \tag{5.6}$$

Impulse factor,

$$Imf = \frac{Peak\,Value}{\frac{1}{N}\sum_{i=1}^{N}|x_i|} \tag{5.7}$$

Shape factor,

$$Shf = \frac{RMS\,Value}{\frac{1}{N}\sum_{i=1}^{N}|x_i|} \tag{5.8}$$

Weibull negative log-likelihood value was used recently for feature extraction from vibration signals. The Weibull negative log-likelihood value (Wnl) and the normal negative log-likelihood value (Nnl) of the time domain vibration signals are used as input features along with the other features defined above in this study. The negative log-likelihood function is defined as:

$$-\Lambda = \sum_{i=1}^{N} Log\left[f\left(x_i, \theta_1, \theta_2\right)\right]$$

where $f(x_i, \theta_1, \theta_2)$ is the probability density function (pdf). For Weibull negative log-likelihood function and normal negative log-likelihood function, the $pdfs$ are computed as follows:

Weibull pdf :

$$f\left(x_i, \beta, \eta\right) = \beta\eta^{-\beta}\,|x_i|^{\beta-1}\,exp\left[-\left(\frac{|x_i|}{\eta}\right)^{\beta}\right] \tag{5.9}$$

Where β and η are the shape and the scale parameters respectively.

Normal pdf :

$$f\left(x_i, \mu, \sigma\right) = \frac{1}{\sigma\sqrt{2\pi}}\,exp\left\{\frac{\left[-\left(x_i - \mu\right)^2\right]}{2\sigma^2}\right\} \tag{5.10}$$

Where μ and σ are the mean and the standard deviation respectively.

There are still three time domain parameters, i.e. Activity, mobility and complexity, can be used for feature extraction [Hjorth, 1970; Xinyang Li et al., 2011]:

$$Activity = \mathrm{var}(x(t)) \tag{5.11}$$

$$Mobility = \sqrt{\frac{Activity(\dot{x})}{Activity(x(t))}} \qquad (5.12)$$

$$Complexity = \sqrt{\frac{Mobility(\dot{x})}{Mobility(x(t))}} \qquad (5.13)$$

The above three parameters are often referred as Hjorth parameters and have been widely applied [Cecchin et al., 2010; Obermaier et al., 2001]. There are also other parameters can be used for feature extraction: Time-Domain Morphology and Gradient [Mazomenos et al., 2012], Correlation, Covariance and Convolution [Vachtsevanos et al., 2006]. For details of these techniques, the corresponding literatures can be found in above mentioned references.

5.3.2   Feature Extraction in Frequency Domain

In many situations, especially with rotating machinery [Eisenmann & Eisenmann, 1998], the frequency domain data of measured time signals, such as vibration, carries a great deal of information useful in diagnosis [Vachtsevanos et al., 2006]. Frequency domain methods are difficult to use in that they contain more information than is necessary for fault detection. There is no method to select the frequency bandwidth of interest, and they are usually noisy in anti-resonance regions [Ewins, 1995; Marwala, 2012]. However, frequency domain methods still have some advantages: 1) the measured data comprise the effects of out-of-frequency-bandwidth modes; 2) one measurement offers ample data; 3) modal analysis is not necessary and consequently modal identification errors are circumvented; and, 4) frequency domain data are appropriate to structures with high damping and modal density [Marwala, 2012]. A fault on a component of a machine might be indicated by the base rotational frequency, two times of this frequency or n times of this frequency. This principle can be used in fault diagnosis and prognosis. The main algorithm of frequency of a signal is Fourier Transform which is introduces in this section.

The Fast Fourier Transform (FFT) is basically a computationally efficient technique for calculating the Fourier transform which exploits the symmetrical nature of the Fourier transform [Marwala, 2012]. The theory of FFT is retrieved here from the literature [Ewins, 1995; Marwala, 2012]. If the FFT is applied to the response, the following expression is obtained:

$$X(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} x(t) e^{-i\omega t} dt \qquad (5.14)$$

Similarly, the transformed excitation can be written as:

$$F(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \qquad (5.15)$$

Frequency Response Function (FRF) $\alpha_{ij}(w)$ of the response at position i to the excitation at j is the ratio of the Fourier transform of the response to the transform of the excitation:

$$\alpha_{ij}(\omega) = \frac{X_i(\omega)}{F_j(\omega)} \tag{5.16}$$

The FRF matrix is related to the spatial properties by the following expression:

$$[\alpha(\omega)] = \left[-\omega^2 [M] + j\omega [C] + [K] \right]^{-1} \tag{5.17}$$

Here α is the frequency response function, ω is the frequency, $[M]$ is the mass matrix, $[C]$ is the damping matrix, $[K]$ is the stiffness matrix and $j = \sqrt{-1}$. The above transform is for the continue signals. For the discrete signals, the frequency response function can be expressed as:

$$X(k) = \sum_{n=1}^{N} x(n) e^{-j 2\pi (k-1)(n-1)/N} \quad k = 1, 2, \cdots, N \tag{5.18}$$

where N is the number of time series $x(n)$. Fig. 5.1 shows a vibration signal with the sample rate 4096 in one second while Fig. 5.2 shows the corresponding frequency response function. From Fig. 5.2, the base frequency of this vibration signal is 46 Hz, the second order frequency is 92 Hz and the third order frequency is 138 Hz. There are also some other features can be extracted from the FRF figure to find the characteristics of the signals for fault diagnosis and prognosis. For example, power spectral density (PSD):

$$\psi_x = \frac{1}{N} X(k) X^{'}(k) = \frac{1}{N} \left| X(k) \right|^2 \tag{5.19}$$

is for fault diagnosis and prognosis which is easier to see details in the frequency response than using $X(k)$ [Vachtsevanos et al., 2006].
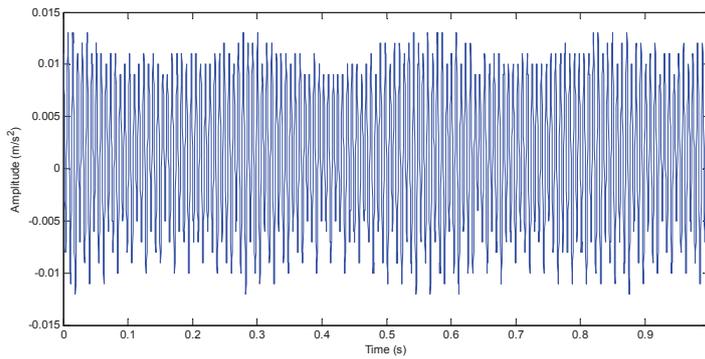
Fig. 5.1 Vibration Signal in Time Domain
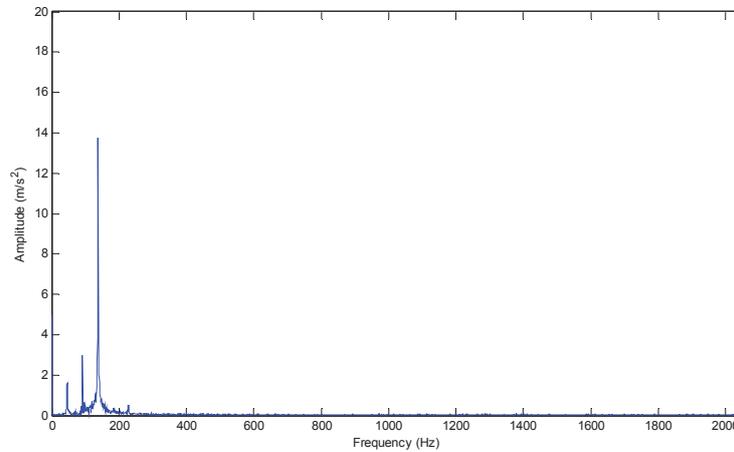
Fig. 5.2 Frequency Response Function of Vibration Signal in Fig. 5.1

5.3.3 Feature Extraction in Time-Frequency Domain

Although FFT based methods are powerful tools for fault diagnosis and prognosis, they are not suitable for non-stationary signals. For analysis in the time-frequency domain, the Wigner-Ville distribution (WVD) and the short time Fourier transform (STFT) are the most popular methods for non-stationary signal analysis. However, WVD suffers from interference terms appearing in the decomposition, and STFT cannot provide good time and frequency resolution simultaneously because it uses constant resolution at all frequencies. Moreover, no orthogonal bases exist for STFT that can be used to implement a fast and effective STFT algorithm [Okamura, 2011; Vachtsevanos et al., 2006]. The methods for time-frequency analysis are compared in Table 5.2 [Vachtsevanos et al., 2006]. This section mainly introduces Wavelet transform for time-frequency analysis and feature extraction.

Wavelet transform is a time-frequency decomposition of a signal into a set of "wavelet" basic function. Wavelet analysis has proved its great capabilities in decomposing, denoising, and signal analysis which made the analysis of non-stationary signals achievable as well as detecting transient feature components as other methods were inept to perform since wavelet can concurrently impart time and frequency structures. Wavelet Transform (WT) gives good time and poor frequency resolution at high frequencies, and good frequency and poor time resolution at low frequencies. Analysis with wavelets involves with breaking up a signal into shifted and scaled versions of the original (or mother) wavelet, i.e., one high frequency term from each level and one low frequency residual from the last level of decomposition. There are three categories of this transformation: Continuous Wavelet Transform (CWT), Discrete Wavelet Transform (DWT) and WPD.

Table 5.2 Comparing Different Time-Frequency Analysis Methods

Methods	Resolution	Interference	Speed
Wavelet Transform (WT)	Good frequency and low time resolution for low-frequency components; low frequency and high time resolution for high-frequency components	None	Fast
Short time Fourier transform (STFT)	Depending on window function used, either good time or good frequency resolution	None	Slower than CWT
Wigner-Ville distribution (WVD)	Good time and frequency resolution	Severe	Slower than STFT
Choi-Williams distribution (CWD)	Good time and frequency resolution	Less than WVD	Very slow
Cone-shaped distribution (CSD)	Good time and frequency resolution	Less than WVD	Very slow

5.3.3.1 Continuous Wavelet Transform (CWT)

A CWT is used to divide a continuous-time function into wavelets. Unlike Fourier transform, the continuous wavelet transform possesses the ability to construct a time-frequency representation of a signal that offers very good time and frequency localization [Soman & Ramachandran, 2005]. The continuous wavelet transform of a time function $x(t)$ is given by following equation:

$$CT(a,b) = \int_{-\infty}^{\infty} x(t)\psi_{(a,b)}^{*}(t)dt \tag{5.20}$$

where $\psi_{(a,b)}^{*}(t)$ is a continuous function in both the time domain and the frequency domain called the mother wavelet and * represents operation of complex conjugate. $\psi_{(a,b)}^{*}(t)$ can be expressed as:

$$\psi_{(a,b)}^{*}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \quad where \quad a,b \in R, a \neq 0 \tag{5.21}$$

The main purpose of the mother wavelet is to provide a source function to generate the daughter wavelets which are simply the translated and scaled versions of the mother wavelet. As seen in Eq. (5.21), the transform signal $CT(a,b)$ is defined on $a-b$ plane, which a and b are used to adjust the frequency and the time location of the wavelet in Eq. (5.21). A small a produces a high-frequency wavelet when high frequency resolution is needed and the reverse is also true. The WT's superior time-localization properties stem from the finite support of the analysis wavelet: as b increases, the analysis wavelet transverses the length of the input signal, and a increases or decreases in response to changes in the signal's local time and frequency content. Finite support implies that the effect of each term in the wavelet

representation is purely localized. This sets the WT apart from the Fourier Transform, where the effects of adding higher frequency sine waves are spread throughout the frequency axis.

5.3.3.2   Discrete Wavelet Transform (DWT)

In numerical analysis and functional analysis, DWT is a wavelet transform for which the wavelet $\psi_{(a,b)}$ is discretely sampled. As with CWT, a key advantage it has over Fourier transforms is temporal resolution: it captures both frequency and location information (location in time). Usually, the DWT can be derived from discretization of CWT. The most common discretization is dyadic method:

$$DT(a,b) = \int_{-\infty}^{\infty} x(t)\psi_{(j,k)}^{*}(t)dt \tag{5.22}$$

$$\psi_{(j,k)}^{*}(t) = \frac{1}{\sqrt{2^j}}\psi\left(\frac{t-2^j k}{2^j}\right) \tag{5.23}$$

where a and b are replaced by 2^j and $2^j k$ respectively [Daubechies, 1988; Mallat, 1989]. An efficient way to implement this scheme using filters was developed by Mallat [1989]. The original signal $x(t)$ passes through two complementary filters and emerges as low frequency called approximations $A_j(t)$ and high frequency called details $D_i(t)$ as shown in Eq. (5.24). The decomposition process can be iterated, with successive approximations being decomposed in turn, such that a signal can be broken down into many lower-resolution components.

$$f(t) = \sum_{i=1}^{i=j} D_i(t) + A_j(t) \tag{5.24}$$

Where $D_i(t)$ denotes the wavelet detail and $A_i(t)$ stands for the wavelet approximation at the jth level. DWT analysis is more efficient still with the identical accuracy [Goumas et al., 2001].

As discussed above, DWT can decompose the signal into two parts: low-frequency A_1 and high frequency D_1. In the process of decomposition, the lost information belonging to the low frequency part is captured by the high frequency part. In the next level of decomposition, this method will also decompose A_1 into two parts: low-frequency A_2 and high frequency D_2. The lost information belonging to low frequency A_2 is capture by the high-frequency D_2, and thus, a deeper level decomposition can be done. The 3-layer structure of signal based on DWT is shown in Fig. 5.3 in which only approximation version is decomposed.
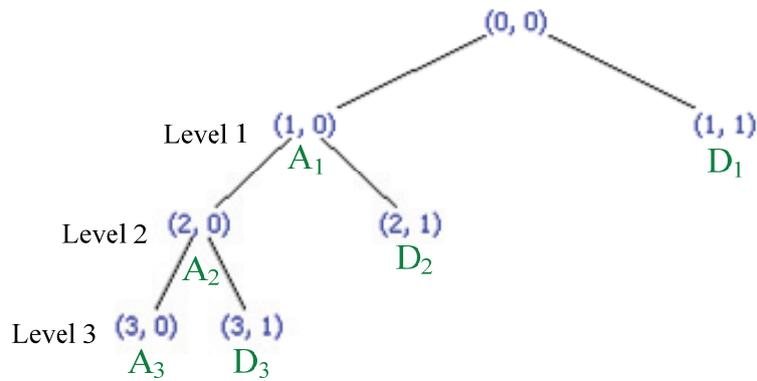
Fig. 5.3 3-layer Signal Decomposition by Discrete Wavelet Transform

For the case of signal with the maximum frequency 2048 Hz, D_1, D_2, D_3 and A_3 represent the frequency 1024~2048 Hz, 512~1024 Hz, 128~512 Hz and 0~128 Hz respectively. The decomposed signals by DWT from vibration signal (Fig. 5.1) are shown in Fig. 5.4.
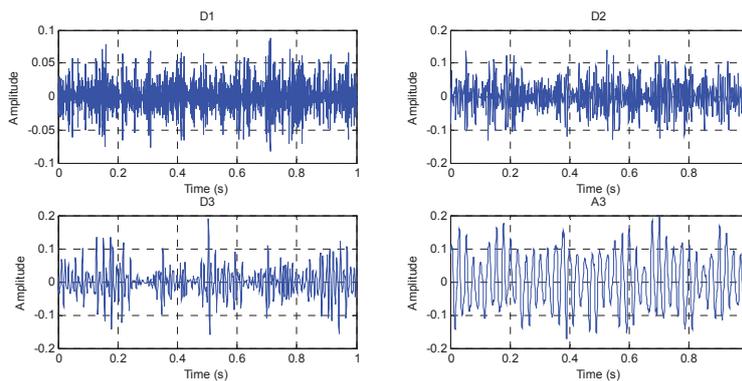
Fig. 5.4 Decomposed Signals by DWT

5.3.3.3   Wavelet Packet Decomposition

The structure of wavelet packet Decomposition (WPD) is similar to DWT. The typical 3 layers signal decomposition by WPD is shown in Fig. 5.5. Both have the framework of multi-resolution analysis. The main difference in the two techniques is the WPT can simultaneously break up detail (D_i) and approximation (A_i) versions [Li et al., 2003] while DWT only breaks up as an approximation version. Therefore, the WPD have the same frequency bandwidths in each resolution and DWT does not have this property. The mode of decomposition does not increase or lose the information within the original signals. Therefore, the signal with great quantity of middle and high frequency signals can offer superior time-frequency analysis. The WPT suits signal processing, especially non-stationary signals

because the same frequency bandwidths can provide good resolution regardless of high and low frequencies.
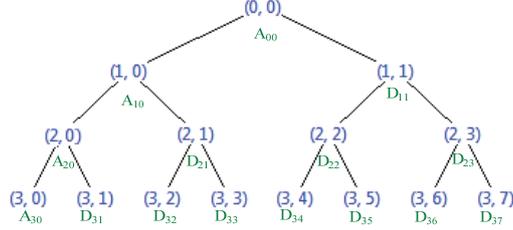
Fig. 5.5 Decomposed Signals by WPD

Wavelet packets consist of a set of linearly combined usual wavelet functions which inherit the attributes of their corresponding wavelet functions such as orthonormality and time–frequency localization. A wavelet packet is a function with three indices of integers, i, j and k which are the modulation, scale and translation parameters, respectively [Shinde, 2004],

$$\psi_{j,k}^{i}(t) = 2^{\frac{j}{2}}\psi^{j}\left(2^{i}t - k\right) \quad i = 1,2,3,\cdots \tag{5.25}$$

The wavelet functions ψ^{j} can be obtained from the following recursive relations:

$$\psi^{2j}(t) = \sqrt{2}\sum_{-\infty}^{\infty}h(k)\psi^{i}(2t - k) \tag{5.26}$$

$$\psi^{2j+1}(t) = \sqrt{2}\sum_{-\infty}^{\infty}g(k)\psi^{i}(2t - k) \tag{5.27}$$

The original signal $f(t)$ after j level of decomposition can be stated as:

$$f(t) = \sum_{i=1}^{2j}f_{j}^{i}(t), \tag{5.28}$$

while the wavelet packet component $f_{j}^{i}(t)$ can be stated by a linear combination of wavelet packet functions $\psi_{j,k}^{i}(t)$ in such a way:

$$f_{j,k}^{i}(t) = \sum_{-\infty}^{\infty}c_{j,k}^{i}(t)\psi_{j,k}^{i}(t) \tag{5.29}$$

where the wavelet packet coefficients $c_{j,k}^{i}(t)$ can be obtained from,

$$c_{j,k}^{i}(t) = \int_{-\infty}^{\infty}f(t)\psi_{j,k}^{i}(t)dt \tag{5.30}$$

providing that the wavelet packet functions are orthogonal:

$$\psi_{j,k}^{m}(t)\psi_{j,k}^{n}(t) = 0 \quad if\ m \neq n \tag{5.31}$$

Since different types of wavelet functions have different time–frequency structures, a function with a time–frequency structure matching superlatively that of the transient component must be used to effectively detect the transient component. In general, the smooth wavelets are better for regular, stationary, periodic data and the compact wavelets are better for non-stationary, transient data [Staszewski, 1997]. As a result, Daubechies 4 (Db4) wavelet function has been chosen for this case after several trials as it is often chosen arbitrarily for signal analysis and synthesis by experiments in many papers in the field (e.g., [Vafaei & Rahnejat, 2003]) there is no computational logic behind the selection of Daubechies order.

The dilation equations may be used to generate orthogonal wavelets. The scaling function $\varphi(t)$ is a dilated (horizontally expanded) version of $\varphi(2t)$. The dilation equation in general has the form [Vachtsevanos et al., 2006]:

$$\varphi(t) = c_0\varphi(2t) + c_1\varphi(2t-1) + c_2\varphi(2t-2) + c_3\varphi(2t-3) \tag{5.32}$$

The Daubechies D4 wavelet coefficients have values:

$$c_0 = \frac{(1+\sqrt{3})}{4\sqrt{2}}, c_1 = \frac{(3+\sqrt{3})}{4\sqrt{2}} \quad c_2 = \frac{(3-\sqrt{3})}{4\sqrt{2}} \quad c_3 = \frac{(1-\sqrt{3})}{4\sqrt{2}} \tag{5.33}$$

Thus, a particular family of wavelets is specified by a particular set of numbers, called the wavelet filter coefficients. The above set of numbers c_0, c_1, c_2 and c_3 are called the Db4 wavelet filter coefficients.

In general, for an even M number of wavelet filter coefficients c_k $(k = 1, 2, \cdots, M-1)$, the scaling function is defined by:

$$\phi(t) = \sum_{k=1}^{M-1} c_k \phi(2t-k) \tag{5.34}$$

and the corresponding wavelet is derived as:

$$w(t) = \sum_{k=1}^{M-1} (-1)^k c_k \phi(2t+k-M+1) \tag{5.35}$$

It is observed that the scaling function has a low-pass form, whereas the wavelet function has a high-pass form. Thus, the wavelet function is essentially responsible for extracting the detail (high-frequency components) of the original signal.

5.3.3.4 Wavelet-based Features

There are several types of features can be extracted from wavelet-based methods, which can be categorized roughly into wavelet coefficients-based, wavelet energy-based, singularity-based, and wavelet function-based methods. All these features are retrieved from the literature [Vachtsevanos et al., 2006].

The wavelet coefficients c_k can be used to extracted features for fault detection and prognosis. The standard deviations of the coefficients are good features for feature extraction. For example, for each of the signal, wavelet packet is applied up to the fourth level thus giving 16 signal coefficient sets. The wavelet packet coefficients

and their corresponding standard deviations for medium-worn fault sampled at second configuration are shown in Fig. 5.6. At the end, the standard deviation of wavelet packet coefficients of pre-processed signals is used as features for fault diagnosis and prognosis.
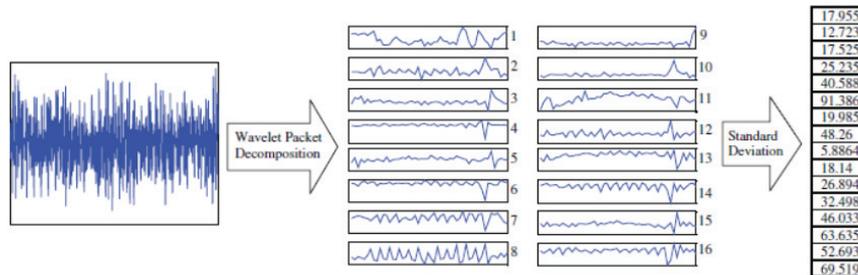
Fig. 5.6. Wavelet Packet Coefficients and Their Relevant Standard Deviation

Energy-based features: The most important advantage of wavelets versus other methods is their ability to provide an image visualization of the energy of a signal, making it easier to compare two signals and identify abnormalities or anomalies in the faulty signal. Based on these observations, suitable features can be designed either using image-processing techniques or simply exploiting the energy values directly. For example, in a study of a helicopter 's planetary gear system [Saxena & Vachtsevanos, 2005], it was observed that the energy distribution among the five planets became asymmetric as a fault (crack) appeared on the gear plate. This observation led to a feature based on the increasing variance among energy values associated with the five planets as time evolves. Similarly, other features can be designed that characterize a visible property in a numerical form.

Singularity-Based Features: Singularities can be discerned from wavelet phase maps and can be used as features for detecting discontinuities and impulses in a signal. Singularity exponents, extracted from the envelope of vibration signals, have been used to diagnose breakers' faults [Yang & Liao, 2001]. Other applications reported in the technical literature relate to detection of shaft center orbits in rotating mechanical systems [Peng et al., 2002].

Some of the most interesting applications of wavelet-based features include fault detection in gearboxes [Chen & Wang, 2002; Hambaba & Huff, 2000; Yen & Lin, 1999, 2000; Zheng et al., 2002], Fault detection in rolling element bearings [Altmann & Mathew, 2001; Shibata et al., 2000], and fault diagnosis in analog circuits[Aminian, 2001], among others.

Wavelet energy-based features often cannot detect early faults because slight changes in the signal result in small energy changes. Coefficient-based features are more suitable for early fault detection. Similarly, singularity- based methods are not very robust to noise in the signal, and denoising must be carried out before calculating any such features [Vachtsevanos et al., 2006].

5.4   Feature Selection

There might be too many features extracted from the signals and collected from sensors which make extraction of useful and understandable information from these features become difficult. Therefore the dimensionality of the features needs to be reduced. Feature selection is primarily performed to select relevant and informative features which can reduce the dimensionality of features effectively. It can have the other motivations, including [Guyon & Elisseef, 2006]:

1) *General data reduction*, to limit storage requirements and increase algorithm speed;
2) *Feature set reduction*, to save resources in the next round of data collection or during utilization;
3) *Performance improvement*, to gain in predictive accuracy;
4) *Data understanding*, to gain knowledge about the process that generated the data or simply visualize the data

Many data mining algorithms can be used to carry out feature selection: neural network ensemble (NNE) [Hansen & Salamon, 1990], neural network (NN) [Liu, 2001; Siegelmann & Sontag, 1994], boosting regression tree (BRT) [Friedman, 2001, 2002; Smola & Scholkopf, 2003], support vector machine (SVM) [Schölkopf et al., 1999; Steinwart & Christmann, 2008], random forest with regression (RF) [Breiman, 2001], standard classification and regression tree (CART) [Speybroeck, 2012], k nearest neighbour neural network (kNN) [Shakhnarovich et al., 2005], wrapper approach integrated with the genetic or the best-first search algorithm [Espinosa et al., 2005; Tan et al., 2006] and principal component analysis (PCA) [Jolliffe, 2002]. All these algorithms are widely used for feature selection. Zhang and Kusiak applied all these algorithm for parameter selection in wind turbine condition monitoring and compared these algorithm [Kusiak & Verma, 2011; Kusiak & Zhang, 2010; Zhang & Kusiak, 2012].

PCA is an unsupervised learning approach for dimensionality reduction that uses correlation coefficients of the parameters to combine and transform them into a reduced dimensional space [Miranda et al., 2008]. The concept of Principal Component Analysis (PCA) was invented in 1901 by Karl Pearson [Pearson, 1901]. It is a mathematical procedure that uses an orthogonal transform to convert a set of observations of possibly correlated variables into a set of values of uncorrelated variables called principal components. This transform is defined in such a way that the first principal component has as high a variance as possible, which means accounting for as much of the variability in the data as possible, and each succeeding component in turn has the highest variance possible under the constraint that it be uncorrelated with the preceding components. It can reduce data dimension and eliminate multi-collinearity. Currently, PCA mostly used to reduce the dimension while maintain the main information in data mining analysis and making models. This section mainly introduces the principle of PCA.

PCA computes a new set of uncorrelated multivariate (vector) samples by a transform of coordinate rotation from original correlated multivariate samples. A

matrix composed by n rows which means n samples are collected and m columns which represent the number of features are expressed as bellowing:

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{bmatrix} \tag{5.36}$$

PCA can obtain a new set of vector according to the following steps:

1) Calculate the correlation coefficient matrix

The correlation coefficient matrix is calculated according to the following equation:

$$R = Cor(i,j) = \frac{(n-1)\cdot Cov(i,j)}{\sqrt{\sum_{k=1}^{n}\left(x_i(k)-\mu_i\right)^2 \sum_{k=1}^{n}\left(x_j(k)-\mu_j\right)^2}} = \left(r_{ij}\right)_{m\times n} \tag{5.37}$$

where n is the number of samples. The dimension of the correlation matrix R is $m \times n$. $Cov(i,j)$ means the covariance which matrix is $m \times n$ can be expressed as:

$$Cov(i,j) = \frac{1}{(n-1)}(x_i - \mu_i)(x_j - \mu_j) \qquad i,j = 1,2,...,m \tag{5.38}$$

where μ_i and μ_j are the averages of the ith and j^{th} rows of matrix X respectively.

2) Calculate the eigenvectors and eigenvalues of the matrix R

The m eigenvalues λ_i which have the constraint as $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$ and their responding eigenvectors V_i are calculated from correlation matrix. λ_i and V_i satisfy the following equation:

$$AV_i = \lambda_i \cdot V_i \qquad i = 1,2,\cdots,m \tag{5.39}$$

where A is a $m \times n$ covariance matrix or correlation matrix and the vector V_i can be expressed as $V_i = [V_{1i}, V_{2i}, \cdots, V_{mi}]$.

3) Generates the new samples

A new set of uncorrelated multivariate (vector) samples are computed according to the following equation:

$$X_{new} = V^T \cdot X \tag{5.40}$$

where X_{new} is the new uncorrelated multivariate (vector) sample, and X is the original correlated multivariate (vector) samples. Both of them are $n \times m$ matrices whose row vectors represent a single channel sample. V is eigenvectors matrix which is also called the weight matrix. Each column of V is one principle component. X_{new} is the principal component scores. Each row of X_{new} is the scores for one principal component. Each λ_i is variance of the scores for one principal component. Most of time, only first several components in X_{new} are selected as principal components according to the variance threshold. The case study of how to

apply PCA to reduce dimensionality of features will be presented in Chapter 6 with the case study of fault diagnosis.

5.5   Summary

This Chapter introduced the techniques of signal preprocessing, feature extraction and feature selection. Signal preprocessing and feature extraction are mainly for time series such as vibration signals and electrical signals. The features can be extracted in time domain, frequency domain and time-frequency domain in which the features extracted based on wavelet transform has special advantages and becomes very popular. The extracted features might be too many to manage for condition monitoring because there are too many sensors and many features can be extracted from one signal. Therefore, dimensionality reduction and feature selection become important for ease of management for these features but not reduce the useful information. All these processes are the preparation for fault diagnosis and prognosis which are crucial parts of condition monitoring.

6 Fault Diagnosis based on Data Mining Techniques

6.1 Introduction

Fault diagnosis has become the subject of numerous investigations over the past two decades. Researchers in many disciplines, such as medicine, engineering, the sciences, business, and finance, been developing methodologies to detect fault (failure) or anomaly conditions, pinpoint or isolate which component or object in a system or process is faulty, and decide on the potential impact of a failing or failed component on the health of the system [Vachtsevanos et al., 2006]. Fault diagnostic algorithms must have the ability to detect system performance, degradation levels, and faults (failures) based on physical property changes through detectable phenomena. Referring the fault diagnosis and condition monitoring, the following concepts need to be defined and distinguished [Vachtsevanos et al., 2006]:

- *Fault diagnosis*. Detecting, isolating, and identifying an impending or incipient failure condition—the affected component (subsystem, system) is still operational even though at a degraded mode.
- *Failure diagnosis*. Detecting, isolating, and identifying a component (subsystem, system) that has ceased to operate.
- *Fault (failure) detection*. An abnormal operating condition is detected and reported.
- *Fault (failure) isolation*. Determining which component (subsystem, system) is failing or has failed.
- *Fault (failure) identification*. Estimating the nature and extent of the fault (failure).

Therefore, the aim of fault diagnosis is to detect abnormal condition of machine before the failure happens, and also identify which component of the machine will become failure. To evaluate the techniques for fault diagnosis of a condition monitoring system, several qualification factors can be used [Vachtsevanos et al., 2006]:

- *Isolability*. A measure of the model's ability to distinguish between certain specific failure modes. Enabling technologies include incidence matrices involving both deterministic (zero-threshold) and statistical (high-threshold) isolability.
- *Sensitivity*. A qualitative measure characteristic of the size of failures. This factor depends on the size of the respective elements in the system's matrices, noise properties, and the time to failure. Filtering typically is used to improve sensitivity, but it is rather difficult to construct a straightforward framework.
- *Robustness*. This factor refers to the model's ability to isolate a failure in the presence of modeling errors. Improvements in robustness rely on algebraic cancelation that desensitizes residuals according to certain modeling errors.

There are many techniques can be used for fault diagnosis. The development of model-based fault diagnosis began in the early of the 1970s [Dirilten, 1972; Hayes, 1971]. This method of fault detection in dynamic systems has been receiving more and more attention over the last two decades [Schubert et al., 2011; Soman et al., 2012; Van den Kerkhof et al., 2012]. It has much to offer in addressing system-based fault diagnosis issues for complex systems [De Kleer & Williams, 1987; Isermann, 2005]. It is used to detect any discrepancy between the system outputs and model outputs. It is assumed that this discrepancy signal is related to a fault. This method is perfect when the mathematical model or physical model is accurate and the system outputs are no noise. However, the same difference signal can respond to model plant mismatches or noise in real measurements, which are erroneously detected as a fault. What's more, sometimes, it is impossible to model nonlinear systems by analytical equations [Mendonqa, 2006]. Therefore, the model-based fault diagnosis techniques are not very good for some cases such as non-linear system which mathematical model is not available.

Case-based Reasoning (CBR) [Aamodt & Plaza, 1994; Reisbec & Schank, 1989] offers a reasoning paradigm that is similar to the way people routinely solve problems which is another method can be used for fault diagnosis. CBR began to be applied in fault diagnosis in 1990s [Grant et al., 1996; Patterson & Hughes, 1997], and become very popular afterwards [Fu et al., 2011; Tsai, 2009]. The cyclic process of CBR can be described as following. When a new problem happens, one or more similar cases are retrieved from the case base. A solution suggested by the matching cases then is reused and tested for success. Unless the retrieved case is a close match, the solution probably will have to be revised, producing a new case that can be retained. Currently, this cycle rarely occurs without human intervention and most CBR systems are used mainly as case retrieval and reuse systems [Watson & Marir, 2009]. The CBR designer is faced with two major challenges: coding of cases to be stored into the case library or case base and adaptation, that is, how to reason about new cases so as to maximize the chances of success while minimizing the uncertainty about the outcomes or actions. Additional issues may relate to the types of information to be coded in a case, the type of database to be used, and questions relating to the programming language to be adopted [Vachtsevanos et al., 2006].

It is obviously that the Model-based fault diagnosis techniques can detect and identify any faults even for unanticipated ones. But these methods need accurate mathematical model or physical model which is usually not available for complex machines. Therefore, data-driven methods could be better solution for fault diagnosis when the model is unavailable and the CBR does not work well.

In contrast to model-based approaches, data-driven fault diagnostic techniques rely primarily on process and data which are from sensors specifically designed to respond to fault signals, to model a relationship between fault features or fault characteristic indicators and fault classes. Such "models'' may be cast as expert systems or artificial neural networks or a combination of these computational intelligence tools. They require a sufficient database (both baseline and fault conditions) to train and validate such diagnostic algorithms before their final online implementation. They lack the insight that model-based techniques provide

regarding the physics of failure mechanisms, but they do not require accurate dynamic models of the physical system under study. They respond only to anticipate fault conditions that have been identified and prioritized in advance in terms of their severity and frequency of occurrence, whereas model-based methods may be deployed to detect even unanticipated faults because they rely on a discrepancy or residual between the actual system and model outputs [Vachtsevanos et al., 2006]. In the past few years, many Computational Intelligence (CI) techniques have been applied as tools for fault diagnosis [Sun et al., 2012; Wang, 1996]. This Chapter mainly introduces data mining techniques especially of CI techniques application in fault diagnosis. Some case studies will be used to show how these techniques work in fault diagnosis.

6.2 Fault Diagnosis based on SBP

The pattern classification theory has become a key factor in fault diagnosis. Some classification methods for equipment performance monitoring use the relationship between the type of fault and a set of patterns which is extract from the collected signals without establishing explicit models. Currently, ANN is one of the most popular methods in this domain. The principle of ANN has been introduced in Section 3.2 which included Back-propagation (BP), Self-organization Mapping (SOM). The application of artificial neural network models lies in the fact that they can be used to infer a function from observations. This is particularly useful in applications where the complexity of the data or task makes the design of such a function by hand impractical. This attribution is very nontrivial in diagnostic problems. BP neural network is a main type of ANN used to solve fault diagnosis and prognosis problems.

ANN can deal with complex non-linear problem without sophisticated and specialized knowledge of the real systems. It is an effective classification techniques and low operational response times needed after training. The relationship between the condition of component and the features is not linear but non-linear. BP neural network does not need to know the exact form of analytical function on which the model should be built. This means neither the functional type nor the number and position of the parameters in the model-function need to know. It can deal with multi-input, multi-output, quantitative or qualitative, complex system with very good abilities of data fusion, self-adaptation and parallel processing. Therefore, it is very suitable to select as a method of fault diagnosis.

Fig. 6.1 shows the procedure of fault diagnosis based on BP network. There are mainly three phases of this method. The first phase is training phase to establish an ANN model for a specific type of fault. The training data could be history data or collected data from sensors. The collected raw signals, such as vibration signals and acoustics signals are very hard to be used to train ANN model, and thus need extract features from these signals. The signals of vibration and acoustics may contain noise electrically or mechanically, thus the signals need to be processed to filter out the noise, improve signal-to-noise ratio and amplify the weak signals. Then the extracted features can be used to training ANN to establish the model of

the fault. Once the ANN model is established, it can be used to judge if the machine has fault and identify which component will be failure. This phase called test phase. The data here used to test ANN model must be the same kind of features as the training data. Thus, the techniques to be used for signal processing and feature extraction must be same as that of the training data. The last phase is maintenance decision making based on the test results of ANN model. This phase will be complete in Chapter 8.
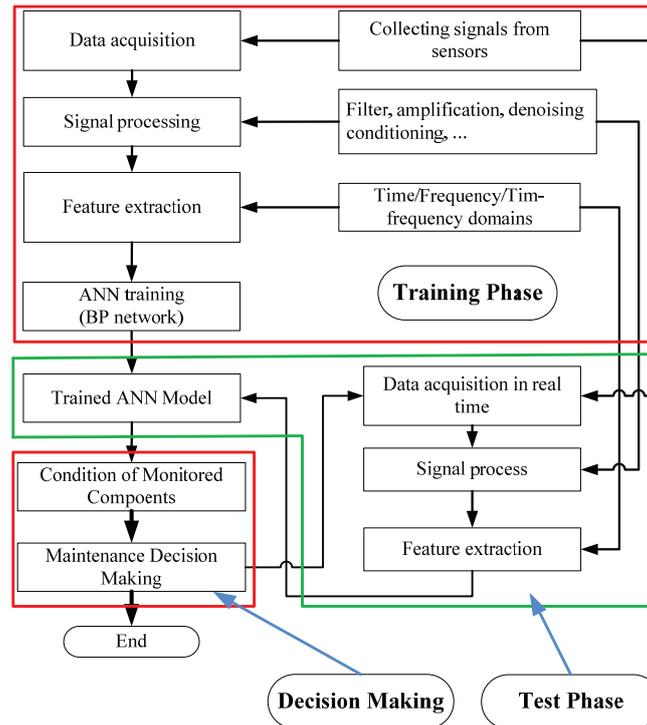
Fig. 6.1 Procedure of Fault Diagnosis BP Network

6.3 Fault Diagnosis based on SOM

Unsupervised learning [Jain et al., 1999; Oja, 2002] is another method of data classification and clustering in addition to the supervised methods (for example BP network) in the field of data analysis. Supervised methods mostly deal with training classifier for known symptoms, while unsupervised learning (clustering) provides exploratory techniques for finding hidden patterns in the data. With huge volumes of data being generated from different systems every day, what makes a system intelligent is its ability to analyze the data for efficient decision-making based on known or new cluster discovery. Unsupervised data clustering is an intelligent tool for delving deep into the unknown and unexplored data. It is a tool that brings out the hidden patterns and association between different variables in a multivariate dataset. When the knowledge of the data is not well known and

explored, the unsupervised learning method can be used to analyze the data. That means in the field of fault diagnosis, it can be used to understand the data and cluster the fault hidden in database.

Self-Organizing Mapping (SOM) is a competitive learning network, it uses self-learning mode of non-supervision and non-direction, and its algorithm is simple with function of sidewise association [Brando et al., 2007]. It is one of the most popular unsupervised learning algorithms which can be used to explore useful information from not well-known data, and thus can be applied in fault diagnosis when the knowledge of the history data is not known. The principle of SOM is introduced in Section 3.2.2. Fig. 6.2 shows the procedure of applying SOM in fault diagnosis which mainly has three phases: training phase, test phase and decision making phase. The whole procedure of SOM application in fault diagnosis is similar with that of SBP application. In the training phase, the sensors are used to collect the data from the monitored mechanical equipment. The data could be the signals such as vibration and acoustics or the time series such as temperature. The former signals need to be processed in order to extract useful features while the later data can be used as features. The features, then, can be used to train SOM for establish classifier model of different type of faults. Once the classifier is established, the test phase can be done. The features used to test the classifier must be the same type with the training data. The finally phase is maintenance decision making based on the test results of SOM classifier.

6.4 Fault Diagnosis based on Semi-supervised Learning

Fault diagnosis for mechanical equipment is the essence of pattern recognition problem over the condition monitoring data, in the process of which the balanced fault and fault-free data and the features definition are the basis for data-driven diagnosis model such as BP model. However, the collection of fault data is very difficult because of its expensive costs and stochastic causes for offline system. Mostly, the fault data with label (type of fault) is collected by test rig in the lab. However, data-driven model such as BP network, cannot inherently support the transplant model of fault diagnosis if the test rig is not enough similar with real system. Typically for a long time running machine, there are lots of unlabeled samples of condition monitoring data which may contain valuable information of normal or abnormal conditions. Traditional supervised classifier cannot explore these data, but it is very improvident to just throw them [Yuan, 2012].

Conventional fault diagnosis methods using supervised learning are good at solving the problems of condition monitoring (CM) data with labels, but not well at utilizing unlabelled CM data. Semi-supervised learning algorithm can be implemented in fault identification by using labelled data and unlabelled data collected from sensors. Manifold Regularization (MR) is one of the most popular semi-supervised algorithms which principle is introduced in Section 3.3. MR has the capacity of learning intrinsic geometric structure of complexity nonlinearity fault samples and exploiting the intrinsic geometric distribution property embedded

in the high-dimensional fault patterns. Thus, the well-trained model can be utilized to further conditions based monitoring as well as fault diagnosis and prognostics.
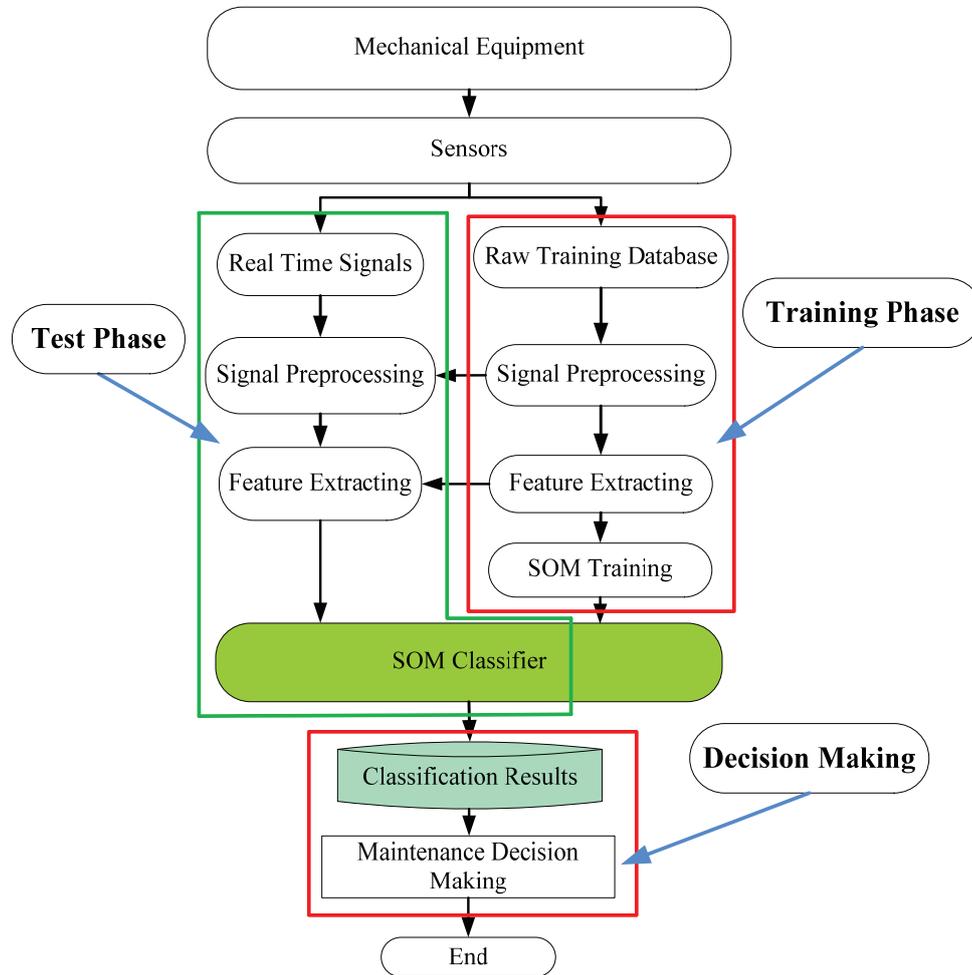
Fig. 6.2 Procedure of SOM in Fault Diagnosis

To show the advantages of semi-supervised learning with additional unlabelled dataset, a toy example called two-moon problem is presented.

Fig. **6.3** shows the solution of two-moon problem without unlabelled dataset. In this figure, the number of labelled training dataset is 50 and the number of test dataset is 200. From the figure, there are some dataset are misclassified. Fig. 6.4 shows the solution of two-moon problem with unlabelled dataset. The difference between

Fig. **6.3** and Fig. 6.4 is that the latter figure utilizes 500 unlabelled dataset with the 50 labelled dataset to train the model. Comparing the two figures, Fig. 6.4 is much better for two-moon problem which almost no test dataset is misclassified.

The fault diagnosis process of semi-supervised learning is shown in Fig. 6.5. The manifold regularization based on semi-supervised manifold learning for fault diagnosis system can be described as follows:

1) Building up general condition monitoring system to collect the labelled and unlabelled data from both local monitoring machines and the test rig;
2) Implementing feature extraction and feature selection from the labelled and unlabelled examples according to the criteria which determines the features set that represent the geometric structure well;
3) Constructing a data adjacency graph with labelled and unlabelled nodes using graph kernel, which describes an intrinsic manifold, and regulating classification decision boundary with manifold regularization algorithm, and then classifying the online patterns in the features space with classified labels;
4) Obtaining diagnosis information by classification of the results, then determining the failure causes, and putting the corresponding decision or control measures back to local condition monitoring system.
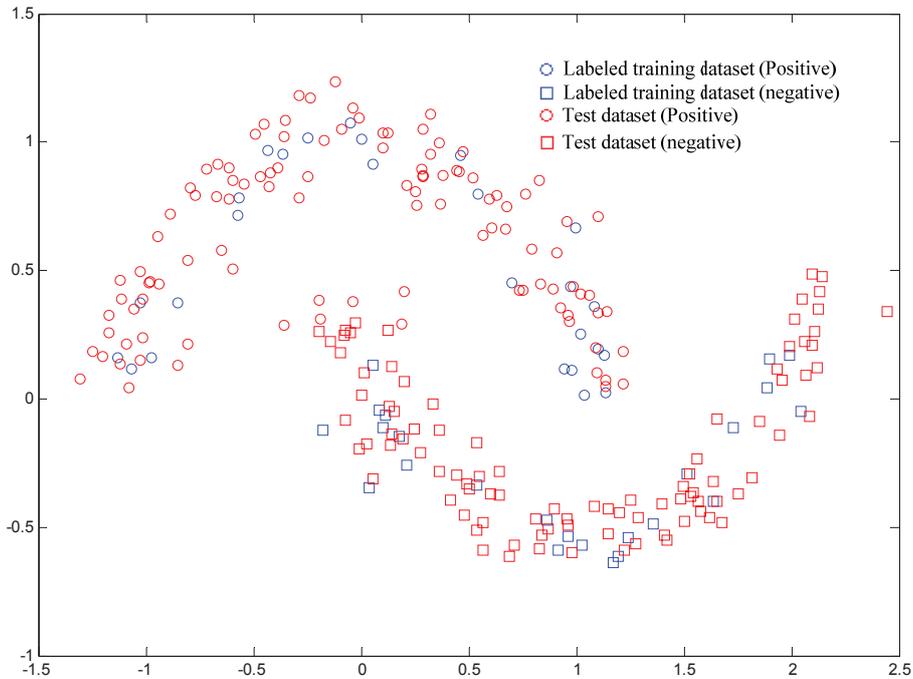
Fig. 6.3 Solution of Two-moon Problem without Unlabelled Dataset
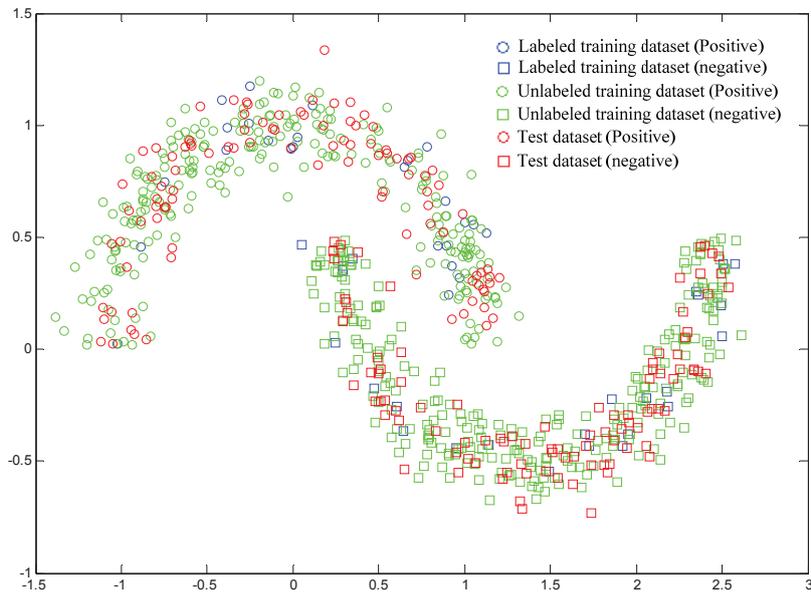
Fig. 6.4 Solution of Two-moon Problem with Unlabelled Dataset
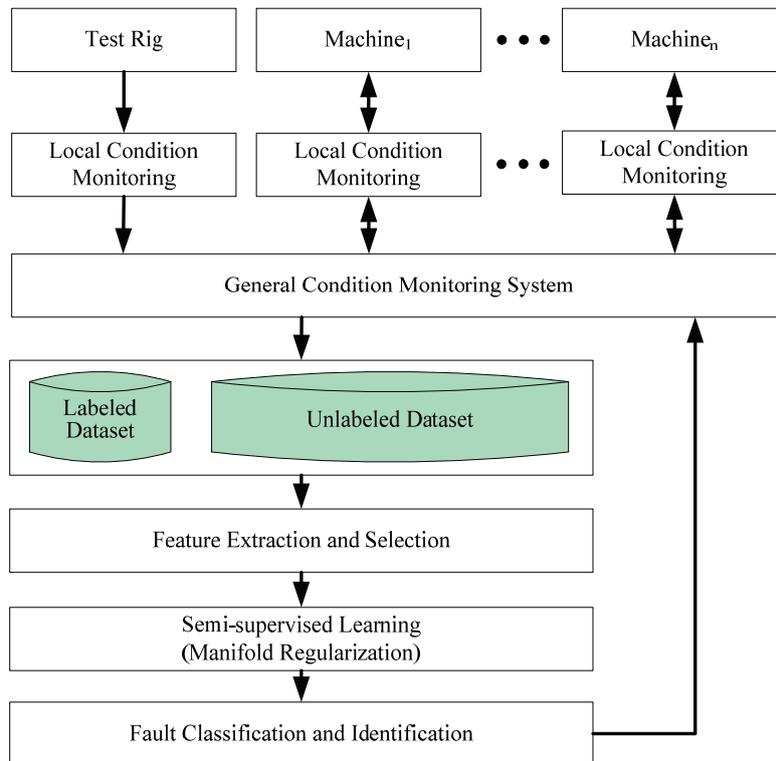
Fig. 6.5 Procedure of Semi-supervised Learning in Fault Diagnosis

6.5 Fault Diagnosis based on Association Rules

Association rules mining is a kind of data mining techniques which can discover significant association rules between items in database [Agrawal et al., 1993]. The basic concept and process of association rules are introduced in Section 3.4. This part will propose an Association Rule-based Fault Diagnosis which structure is shown in Fig. 6.6.
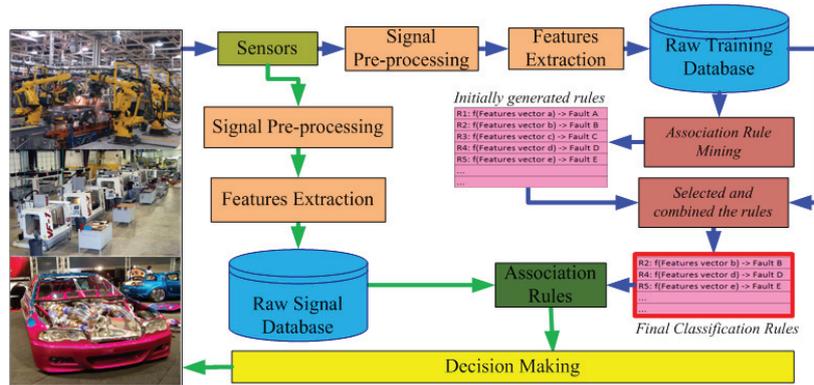
Fig. 6.6 The Structure of Association Rule-based Fault Diagnosis

Whatever a machines, cars or Robots, after long time running, their performance may become degradation or failure. Some suitable kinds of sensors should be selected to monitor their conditions. The data should be pre-processed before features extraction because the raw data from sensors may contain noise. After extracting the features, all the data are stored in a database called "Raw Training Database" which can be used to mine the association rules. For each kind of fault, several rules can be mined from the training data. Then, select and combine all the rules together as the whole association rules which can classify the fault or judge the condition of monitored equipment. Finally, the features extracted from pre-processed real time data can be used to diagnose the fault using the association rules generated above. According to result from association rules, the maintenance or control decision can be made correctly and efficiently.

6.6 Case Study 1: Fault Diagnosis Integration of WPD, PCA and BP Network

To demonstrate how the BP network works in fault diagnosis for mechanical machines, a lab setup is established in Knowledge Discovery Lab (KDL) in NTNU. The first case will show fault diagnosis integrating WPD, PCA and BP network. This case study is retrieved from [Zhang et al., 2013].

6.6.1 Experimental Setup

Fig. 6.7 shows the hardware of the experimental setup which includes a blower, three vibration sensors, power supply for sensors, connector, DAQ card and a computer. In this setup, the blower is selected as our monitoring object and a kind of vibration sensors (Kistler: Type 8702B100) are chosen to collect the signals from the blower. Three sensors are mounted on the blower in three directions which can collect the vibration signals in different directions (Fig. 6.8). The signals are collected from the sensors and processed using some processing method like filter, de-noising and compression. Then the features are extracted in wavelet domain which can be used to train and query BP network. After training, the system can judge the real states of monitored components using real time signals.
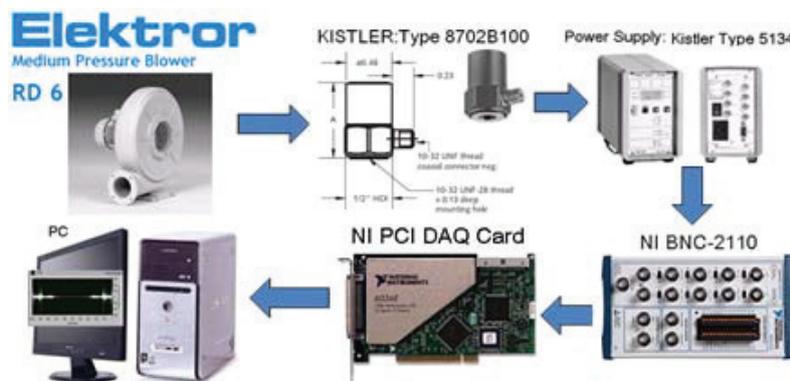
Fig. 6.7 Hardware of Experimental Setup
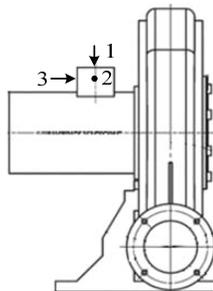
Fig. 6.8 Sensors Setup on Blower

6.6.2 Experimental Procedure

In the present study, four different degradation's levels of unbalance are simulated using three different parts (Fig. 6.9) which are mounted in the axis end of the blower. The unbalance degradation (condition) contains 0, 0.3, 0.7 and 1 which represents the performance states from perfect to absolutely failure (unbalance). In the first case, power on the blower, collect and store signals with sample rate 1024 per second from the sensors without mounting any simulation part. Next, power off the blower and mount first part in the axis end and then, power on the blower,

114

collect and store the signals from sensors. Repeat this process until collect all the degrading signals simulated by simulation parts. Fig. 6.10 shows the signals of the second sensor from perfect state to absolutely failure.

Fig. 6.9 Parts for Simulation Degradations
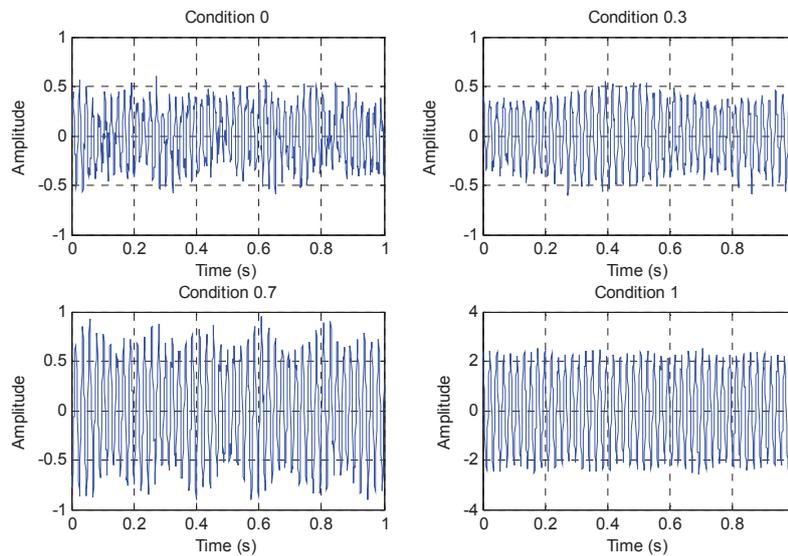
Fig. 6.10 Raw Signals with Different Degradations

6.6.3 Features Extraction in Wavelet Domain

Wavelet packet method [Li et al., 2003] which is a generalization of wavelet decomposition offers a richer range of possibilities for signal analysis. Contrary to WT, the wavelet packets contain a complete set of decompositions and details at every level and hence providing a higher resolution in the high frequency region, i.e., the wavelet detail component at each level is further decomposed to obtain its approximation and detail components. The principle of Wavelet Packet Decomposition (WPD) was introduced in Section 5.3.3. In this experiment, the structure of Wavelet Packet Decomposition (WPD) algorithm broke up to 4 resolution levels is shown in Fig. 6.11 In the figure, the node (4, 0) presents the symbol for a subspace that stands for the 4th resolution and the 0th subspace. For

this case, each node present the frequency bandwidth 64 Hz which means a node (4, 0) presents the signal character of the bandwidth between 0 Hz and 64 Hz.
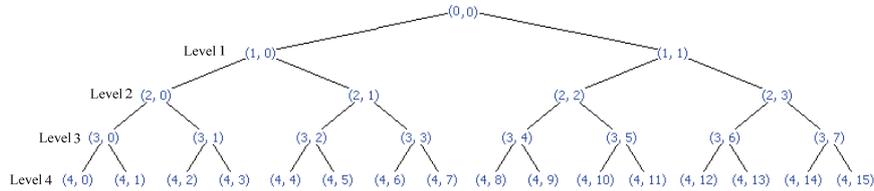
Fig. 6.11 Tree structures of wavelet packet transform (4 levels)

For each signal, wavelet packet was applied up to the fourth level, thus giving 16 signal coefficients. The wavelet packet coefficients (Eq.(5.30)) and their corresponding standard deviations for one signal are shown in Fig. 6.12. In the end, the Standard Deviation of Wavelet Packet Coefficients (SDWPC) of processed signals is selected as feature vector which is used to train ANN after PCA analysis.
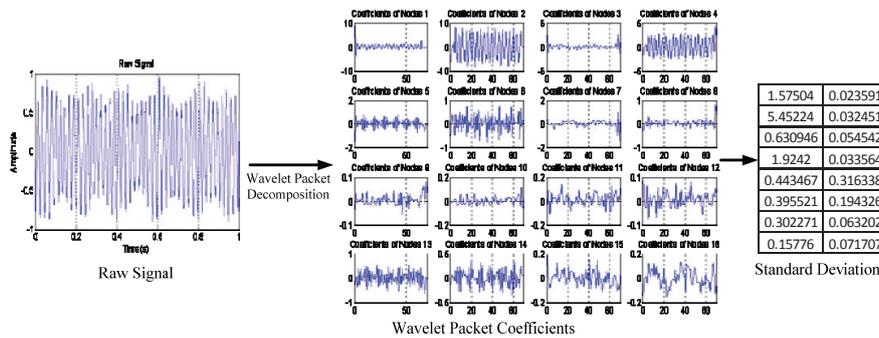
Fig. 6.12 Wavelet Packet Coefficients (WPC) and Their Relevant Standard Deviation

There are three vibration sensors mounted on the blower and for signal from each sensors, 16 parameters are extracted and thus overall 48 features for each time signals. Therefore, Principal Component Analysis (PCA) is employed to reduce the dimension of the features.

6.6.4 Principal Component Analysis (PCA)

PCA is a good option to reduce dimension of the features and its principle was introduced in Section 3.5. In this experiment, 200 samples for each condition are collected as training data and are analyzed by PCA. There are 48 variables (SDWPC) in each sample. Now the original sample matrix's dimension is 800×48. Then, these data are analyzed by PCA. The variance for each component is shown in Table 6.1 (only first 8 values are shown) and the first four principal components were displayed in Fig. 6.13. If the value of threshold is set to $\varepsilon = 1$, only first principal component was selected as feature to train ANN. If the value of threshold is set to $\varepsilon = 0.5$, only first two principal components were selected as features to train ANN.

116

Table 6.1 Variance for each component

Component No.	1	2	3	4	5	6	7	8	…
Variance	5625.841	0.681	0.424	0.133	0.053	0.005	0.002	0.001	…
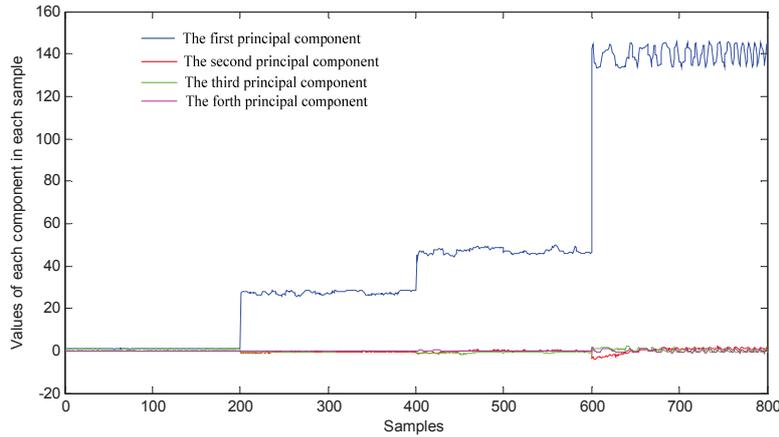
Fig. 6.13 The first four Principal Components

6.6.5 Fault Diagnosis using BP Network

The PCA new features from SDWPC of vibration signals are used to estimate the fault status of components and machines. The input nodes of BP neural network come from the test signal sensors. BP neural network made up of one input layer, one output layer and one hidden layers of nodes. And it has been proved that such three layers' BP neural network model can approach any continuous functions at any precision. The values of output are from 0 to 1 which represent from perfect condition to complete failure of specific kinds of fault.

For convenience of handling the signal collection, signal processing and interface things, the Labview are selected as program software in this case study. However, the capability of mathematical calculation of Labview is not as good as Matlab. Therefore, both kinds of software are combined. The procedure of fault diagnosis and prognosis integrating BP Network, PCA and WPC is shown in Fig. 6.14 which is modified from Fig. 6.1. The historic data is collected and processed which are fist two steps. Then, the features in wavelet domain (SDWPC) are extracted from the processed signals. These features are analyzed by the PCA which can generate new features called principal component which can used to train ANN. After training, the signals in real time are collected and used to query the BP network, and then the condition of the monitored components can be obtained.

```
┌──────────────────┐          ┌──────────────────────────┐
│ Data acquisition │◄─────────│ Collecting signals from  │
│                  │          │         sensors          │
└──────────────────┘          └──────────────────────────┘
        │
        ▼
┌──────────────────┐          ┌──────────────────────────┐
│ Signal processing│◄─────────│ Filter, amplification,   │
│                  │          │ denoising conditioning,..│
└──────────────────┘          └──────────────────────────┘
        │
        ▼
┌──────────────────┐          ┌──────────────────────────┐
│ Feature extraction│◄────────│ Wavelet packet coefficients│
└──────────────────┘          └──────────────────────────┘
        │
        ▼
┌────────────────────────────┐
│ Principal Component Analysis│
└────────────────────────────┘
        │
        ▼
┌──────────────────┐          ┌──────────────────────────┐
│ ANN training     │          │ Data acquisition in real │
│ (BP network)     │          │          time            │
└──────────────────┘          └──────────────────────────┘
        │
        ▼
┌──────────────────┐          ┌──────────────────────────┐
│ Classfication    │          │ Signal process           │
└──────────────────┘          └──────────────────────────┘
        │
        ▼
┌──────────────────┐          ┌──────────────────────────┐
│ Predict the condition of the│ Feature extraction       │
│ monitored components        │                          │
└──────────────────┘          └──────────────────────────┘
        │
        ▼
┌──────────────────┐
│ Maintenance Decision Making │
└──────────────────┘
        │
        ▼
     ( End )
```
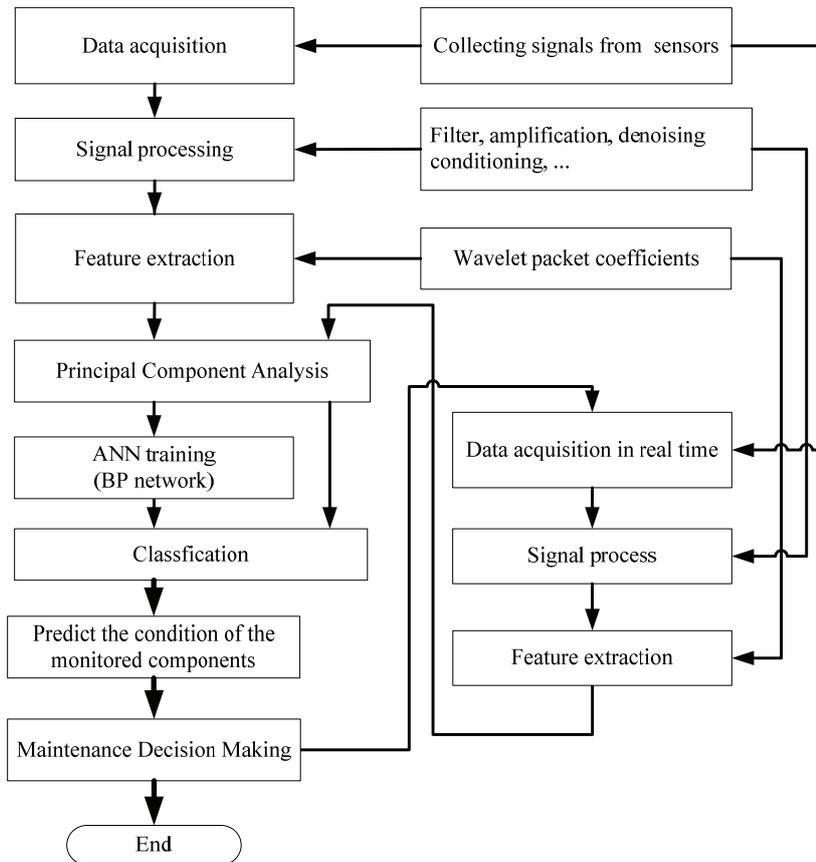
Fig. 6.14 Procedure of Fault Diagnosis Integrating BP Network, PCA and WPC

6.6.6 Results and Discussion

In this case study, four conditions are defined for the monitored component which are 0, 0.3, 0.7 and 1. They represent from perfect performance (condition 0) to completely failure (condition 1) discretely. For each condition, 200 training signals are collected and processed. The new feature vectors are generated using PCA from SDWPC. These new features are put into BP network for training. Finally, test signals are collected and processed like the training data. In this experiment, for each condition, 20 samples are collected which used to test trained BP network for verification.

For each testing data, the output of BP network and the nominal values which can be called "error from nominal value" (average value of testing data for each condition) are compared. The values of these errors are shown in Fig. 6.15-Fig. 6.18. There are two curve-lines for each figure. One represents only using the features of SDWPC as inputs while the other represents using the new features generated by PCA from the features of SDWPC as inputs to BP network.
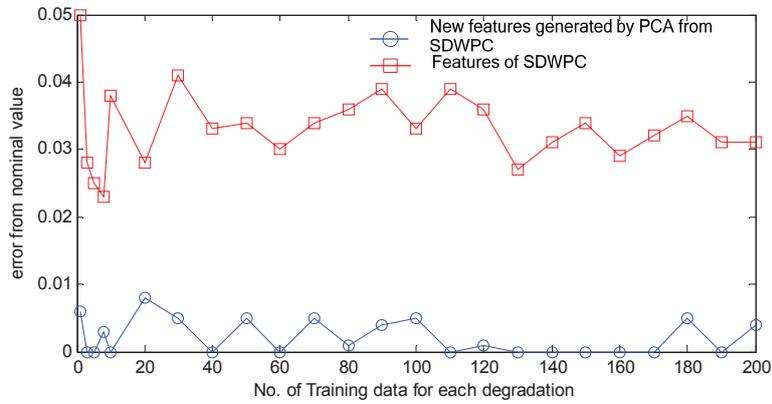
Fig. 6.15 Errors of Condition 0
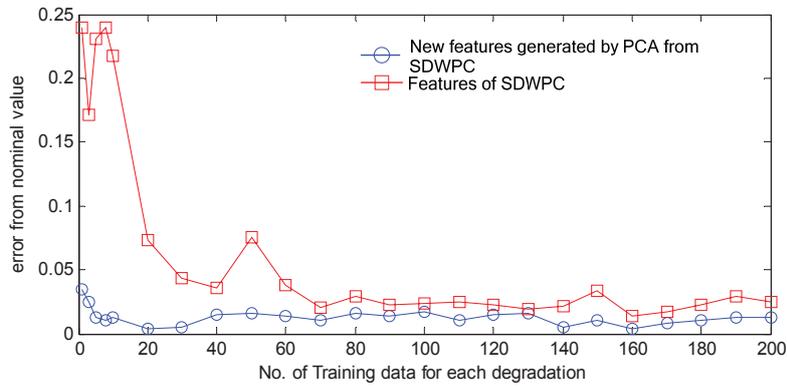
Fig. 6.16 Errors of Condition 0.3
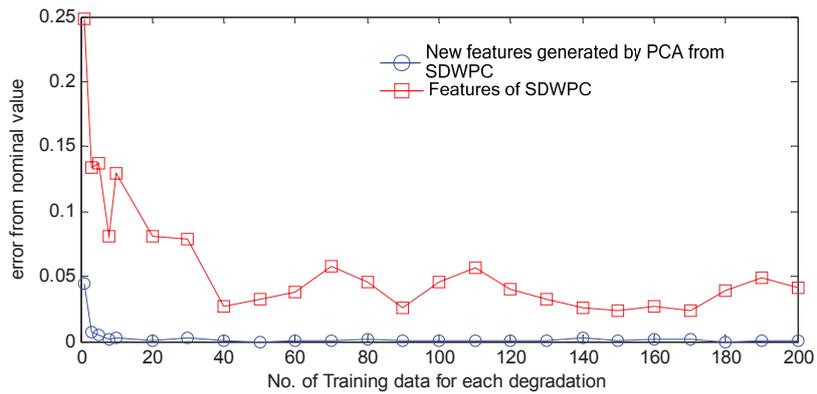
Fig. 6.17 Errors of Condition 0.7
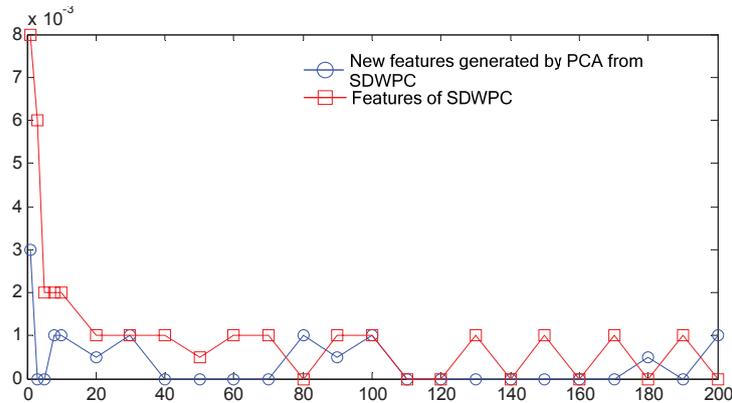
Fig. 6.18 Errors of Condition 1

All these four figures show the differences between the predicted values and nominal values of four different conditions using the features of SDWPC and new features generated by PCA from SDWPC. Fig. 6.15 shows the result of condition 0. The error is much smaller of the result using the new features generated by PCA from SDWPC compared to using the features of SDWPC as inputs of ANN. Fig. 6.16 and Fig. 6.18 show the results of condition 0.3 and condition 1 respectively. When the number of training sets is very small, the results using new features generated by PCA from SDWPC are much better than using features of SDWPC in these two figures. However, with the number of the training data increasing, the results of using both features are almost the same in these two figures and both of them are correct and precise. Fig. 6.17 shows the result of condition 0.7. In this figure, in both kinds of features, the performance is very effective and corrective whatever the number of training data is, but the result of using the new features generated by PCA from SDWPC is much better than using features of SDWPC. We can see from Fig. 6.16 and Fig. 6.17, when the condition is neither perfect nor completely failure, the result of using SDWPC is not believable if the number of training data is very small because the 'error from nominal value' is large. But it is still believable of using new features generated by PCA from SDWPC to training and testing ANN in these conditions. We can see from the four figures, the precision is better of using new features generated by PCA from SDWPC than using features of SDWPC in any condition and in any number of training data.

6.7 Case Study 2: Fault Diagnosis Integration of WPD, FFT and BP Network

This case study is retrieved from [Zhang et al., 2012]. The experimental setup and Experimental Procedure are the same as Section 6.6.1 Experimental Setup and Section 6.6.2 Experimental Procedure. However, the data analysis and diagnostic algorithm are different.

6.7.1   Feature Extraction

The feature extraction algorithm is combining WPD and FFT. For WPD, not like previous case, the vibration signals are decomposed to 3 levels and for each level, the approximation part is not decomposed in order to reduce the dimension of the parameters without omitting much information. The structure of wavelet packet decomposition is shown in Fig. 6.19.
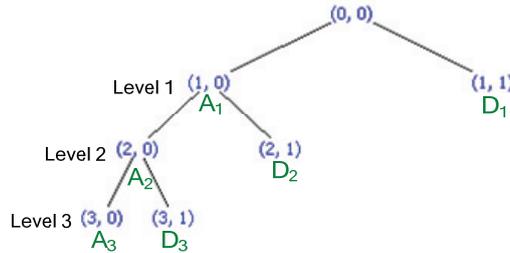
Fig. 6.19 3-layer Structure of Wavelet Packet Decomposition

For this case, the signal maximum frequency is 512 Hz, and thus D1, D2, D3 and A3 represent the frequency 256~512 Hz, 128~256 Hz, 64~128 Hz and 0~64 Hz respectively in Fig. 6.19. In this experiment, only these four parts are analyzed to judge the degradation of the performance. The decomposed signals by WPD from the different degrading signals are shown in Fig. 6.20-Fig**.** 6.23.
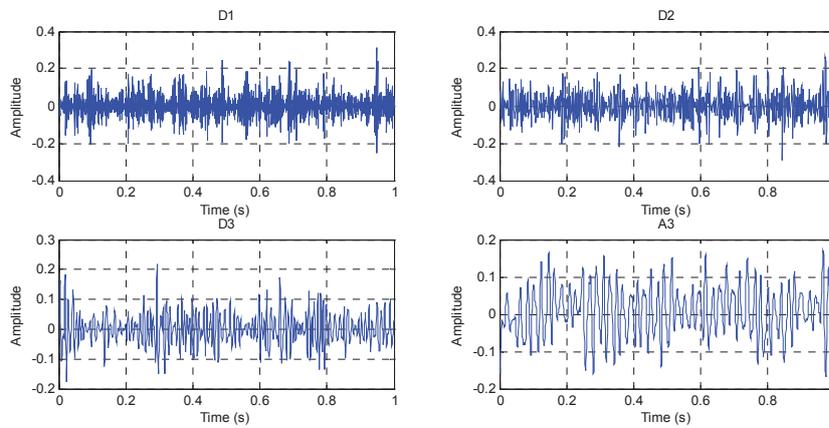
Fig. 6.20 Decomposed Signal of Condition 0
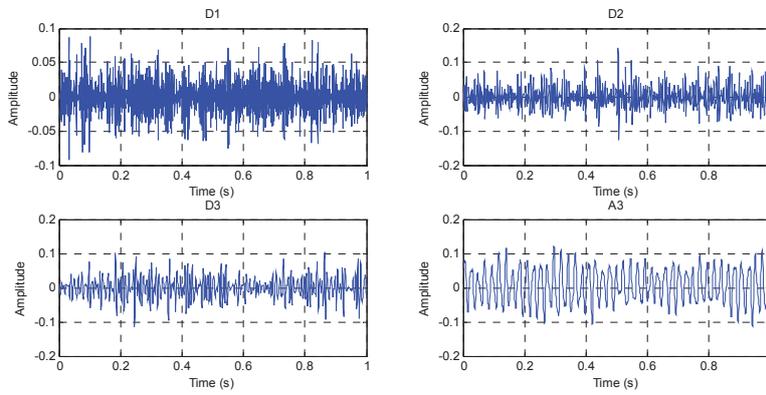
Fig. 6.21 Decomposed Signal of Condition 0.3
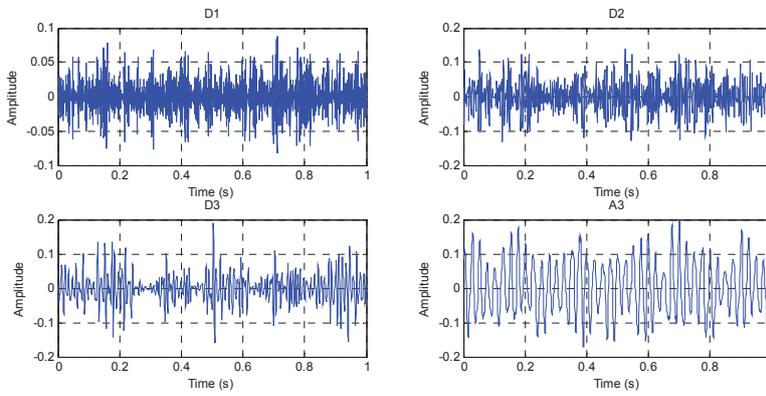
Fig. 6.22 Decomposed Signal of Condition 0.7
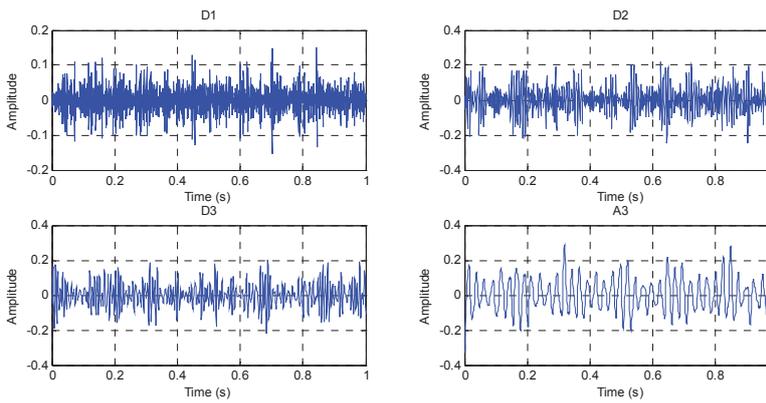
Fig. 6.23 Decomposed Signal of Condition 1

6.7.2   Fast Fourier Transform to WPD Signals

The principle of FFT was introduced in Section 5.3.2. In Section 6.7.1, the original signal was decomposed as on approximation and details. Then, the decomposed signals are transformed with FFT which are shown in Fig. 6.24-Fig. 6.27 which present different conditions from condition 0 to condition 1. From the result of FFT, some kinds of features can be chosen. In this paper, the peaks for each part are selected as features to judge the condition of monitored equipment.
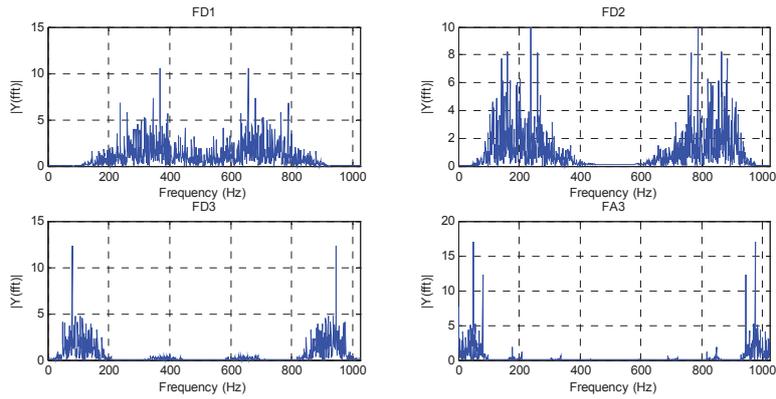
Fig. 6.24 FFT for Each Version Signal of Condition 0
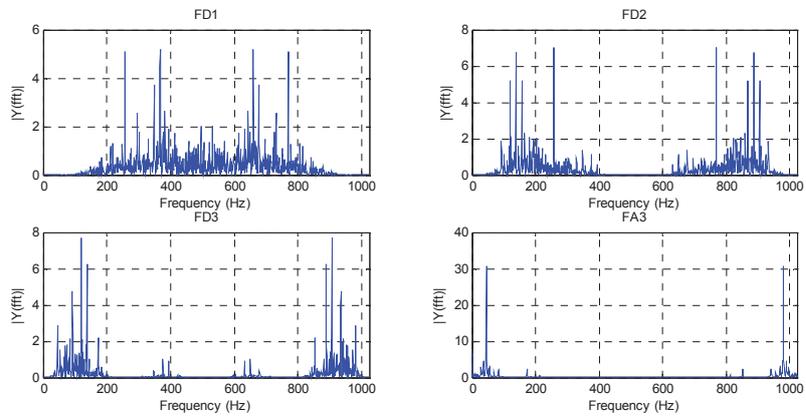
Fig. 6.25 FFT for Each Version Signal of Condition 0.3
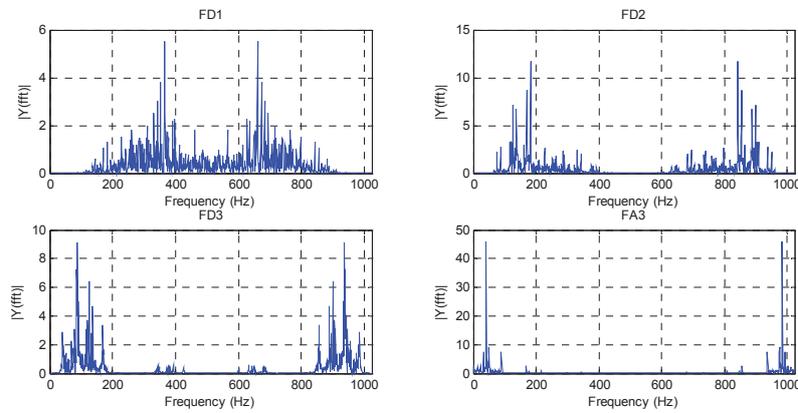
Fig. 6.26 FFT for Each Version Signal of Condition 0.7
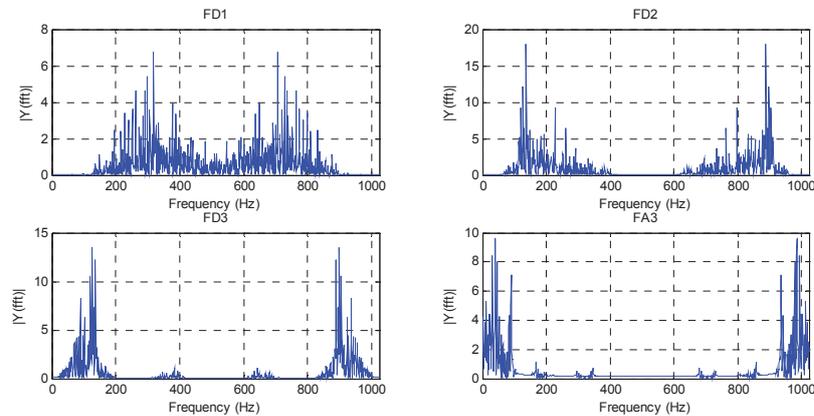
Fig. 6.27 FFT for Each Version Signal of Condition 1

6.7.3 Fault Diagnosis Procedure of Integrating WPD, FFT and BP Network

Fig. 6.28 shows the procedure of fault diagnosis integrating WPD, FFT and BP Network which is modified from Fig. 6.1. The historic data is collected and processed which are first two steps. Then, the processed signals can be decomposed by WPD. Each part of decomposed signals can be transformed using FFT and the peak value for each of them is selected as feature to train BP network. After training, the signals in real time are collected and used to query the BP network, and then the condition of the monitored components can be obtained. Finally, the remaining useful life is evaluated for decision making of maintenance according to the condition.
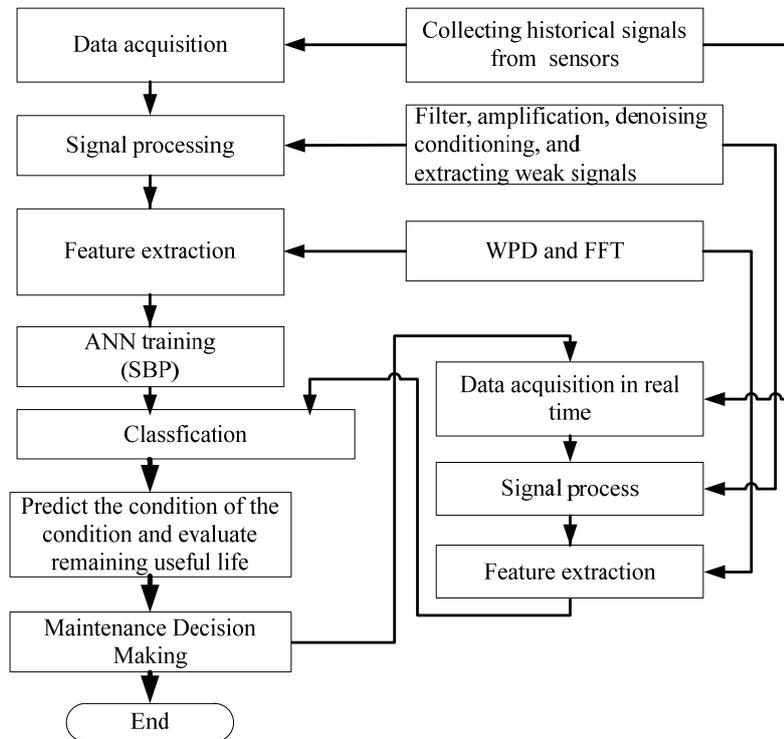
Fig. 6.28 Procedure of Diagnosis Integrating WPD, FFT and BP Network

6.7.4   Experiment and Results

In this case study, four conditions for the monitored component are defined which include 0, 0.3, 0.7 and 1 which represent from perfect performance to completely failure discretely. For each condition, 200 training signals were collected and processed. The training signals are pre-processed firstly and then decomposed by WPD. For each part of decomposed signal, calculating the peak value in its frequency domain transformed using FFT which called PFD1, PFD2, PFD3, and PFA3. In this case, there are three sensors and thus there are 12 parameters are input to input nodes of BP network and one output value which represents condition of the monitored component (Called C). A part of training data is shown in Table 6.2. After training, test data or query data obtaining from real system can be used to test or query BP network. In this case, 20 sets of test data (Table 6.3) are used to test BP network.

There is no mathematical method to select the best structure of the BP network, but the three layers SBP structure was validated its powerful function to build a complex model. The SBP structure in this experiment is set to three layer 12×20×1 networks. 12 means the number of input parameters (features in this experiment), 20 means the number of the hidden layer nodes and 1 means only one output in this BP network structure (condition). Its maximum training epoch is set to 5000. For each condition, 80 training sets are used to train ANN and 20 sets of features are chosen to test it. Table 6.3 shows the results of the test data. As mentioned before,

125

there are 20 sets of test data in which there are 5 sets of them for each condition. The nominal condition is called NC while the output condition of test is called TC in this table. From this table, the results are 100% correct in the above parameter sets. However, the output is not exactly the same as the nominal condition and there are deviations between them. The precision of the output is discussed next section.

Table 6.2 Part of Training Data

Sensor 1				Sensor 2				Sensor 3				C
PFD1	PFD2	PFD3	PFA3	PFD1	PFD2	PFD3	PFA3	PFD1	PFD2	PFD3	PFA3	
4.20	3.18	3.768	49.05	4.07	3.756	3.26	95.17	4.325	4.323	2.816	101.08	0
4.46	2.965	2.788	20.38	4.54	3.404	3.346	102.0	3.891	4.108	3.248	107.36	0
4.58	4.039	3.874	317.6	6.15	3.603	3.704	4170	4.663	3.55	5.447	1094.9	0.3
3.42	3.802	3.227	314.3	3.46	3.765	3.659	4220	4.261	3.42	4.659	1132.5	0.3
4.87	4.238	5.951	482.2	5.19	4.184	4.617	6975	3.523	3.845	2.723	1889.8	0.7
4.49	3.745	4.178	395.6	4.03	4.412	4.289	6828	4.781	3.705	3.022	1861.4	0.7
6.41	3.007	18.46	1933	4.94	3.053	5.048	2035	5.095	2.919	5.601	6189.9	1
4.54	4.304	18.43	1936	4.72	4.23	4.73	2103	4	4.506	6.062	6391.8	1
…	…	…	…	…	…	…	…	…	…	…	…	…

6.7.5 Discussion

In this section, three issues will be discussed. The first one is how many training sets should be used in order to achieve enough accurate condition of the machine from BP network. The second one is attempting to discuss the relationship between the accuracy and the number of hidden layer nodes. The last issue is convergent time of the BP network training.

To discuss the first issue, the numbers of training sets for each condition are changed from 1 to 200. The number of hidden layer nodes is set to 20 and the number of training epoch is set to 5000. For each testing data, compare the output of ANN and the nominal value which is called "error from nominal value" which is average value of testing data for each condition. The values of these errors are shown in Fig. 6.29. We can see from this figure, the result is believable whatever the condition of the component is when the number of training data is larger than 20. For condition 0 and condition 1, the result is still believable even if the number of training data is smaller than 20. It is clear that the result will be believable if there are only two conditions (0 and 1 or good and fault) even if the number of training data is very small. But if there are more conditions, the number of training data should be increased. Therefore, the number of conditions should be considered in designing of how many training sets are used to trained BP neural networks.

Table 6.3 Test Data and the Results

Sensor 1				Sensor 2				Sensor 3				Results		Deviation
PFD1	PFD2	PFD3	PFA3	PFD1	PFD2	PFD3	PFA3	PFD1	PFD2	PFD3	PFA3	NC	TC	
3.71	3.382	2.941	37.608	4.636	4.582	3.018	99.027	4.095	3.719	4.749	107.685	0	0.04	0.038
4.755	3.079	3.049	30.693	6.705	4.092	3.187	81.135	3.951	3.525	3.583	105.698	0	0.05	0.046
4.29	4.083	2.418	20.416	4.258	4.069	3.583	85.343	6.563	3.09	4.254	100.095	0	0.04	0.041
4.803	3.506	3.056	39.464	4.561	3.792	2.952	76.756	4.398	4.545	4.052	111.304	0	0.03	0.033
4.114	3.856	2.557	32.6	4.77	4.005	3.258	112.054	4.752	3.517	3.48	116.38	0	0.04	0.038
4.133	3.684	3.162	275.02	4.114	3.736	3.227	4135.45	4.701	3.452	4.835	1199.69	0.3	0.28	0.017
4.433	3.475	3.589	280.2	3.944	3.532	3.121	4174.94	4.215	3.701	3.25	1223.66	0.3	0.28	0.025
5.301	3.352	3.539	280.78	4.89	4.044	4.023	4280.03	5.667	4.422	4.008	1259.53	0.3	0.29	0.014
5.346	6.322	4.353	257.69	4.934	3.491	3.361	4175.62	5.982	5.922	3.044	1212.37	0.3	0.28	0.02
3.852	3.516	3.548	303.43	4.874	3.825	3.852	4193.24	3.817	3.952	3.428	1233.34	0.3	0.29	0.011
4.699	3.421	3.31	311.82	4.327	4.911	5.273	7101.38	4.158	3.558	3.183	2098.35	0.7	0.71	0.005
4.087	4.644	3.008	278.09	3.865	3.482	5.644	7211.46	5.392	4.981	3.51	2147.95	0.7	0.64	0.059
3.978	3.719	3.463	286.09	4.321	3.635	5.177	7122.77	4.196	3.682	3.883	2094.52	0.7	0.7	0.002
4.347	2.976	3.434	279.05	5.284	5.405	4.546	7157.75	3.978	4.449	3.333	2126.02	0.7	0.67	0.033
4.44	3.505	3.345	262.11	5.521	3.628	4.63	7080.4	3.991	3.587	4.037	2082.84	0.7	0.67	0.031
3.603	4.235	8.397	910.92	5.633	5.258	9.274	21669.1	3.839	3.875	5.985	6824.31	1	1	0.002
5.451	3.87	6.187	885.07	4.922	6.128	12.67	21416.9	5.687	3.643	8.407	6661.59	1	1	0
5.957	3.575	8.829	918.7	5.764	5.873	8.867	21244.5	4.995	4.161	4.444	6594.82	1	1	0.002
4.818	3.049	8.352	882.03	6.918	5.931	10.12	20684.6	5.035	3.511	8.835	6461	1	1	0.002
3.745	3.083	8.32	885.89	6.902	5.976	10.9	20605	5.183	3.216	8.642	6455.84	1	1	0.002
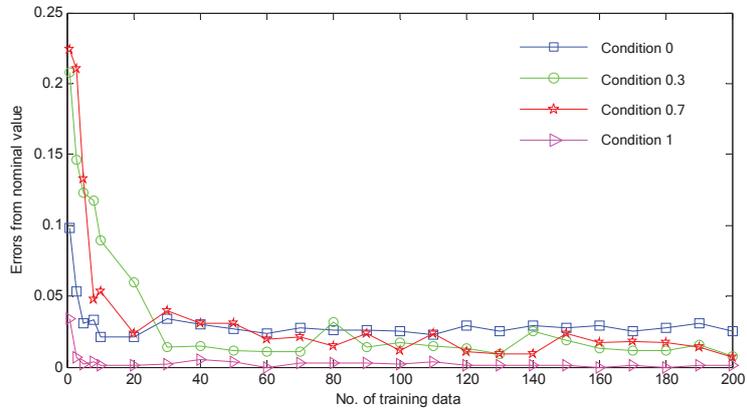
Fig. 6.29 Errors for Each Condition
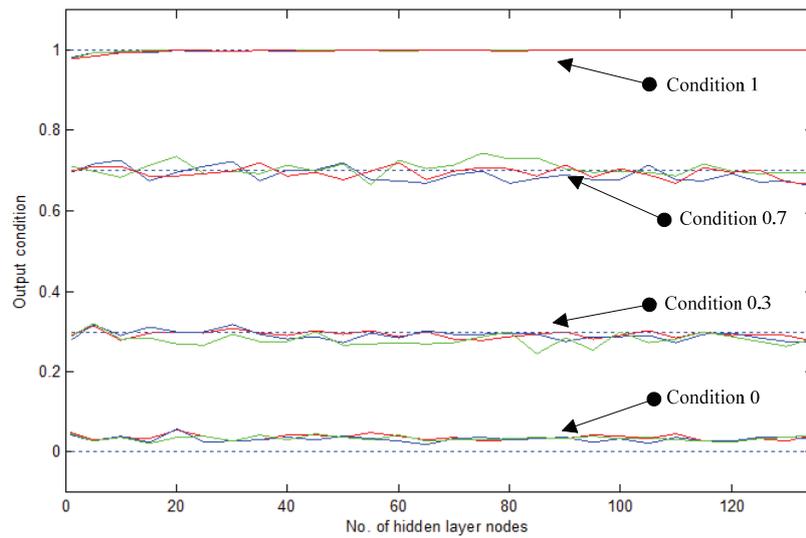
Fig. 6.30 Output with Different Number of Hidden Layer Nodes

To discuss the second issue, the number of hidden layer nodes is changed from 5 to 135. The number of training data is set to 80 and the number of maximum training epoch is set to 5000. For each training process, several test sets for every condition are used to test the trained SBP networks. The results are shown in Fig. 6.30. From the figure, with the increasing of the number of hidden layer nodes, the fluctuations of the output for each condition are small. So the changing of the number of hidden layer nodes does not affect the accuracy of the output. What's more, there is no mathematical method to prove what the number of it is best. Therefore, the number of hidden layer nodes does not need to be considered much.

To discuss the last issue, the number of hidden layer nodes is set as 20 and the training epoch is set as 2000. Fig. 6.31 shows the BP network training time with the number of training data increasing from 10 to 200. From the figure, training time is not apparently increasing with the increasing of training data sets. Therefore, when the BP network is need, we should use as many as possible data sets to complete the training. Fig. 6.32 shows the training time changes with the increasing of hidden layer nodes. The number of training data sets is set as 200 and the training epoch is set as 2000. From this figure, the training time increase gradually with the increasing of hidden layer nodes. Therefore, when a BP network need to trained, the number of hidden layer nodes should be considered. However, from the experience of previous work, the numbers of the hidden layer neurons depends both on the input layer number and the output layer neuron number but the numbers can not be too many [Meng & Meng, 2010].
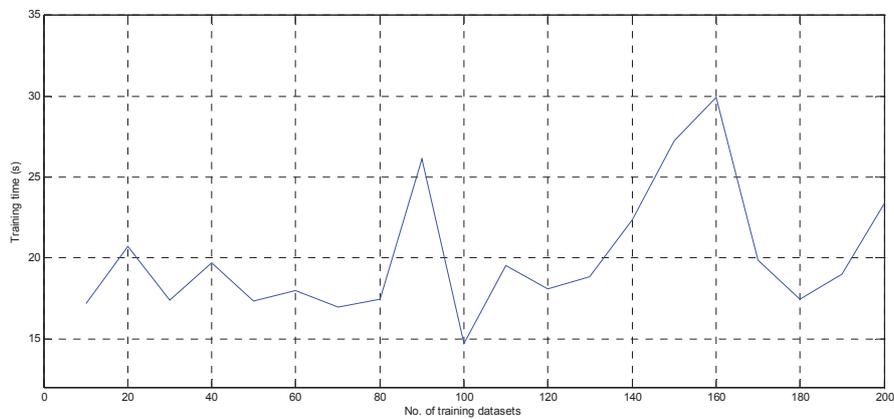
Fig. 6.31 BP Network Training Time with the Increasing of Training Data
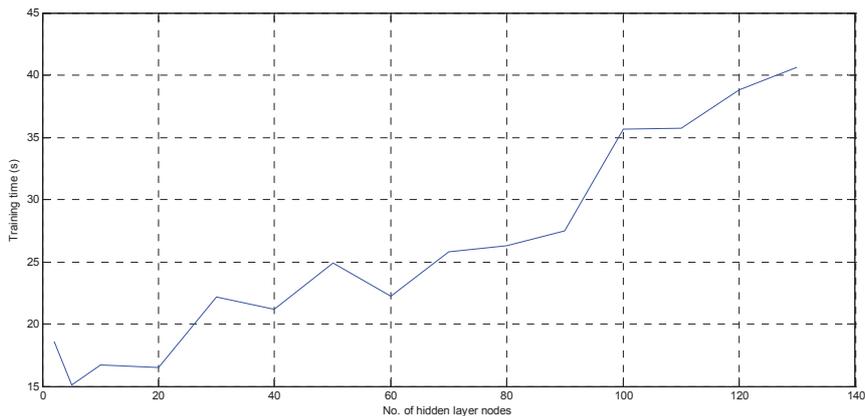
Fig. 6.32 BP Network Training Time with the Increasing of Hidden Layer Nodes

6.8 Case Study 3: Fault Diagnosis based on Self-organizing Map

SOM is a type of Artificial Neural Network (ANN) which is trained by unsupervised learning to map a high dimensional dataset into low dimensional space. It is very suitable for classification and clustering. The principle of SOM has been introduced in Section 3.2.2. This example shows how the SOM works in fault diagnosis, i.e. fault classification which is retrieved from [Zhang & Wang, 2011].

6.8.1 Experimental Setup

The experimental set-up consists of a centrifugal pump designed for a pressure increase of 6.6 bars at 90m^3/h and at an operating speed of 3000 rpm. The drive unit is a 3 phase induction motor with an output of 26 kW. The pump rig is rigidly mounted in a relatively noise-free environment. It is designed to lift and circulate water. Both the motor and the pump are equipped with ball bearings. Vibration measurements were taken in axial direction at the free ends of both the motor and the pump. Measurements were also taken at the vertical and the horizontal directions on the bearing housing at both the pump and the motor drive and free ends. Along the vertical direction on the pump casing, another measurement was taken close to the impeller (Fig. 6.33). The following types of vibration measurements were carried out on the pump rig:

- High Frequency Domain (HFD) parameter (5-60 kHz).
- Low Frequency (LF) spectrum (0- 400 Hz).
- High Frequency (HF) spectra (0-8 kHz).

For the respective measuring point, the frequency components in Table 6.4 will be registered.
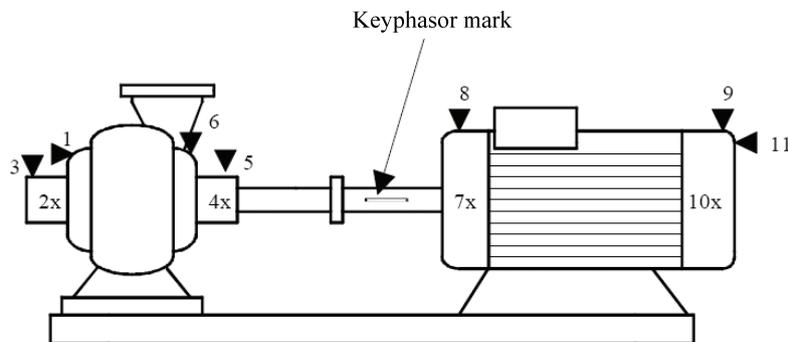
Fig. 6.33 Vibration Measurement Points

Table 6.4 Measurement Points and Their Corresponding Vibration Types

Equipment	Measurement Points	Type of vibration measurement
Pump	Free-end axial (1) Free-end horizontal (2) Drive-end horizontal (4)	Low Frequency Spectrum
	Free-end vertical (3) Free-end vertical (5)	High Frequency Domain Low Frequency Spectrum High Frequency Spectrum
	Pump Casing (6)	High Frequency Domain High Frequency Spectrum
Motor	Drive-end horizontal (7) Free-end horizontal (10) Free-end axial (11)	Low Frequency Spectrum
	Drive-end vertical (8) Free-end vertical (9)	High Frequency Domain Low Frequency Spectrum High Frequency Spectrum

6.8.2 Fault Types of Centrifugal Pump System

There are several types of faults in the centrifugal pump system have different symptoms with different failure. The most important problems to be monitored are introduced as the following:

Leakage from worn wearing-ring (L)

Inner leakage as a result of worn wearing-ring in a pump will have a result that a large part of the delivered capacity, which the pump delivers, will go directly back to the suction side of the pump. The efficiency will be lower, and the pump will no longer be able to produce the same pressure at a given capacity.

This problem is simulated by exchanging one of the two rings with another which had a clearance of 1.0 mm instead of the recommended clearance of 0.25 mm. Clearance is measured as the maximum distance between the inner side of the ring and the impeller in radial direction.

Unbalance on impeller (U)

The efficiency of the pump is decreased as a result of the unbalance on the impeller, and this leads to high current consumption at speed and flow. This symptom includes high and steady once per revolution component (1x) at both

bearing. Approaching a phase difference of 90 digress between the 1x components in the vertical and horizontal directions for both bearing, and approaching zero phase difference between 1x vibration at the pump's free-end and drive-end bearings.

This problem is simulated by exchanging the impeller with another which has a steel weight of 0.114 kg mounted on the suction side, at a radius of 100 mm. the shape of the weight was designed to give minimum disturbance to the flow around the impeller.

Unbalance on coupling (N)

The symptoms include the high and steady once per revolution component (1x) at the bearing on both side of the coupling, approaching a phase difference of 90 degrees between 1x vibration at the pump's free-end and drive-end bearings.

This problem is simulated by mounted a steel weight of 0.102 kg on the periphery of the coupling at a radius of 80 mm.

Misalignment between Motor and Pump (M)

Misalignment manifests itself as coupling misalignment, and can therefore cause deflection forces to be generated in the rotor, friction in seals and casings, and bearing failure, etc. This can result in high current consumption at high flows and rpm. Misalignment is characterized by steady 1x and 2x frequency amplitude components ,approaching a phase difference of 90 degrees between 1x vibration component at the pump's free-end and drive-end bearings. It can be accompanied by large axial vibration, up to about 50% of the radial level.

This problem is simulated by moving the motor in both vertical and horizontal directions; as a result a combination of parallel and angular misalignment exists. A laser alignment monitoring instrument was used to verify the amount of misalignment present.

Bearing Damage (B)

The effect could be measured by high frequency domain parameter and the total revolution-level for a high frequency.

This problem is simulated by exchanging the ball bearing at the pump's free-end with another bearing that has a small cavity on the inner ring.

Cavitation (C)

Cavitation is one of the most frequent occurring problems in centrifugal pump system which is unfavorable operational state as a result of pressure losses along the pipe at the suction side at high flow rates. The most important effects of cavitation are increased hydraulic losses, noise and vibration, and massive wear of surface.

In experiment, a flow valve is used to regulate the flow on the suction side to a pressure below the pump's Net Position Suction Head (NPSH).

When a problem is going to occur or has occurred, a certain number of symptoms or parameters will demonstrate themselves in a certain way, i.e., some frequency component value will change significantly when there is a bearing problem, these parameters are then called "features" for monitoring or detecting that problem. In this system, there are 56 parameters can be collected which is shown in Table 6.5. A part of data used to Train SOM network while the others are used to test SOM classifier. In this case study, 100 training data sets are used to train SOM network and 20 data sets are used to test these network.

Table 6.5 Parameters Calculated from Vibration Signals

Parameters	SCFLOW, SCSPEED, CPFATOTL, CPFA1X, CPFA2X, CPFHTOTL, CPFH1X, CPFH2X, CPFH36X, CPFHSYNC, CPFHSUBS, CPFHNONS, CPFRAT1X, CPFVTOTL, CPFV1X, CPFV2X, CPFV36X, CPFVSYNC, CPFVSUBS, CPFVNONS, CPFVTOTH, CPFVHFD, CPDHTOTL, CPDH1X, CPDH2X, CPDH36X, CPDHSYNC, CPDHSUBS, CPDHNONS, CPDRAT1X, CPDVTOTL, CPDV1X, CPDV2X, CPDV36X, CPDVSYNC, CPDVSUBS, CPDVNONS, CPDVTOTH, CPDVHFD, CMDH1X, CMDV1X, CMDRAT1X, CMFATOTL, CMFA1X, CMFA2X, CPFVHPHA, CPDVHPHA, CPFDPHA, CMDVHPHA, CPMAPHA, CPCATOTH, CPCASHFD, CPOWER, CTORQUE, CDELTAP
Terminology	SCFLOW - normalized flow, SCSPEED - normalized speed, CX…X-normalized data, X…X - raw data, P - pump, M - Motor, F - free end, D - drive end, H - horizontal direction, V - vertical direction, A -axial direction, CAS - pump casing, 1X, 2X, …, 6X - first, second up to sixth fundamental frequency, PHA - phase angle, TOTL-total low frequency, TOTH - total high frequency, SYNC - synchronous frequency, SUBS - sub synchronous frequency, NONS - non-synchronous frequency, FREQ - frequency, P1 - inlet pressure, P2 - outlet pressure, PH - phase, DELTAP - difference between P1 and P2, VOLTAGE - voltage, CURRENT - current, SPEED - rotational speed, FLOW - flow rate, CPFRAT1 - difference between CPFV1X and CPFH1X.

6.8.3 Experiment and Results

Fig. 6.34 shows the visualization of SOM for U-matrix and first 5 variables. Fig. 6.34(a) shows the U-matrix which is short for unified distance matrix which means the Euclidean distance between the SOM node vectors of the neighboring neurons is depicted in a gray scale image. From this figure, it is easy to see that the maps are classified as several different groups. Fig. 6.34(a)-(f) displays the distributions of the variables from Variable1 to Variable5. It is noticeable that the groups in variables are not the same as ones in U-matrix map and are not the same themselves. This is because the effects of each variable for different type of faults are different. For each variable, the map is classified as several groups clearly according to its values. Combining all the maps together, the general map like U-matrix could be generated which presents the whole groups of clustering, and here presents the types of faults.

Fig. 6.35 shows the results of SOM classification of Centrifugal Pump System. The labels "B", "C", "L", "M", "N", "U" and "M" present the types of faults. In this map, the neurons with the fault type means the inputs with the same fault are mapped into this node. It is noticeable that there are probably more than one fault types locating in the same node which means the input data represent corresponding fault types. It is also noticeable that the neurons representing the same fault type may be not in the same area, which means a type of fault may be caused by different parameters. The numbers located in the neurons mean the sequence of the test data sets. From the map, the fault type of the test data could be classified very clearly.
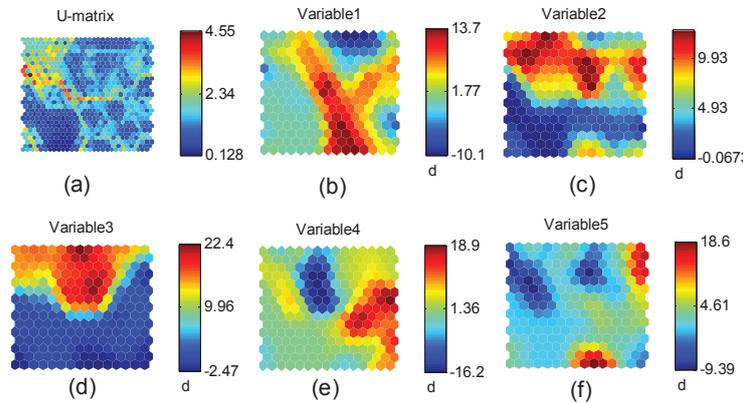
Fig. 6.34 Visualization of SOM
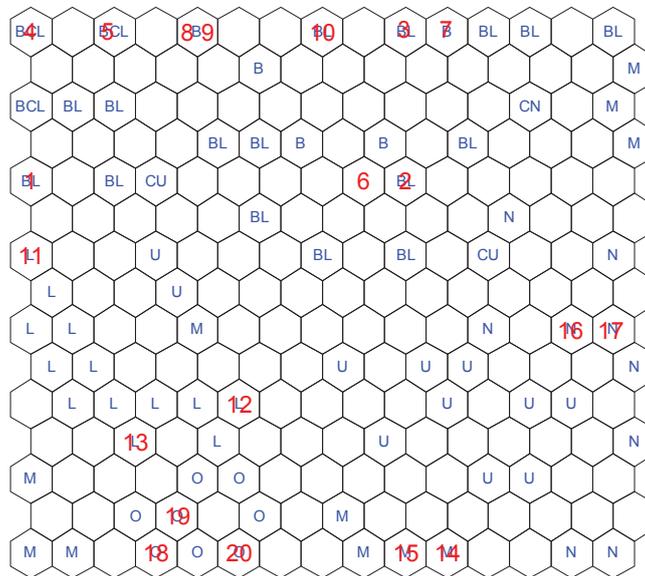
Fig. 6.35 Classification Result of SOM

134

6.9 Summary

This Chapter mainly describes how the data mining techniques work in fault diagnosis of mechanical machines. These data mining techniques include: BP network, SOM, Semi-supervised learning and association rules. Some case studies are used to verify these techniques except semi-supervised learning because there are no data for case study and a two-moon problem was used to show how it works. From these case studies, the data mining techniques are suitable to diagnose the faults of mechanical equipment. The discussion for each case study is presented in following paragraphs.

Case Study 1 and Case Study 2 described two examples integrating BP network with other two techniques. The former described case study with the method of integrating BP network, PCA and WPD while the later described the case study of methods integrating BP network, WPD and FFT. To verify the correctness and effectiveness of these two methods, Blower Fault Diagnosis System was established. These methods demonstrated high effectiveness in diagnosing machine faults. They can classify the condition of the monitored components.

In former case study, PCA was applied to reduce the input dimension (number of variables) of BP network without omitting the useful information. BP network model may become over specified, i.e. more input variables than is strictly necessary, due to including superfluous variables which are uninformative, weakly informative, or redundant [May et al., 2011]. In this case, the total volume of the modeling problem domain increases exponentially with the linearly increasing of variable dimensionality which is called curse of dimensionality [Bellman, 1961]. This will cause many problems such as: computational burden increasing which is a significant influence in determining speed of training and training difficulty due to inclusion of redundant and irrelevant input variables. By reducing the dimensionality of variables, PCA can solve these problems and improve the effectiveness of BP network training. Therefore, the method provides a faster, more effective and more precise solution for fault diagnosis and prognosis. The latter case study applies FFT after WPD to extract features which does not too many variables, and thus the PCA is not applied.

In these two cases, the minimum bandwidth 0~64 Hz is chosen in WPD because the fundamental frequency of the vibration signal is 47.5 Hz. In a real system, the minimum bandwidth of WPD (which means how many levels should be decomposed) should be selected according to the real fundamental frequency. There is only one type of fault (unbalance) simulated. In the future, multi-fault diagnosis should be a research topic. These two methods can also be applied to decide many other faults such as wear, crack, and fatigue of bearings and gearbox which faults can be reflected by vibration signals. To apply these methods, the fundamental frequency has to be known firstly and thereafter the sample rate of vibration signals, the level of wavelet decomposition, and the structure of BP network can be determined properly. The degradation information could be very useful for maintenance decision making, and thus, how to apply this degradation information in maintenance decision making should be a research issue as well in the future.

Case Study 3 described a Self-organizing Map (SOM) classifier applying in fault isolation for a machine (here is a centrifugal pump), which could be called as pattern clustering as well. The Self-Organizing mapping describes a mapping from a higher dimensional input space to a lower dimensional space. In the experiment, SOM maps 56 dimensional variables into a two dimensional space (15×15). The result of SOM method is very effective, clear and easy to understand. The results of the experiment shows SOM is very suitable for solving this kind of problem like fault isolation, fault classification and pattern recognition.

From this case, the way to continuously monitor the condition of machines, components, systems and processes have been found. The first stage is determining the number of neurons of a two-dimensional SOM lattice and the number of clusters of the conditions and fault type according to the real machines or systems. The second stage is to train SOM which is mapping the many of variables to the predefined SOM neurons, that is, finding the groups of lattice which each of them represent a kind of condition or fault. The third stage is finding the location of the test data or real time data inside the lattice. Finally, the conditions or faults of machines, components, systems or processes could be determined according to the trained lattice and the location of the test data or real time data.

In this case, for each type of fault, only two conditions (normal or failure) are considered. However, this is not enough and not fitting the real situation. In the future, more conditions should be considered for each type of fault. Here, only offline data are used to test trained SOM neurons, but in the future, the real time data should be used for real time monitoring, control and maintenance. Finally, in the future, SOM could be applied in these kinds of problems such as pattern recognition combining with other machine learning method such as Support Vector machine (SVM) and Supervised Back-propagation (SBP).

7 Fault Prognosis based on Artificial Neural Network

7.1 Introduction

Prognosis is the ability to predict accurately and precisely the Remaining Useful Life (RUL) of a failing component or subsystem. The task of the prognostic module is to monitor and track the time evolution (growth) of the fault. In the industrial and manufacturing arenas, prognosis is interpreted to answer the question: "what is the RUL of a machine or a component once an impending failure condition is detected, isolated, and identified?" It is a basis of a Condition-Based Maintenance (CBM) system and presents major challenges to CBM system designer primarily because it entails large-grain uncertainty. Long-term prediction of the fault evolution to the point that may result in a failure requires means to represent and manage the inherent uncertainty. Moreover, accurate and precise prognosis demands good probabilistic models of the fault growth and statistically sufficient samples of failure data to assist in training, validating, and fine-tuning prognostic algorithms. Fault prognosis has been approached through probabilistic, artificial intelligence and other methodologies. Specific techniques include fuzzy-adaptive Kalman predictor [Tian et al., 2011], Autoregressive Model [Xin et al., 2012], fuzzy-filtered neural networks [Li et al., 2013] and Case-Based Reasoning [Berenji, 2006]. However, there are still some challenge in this area [Vachtsevanos et al., 2006]:

- How we can infer the actual crack dimension over time in the absence of the techniques of measuring creak length directly?
- How do we predict accurately and precisely the temporal progression of the fault?
- How do we prescribe the uncertainty bounds or confidence limits associated with the prediction?
- Once we have predicted the time evolution of the fault and prescribed the initial uncertainty bounds, how do we improve on such performance metrics as prediction accuracy, confidence, and precision?

The techniques of fault prognosis can be classified into three categories: model-based, probability-based and data-driven methodologies. The model-based techniques can predict any fault of the machines or components if the accurate physical model or mathematical model is available. The advantages of this technique are very apparent: it can predict any type fault in any component in any stage of faults of a machine. However, determining a complete dynamic model in terms of differential equations that relate the inputs and outputs of the system being considered may be impractical in some instances since the machine becomes more and more complex and integration. Often, historical data from previous failures for a given class of machinery can be used to establish probabilistic model [Hu et al., 2011] based on statistic methods. These methods require less detailed information than model-based techniques because the information needed for prognosis resides in various probability density functions (PDFs), not in dynamic differential

equations. Advantages are that the required PDFs can be found from observed statistical data and that the PDFs are sufficient to predict the quantities of interest in prognosis. Moreover, these methods also generally give confidence limits about the results, which are important in giving a feeling for the accuracy and precision of the predictions. In many instances, one has historical fault/failure data in terms of time plots of various signals leading up to failure, or statistical data sets. In such cases, it is very difficult to determine any sort of model for prediction purposes. In such situations, nonlinear network approximators can be used for prediction of failure which provides desired outputs directly in term of data using well-established formal algorithms. This is so-called data-driven technique fault prognosis. This Chapter will describe the process of fault prognosis based on neural network.

7.2   Procedure of Fault Prognosis based on Artificial Neural Network

As mentioned before, most of big companies have huge history data which is not effectively used currently. This kind of data can be used to predict and identify the fault of machines before the failure happens. Fig. 7.1 shows how the history data can be used in fault prognosis by ANN. This figure and the following sections just take the SCADA data of wind turbines as an instance of research objects. These kinds of history data normally contain the performance parameters such as temperatures, vibrations, speed and lubrication, etc., and alarm/fault/warning list of all components of the machines. The first step of fault prognosis is to select right parameters to be analyzed. For a specific fault/failure, there is normally one or more performance parameters could be the indicator to determine if the fault/failure happens. For instance the temperature of bearing can be the indicator of the bearing defect. It is not too difficult to choose the right indicator for a specific fault through data analysis or experience. Besides, the related performance parameters with the indicator should be also selected through data analysis, experience or some algorithms such as boosting tree algorithm [Kudo & Matsumoto, 2004] and wrapper with genetic search [Kohavi & John, 1997]. Then the ANN model in normal condition can be trained in which the selected performance parameters could be the input while the indicator of the fault could be the output of the ANN model. The trained ANN is so-called ANN model of normal behavior. The second step is to establish ANN predictor for fault prognosis. In this step, the history data with fault will be used. With the ANN normal behavior and the selected performance parameter values, the theoretical values of the indicator can be estimated and compared with real values from the history data. Through the comparison and how early the customer wants to have early warning, close alarm and emergency stop, the thresholds of these levels can be set. Fig. 7.5 could be an example of these functions. Finally, the ANN model with these thresholds can be the fault predictor of a machine.

```
┌─────────────────────────────────────────────────────────────────┐
│            Mechanical System and/or Electrical System             │
└─────────────────────────────────────────────────────────────────┘

┌───────────────────────────────┐     ┌──────────────────────────────┐
│ Existing History Data:        │     │   Online Condition Monitoring │
│  • Temperature, vibration,    │     │        (Online Data)          │
│    speed...                   │     │                               │
│  • Fault, Alarm, warning...   │     │                               │
└───────────────────────────────┘     └──────────────────────────────┘

┌───────────────────────────────────────────────────┐
│              Parameter Selection                   │
└───────────────────────────────────────────────────┘

┌────────────────────┐   ┌────────────────────┐      ┌──────────────────┐
│ ANN Training using │   │   Data with Fault  │      │    Real Time     │
│   Healthy Data     │   │                    │      │   Performance    │
└────────────────────┘   └────────────────────┘      │ parameters and   │
                                                      │ Indicator Values │
┌────────────────────┐  ┌───────────┐                └──────────────────┘
│  ANN Model for     │←─│Performance│
│  Normal Behavior   │  │ Parameter │
└────────────────────┘  │  Values   │
                        └───────────┘
┌────────────────────┐   ┌────────────────────┐
│ Estimated Indicator│   │ Real Indicator     │
│      Values        │   │     Values         │
└────────────────────┘   └────────────────────┘

┌───────────────────────────────────────────────────┐
│                   Comparison                        │
└───────────────────────────────────────────────────┘

┌─────────────────────────────────────────────────────────────────┐
│              ANN Fault Predictor and Identifier                   │
└─────────────────────────────────────────────────────────────────┘

┌─────────────────────────────────────────────────────────────────┐
│      Fault Prediction and Identification, Early warning           │
└─────────────────────────────────────────────────────────────────┘
```

Fig. 7.1 Procedure of Fault Prognosis

7.3 Fault Prognosis based on Indicator Prediction by ANN for Wind Turbine Monitoring

Renewable energy sources are playing an important role in the global energy mix, as a means of reducing the impact of energy production on climate change. Wind energy is the most developed renewable energy technologies worldwide with more than 282.48 GW installed capacity at the end of 2012 [GWEC, 2013]. Certain forecasts indicate that the share of wind in Europe's energy production will reach up to 20% in the close future [Krohn et al., 2007]. Today, large wind turbines (2-6MW) are becoming established as economically viable alternatives to traditional fossil-fuelled power generation. In some countries, such as Denmark, Germany and Spain, wind turbines have become a key part of the national power networks [Pinar Pérez et al., 2013].

Condition monitoring of wind turbines is of increasing importance as the size and remote locations of wind turbines used nowadays makes the technical availability of the turbine very crucial. Unexpected faults, especially of large and crucial components, can lead to excessive downtime and cost because of restricted turbine accessibility especially for some remote controlled wind farms on mountain and offshore wind farms. However, even smaller issues and faults of auxiliary equipment like pumps or fans can also cause expensive turbine downtime due to the same causes. From an operator's point of view it is therefore worth increasing the effort spent to monitor the turbine condition in order to reduce unscheduled downtime and thus operational costs. The key part of wind turbine monitoring system is to detect and predict fault (fault diagnosis and prognosis) of turbines as early as possible so that the maintenance staff can manage and prepare the maintenance action in advance.

Most wind turbines installed nowadays are integrated with SCDA system which can monitor the main components. SCADA system typically monitors parameters such as temperatures of bearings, lubricating oil, windings and vibration levels of driven train [Becker & Poste, 2006]. This monitored data is collected and stored via a SCADA system that archives the information in a convenient manner, usually for all of the turbines in the wind farm. This data quickly accumulates to create large and unmanageable volumes that can hinder attempts to deduce the health of a turbine's components. It would prove beneficial, from the perspective of utility companies, if the data could be analyzed and interpreted automatically to support the operators in identifying defects. One main function of SCADA data analysis is fault detection and predict as early as possible to support the decision of maintenance action and operation.

Model based methods require a comprehensive physical or mathematical model which is normally unavailable. Success of data based methods is conditioned by the significance of historical data and the mathematical method used to detect the patterns in data. For wind turbine systems where an important amount of data is stored regularly by SCADA system and process model is not available, the use of data driven methods is preferred [Nassim, 2011].

This section describes Artificial Neural Network (ANN) that can be used to predict and identify incipient faults in the main component of a turbine, such as main bearing, gearbox and blades, through the analysis of this SCADA data. BP network is one type of ANN which can solve the non-linear problems without sophisticated and specialized knowledge of the real systems. It is suitable to be applied in fault detection and predict and the principle of BP network was described in Section 3.2.1.The SCADA data sets are already collected and stored, and therefore, no new installation of specific sensors or diagnostic equipment is required. The technique developed normal behavior model by ANN and SCADA data analysis which can calculate the theoretical value of related parameters and compare to the real measurement of the same parameters. The parameters mentioned above can be indicator of abnormal behavior of incipient component failure. In this way, only interesting information is highlighted to the operator, therefore significantly reducing the volume of data they are faced with. This section just take the main shaft rear bearing monitoring as an instance to show how the technique works.

7.3.1 SCADA Dataset Description

An operational wind farm typically generates vast quantities of data which is well known SCADA data.

- The SCADA data contain information about every aspect of a wind farm, from power output and wind speed to any errors registered within the system. Thus by keeping track of both wind speed and power output parameters, the overall health of the turbine can be supervised.
- SCADA data may be effectively used to "tune" a wind farm, providing early warning of possible failures and optimizing power output across many turbines in all conditions

It is common for "condition monitoring" to be applied to a wind farm. However, this involves the addition of extra instrumentation, involving wind farm down time, extra cost and potential warranty implications. As distinct from condition monitoring, performance monitoring using existing instrumentation to analyze SCADA data of wind turbines is no extra instrumentation, no down time and no cost. It has the advantage of using data already routinely gathered. By making use of specially-designed software tools, a great deal of information may be gathered and analyzed to provide a detailed look at the performance of the wind farm.

Typical parameters recorded by SCADA on wind turbines could be broadly categorized into following types which could be used in fault detection and diagnosis activity [Verma & Kusiak, 2012].

- Wind parameters, such as wind speed and wind deviations;
- Performance parameters, such as power output, rotor speed, and blade pitch angle;
- Vibration parameters, such as tower acceleration and drive train acceleration; and

- Temperature parameters, such as bearing temperature and gearbox temperature.

Specifically the SCAD data recorded and used for condition monitoring from wind turbines are as follows:

- Active power output (10 min max/min/average)
- Anemometer-measured wind speed (10 min max/min/average)
- Turbine speed (10 min max/min/average)
- Nacelle temperature (10 min max/min/average)
- Turbine rear bearing temperature (10 min max/min/average)
- Turbine rear vibration (10 min RMS max/min/average)
- Turbine front bearing temperature (10 min max/min/average)
- Turbine front vibration (10 min RMS max/min/average)

The technique presented utilizes only some types of the data mentioned above. The parameters listed above are typical of data collected and stored by commercial wind turbine SCADA systems. This means the approach developed in this section can be widely applied by wind farm operators.

7.3.2   Modeling of SCADA Parameter Normal Behavior

A parameter of main shaft rear bearing in the SCDA data, i.e. turbine rear bearing temperature, gives an indication of how hot of the bearing are running, and therefore offer the possibility to detect rear bearing overheating. The straightforward threshold check which has already been applied in real wind farm, could be used to flag up temperature exceeding a certain limit, this might be too late to avoid significant damage to the main shaft rear bearing. The desired functionality should take into consideration any relevant aspects of turbine operation. This approach would allow temperatures to be detected that are too high in the context of the concurrent level of power generation, leading to a quicker and more effective identification of abnormal behavior.

7.3.2.1   Parameter Selection

In order to establish the normal behavior ANN model of main shaft rear bearing temperature, the variables that can affect this temperature must be taken into consideration to build an accurate model. Wind turbines can only aerodynamically capture a proportion of the energy in the incident wind [Hansen, 2007]. This energy is converted by the rotor blades into mechanical power and is transmitted directly to the generator by the main shaft because the turbine monitored in this paper is direct-driven turbine without gearbox. Zaher et al. [Zaher et al., 2009] established the normal behavior model of gearbox bearing temperature and cooling oil temperature and Sanz-Bobi et al. [Garcia et al., 2006] built the same model in addition to cooling oil thermal difference. The former model utilized active power and nacelle temperature while the later also utilized the operation of fans used to cool the gearbox. However, there is no gearbox in direct-driven wind turbine and thus avoid to faults of comprehensive gearbox. The main shaft bearings are the key

components of this type of wind turbine. Therefore, one of the key components, main shaft rear bearing, is main monitoring object in this section.

Accordingly, the parameters may affect the rear bearing temperature contain: active power output, nacelle temperature, turbine speed and cooling fan status. Unfortunately, the cooling fan status is not available in current SCADA data and thus the parameters selected to establish ANN model for the parameter of main shaft rear bearing temperature can be chosen as seen in Table 7.1.

Table 7.1 Input and Outputs of ANN Model

Model Output	Input
Rear Bearing Temperature	Rear bearing temperature (t-1)
	Active power output (t)
	Nacelle temperature (t)
	Turbine speed (t)

7.3.2.2   Training ANN Model

The models are trained using the parameters discussed in Table 7.1. In order to get an accurate representation for the parameter under study, the range of the parameters as inputs to ANN should be as varied as possible while still ensuring the turbines are in normal operational condition in ANN training process. This was achieved through selection the period of training data with many conditions as possible: the starting and stopping of turbine, big changes of turbine speed, with and without active power output. Therefore, three months from 01.01.2009-01.04.2009 SCADA data are chosen as training data for ANN rear bearing normal behavior model as seen in Fig. 7.2. This amounts to roughly 13,000 data points for each input. The training process then attempts to capture the nonlinear relationship between these parameters, i.e. the associated rear bearing and nacelle temperatures for the turbine speed and corresponding power output. The number of training cycles used, also known as epochs, was 1000. Determining the architecture for the network is an iterative process and depends solely on the structure that yields the best accuracy when tested. The final architecture used for rear bearing model was 5-10-1.

The trained model was tested in new data from a healthy wind turbine which had not been used in training process of ANN. Fig. 7.3 shows the input data used to test rear bearing ANN model from the same turbine of the date from 26.05.2009 to 26.07.2009. The test data shown here was also very varied. Fig. 7.4 shows the output of rear bearing ANN model (EstimatedTemp), the real temperature of that (BearTemp) and the difference between these values. The average difference between actual and estimated value is 0.026 ^oC, and the root mean square error is 0.2, which is considered to be an acceptable level for successful fault detection and prediction. This means that the output of ANN model can be used directly as a comparison with the actual temperature to assess if a fault is present. If the difference between the estimated value and the actual value increases for a

continuous number of instances, i.e. a prolonged period of time and not a minor fluctuation, then this would flag as a fault.
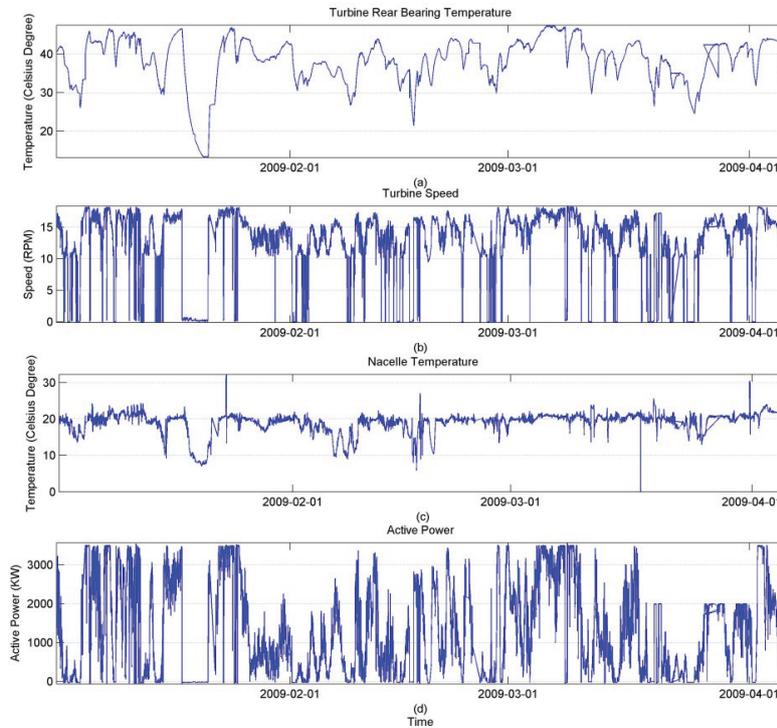
Fig. 7.2 Neural Network Turbine Rear Bearing Temperature Model Training Data
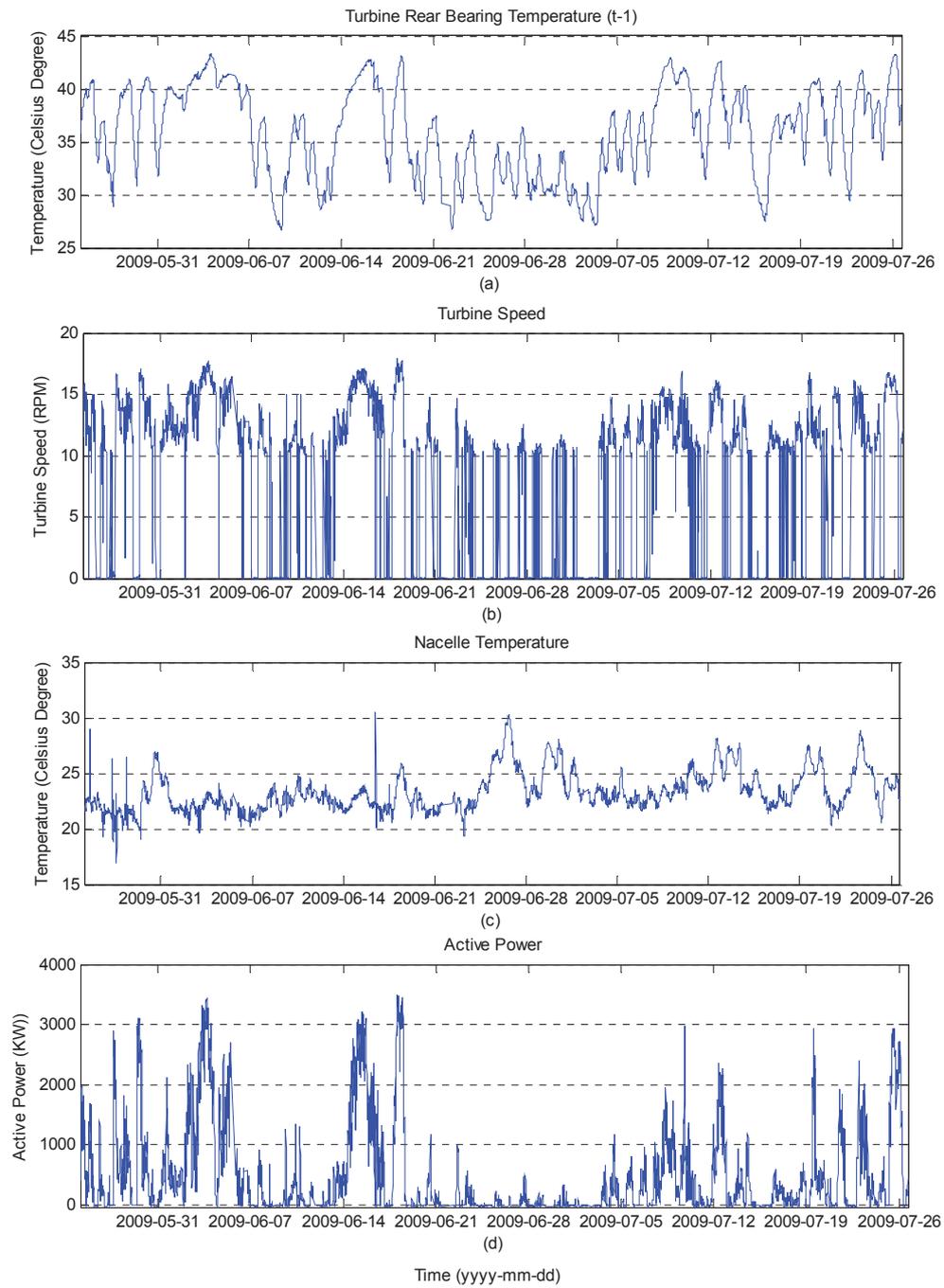
Fig. 7.3 Rear Bearing Model Testing Input Data
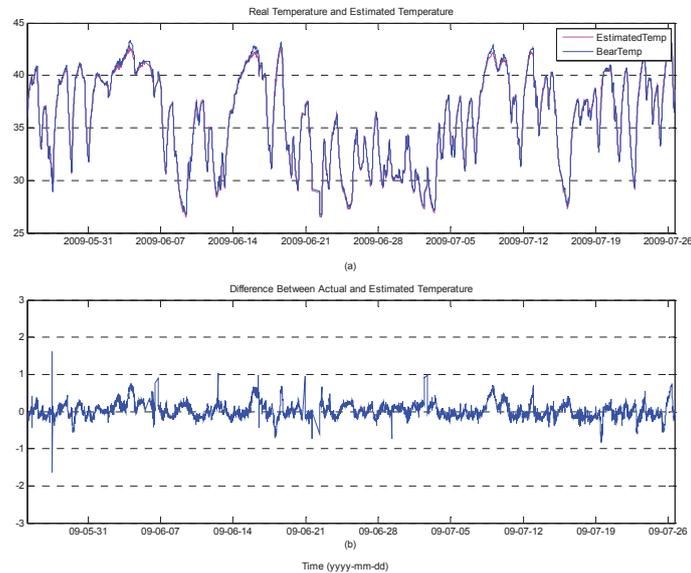
Fig. 7.4 Rear Bearing Model Output in Normal Condition

7.3.3   Prediction and Detection of Rear Bearing Fault

Once the normal behavior of rear bearing ANN model was trained, it can be used to detect and predict the corresponding fault of rear bearing by comparing estimated and actual temperature. Fig. 7.5(a) shows the evolution of rear bearing temperature from the period of July 2010 to March 2011 which contains eight months where it eventually fails. Fig. 7.5(b) shows the difference trend between the estimated and actual temperature of rear bearing in this period. The first important deviation from the model estimates occurred from the start of October 2010, i.e. point ①. The frequency of deviation and their duration increased in the following months. From point ②, the deviation from the model estimates increased to 4 ^{0}C and lasted to point ③ where the turbine was stopped because of overheating. Then, the operator of wind farm tried to solve the problem two times in point ③ and point ④, but not successful and finally the turbine was completely stopped because of the same overheating. From this figure, the method can give the operator a warning as early as three months in point ① before the failure happens. With the evolution of the failure, the deviation from model estimation increases and the alarm can be given to operator when the deviation reaches the level of point ②. Therefore, the alarm can be given as early as 10 days before the failure happens.

The results produced by ANN model for rear bearing fault detection and prediction are very positive. They can provide an early warning of problems developing in the bearing before the absolute temperature becomes apparently high. The results of the fault detection can be used to help the operator to make the schedule of

maintenance actions before the failure happens to reduce the maintenance cost, reduce the unanticipated downtime and improve the reliability of the wind turbine.
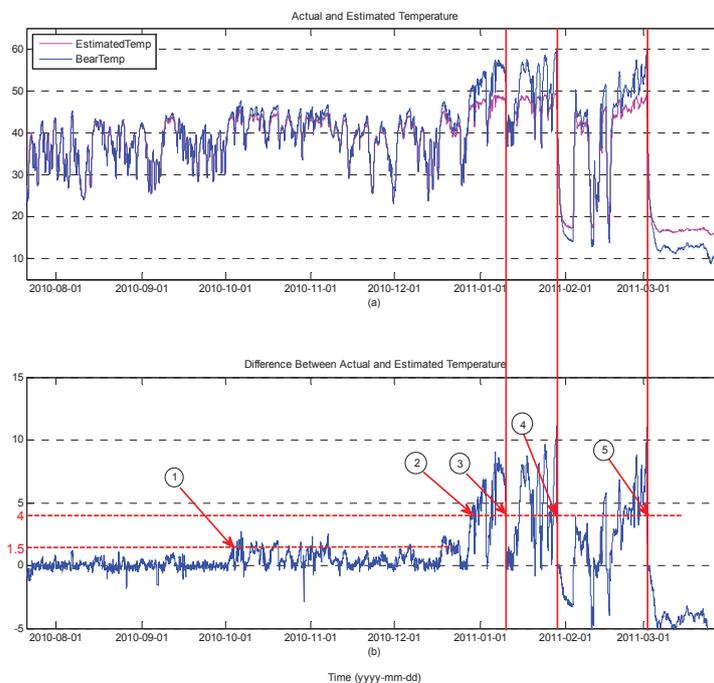
Fig. 7.5 Fault Detection Results of Rear Bearing

7.3.4   Discussion

This section mainly discusses whether the model established from the SCADA data of one turbine as in Fig. 7.2 can apply in fault detection for other turbines. Then, another turbine is selected to test for this purpose. However, there is no same fault in this wind farm and thus the SCADA data from a new turbine in normal condition. If the differences between actual and estimated temperatures are located within 1.5 ^0C as in Fig. 7.4, the conclusion can be drawn that the ANN model for rear bearing can be applied for fault detection and prediction for other turbines.

Fig. 7.6 shows the three months' SCADA data of a new turbine in same wind farm in normal condition. The data presented in this figure is also very varied: the turbine speed is varied with starting and stopping, the active power changes from 500 KW to more than 3000 KW, and the temperatures are also varied. These three months' SCADA data supposed to be large varied of the parameters in normal condition. Fig. 7.7 shows the results of ANN model using the SCADA data from the new turbine. The estimated rear bearing temperature is very close to the actual value. The maximum difference between estimated and actual temperatures is less than 1.5 ^0C which is an early warning level as shown in Fig. 7.5. This means the

new turbine is in normal condition. Therefore, the ANN model of rear bearing using SCADA data from one turbine can be applied for other turbines in same type.
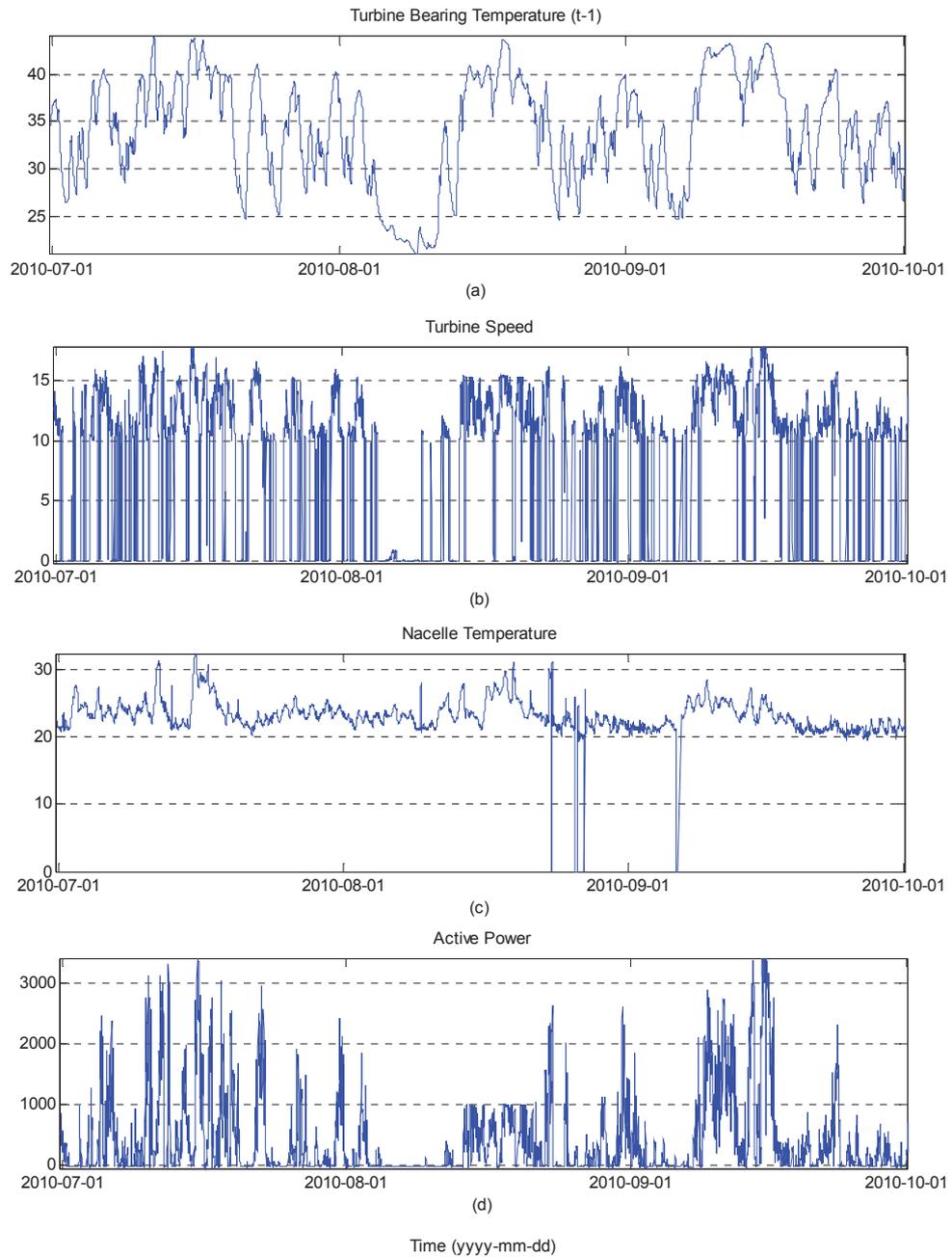
Fig. 7.6 Rear Bearing Model Testing Input Data of New Turbine

148

Fig. 7.7 Rear Bearing Model Output in Normal Condition of New Turbine

7.4 Summary

This Chapter described ANN technique for early fault prediction and identification for the main components of wind turbines especially for the bearings based on the existing SADA data collected by commercial SCADA system. The result shows that it can deal with large volume SCADA data and give the operators of wind farm very early warning and close alarm to help them make the right maintenance schedule and action decision in advance. In this way, the information presented to the operator is dramatically reduced without omitting useful information. The maintenance and operation cost can also be reduced by optimize the maintenance plan, staff and preparation of tools according to the early warning and alarm. The instance presented in this section only established a normal behaviour ANN model for one component, i.e. main shaft rear bearing. In the future, more components need to establish normal behaviour model. In this section, the ANN model established by one turbine tested in new one only in normal condition, and in the future, the test should be done in different conditions contain normal level, warning level and alarm level.

Chapter 7: Fault Prognosis based on Artificial Neural Network

8 Maintenance Scheduling Optimization based on Data Mining Techniques

8.1 Introduction

The range of maintenance cost is from 15% for manufacturing companies and 40% for iron and steel industry of the whole cost of manufactured parts and machines [Mobley, 1990]. The corresponding cost in United Stated is more than 200 billion dollars every year [Chu et al., 1998]. This shows the significance of maintenance in the viewpoint of economy.

Generally, there are three different types of maintenance strategies. The first one is called Corrective Maintenance (CM) which is similar to repair work, is undertaken after a breakdown or when obvious failure has been located. However, CM at its best should be utilized only in non-critical areas where capital costs are small, consequences of failure are slight, no safety risks are immediate, and quick failure identification and rapid failure repair are possible. The second one is called preventive maintenance which is scheduled without the occurrence of any monitoring activities. The scheduling can be based on the number of hours in use, the number of times an item has been used, or the number of kilometers the items has been used, according to prescribed dates. The preventive maintenance may cause much more or much less maintenance activities, which may cause more maintenance cost or hazard of personnel and equipment. The last one is called Predictive Maintenance (PM) which is a set of activities that detect changes in the physical condition of equipment (signs of failure) in order to carry out the appropriate maintenance work for maximizing the service life of equipment without increasing the risk of failure. PM is a dynamic schedule according to the state of machines from continuous and/or periodic inspection. It utilizes the product degradation information extracted and identified from on-line sensing techniques to minimize the system downtime by balancing the risk of failure and achievable profits.

PM has some advantages over other maintenance policies: 1) Improving availability and reliability by reducing downtime; 2) Enhancing equipment life by reducing wear from frequent rebuilding, minimizing potential for problems in disassembly and reassembly and detecting problems as they occur; 3) Saving maintenance costs by reducing repair costs, reducing overtime and reducing parts inventory requirements; 4) Decreasing number of maintenance operations causes decreasing of human error influence. However, there are still some challenges of PM: 1) Initiating PM is costly because the cost of sufficient instruments could be quite large especially if the goal is to monitor already installed equipment; 2) The goal of PM is accurate maintenance, but it is difficult to achieve for the complexity of equipment and environment; 3) Introducing PM will invoke a major change in how maintenance is performed, and potentially to the whole maintenance organization in a company. Organizational changes are in general difficult. The objective of maintenance scheduling optimization is to optimize maintenance

scheduling in order to maximize the whole profit, ensure safety and increase availability.

Mathematically, the maintenance scheduling problem is a multiple-constraint, non-linear and stochastic optimization problem. This kind of problem has been studied for several decades and many kinds of different methods have been applied to solve it. Two methods for PM optimization had been developed during 1980s. The first method [Perla, 1984; Walker, 1987] performs cost/benefit analysis of each analyzed piece of manufacturing equipment. It is based on identifying important equipment firstly, and then predicting its future performance with and without changes in the regularly scheduled maintenance program. The second approach is the Reliability-Centered Maintenance (RCM) [Crellin, 1986; Hook et al., 1987; Vasudevan, 1985]. This methodology was adopted from the commercial air transport industry. It is based on a series of orderly steps, including identification of system/subsystem functions and failure modes, prioritization of failures and failure modes (using a decision logic tree), and finally selection of PM tasks that are both applicable (i.e. have the potential of reducing failure rate) and effective (i.e. economically worth doing). In the last two decades, many kinds of intelligent computational methods, such as the artificial neural network method, simulated annealing method, expert system, fuzzy systems and evolutionary optimization, have been applied to solve the maintenance scheduling problem and obtained many very exciting results [Huang, 1998; Miranda et al., 1998; Satoh & Nara, 1991; Sutoh et al., 1994; Yoshimoto et al., 1993]. And also, with the rapid development of the evolutionary theory, genetic algorithms (GAs) had become a very powerful optimization tool and obtained wide application in this area [Arroyo & Conejo, 2002; Back et al., 1997; Huang et al., 1992; Lai, 1998; Lee & Yang, 1998; Y. Wang & Handschin, 2000]. In recently years, several new intelligent computational methods such as Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) have been applied in preventive maintenance scheduling [Benbouzid-Sitayeb et al., 2008; Pereira et al., 2010; Yare & Venayagamoorthy, 2010].

All the above methods of maintenance scheduling are based on the specified time periods other than based on the condition of the equipment or facilities. PM is a good strategy which could be used to improve reliability and increase useful life of the equipment and reduce the cost of maintenance according to the condition of machine. When the condition of a system, such as its degradation level, can be continuously monitored, PM policy can be implemented, according to which the decision of maintaining the system is taken dynamically on the basis of the observed condition of the system. Recently, genetic algorithms, Monte Carlo method, Markov and semi-Markov methods are applied in PM [Amari et al., 2006; Barata et al., 2001, 2002; Be′ renguer et al., 2000; Grall et al., 2008; Marseguerra et al., 2002]. However, there are very few literatures on applying the intelligent computational methods in predictive maintenance based on the conditions (degradation) of monitored machines.

This Chapter will build PM scheduling models and optimize it using Swarm Intelligence algorithms.

8.2 Predictive Maintenance Scheduling Optimization Based on Swarm Intelligence

Swarm Intelligence (SI) is an innovative distributed intelligent paradigm for solving optimization problems that originally took its inspiration from the biological examples by swarming, flocking and herding phenomena in vertebrates. Particle Swarm Optimization (PSO) incorporates swarming behaviors observed in flocks of birds, schools of fish, or swarms of bees, and even human social behavior, from which the idea is emerged. Ant Colony Optimization (ACO) deals with artificial systems that are inspired from the foraging behavior of real ants, which are used to solve discrete optimization problems. Bee Colony Algorithm (BCA) is an optimization algorithm based on the intelligent foraging behavior of honey bee swarm, proposed by Karaboga in 2005 [Karaboga, 2005]. These optimization algorithms are metaheuristic which can solve the difficult optimization problems even the problem is NP problem. They can easily to be applied in maintenance scheduling optimization.

The maintenance scheduling mentioned in this Chapter does not refer to the scheduling for one machine or one component in its life cycle but to number of machines or components in specific time duration in order to reduce cost and increase productivity or profit. Fig. 8.1 shows the scheme of maintenance scheduling optimization. The results of fault diagnosis and prognosis are the key information of the maintenance scheduling optimization. The objective of maintenance optimization is to maximize or minimize the fitness function with some constraints such as crew constraint and maintenance window. The following sections will show how the swarm intelligence techniques work in maintenance scheduling optimization through some case studies.
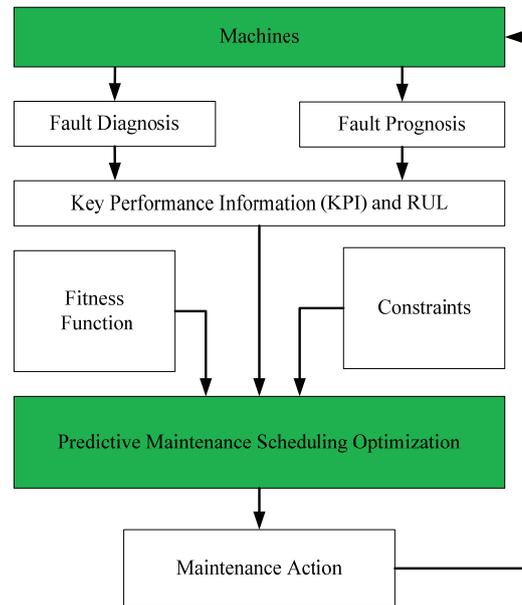
Fig. 8.1 Maintenance Scheduling Optimization Scheme

8.3 Generating Unit Maintenance Scheduling (GMS) using PSO

Power generating companies must generate sufficient electrical power to cater for
the varying demands of consumers. Electricity cannot be easily and cheaply stored,
so it must be continuously generated based on the customers' demand. With the
increasing demand of electricity, the generating unit maintenance scheduling (GMS)
of power system has become a complex, multi-object-constrained optimization
problem. Within the last three decades, several techniques have appeared in the
literature addressing such optimization problems under different scenarios
[Marwali & Shahidehpour, 2000; Negnevitsky & Kelareva, 1999]. The primary
goal of the GMS is the effective allocation of generating units for maintenance
while ensuring high system reliability, reducing production cost, prolonging
generator life time subject to some units and system constraints [Yare et al., 2008].

In order to obtain an approximate solution of a complex GMS, some new concepts
have been proposed in recent years. They include applications of probabilistic
approach [Billinton & Abdulwhab, 2003], simulated annealing [Satoh & Nara,
1991], decomposition technique [Yellen et al., 1992] and genetic algorithm (GA)
[Firmo & Legey, 2002]. A flexible GMS that considered uncertainties is proposed
with a fuzzy 0-1 integer programming technique adopted and applied to the Taiwan
power system. The application of GA to GMS has been compared with and
confirmed to be superior to other conventional algorithms such as heuristic
approaches and branch-and-bound (B&B) in the quality of solutions [Firmo &
Legey, 2002]. However, the application of particle swarm optimization (PSO) and

their variants to GMS has not been fully explored in the literature. This section is retrieved from [Zhang & Wang, 2010].

8.3.1 Fitness function and Constraints of GMS

Generally, there are two main categories of objective functions in GMS problems, namely based on reliability and economic cost. The reliability criteria of levelling reserve generation for the entire period of study is considered in this paper. As an objective function of the GMS problems, we establish annual supply reserve ratio levelling one of the deterministic index. Because algorithms for levelling supply reserve ratio is easy to implement without considering probabilistic simulation procedures operation cost, it is possible to establish an annual GMS problem (52-week horizon). However, it has a weak point in not considering probabilistic conditions such as generators' forced outage. Actually generation companies have been utilizing minimizing annual supply reserve ratio more than probabilistic index methods. For our research, we just focus on PSO algorithm accessing to GMS problem, so it is enough to formulate the objective function as an annual supply ratio levelling.

The problem studied here was solved by minimizing the annual supply reserve ratio. The problem has a number of units and system constraints to be satisfied which were described as follows:

- Load constraints – total capacity of the units running at any interval should be not less than predicted load at that interval.
- Crew constraint – for each period, the capacity of maintenance units cannot exceed the maximum available maintenance capacity considering crew in this period.
- Start week of maintenance – Each unit has its maintenance periods, the maintenance schedule cannot exceed these periods.
- Maintenance window constraints – defines the starting of maintenance at the beginning of an interval and finish at the end of the same interval which may contain one or several weeks. The maintenance cannot be aborted or finished earlier than scheduled.

The objective function to be minimized is given by Eq. (8.1) subject to the constraints given be Eq. (8.2)-(8.5).

$$
\begin{aligned}
it &= Min \sum_{t=1}^{T} \left[\left(\frac{Ac_t - L_t}{L_t} \right) - \frac{1}{T} \sum_{t=1}^{T} \left(\frac{Ac_t - L_t}{L_t} \right) \right]^2 \\
&= Min \sum_{t=1}^{T} \left[\left(\frac{IC - SL_t - L_t}{L_t} \right) - \frac{1}{T} \sum_{t=1}^{T} \left(\frac{IC - SL_t - L_t}{L_t} \right) \right]^2
\end{aligned}
\tag{8.1}
$$

Subject to load constraint

$$IC - SL_t > SL_t \tag{8.2}$$

Subject to crew constraint

$$SL_t < CR_t,? \quad SL_t = \sum_{t=1}^{N} P_{jt} C_j \tag{8.3}$$

Subject to start week of maintenance

$$S_j^{min} < S_j < S_j^{max} \tag{8.4}$$

Subject to maintenance window

$$\begin{cases} t \le S_j \, or \, t \ge S_j + M_j & no \, maintenance \\ S_j < t < S_j + M_j & maintenance \end{cases} \tag{8.5}$$

Where:

T : Length of the maintenance planning scheduling (normally 52 weeks);

AC_t : Available generation capacity at t^{th} week;

L_t : Load demand at t^{th} week;

IC : Total Installed Capacity;

SL_t : Capacity loss in t^{th} week because of maintenance;

CR_t : Maximum available maintenance capacity at t^{th} week considering crew [MW];

S_j : Starting week for maintenance scheduling of jth unit;

S_j^{min} : Feasible minimum starting week for maintenance scheduling of jth unit;

S_j^{max} : Feasible maximum starting week for maintenance scheduling of jth unit;

C_j : Capacity of jth unit;

$P_{jt} = P\{t \mid S_j < t < S_j + M_j\}?$: Whether the jth unit maintenance in t^{th} week.

8.3.2 Improved PSO (IPSO) Algorithm

PSO performs well in the early iterations, but they have problems approaching a near-optimal solution. If a particle's current position accords with the global best and its inertia weight multiply previous velocity is close to zero, the particle will only fall into a specific position. If their previous velocities are very close to zero, all the particles will stop moving around the near-optimal solution, which may lead to premature convergence of algorithm. All the particles have converged to the best position discovered so far which may be not the optimal solution. So, an improved PSO (IPSO) is proposed here.

In IPSO, before updating the velocities and positions in every iteration, the particles are ranked according to their fitness values in descending order. Select the first part of particles (suppose mutation rate is α, fist part is $(1-\alpha)$ and put them into the next iteration directly. Regenerate the rest part of particles (α) randomly. In this case, we can regenerate the positions and velocities according to the following equations instead of Eq. (3.23)-(3.24):

$$x_{id} = round\left(rand \times \left(S^{max}(j) - S^{min}(j)\right) + S^{min}(j)\right) \tag{8.6}$$

$$v_{id}(t) = v_{max} - round\left(rand \times 2v_{max}\right) \qquad v_{id}(t) \in [-v_{max}, v_{max}] \tag{8.7}$$

8.3.3   Case Study and Results

In order to investigate the performance of IPSO for the GMS problem, a test system comprising 32 units over a planning period of 52 weeks was used. The case study is described below and implemented in a MATLAB environment.

There are 32 generating units, annual peak load demand is 2,850 MW, and installed capacity is 3,450 MW. The weekly peak loads in present of annual peak are shown in Table 8.1. The specific data of the generators are shown in Table 8.2 which include capacity (MW), maintenance period and load constraints. The value of crew constraint is constant at 800 MW.

To implement PSO and IPSO, a population size of 150 particles was chosen to provide sufficient diversity into the population taking into account the dimensionality and complexity of the problem. This population size ensured that the domain was examined in full but at the expense of an increase in execution time. The other parameters of PSO and IPSO were: c1 = c2 = 2.0, ω = 1.2 - 0.8 with linearly decreasing, total iteration = 300 and V ∈ [-3, 3].

Annual supply reserve ratio values by the change of the number of iteration are shown in Fig. 8.2. We compared simulation results between the PSO and IPSO algorithms. We can see from this figure, the IPSO algorithm has a better performance than PSO in GMS problems to find optimal solutions. The particles of IPSO have a higher possibility to find optimal solution than those of PSO. The optimal solutions of GMS problems using PSO and IPSO are shown in Table 8.3. It contains global particles of PSO and IPSO respectively which have a best maintenance period satisfying maintenance continuity and crew constraints, etc.

Chapter 8: Maintenance Scheduling Optimization based on Data Mining Techniques

Table 8.1 Weekly Peak Load in Percent of Annual Peak (%)

Week	Load	Week	Load	Week	Load	Week	Load
1	86.2	14	75.0	27	75.5	40	72.4
2	90.0	15	72.1	28	81.6	41	74.3
3	87.8	16	80.0	29	80.1	42	74.4
4	83.4	17	75.4	30	88.0	43	80.0
5	88.0	18	83.7	31	72.2	44	88.1
6	84.1	19	87.0	32	77.6	45	88.5
7	83.2	20	88.0	33	80.0	46	90.9
8	80.6	21	85.6	34	72.9	47	94.0
9	74.0	22	81.1	35	72.6	48	89.0
10	73.7	23	90.0	36	70.5	49	94.2
11	71.5	24	88.7	37	78.0	50	97.0
12	72.7	25	89.6	38	69.5	51	100
13	70.4	26	86.1	39	72.4	52	95.2

Table 8.2 Data of Generators

Generator (Unit)	Capacity (MW)	Maintenance Window	Maintenance period	Generator (Unit)	Capacity (MW)	Maintenance Window	Maintenance period
1	12	1-52	2	17	76	18-29	3
2	12	1-52	2	18	76	18-29	3
3	12	1-52	2	19	76	18-29	3
4	12	1-52	2	20	100	18-29	3
5	12	1-52	2	21	100	18-29	3
6	20	18-29	2	22	100	18-29	3
7	20	18-29	2	23	155	1-52	4
8	20	18-29	2	24	155	1-52	4
9	20	18-29	2	25	155	1-52	4
10	50	41-52	2	26	155	1-52	4
11	50	41-52	2	27	197	1-52	4
12	50	41-52	2	28	197	1-52	4
13	50	1-27	2	29	197	1-52	4
14	50	1-27	2	30	350	1-52	5
15	50	1-27	2	31	400	1-52	6
16	76	18-29	3	32	400	1-52	6
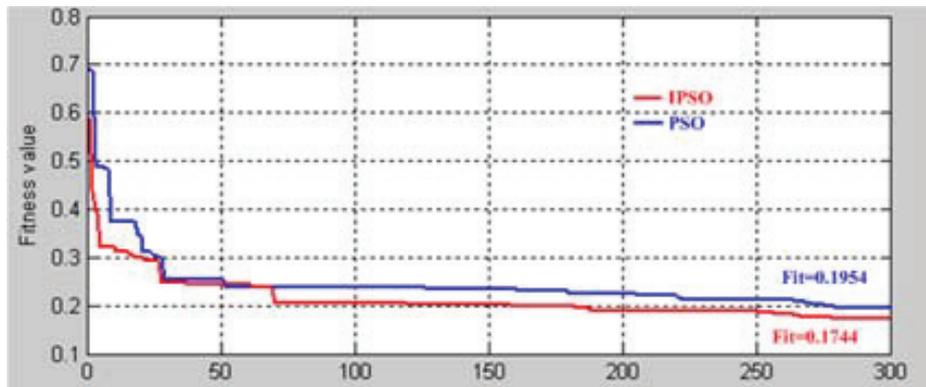
Fig. 8.2 Fitness Value by the Change of the Number of Iteration

Table 8.3 Result (Maintenance period)

Unit	Maintenance period (week)		Unit	Maintenance period (week)	
	PSO	IPSO		PSO	IPSO
1	35,36	14, 15	**17**	26,27,28	26,27,28
2	21,22	21,22	**18**	18,19,20	26,27,28
3	31,32	8,9	**19**	26,27,28	18,19,20
4	21,22	35,36	**20**	26,27,28	26,27,28
5	17,18	21,22	**21**	26,27,28	26,27,28
6	27,28	21, 22	**22**	21,22,23	18,19,20
7	21,22	27,28	**23**	14,15,16,17	34,35,36,37
8	27,28	21,22	**24**	14,15,16,17	29,30,31,32
9	18,19	27,28	**25**	34,35,36,37	14,15,16,17
10	41,42	50,51	**26**	14,15,16,17	6,7,8,9
11	41,42	50,51	**27**	10,11,12,13	10,11,12,13
12	41,42	41,42	**28**	40,41,42,43	14,15,16,17
13	6,7	14,15	**29**	38,39,40,41	37,38,39,40
14	14,15	3,4	**30**	29,30,31,32,33	9, 10, 11, 12, 13
15	21,22	14, 15	**31**	34,35,36,37,3839	38, 39, 40, 41, 42, 43
16	18,19,20	21,22,23	**32**	8,9,10,11,12,13	31, 32, 33, 34, 35, 36

8.4 Dynamic Condition-Based Maintenance Scheduling using BCA

8.4.1 Model of Condition based PM

In order to show the general idea of applying BCA in condition-based PM scheduling, a manufacturing model is built assuming the features of the system that

we analyze. There are several assumptions of the manufacturing system as following:

1) The manufacturing system is subjected to deterioration.
2) Periodically the system is under inspection and each inspection reveals the system deterioration state perfectly.
3) Machine inspection is planned at the beginning of each period.
4) The inspection time is very short and can be ignored compared to the whole period.
5) Following an inspection based on the current state of machine (S_i), one of the following action is taken:
   - $0 \leq S_i < S_k$: no maintenance is performed (S_k is PM threshold);
   - $S_k \leq S_i < S_n$: PM is planed (S_n is CM threshold) but is not always performed ;
   - $S_i \geq S_n$: CM has to be performed.
6) Following a PM or CM, the machine is restored to an as-good-as-new condition.
7) The duration of PM action is much less than that of CM action for a same machine.

8.4.1.1 Modeling of Manufacturing System

The manufacturing system has a number of machines marked as M , and for each machine, the productivity is $Prod_i$. Therefore, the maximum productivity of this system can be expressed as Eq. (8.8) while its real total productivity can be expressed as Eq. (8.9).

$$Prod_{max} = \sum_{i=1}^{M} Prod_i \qquad (8.8)$$

$$Prod_{tot} = \sum_{i=1}^{M} (Prod_i \bullet \omega_i) \qquad (8.9)$$

where: $Prod_i$ is the productivity of ith machine. ω_i is the coefficient of ith machine productivity. The value of ω_i is 1 if the ith machine is not under any kind of maintenance, the value is 0 if it is under CM action, and the value is 0.5 if the machine is under a PM action in a period.

8.4.1.2 Modeling of Equipment Inspection

The value of the state can belong to arrange from 0 to 1 which represent the perfect state to the totally failure of the component. The state of machine is can be discretized as $S_1, S_2, \cdots, S_n$ which S_1 can be set equal 0 while S_n can be set equal 1 or a value very closed to 1 (for example 0.98 which is the CM threshold). In this model, the beginning condition is considered for each interval. During each period, degradation of each machine is independent and random distribution according to Poisson distribution. At the start of each period, there is the inspection of a machine and the obtaining of the value of the state S of a machine as shown in Fig.

8.3. The states for all the machines can be used as parameters in predictive maintenance scheduling.
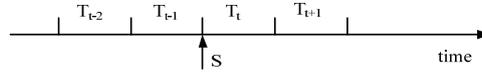
Fig. 8.3 Inspection Point Schematic Diagram

8.4.1.3 Deterioration Model for Each Machine

Deterioration means a process where the important parameters of a system gradually worse. If left unattended, the process will lead to deterioration failure. Therefore, the deterioration has to be considered when a maintenance policy needs to be employed. Fig. 8.4 shows the deterioration model of a machine. The state S of a machine can be a value among $[S_1, S_n]$. In Fig. 8.4, $S_i (i = 1, 2, \cdots, n)$ is the predefined state of a machine, S_k is while S_n is CM threshold. P_{ij} is the transition probability for the state from S_i to S_j in one period. The PM should be planned when the state is between S_k and S_n. If the state goes to S_n, the CM action must be performed which means $P_{n1} = 1$. The state transition matrix P can be expressed as Eq. (8.10) with the constraint of Eq. (8.11).
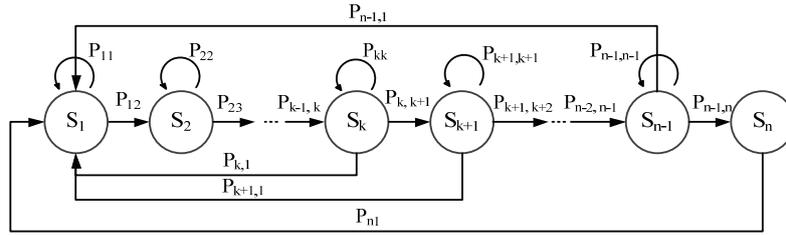
Fig. 8.4 Degradation Model for One Machine

$$P = \begin{bmatrix} P_{11} & P_{12} & \cdots & P_{1n} \\ P_{21} & P_{22} & \cdots & P_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ P_{n1} & P_{n2} & \cdots & P_{nn} \end{bmatrix} \tag{8.10}$$

$$\sum_{j=1}^{n} P_{ij} = 1 \quad \forall i = 1, \cdots, n \tag{8.11}$$

This model is very similar with the Markov model in lack of a random variable of inspection time. With the Markov, the mean time between CM and mean time between PM can be estimated [Amari et al., 2004]. But with the Markov model, the accumulative error is very difficult to eliminate. The result is only the mean time between CM and mean time between PM rather than the real plan or scheduling of CM or PM. With that result, the maintenance action CM and PM could be much more or less than it necessary because of uncertainty of mechanical products. Therefore, the inspection action is performed in the beginning of every

period as mentioned in section 8.4.1.2. What's more, in this model, there is no any CM or PM action when the state of the machine is in the range between S_1 and S_{k-1}. The PM plan is made when the state of the machine is in range between S_k and S_{n-1}, and as mention above, the CM action is performed if and only if the state of machine reach or exceed S_n. To simplify the analysis, for the element values in the state transition matrix in Eq. (8.10), from the S_1 to S_{k-1}, only P_{ii} and $P_{i,i+1}(i=1,2,\cdots,k-1)$ have positive values and others are all zero, while from the S_k to S_{n-1}, only P_{ii}, $P_{i,i+1}$ and $P_{i1}(i=k,k+1,\cdots,n-1)$ have positive values and the others are all zero as well. The new equation can be expressed as Eq. (8.12).

$$P = \begin{bmatrix} P_{11} & P_{12} & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & P_{22} & P_{23} & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & P_{33} & P_{34} & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ P_{k1} & 0 & 0 & 0 & 0 & P_{kk} & P_{k,k-1} & \cdots & 0 & 0 \\ P_{k+1,1} & 0 & 0 & 0 & 0 & 0 & P_{k+1,k+1} & \cdots & 0 & 0 \\ P_{k+2,1} & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ P_{n-1,1} & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & P_{n-1,n-1} & P_{n-1,n} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \tag{8.12}$$

The ideal values of all the elements in Eq. (8.12) for the perfect deterioration model are express from Eq. (8.13) and Eq. (8.14).

$$P_{ii} = 0 \quad \& \quad P_{i,i+1} = 1, \quad i = 1 \cdots k \tag{8.13}$$

$$P_{ii} = 0 \quad \& \quad (P_{i,i+1} = 1 \quad or \quad P_{i1} = 1), \quad i = k, k+1 \cdots n-1 \tag{8.14}$$

For the state of S_n in Eq. (8.12), $P_{n1} = 1$ and all values of other elements are 0 which mean that when the state reach S_n, CM has to be performed. These values could be a real situation of a manufacturing machine but it is difficult make the values reality. To achieve this point, the values of states from S_1 to S_n should be adjusted after a number of periods by statistics.

8.4.1.4 Modelling of Cost Function

There are many types of costs for each period which are analysed one by one as in this section. All the costs calculated in this section are just for only one period.

Production Cost: it is due to the amount products produced by the manufacturing system which means how much money it need cost to produce the amount of products.

$$C_{prod} = Prod_{tot} \bullet C_{piece} \tag{8.15}$$

where C_{prod} represents the production cost for a period while C_{piece} represents the cost for producing one piece.

Maintenance Cost: it is due to the performing PM and CM, which means the how much money needed to perform the PM and CM.

$$C_M = \sum_{i=1}^{M} (CM_i \bullet C_{ci} + PM_i \bullet C_{pi}) \qquad (8.16)$$

where CM_i represents if the ith machine is under the CM (0 means no CM action while 1 means under that action). PM_i represents if the ith machine is under the PM (0 means no PM action while 1 means under that action). C_{ci} and C_{pi} represent the costs of one CM and PM action respectively for i^{th} machine.

Total Cost: it is the total cost for one period.

$$C_{tot} = C_{prod} + C_M + CI \qquad (8.17)$$

where C_{tot} is the total cost in one period while CI is the inspection cost. Because in this model all machines are inspected for every period, the value of CI is fixed.

8.4.1.5 Modelling of Profit for the Manufacturing System

After above analysis, the total profit for one period can be calculated using Eq. (8.18). This equation could be an objective function for optimization. The total number of produced products in the period should be more than a minimum number which can be describe as Eq. (8.19). Furthermore, the number of CM and PM have a limitation because of the resources limitation, such as repairers and tools limitation.

$$Profit = Prod_{tot} \bullet Pr - C_{tot} = Prod_{tot} \bullet Pr - (C_{prod} + C_M + CI)$$
$$= Prod_{tot} \bullet Pr - [Prod_{tot} \bullet C_{piece} + \sum_{i=1}^{M} (CM_i \bullet C_{ci} + PM_i \bullet C_{pi}) + CI] \qquad (8.18)$$

$$Prod_{tot} \geq Prod_{min} \qquad (8.19)$$

$$\sum_{i=1}^{M} (CM_i + PM_i) \leq M_{max} \qquad (8.20)$$

where Pr is the price for one piece of product, $Prod_{min}$ is the minimum amount products limitation of one period, and M_{max} is a limitation of the maximum maintenance action can be performed. In this model, to find the optimal dynamic predictive maintenance plan for each period, Eq. (8.18) could be an objective function and Eq. (8.19) and Eq. (8.20) could be two constraints. The aim is to make PM maintenance scheduling to obtain maximum *Profit* with two constraints of Eq. (8.19) and Eq. (8.20).

8.4.2   Numerical Examples

In order to investigate the performance of BCA for the condition-based PM scheduling problem, a test system comprising 30 machines is used. According to the conditions at the start of period, a dynamic PM scheduling is made period by period. The case study is described below and implemented in a MATLAB environment. In this case, the number of machine is 30, and the machine parameters are shown in Table 8.4. In the table, the $Prod_i$ means the one day productivity for i^{th} machine. The problem of this case could be described as: making a PM and CM scheduling decision for 30 machines in a week according to the initial state of each machine.

There is no mathematical method to select the best population size of the BCA. However, there are some empirical parameters from experience. In this example, the value of population is set to 20. The profit (fitness value) for one period (a week) by the change of the number of iteration is shown in Fig. 8.5. The result of PM decision and CM decision are shown in Table 8.5. In the table, the values of PM are 0 or 1 which mean perform or not perform the PM action. The CM value is the same mean as PM. The optimal fitness value of this numerical example is 3081390. The result shows that BCA can make the dynamic PM scheduling optimization very effective and clear with PM model.
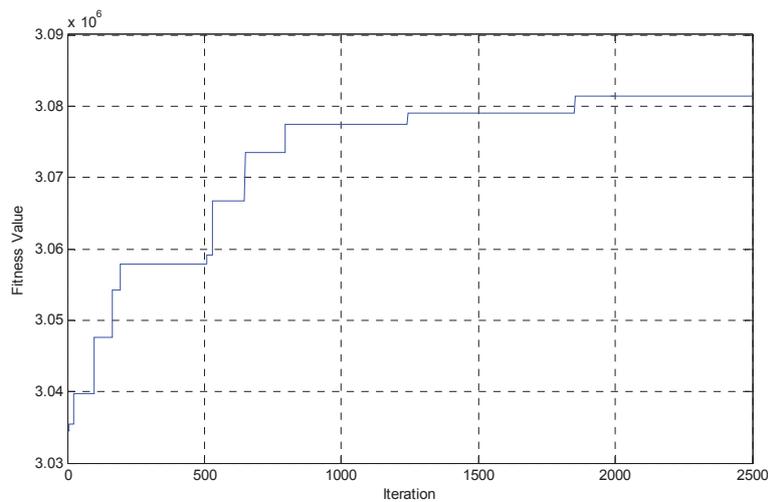
Fig. 8.5 Fitness Value by the Change of Iterations

Table 8.4 Machine Parameters

Machine	$Prod_i$	C_{piece}	Pr	CI	S_i	C_{pi}	C_{ci}
M1	200	70	140	200	0.46	4000	10000
M2	200	70	140	200	0.51	4000	10000
M3	200	70	140	200	0.94	4000	10000
M4	200	70	140	200	0.55	4000	10000
M5	200	70	140	200	0.93	4000	10000
M6	200	70	140	100	0.96	6000	12000
M7	200	70	140	100	0.42	6000	12000
M8	200	70	140	100	0.66	6000	12000
M9	200	70	140	100	0.35	6000	12000
M10	200	70	140	100	0.35	6000	12000
M11	300	60	140	210	0.86	6500	16000
M12	300	60	140	210	0.95	5500	12000
M13	300	60	140	210	0.72	7800	16000
M14	300	60	140	210	0.82	8000	15000
M15	300	55	140	150	0.71	9000	17000
M16	300	55	140	150	0.64	6000	13000
M17	300	55	140	180	0.95	7000	15000
M18	300	55	140	180	0.75	8000	16000
M19	300	55	140	170	0.92	8000	16000
M20	300	55	140	170	0.66	8000	16000
M21	400	70	140	220	0.33	9000	20000
M22	400	70	140	220	0.5	10000	17000
M23	400	70	140	220	0.91	7500	20000
M24	400	70	140	220	0.6	10000	16000
M25	400	70	140	220	0.37	8400	20000
M26	400	70	140	220	0.97	10000	18000
M27	400	60	140	200	0.4	9000	18000
M28	400	60	140	200	0.46	9000	17000
M29	400	60	140	200	0.91	9000	18000
M30	400	60	140	200	0.7	9000	16000

Table 8.5 Results of PM and CM by BCA

Machine	PM	CM	Machine	PM	CM	Machine	PM	CM
M1	0	0	M11	1	0	M21	0	0
M2	0	0	M12	0	1	M22	0	0
M3	1	0	M13	1	0	M23	1	0
M4	0	0	M14	1	0	M24	0	0
M5	1	0	M15	1	0	M25	0	0
M6	0	1	M16	0	0	M26	0	1
M7	0	0	M17	0	1	M27	0	0
M8	0	0	M18	1	0	M28	0	0
M9	0	0	M19	1	0	M29	1	0
M10	0	0	M20	0	0	M30	1	0

8.5 Routing and Scheduling Optimization of Maintenance Flee (RSOM) for Offshore Wind Farm

Wind energy industry has experienced an extensive and worldwide growth during the past years. Certain forecasts indicate that the share of wind in Europe's energy production will reach up to 20% in the close future [Krohn et al., 2007]. The efficient operation of installed turbines has an increasing significance. Among operational decisions, the planning and scheduling of maintenance tasks is decisive regarding both turbine availability and operational costs. Considering the spread of offshore installations and the fact that their operational costs including specialized support resources for offshore operations, such as service vessels and personnel, can be estimated to be five to ten times more expensive than that of the onshore farms [Bussel & Zaaijer, 2001; Markard & Petersen, 2009], maintenance scheduling will receive even more emphasis. Meanwhile, the support resources are often restricted by the environmental conditions at the site, and certain operations are allowed only in short weather windows. Missing the weather window may lead to production interruption and economic loss.

The Chapter aims to investigate an operational decision problem, i.e. routing and scheduling of a maintenance fleet for offshore wind farms which can be used to avoid a time-consuming process of manually planning the scheduling and routing with a presumably suboptimal outcome. Mathematical model of RSOM is retrieved from a literature [Dai, 2014] and then a swarm intelligence, i.e. Ant Colony Optimization (ACO) is modified as Duo-ACO to be applied to solve this problem.

8.5.1 Mathematical Model of RSOM

Let there are n offshore wind turbines (OWTs) indexed by i. Associate to the delivery location of OWT i a node i, and to its pick up location a node $n+i$. Also associate to the harbor, nodes 0 and $2n + 1$. The definitions of the variables can be given as following:

Chapter 8: Maintenance Scheduling Optimization based on Data Mining Techniques

Sets:

Z^-: the set of delivery nodes, $Z^- = \{1, 2, 3, \cdots, n\}$.

Z^+: the set of pick up nodes, $Z^+ = \{n+1, n+2, \cdots, 2n\}$.

$Z = Z^- \cup Z^+$.

$Z^V \subseteq Z$: the set of nodes that require the vessel present during the maintenance operations.

N: the set of all the nodes; $N = Z \cup [0, 2n+1]$.

V: the set of service vessels.

T: the set of days in the planning period; $T = \{1, 2, \cdots\}$ represents the length of the period.

Constants

T_{vij}: the time (hours) for vessel v traversing arc (i, j).

C_v: the traveling cost of vessel v per hour.

T_i^M: the time needed for performing the maintenance task on turbine i; $T_0^M = T_{2n+1}^M = 0$.

L_i: the weight of spare parts and equipment for maintenance on turbine i.

P_i: the required personnel number for maintenance on turbine i.

T_{vd}^{MAX}: the maximum working hours on day d for vessel v, which is used as the weather limitation for different vessels.

L_v^{MAX}: the load capacity of vessel v.

P_v^{MAX}: the personnel capacity of vessel v.

T_i^{LATE}: the latest day to perform the maintenance task on turbine i without incurring a penalty cost.

C_i^{PE}: the penalty cost per day for the delaying maintenance task on turbine i beyond T_i^{LATE}.

Decision variables

$$x_{vijd} = \begin{cases} 1, & \text{vessel } v \text{ travels from node } i \text{ to node } j \text{ on maintenance day } d \\ 0, & \text{otherwise} \end{cases}$$

y_i: the number of delayed days for maintenance task on turbine i.

t_{vid}: the time at which vessel v visits turbine i on maintenance day d.

167

k_{vid}: the total load weight on vessel v just after it leaves node i on maintenance day d.

q_{vid}: the total personnel number on vessel v just after it leaves node i on maintenance day d.

Objective function

$$\min\left\{\sum_{v\in V}\sum_{d\in T}C_v t_{v(2n+1)d} + \sum_{i\in Z^-}C_i^{PE}y_i\right\} \tag{8.21}$$

Constraints

$$\sum_{j\in N}\sum_{v\in V}\sum_{d\in T}x_{vijd} = 1, \qquad \forall i\in Z, \tag{8.22}$$

$$\sum_{i\in N}x_{v0id} = 1, \qquad \forall v\in V, d\in T, \tag{8.23}$$

$$\sum_{j\in N}x_{vjid} = \sum_{j\in N}x_{vijd}, \qquad \forall v\in V, d\in T, i\in N, \tag{8.24}$$

$$\sum_{i\in N}x_{vi(2n+1)d} = 1, \qquad \forall v\in V, d\in T, \tag{8.25}$$

$$\sum_{j\in N}x_{vjid} = \sum_{j\in N}x_{v(n+i)jd}, \qquad \forall v\in V, d\in T, i\in Z^-, \tag{8.26}$$

$$\sum_{v\in V}\sum_{d\in T}x_{vi(i+n)d} = 1, \qquad \forall i\in Z^V, \tag{8.27}$$

$$t_{v(n+i)d} - t_{vid} \geq T_i^M, \qquad \forall i\in Z^-, v\in V, d\in T, \tag{8.28}$$

$$\sum_{j\in N}\sum_{v\in V}\sum_{d\in T}(d\bullet x_{vijd} - y_i) \leq T_j^{LATE}, \qquad \forall i\in Z^-, \tag{8.29}$$

$$(t_{vid} + T_{ij} - t_{vjd})x_{vijd} \leq 0, \qquad \forall i,j\in N, v\in V, d\in T, \tag{8.30}$$

$$\sum_{i\in Z^-}\sum_{j\in N}L_i x_{ijvd} \leq L_v^{MAX}, \qquad \forall v\in V, d\in T, \tag{8.31}$$

$$(k_{vid} - L_j - k_{vjd})x_{vijd} = 0, \qquad \forall i\in Z^-, j\in N, v\in V, d\in T, \tag{8.32}$$

$$(k_{vid} - k_{vjd})x_{vijd} = 0, \qquad \forall i\in N\backslash Z^-, j\in N, v\in V, d\in T, \tag{8.33}$$

$$(q_{vid} - P_j - q_{vjd})x_{vijd} = 0, \qquad \forall i\in Z^-, j\in N, v\in V, d\in T, \tag{8.34}$$

$$(q_{vid} + P_j - q_{vjd})x_{vijd} = 0, \qquad \forall i\in Z^+, j\in N, v\in V, d\in T, \tag{8.35}$$

$$0 \leq k_{vid} \leq L_v^{MAX} \qquad \forall i\in N, v\in V, d\in T, \tag{8.36}$$

$$0 \leq q_{vid} \leq P_v^{MAX} \qquad \forall i\in N, v\in V, d\in T, \tag{8.37}$$

$$t_{v(2n+1)} \leq T_{vd}^{MAX} \qquad \forall v\in V, d\in T, \tag{8.38}$$

$$t_{v0d} = 0 \qquad \forall v\in V, d\in T, \tag{8.39}$$

168

$$y_i \geq 0, \quad \forall i \in Z^- \tag{8.40}$$

Constraints -

1) Eq. (8.22) ensure that each OWT is visited only once for delivery and once for pick up.
2) Eq. (8.23) and (8.25) ensure that each vessel leaves and returns the harbor only once every day.
3) Eq. (8.24) and (8.26) ensure flow conservation at each node.
4) Eq. (8.27) means that if the vessel needs to present during the maintenance operation on one OWT, it will only leave the OWT when the operation is completed.
5) Eq. (8.28) is precedence constraints which force the pickup is not done before completing the maintenance operation on the same OWT.
6) Eq. (8.29) is soft constraints which require that the maintenance task is performed within the preferred time.
7) Eq. (8.30) keeps the travelling time compatibility of each vessel.
8) Eq. (8.31) ensures the service vessels are not overloaded.
9) Eq. (8.32) expresses the compatibility requirements between routes and vessel loads.
10) Eq. (8.33) ensures that no extra load added when the vessels pick up from OWTs.
11) Eq. (8.34) and (8.35) describe the compatibility requirements between routes and personnel number on the vessels.
12) Eq. (8.36) and (8.37) guarantee that neither of load or personnel number exceeding the vessel limitations.
13) Eq. (8.38) imposes a maximal working time of the service vessels on each day.
14) Eq. (8.39) means the time is counted from the vessels leaving the harbor.
15) Eq. (8.40) set the delayed maintenance day to be non-negative.

8.5.2   Application of Duo-ACO in RSOM Problem

ACO is a meta-heuristic technique which is inspired by the foraging behavior of some ant species [Marco Dorigo et al., 2006]. It is a very good algorithm for solving optimization problem typically Travelling Salesman Problem (TSP). In classical TSP problem, there are many cities and only one salesman. If there are two sales man to travel all these cities and each city can and only can be traveled once, how to solve this Duo-TSP problem? The RSOM problem may have two or more vessels which is very similar with duo-TSP problem. This section describes the principle of Duo-ACO.

The idea of Duo-ACO is evolved from basic ACO which is introduced in Section 3.5.1. Duo-ACO has two groups with the same number of ants and each group has its own pheromone (group1, group2 and pheromone1, pheromone2 respectively). The procedure of the algorithm can be written as:

Begin
      Initialization

While stopping criterion not satisfied **do**

    Deploy each ant (k) in a starting city for group1

    Deploy each ant (k) in a starting city for group2

    (The ant with the same sequence (k) of two groups cannot in same city)

      For each ant (same sequence ant (k) for both group)

        Repeat

          Calculate probability of remaining cities selected to be next city for group1

          Choose next city according to probability using roulette wheel selection algorithm for group1

          Calculate probability of remaining cities selected to be next city for group2

          Choose next city according to probability using roulette wheel selection algorithm for group2

        Until all cities are visited

          Update pheromone1

          Update pheromone2

      End for

    Update the best routes (route1 for group1 while route2 for group2)

End while

    Record and output the beat routes (solutions)

End

The implement steps of duo-ACO are shown in Fig. 8.6. In each iteration, the two ants with the same index (k) in two groups select nodes (cities in TSP problem) alternatively according to their probabilities. Accordingly, the pheromones for two groups can be updated respectively. After all ants passed all nodes, the iteration number increases one by one until maximum iteration. Then the best routes of two groups with the same index are recorded as the best solution. Referring to apply Duo-ACO in RSOM problem, the solution of each group represents the route of corresponding vessel. The d_{ij} in Eq. (3.21) is replaced by Reciprocal of Eq. (8.21) in the case of RSOM problem.

8.5.3 Numerical Examples

To examine the effectiveness of Duo-ACO application in RSOM problem, several case studies are presented in this section. Fig. 8.7 shows an example of offshore wind farm with 64 wind turbines. The states of turbines can be "Replacement", "Repair" and "No service demand" according to the results of condition monitoring system especially of fault diagnosis and prognosis. There are two vessels can be used as maintenance flee and the parameters of them are shown in Table 8.6. The parameters of turbines and the maximum working hours for each day are shown in Table 8.7 and Table 8.8 respectively. The maximum working hours in Table 8.8 can be obtained from weather condition forecasting.
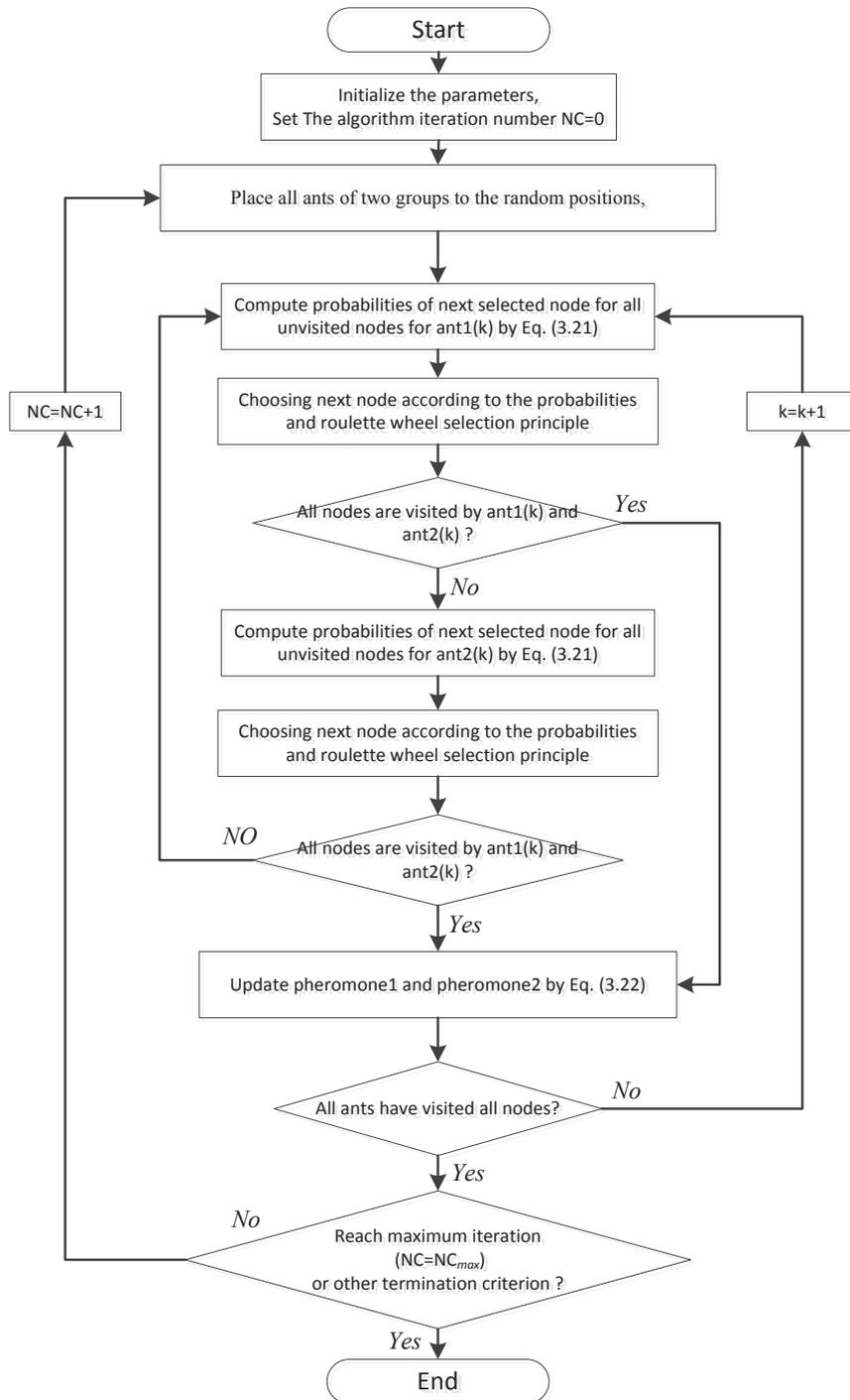
Fig. 8.6 The implementation steps of Duo-ACO

The process of the program was shown in Fig. 8.6. There are two groups of ants and each of them represents a vessel. The routing of ant represents the routing and scheduling of maintenance. From the experience, the number of ants in each group should be approximately the number of nodes the ants be visited which are offshore turbines in this case. Therefore, the parameters of Duo-ACO are set as: number of ants of each group is 10, the maximum iteration is 300, important coefficient of pheromone α and β are set as 1 and 5 respectively, and the pheromone evaporation coefficient is 0.1.

The results of maintenance scheduling and routing with 8 offshore turbines are shown in Table 8.9. Vessel1 and vessel2 visit and maintain 5 turbines and 3 turbines respectively. The routing number here is the same mean as Table 8.7. The result is that the two vessels can visit and maintain these turbines within one day (5.4915 and 8.7097 hours respectively) and the objective value of Eq. (8.21) is 3848.5.
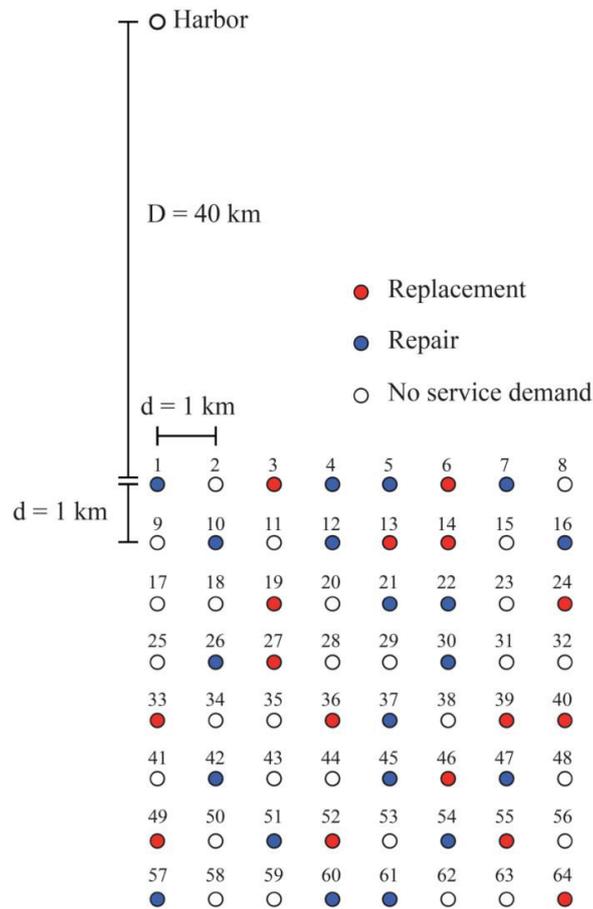
Fig. 8.7 The offshore wind farm example

Table 8.6 Parameters of Maintenance Vessels

Vessels	Speed (S, km/h)	Load Capacity (L, kg)	Personal Capacity (P)	Cost (C, €/h)
Vessel1	33	1500	12	225
Vessel2	20	26000	12	300

Table 8.7 Parameters of 8 Turbines

Unit	Turbines	Task type	Time window (day) T_i^{LATE}	Penalty cost (euro/day) C_i^{PE}	Required load (kg) L_i	Required personnel P_i	Task duration (hours) T_i^M
1	T12	Repair	2	1600	200	2	3
2	T19	Replacement	1	3000	200	2	2
3	T30	Repair	3	1200	100	2	1
4	T36	Replacement	2	2000	800	5	3
5	T39	Replacement	3	2000	300	2	2
6	T42	Repair	4	1000	200	2	2
7	T52	Replacement	2	2000	800	4	3
8	T60	Repair	4	1000	500	3	4

Table 8.8 Mmaximum Working Hours for Each Day

Time (Day)	Mmaximum Working Hours	
	Vessel1	Vessel2
Day 1	6	10
Day 2	6	10
Day 3	8	12
Day 4	7	11
Day 5	7	11
Day 6	5	8
Day 7	6	10
Day 8	6	10
Day 9	6	10
Day 10	7	11

Table 8.9 Results of Maintenance Routing with 8 Turbines

Vessels	No. of Visited Turbine	Routing	Needed Time (hours of each day)	Objective Value
Vessel1	5	0-1-2-6-3-5-3-2-6-5-1-0	5.4915	3848.5
Vessel2	3	0-7-8-4-7-4-8-0	8.7097	
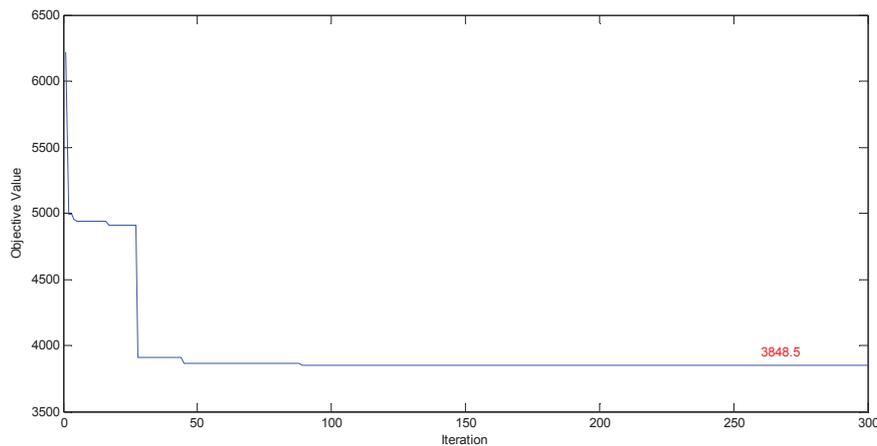
Fig. 8.8 Objective Value Changes with Iteration (8 turbines)

In order to examine the Duo-ACO performance for a large number turbines' wind farm, a new offshore wind farm with 28 turbines are tested. The information of two vessels is the same as shown in Table 8.6 and the maximum working hours for each day is the same as Table 8.8. The conditions and parameters of 28 turbines are shown in Table 8.10. The parameters of Duo-ACO changes because of the increasing the number of wind turbine. The number of ants of each group is set as 30 and the maximum iteration is set as 1000. The results are shown in Table 8.11Table and Fig. 8.9. The vessel1 and vessel2 visit and repair, inspection or replacement 19 turbines and 9 turbines respectively. Vessel1 need four days to visit and maintain all these 19 turbines, and it needs 5.859, 5.7938, 6.9541, and 4.8947 hours for each day which are less than that of the maximum working hours of vessel1 in Table 8.8. Vessel2 need 2 days to visit and maintain 9 turbines, and it needs 7.4814 and 7.0311 hours for each day which are also less than that of maximum working hours of vessels2 in Table 8.8. The objective value of fitness function of Eq. (8.21) is 94641.6 as shown in Table 8.11.

These two numerical examples show how to apply Duo-ACO in scheduling and routing of maintenance fleet for offshore wind farms which is a complex non-linear problem. Example 1 shown the problem solution with 8 offshore turbines while example 2 shows that of 28 offshore turbines and both of examples show the effectively of Duo-ACO application of the scheduling and routing problems of offshore wind farms.

Table 8.10 Parameters of 28 Turbines

Unit	Turbines	Task type	Time window (day) T_i^{LATE}	Penalty cost (euro/day) C_i^{PE}	Required load (kg) L_i	Required personnel P_i	Task duration (hours) T_i^M
1	T3	Replacement	3	2000	800	3	3
2	T4	Repair	6	500	50	2	2
3	T6	Replacement	4	1500	800	3	3
4	T11	Inspection	12	0	20	1	1
5	T12	Repair	4	1600	200	2	3
6	T13	Replacement	2	2500	500	3	2
7	T14	Replacement	2	2000	500	3	2
8	T16	Repair	5	1000	300	3	2
9	T19	Replacement	1	3000	200	2	2
10	T21	Repair	7	1000	50	1	2
11	T23	Inspection	12	0	20	1	1
12	T25	Inspection	10	0	20	1	1
13	T27	Replacement	2	2500	500	2	3
14	T30	Repair	4	1200	100	2	1
15	T36	Replacement	3	2000	800	5	3
16	T38	Inspection	12	0	20	1	1
17	T39	Replacement	4	2000	300	2	2
18	T42	Repair	5	1000	200	2	2
19	T44	Inspection	10	0	20	1	1
20	T45	Repair	8	1000	500	2	2
21	T49	Replacement	1	3000	800	4	3
22	T52	Replacement	2	2000	800	4	3
23	T54	Repair	5	1000	50	1	1
24	T55	Replacement	3	2000	500	3	2
25	T58	Inspection	13	0	20	1	1
26	T60	Repair	6	1000	500	3	4
27	T61	Repair	7	1000	300	2	3
28	T62	Inspection	12	0	20	1	1

Table 8.11 Results of Maintenance Routing with 28 Turbines

Vessels	No. of Visited Turbine	Routing	Needed Time (hours of each day)	Objective Value
Vessel1	19	0-5-4-9-12-21-25-4-12-25-9-5-21-0-22-22-0-24-23-28-27-26-19-23-28-19-24-27-26-0-8-11-16-17-20-18-11-16-8-17-20-18-0	5.859, 5.7938, 6.9541, 4.8947	9641.6
Vessel2	9	0-2-1-13-15-2-1-13-15-0-3-7-6-10-14-14-7-6-10-3-0	7.4814, 7.0311	
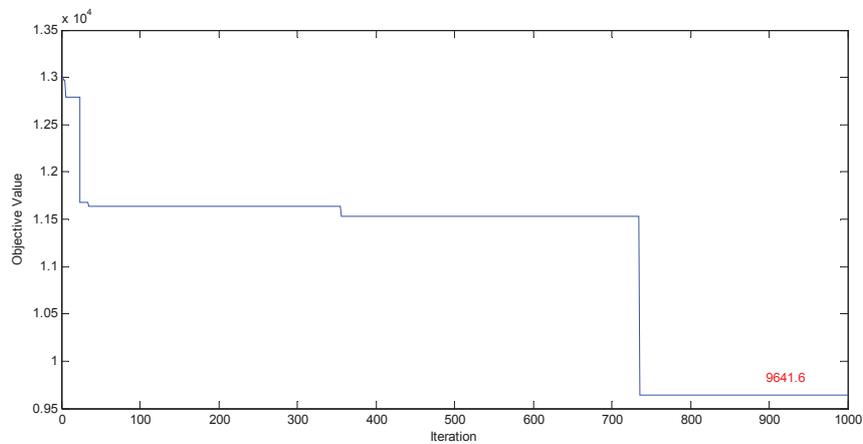
Fig. 8.9 Objective Value Changes with Iteration (28 turbines)

There is also a drawback of the Duo-ACO to solve this problem. With the increasing of turbines, the process to find the solution using Duo-ACO becomes time-consuming. However, this problem is not so time sensitive which means the key point is to find the optimal solution regardless how much time it using. Therefore, the Duo-ACO is a suitable algorithm to solve this non-linear scheduling and routing problem.

8.6   Summary

This Chapter mainly described the maintenance scheduling optimization based on Swarm Intelligence such as PSO, BCA and ACO. For each algorithm, applications in industry or numerical examples were described to indicate how the algorithm works in maintenance scheduling.

For algorithm of PSO, the problem of maintenance scheduling of generating units for reliable operation of a power system with 32 units were tested. In this case, the annual supply reserve ratio was selected as fitness function with constraints of load, crew and maintenance window. The maintenance schedule was not based on the

condition of machines but based on a fixed period (a year) and so it can be seen as preventive maintenance scheduling. PSO was improved with mutation rate α called improved PSO (IPSO) to apply in generating units maintenance scheduling. Both PSO and IPSO can find optimal maintenance schedule of generation units but IPSO has better performance with faster convergence speed and better fitness value.

For application of BCA in predictive scheduling optimization, a model of dynamic model of condition based maintenance was established. The dynamic predictive maintenance model is based on the condition of machines other than fixed period like preventive maintenance. The main effort of predictive maintenance (PM) is to avoid unnecessary maintenance action tasks by taking maintenance action just in case of detecting any evidence of abnormal performance in physical condition. A PM program can significantly decline the maintenance cost by decreasing the number of needless scheduled preventive activities. PM program allows the maintenance function to do only the right things, at the correct time, minimizing spare parts cost, system downtime and time spent on maintenance. Based on the model and condition of each machine, a dynamic scheduling of PM and CM can be done using BCA. The result obtained from the numerical example confirms the trend of successful application using this algorithm in the field of PM, where a dynamic approach has a fundamental importance. Although the desired results have fully achieved, and the analysis has helped to highlight and solve many critical issues, it is clear that more careful analysis should be done when analyzing PM maintenance model. In this Chapter, only one single kind condition for each machine. However, mostly, more than one parameters get together to determine the state of a machine. Therefore, how to get the state of a machine using different parameters could be a future research field. Furthermore, the case study in this Chapter only consider the one period because the limitation of our resources. In the future, the long history period should be considered and the methods for adjusting the state $S_i(i=1,2,\cdots,n)$ could be a good topic to research.

For application of ACO in maintenance scheduling, a model of scheduling and routing of maintenance fleet for offshore wind farms was established. ACO was varied with two groups of ants which called Duo-ACO. Through the numerical examples, Duo-ACO can solve this problem effectively even if the number of turbines increasing. The drawback of the methodology is that it is impossible to know if the optimal solution found by Duo-ACO is the best one.

Chapter 8: Maintenance Scheduling Optimization based on Data Mining
Techniques

9 Conclusion and Future Work

This chapter provides general overall comments and concluding remarks about the work presented in this thesis and some suggestions for future work.

9.1 Summary and Conclusions

The goals of this thesis are to develop a framework of intelligent Condition based Maintenance (CBM) and apply data mining techniques in its phases. CBM is a sufficient maintenance strategy which take maintenance action just before the failure based on the condition of equipment to increase the reliability and availability of the equipment and meanwhile reduce maintenance and operation cost. It can also improve the safety for both equipment and operation staff. There are mainly two tasks of CBM: the one is fault diagnosis and prognosis for the equipment and the other is based on which to optimize the maintenance scheduling.

Chapter 2 presented framework of Intelligent Fault Diagnosis and Prognosis System (IFDPs) for CBM which showed phases of the CBM and data mining techniques applied in the system.

Chapter 3 presented data mining techniques applying in IFDPS, including Artificial Neural Network (ANN), Swarm Intelligence (SI) and Association Rules (AR). The techniques of ANN and AR are supposed to be applied in fault diagnosis and prognosis while the techniques of SI are supposed to be applied in sensor place optimization and maintenance optimization.

Chapter 4 introduced the sensor classification and sensor placement optimization techniques. The presented methods sensor placement optimization is combination of Finite Element Analysis (FEA) and SI algorithm such as PSO and BCO are suitable for component level and machine level of sensor placement optimization. However, the system level sensor placement optimization need to be further researched.

Chapter 5 presented methods of signal processing typically for vibration signals and feature extraction. The vibration signals can be processed in time domain, frequency domain, time-frequency domain and wavelet domain analysis which many features (parameters) can be extracted. The parameters extracted from signals may be too many to be classified or predicted using data mining techniques and thus feature selection techniques need to be used to reduce the dimensionality of the parameters. PCA is an unsupervised learning approach for dimensionality reduction that uses correlation coefficients of the parameters to combine and transform them into a reduced dimensional space. It transforms high dimensionality features to lower dimensionality but not select the features from original features directly. Therefore, the feature selection directly from original features should be researched.

Chapter 6 presented the methods of fault diagnosis, i.e. fault detection and classification, based on data mining techniques such as BP network, SOM and Association Rules. The conclusions have been presented at the end of this chapter. When the history data is available but the physical model and mathematical model are not available or not accurate, the data-driven techniques can be sufficient applied in fault diagnosis.

Chapter 7 presented fault prognosis based on the indicator prediction of the fault using BP network. The traditional methods of data-driven fault prognosis are based on statistics of the history data [Lee et al., 2006]. ANN model is supposed to be used for multi-component, multi-fault prognosis but the case study for wind turbine fault prognosis in this chapter only one component and one fault was used. In the future, the multi-component, multi-fault ANN model should be further researched.

Chapter 8 presented the maintenance optimization based on data mining techniques. Three different models and Swarm Intelligence (variants of PSO, BCA and ACO) were presented in this Chapter. Generating Unit Maintenance Scheduling is a preventive maintenance optimization, while the following two examples are predictive maintenance or so-called CBM, and both of which can use data mining techniques to solve.

9.2 Suggestions of Future Work

The following are proposed for future work:

➢ Developing sensor placement optimization methods in system level, i.e. more than two machines.
➢ Developing methods of feature selection directly from original features.
➢ Developing hybrid methods of model-based and data-driven for fault diagnosis and prognosis to improve accuracy.

References

Aamodt, A., & Plaza, E. (1994). Case-Based Reasoning : Foundational Issues , Methodological Variations , and System Approaches. *AI Communications*, Vol. 7, No. 1, pp. 39–59.

Abraham, A., Grosan, C., & Ramos, V. (2006). *Swarm intelligence in data mining*. Berlin/Heidelberg: Springer.

Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93*, pp. 207–216. New York, USA: ACM Press. doi:10.1145/170035.170072.

Ahmad, R., & Kamaruddin, S. (2012). An overview of time-based and condition-based maintenance in industrial application. *Computers & Industrial Engineering*, Vol. 63, No. 1, pp. 135–149. doi:10.1016/j.cie.2012.02.002.

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, Vol. 38, No. 4, pp. 393–422. doi:10.1016/S1389-1286(01)00302-4.

Al-Shehabi, A. G., & Newman, B. (2001). Aeroelastic vehicle optimal sensor placement for feedback control applications using mixed gain-phase stability. *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, Vol. 3, pp. 1848–1852. IEEE. doi:10.1109/ACC.2001.946005.

Altmann, J., & Mathew, J. (2001). Multiple Band-Pass Autoregressive Demodulation for Rolling-Element Bearing Fault Diagnostics. *Mechanical Systems and Signal Processing*, Vol. 15, No. 5, pp. 963–977. doi:10.1006/mssp.2001.1410.

Amari, S. V, Ph, D., Corporation, R. S., & Mclaughlin, L. (2004). Optimal Design of a Condition-Based Maintenance Model. *Annual Symposium Reliability and Maintainability*, pp. 528–533.

Amari, S. V, Ph, D., Corporation, R. S., & Mclaughlin, L. (2006). Cost-Effective Condition-Based Maintenance Using Markov Decision Processes. *Annual of Reliability and Maintainability Symposium*, Vol. 00, pp. 464–469.

Aminian, F. (2001). Fault Diagnostics of Analog Circuits using Bayesian Neural Networks with Wavelet Transform as Preprocessor. *Journal of*

Electronic Testing: Theory and Applications, Vol. 17, No. 1, pp. 29–36. doi:10.1023/A:1011141724916.

Andersen, T. M., & Rasmussen, M. (1999). Decision support in a condition based environment. *Journal of Quality in Maintenance Engineering*, Vol. 5, No. 2, pp. 89–102. doi:10.1108/13552519910271793.

Andria, G., Savino, M., & Trotta, A. (1994). Application of Wigner-Ville Distribution to Measurements on Transient Signals. *IEEE Transactions on Instrumentation and Measurement*, Vol. 43, No. 2, pp. 187–193. doi:10.1109/19.293418.

Ansari, F. (1998). *Fiber Optic Sensors for Construction Materials and Bridges*. Lancaster: Taylor & Francis.

Arroyo, J. M., & Conejo, A. J. (2002). A parallel repair genetic algorithm to solve the unit commitment problem. *IEEE Transactions on Power Systems*, Vol. 17, No. 4, pp. 1216–1224. doi:10.1109/TPWRS.2002.804953.

Back, T., Hammel, U., & Schwefel, H.-P. (1997). Evolutionary computation: comments on the history and current state. *IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, pp. 3–17. doi:10.1109/4235.585888.

Barata, J., GuedesSoares, C., Marseguerra, M., & Zio, E. (2001). Monte Carlo simulation of deteriorating systems. *Proceedings ESREL*, pp. 879–886.

Barata, J., Soares, C. G., Marseguerra, M., & Zio, E. (2002). Simulation modelling of repairable multi-component deteriorating systems for "on condition" maintenance optimisation. *Reliability Engineering & System Safety*, Vol. 76, No. 3, pp. 255–264.

Baskar, S., & Suganthan, P. N. (2004). A novel concurrent particle swarm optimization. *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No.04TH8753)*, pp. 792–796. IEEE. doi:10.1109/CEC.2004.1330940.

Be′renguer, C., Grall, A., & Castanier, B. (2000). Simulation and evaluation of Condition-based maintenance policies for multi-component continuous-state deteriorating systems. *Proceedings of the Foresi*.

Becker, E., & Poste, P. (2006). Keeping the Condition Monitoring of Wind Turbine Gears. *Wind Energy*, Vol. 7, No. 2, pp. 26–32.

Beigel, M. (1982). Identification Device. US.

Belkin, M., Niyogi, P., & Sindhwani, V. (2006). Manifold Regularization : A Geometric Framework for Learning from Labeled and Unlabeled

Examples. *Journal of Machine Learning Research*, Vol. 7, pp. 2399–2434.

Bellman, R. (1961). *Adaptive control processes: a guided tour*. New Jersey: Princeton University Press.

Benbouzid-Sitayeb, F., Ammi, I., Varnier, C., & Zerhouni, N. (2008). Applying Ant Colony Optimization for the Joint Production and Preventive Maintenance Scheduling Problem in the Flowshop Sequencing Problem. *2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications*, pp. 1–6. IEEE. doi:10.1109/ICTTA.2008.4530343.

Beni, G. (1988). The concept of cellular robotic system. *Proceedings IEEE International Symposium on Intelligent Control 1988*, pp. 57–62. IEEE Comput. Soc. Press. doi:10.1109/ISIC.1988.65405.

Benouaret, M., Sahour, A., & Harize, S. (2012). Real time implementation of a signal denoising approach based on eight-bits DWT. *AEU - International Journal of Electronics and Communications*, Vol. 66, No. 11, pp. 937–943. doi:10.1016/j.aeue.2012.04.001.

Berenji, H. R. (2006). Case-Based Reasoning for Fault Diagnosis and Prognosis. *2006 IEEE International Conference on Fuzzy Systems*, pp. 1316–1321. IEEE. doi:10.1109/FUZZY.2006.1681880.

Bergh, F. van den, & Engelbrecht, A. . (2004). A Cooperative Approach to Particle Swarm Optimization. *IEEE Transactions on Evolutionary Computation*, Vol. 8, No. 3, pp. 225–239. doi:10.1109/TEVC.2004.826069.

Bhushan, M., & Rengaswamy, R. (2000). Design of sensor location based on various fault diagnostic observability and reliability criteria. *Computers & Chemical Engineering*, Vol. 24, pp. 735–741. doi:http://dx.doi.org/10.1016/S0098-1354(00)00331-8.

Billinton, R., & Abdulwhab, A. (2003). Short-term generating unit maintenance scheduling in a deregulated power system using a probabilistic approach. *IEE Proceedings - Generation, Transmission and Distribution*, Vol. 150, No. 4, pp. 463.

Biskas, P. N., Ziogos, N. P., Tellidou, A., Zoumas, C. E., Bakirtzis, A. G., Petridis, V., & Tsakoumis, A. (2006). Comparison of Two Metaheuristics with Mathematical Programming Methods for the Solution of OPF. *Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems*, Vol. 153, pp. 510–515. IEEE. doi:10.1109/ISAP.2005.1599316.

Blackwell, T. M., & Bentley, P. (2002). Don't push me! Collision-avoiding swarms. *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, Vol. 2, pp. 1691–1696. IEEE. doi:10.1109/CEC.2002.1004497.

Bloch, H. P., & Geitner, F. K. (2012). *Machinery failure analysis and troubleshooting*, 4th ed., pp. 1–760. Houston: Butterworth-Heinemann.

Blum, A., & Chawla, S. (2001). Learning from Labeled and Unlabeled Data using Graph Mincuts. *ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 19–26.

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm Intelligence: From Natural to Artificial Systems*. New York: Oxford University Press.

Boucly, F. (2001). *Le Management de la maintenance: évolution et mutation*, pp. 1–307. Paris: Association Française de Normalisation (AFNOR).

Brando, G., Dannier, A., Del Pizzo, A., & Rizzo, R. (2007). Quick identification technique of fault conditions in cascaded H-Bridge multilevel converters. *2007 International Aegean Conference on Electrical Machines and Power Electronics*, Vol. 3, pp. 491–497. IEEE. doi:10.1109/ACEMP.2007.4510549.

Breiman, L. (2001). Random Forests. *Machine Learning*, Vol. 45, No. 1, pp. 5–32. doi:10.1023/A:1010933404324.

Bussel, G. J. W. van, & Zaaijer, M. B. (2001). Reliability , Availability and Maintenance aspects of large-scale offshore wind farms , a concepts study . *In Marine Renewable Energies Conference*, Vol. 113, pp. 119–126. Newcastle.

Callaway, E., Gorday, P., Hester, L., Gutierrez, J. A., Naeve, M., Heile, B., & Bahl, V. (2002). Home networking with IEEE 802.15.4: a developing standard for low-rate wireless personal area networks. *IEEE Communications Magazine*, Vol. 40, No. 8, pp. 70–77. doi:10.1109/MCOM.2002.1024418.

Campos, J. (2009). Development in the application of ICT in condition monitoring and maintenance. *Computers in Industry*, Vol. 60, No. 1, pp. 1–20. doi:10.1016/j.compind.2008.09.007.

Carnero Moya, M. C. (2004). The control of the setting up of a predictive maintenance programme using a system of indicators. *Omega*, Vol. 32, No. 1, pp. 57–75. doi:10.1016/j.omega.2003.09.009.

Reference

Cecchin, T., Ranta, R., Koessler, L., Caspary, O., Vespignani, H., & Maillard, L. (2010). Seizure lateralization in scalp EEG using Hjorth parameters. *Clinical neurophysiology : official journal of the International Federation of Clinical Neurophysiology*, Vol. 121, No. 3, pp. 290–300. doi:10.1016/j.clinph.2009.10.033.

Chen, D., & Trivedi, K. S. (2005). Optimization for condition-based maintenance with semi-Markov decision process. *Reliability Engineering & System Safety*, Vol. 90, pp. 25–29. doi:10.1016/j.ress.2004.11.001.

Chen, D., & Wang, W. J. (2002). Classification of Wavelet Map Patterns Using Multilayer Neural Networks for Gear Fault Detection. *Mechanical Systems and Signal Processing*, Vol. 16, No. 4, pp. 695–704. doi:10.1006/mssp.2002.1488.

Chen, G., Liu, Y., Zhou, W., & Song, J. (2008). Research on intelligent fault diagnosis based on time series analysis algorithm. *The Journal of China Universities of Posts and Telecommunications*, Vol. 15, No. 1, pp. 68–74.

Chen, S. Y., & Li, Y. F. (2002). A method of automatic sensor placement for robot vision in inspection tasks. *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, Vol. 3, pp. 2545–2550. IEEE. doi:10.1109/ROBOT.2002.1013614.

CHERNG, A.-P. (2003). OPTIMAL SENSOR PLACEMENT FOR MODAL PARAMETER IDENTIFICATION USING SIGNAL SUBSPACE CORRELATION TECHNIQUES. *Mechanical Systems and Signal Processing*, Vol. 17, No. 2, pp. 361–378. doi:10.1006/mssp.2001.1400.

Chong, C., Hean Low, M., Sivakumar, A., & Gay, K. (2006). A Bee Colony Optimization Algorithm to Job Shop Scheduling. *Proceedings of the 2006 Winter Simulation Conference*, pp. 1954–1961. IEEE. doi:10.1109/WSC.2006.322980.

Chu, C., Proth, J., & Wolff, P. (1998). Predictive maintenance : The One-unit Replacement Model. *International Journal of Production Economics*, Vol. 54, No. 3, pp. 285–295. doi:http://dx.doi.org/10.1016/S0925-5273(98)00004-8.

Clerc, M., & Kennedy, J. (2002). The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 1, pp. 58–73. doi:10.1109/4235.985692.

Reference

Cooke, R. M. (1992). *Experts in Uncertainty: Opinion and Subjective Probability in Science*, p. 336. New York: Oxford University Press.

Corinthios, M. J. (1971). A fast Fourier transformation for high-speed signal processing. *IEEE Transaction on Computers*, Vol. 20, No. 8, pp. 843–846. doi:http://doi.ieeecomputersociety.org/10.1109/T-C.1971.223359.

Crellin, G. L. (1986). *Use of reliability-centered maintenance for the McGuire Nuclear Station feed water system, EPRI NP-4795*.

Dai, L. (2014). *Safe and efficient operation and maintenance of offshore wind farms*. NTNU.

Dargie, W., & Poellabauer, C. (2010). *Fundamentals of Wireless Sensor Networks: Theory and Practice*, p. 311. West Sussex: JohnWiley & Sons Ltd.

Daubechies, I. (1988). Ortho-normal bases of compactly supported wavelets.pdf. *Communications on Pure and Applied Mathematics*, Vol. 41, pp. 909–996.

De Kleer, J., & Williams, B. (1987). Diagnosing Multiple Faults. *Artificial Intelligence*, Vol. 32, No. 1, pp. 97–130. doi:10.1016/0004-3702(87)90063-4.

Deshpande, V. ., & Modak, J. . (2002). Application of RCM for safety considerations in a steel plant. *Reliability Engineering & System Safety*, Vol. 78, No. 3, pp. 325–334. doi:10.1016/S0951-8320(02)00177-1.

Dieulle, L., BCrenguer, C., Grall, A., & Roussignol, M. (2001). Continuous Time Predictive Maintenance Scheduling for a Deteriorating System.

Dirilten, H. (1972). On the Mathematical Models Characterizing Faulty Four-Phase MOS Logic Arrays. *IEEE Transactions on Computers*, Vol. C-21, No. 3, pp. 301–305. doi:10.1109/TC.1972.5008954.

Djurdjanovic, D., Lee, J., & Ni, J. (2003). Watchdog Agent — an infotronics-based prognostics approach for product performance degradation assessment and prediction, Vol. 17, pp. 109–125. doi:10.1016/j.aei.2004.07.005.

Donoho, D. L. (1995). De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, Vol. 41, No. 3, pp. 613–627. doi:10.1109/18.382009.

Dorigo, M., Birattari, M., & Stutzle, T. (2006). Ant colony optimization. *IEEE Computational Intelligence Magazine*, Vol. 1, No. 4, pp. 28–39. doi:10.1109/MCI.2006.329691.

Reference

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, Vol. 26, No. 1, pp. 29–41. doi:10.1109/3477.484436.

Dorigo, M., & Stützle, T. (2004). *Ant colony optimization*. Cambridge, Mass: MIT Press.

Dragomir, O. E., Gouriveau, R., Zerhount, N., & Dragomir, F. (2007). Framework for a distributed and hybrid prognostic system. In B. Octavian (Ed.), *4th IFAC Conference on Management and Control of Production and Logistics (2007)*, pp. 431–436. doi:10.3182/20070927-4-RO-3905.00072.

Du, M., Cai, J., Liu, L., & Chen, P. (2011). ARRs based sensor placement optimization for fault diagnosis. *Procedia Engineering*, Vol. 16, pp. 42–47. doi:10.1016/j.proeng.2011.08.1049.

Eberhart, R. C., & Shi, Y. (2000). Comparing inertia weights and constriction factors in particle swarm optimization. *Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512)*, Vol. 1, pp. 84–88. IEEE. doi:10.1109/CEC.2000.870279.

Eisenmann, R. C. S., & Eisenmann, R. C. J. (1998). *Machinery Malfunction Diagnosis and Correction*. Englewood Cliffs, NJ: Prentice-Hall.

Eklöv, T., Mårtensson, P., & Lundström, I. (1997). Enhanced selectivity of MOSFET gas sensors by systematical analysis of transient parameters. *Analytica Chimica Acta*, Vol. 353, No. 2-3, pp. 291–300. doi:10.1016/S0003-2670(97)87788-4.

El-Abd, M., & Kamel, M. S. (2006). A hierarchical cooperative Particle Swarm Optimizer. *Proc. of Swarm Intelligence Symposium*, pp. 43–47.

EN 13306: 2001 Maintenance Terminology, European Standard (2001). CEN (European Committee for Standardization), Brussels.

Espinosa, J., Vandewalle, J., & Wertz, V. (2005). *Fuzzy Logic, Identification and Predictive Control*. London: Springer-Verlag.

Estrin, D., Govindan, R., Heidemann, J., & Kumar, S. (1999). Next century challenges. *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking - MobiCom '99*, pp. 263–270. New York, New York, USA: ACM Press. doi:10.1145/313451.313556.

Reference

Ewins, D. J. (1995). *Modal testing: theory and practice*. Letchworth: Research Studies Press.

Faulds, A. L., & King, B. B. (2000). Sensor Location in Feedback Control of Partial Differential Equation Systems. *Proceedings of the 2000 IEEE International Conference on Control Applications*, pp. 536–541.

Firmo, H. T., & Legey, L. F. L. (2002). Generation expansion planning: an iterative genetic algorithm approach. *IEEE Transactions on Power Systems*, Vol. 17, No. 3, pp. 901–906. doi:10.1109/TPWRS.2002.801036.

Friedman, J. H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*, Vol. 29, No. 5, pp. 1189–1232. doi:10.1214/aos/1013203451.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis*, Vol. 38, No. 4, pp. 367–378. doi:10.1016/S0167-9473(01)00065-2.

Frisch, K. von. (1967). *The dance language and orientation of bees*. Cambridge: Harvard University Press.

Fu, X., Zhang, Y., & Zhu, Y. (2011). Rolling bearing fault diagnosis approach based on case-based reasoning. *Journal of Xi'an Jiaotong University*, Vol. 45, No. 11, pp. 79–84.

Garcia, M. C., Sanz-Bobi, M. a., & del Pico, J. (2006). SIMAP: Intelligent System for Predictive Maintenance Application to the Health Condition Monitoring of a Wind Turbine Gearbox. *Computers in Industry*, Vol. 57, No. 6, pp. 552–568. doi:10.1016/j.compind.2006.02.011.

Garey, M. R., & Johnson, D. S. (1979). *Computers and Intractability: a Guide to the Theory of NP-completeness*. San Francisco: W. H. Freeman.

Garfinkel, S., & Holtzman, H. (2005). Understanding RFID Technology in: Applications, Security and Privacy, pp. 15–36. Addison-Wesley.

Gibson, T. D., Prosser, O., Hulbert, J. N., Marshall, R. W., Corcoran, P., Lowery, P., Ruck-Keene, E. a., et al. (1997). Detection and simultaneous identification of microorganisms from headspace samples using an electronic nose. *Sensors and Actuators B: Chemical*, Vol. 44, No. 1-3, pp. 413–422. doi:10.1016/S0925-4005(97)00235-9.

Giraud, C., & Jouvencel, B. (1995). Sensor selection: a geometrical approach. *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and*

Cooperative Robots, Vol. 2, pp. 555–560. IEEE Comput. Soc. Press. doi:10.1109/IROS.1995.526271.

Goldberg, D. E. (1989). *Genetic Algorithms in search, optimization, and machine learning*. Addison-Wesley Publishing Corporation, Inc.

Goumas, S., Zervakis, M., Pouliezos, A., & Stavrakakis, G. S. (2001). Intelligent on-line quality control of washing machines using discrete wavelet analysis features and likelihood classification. *Engineering Applications of Artificial Intelligence*, Vol. 14, No. 5, pp. 655–666. doi:10.1016/S0952-1976(01)00028-8.

Grall, A., Be′renguer, C., & Chu, C. (2008). Optimal dynamic inspection/replacement planning in condition-based maintenance. *Proceedings of the European Safety and Reliability Conference*, pp. 381–388.

Grant, P. W., Harris, P. M., & Moseley, L. G. (1996). Fault diagnosis for industrial printers using case-based reasoning. *Engineering Applications of Artificial Intelligence*, Vol. 9, No. 2, pp. 163–173. doi:10.1016/0952-1976(96)00009-7.

Guardia, M. (1995). Biochemical sensors: The state of the art. *Mikrochimica Acta*, Vol. 120, No. 1-4, pp. 243–255. doi:10.1007/BF01244435.

Gupta, A., & Lawsirirat, C. (2006). Strategically optimum maintenance of monitoring-enabled multi-component systems using continuous-time jump deterioration models. *Journal of Quality in Maintenance Engineering*, Vol. 12, No. 3, pp. 306–329. doi:10.1108/13552510610685138.

Gutierrez-Osuna, R., Nagle, H. T., & Schiffman, S. S. (1999). Transient response analysis of an electronic nose using multi-exponential models. *Sensors and Actuators B: Chemical*, Vol. 61, No. 1-3, pp. 170–182. doi:10.1016/S0925-4005(99)00290-7.

Gutierrez-Osuna, R., Nagle, T., Kermani, B., & Schiffman, S. (2003). Signal Conditioning and Pre-Processing. *Handbook of Machine Olfaction: Electronic Nose Technology*, pp. 1–50. Wiley-VCH.

Guyon, I., & Elisseef, A. (2006). An Introduction to Feature Extraction. In I. Guyon, M. Nikravesh, S. Gunn, & L. Zadeh (Eds.), *Feature Extraction*, pp. 1–25. Berlin/Heidelberg: Springer.

GWEC. (2013). *Global Wind Statistics 2012*, pp. 1–4.

Halim, D., & Moheimani, R. S. O. (2003). An optimization approach to optimal placement of collocated piezoelectric actuators and sensors on

a thin plate. *Mechatronics*, Vol. 13, No. 1, pp. 27–47. doi:10.1016/S0957-4158(01)00079-4.

Hambaba, A., & Huff, E. (2000). Multiresolution error detection on early fatigue cracks in gears. *2000 IEEE Aerospace Conference. Proceedings (Cat. No.00TH8484)*, Vol. 6, pp. 367–372. IEEE. doi:10.1109/AERO.2000.877912.

Hameed, Z., Hong, Y. S., Cho, Y. M., Ahn, S. H., & Song, C. K. (2009). Condition monitoring and fault detection of wind turbines and related algorithms: A review. *Renewable and Sustainable Energy Reviews*, Vol. 13, No. 1, pp. 1–39. doi:10.1016/j.rser.2007.05.008.

Han, Y., & Song, Y. H. (2003). Condition monitoring techniques for electrical equipment-a literature survey. *IEEE Transactions on Power Delivery*, Vol. 18, No. 1, pp. 4–13. doi:10.1109/TPWRD.2002.801425.

Hand, D. J., Mannila, H., & Smyth, P. (2001). *Principles of Data Mining*. London: MIT Press & IEEE.

Hansen, L. K., & Salamon, P. (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 10, pp. 993–1001. doi:10.1109/34.58871.

Hansen, M. O. L. (2007). *Aerodynamics of Wind Turbines*, 2nd ed., p. 208. London, UK: Earthscan.

Hayes, J. P. (1971). A Nand Model ror Fault Diagnosis in Combinational Logic Networks. *IEEE Transactions on Computers*, Vol. C-20, No. 12, pp. 1496–1506. doi:10.1109/T-C.1971.223162.

He, S., Wen, J. Y., Prempain, E., Wu, Q. H., Fitch, J., & Mann, S. (2004). An improved particle swarm optimization for optimal power flow. *2004 International Conference on Power System Technology, 2004. PowerCon 2004.*, Vol. 2, pp. 1633–1637. IEEE. doi:10.1109/ICPST.2004.1460265.

He, S.-G., He, Z., & Wang, G. A. (2011). Online monitoring and fault identification of mean shifts in bivariate processes using decision tree learning techniques. *Journal of Intelligent Manufacturing*. doi:10.1007/s10845-011-0533-5.

Hjorth, B. (1970). EEG analysis based on time domain properties. *Electroencephalography and Clinical Neurophysiology*, Vol. 29, No. 3, pp. 306–310. doi:10.1016/0013-4694(70)90143-4.

Holmberg, K., Adgar, A., Arnaiz, A., Jantunen, E., Mascolo, J., & Mekid, S. (2010). *E-maintenance*, pp. 1–241. London: Springer.

Reference

Hook, T. G., Hughes, E. A., Levline, R. E., Morgan, T. A., & Parker, L. M. (1987). *Application of reliability-centered maintenance to San Onofre Units 2 & 3 auxiliary feed water system, EPRI NP-5430.*

Hu, J., Zhang, L., Ma, L., & Liang, W. (2011). An integrated safety prognosis model for complex system based on dynamic Bayesian network and ant colony algorithm. *Expert Systems with Applications*, Vol. 38, No. 3, pp. 1431–1446. doi:10.1016/j.eswa.2010.07.050.

Hu, S., Zhou, C., & Hu, W. (2000). A New Structure Fault Detection Method Based on Wavelet Singularity. *Journal of Applied Sciences*, Vol. 18, No. 3, pp. 198–201.

Huang, C. J., Lin, C. E., & Huang, C. L. (1992). Fuzzy approach for generator maintenance scheduling. *Electric Power Systems Research*, Vol. 24, No. 1, pp. 31–38. doi:10.1016/0378-7796(92)90042-Y.

Huang, S. (1998). A genetic-evolved fuzzy system for maintenance scheduling of generating units. *International Journal of Electrical Power & Energy Systems*, Vol. 20, No. 3, pp. 191–195. doi:http://dx.doi.org/10.1016/S0142-0615(97)00080-X.

Huang, Y., Mcmurran, R., & Jones, D. R. P. (2008). Probability based vehicle fault diagnosis : Bayesian network method, pp. 301–311. doi:10.1007/s10845-008-0083-7.

Ilie-zudor, E., Kemény, Z., Egri, P., & Monostori, L. (2006). The RFID Technology and Its Current Applications. *Proceeding of The Modern Information Technology in the Innovation Process of the Industrial Enterprise-MITIP*, pp. 29–36.

Intanagonwiwat, C., Govindan, R., & Estrin, D. (2000). Directed Diffusion : A Scalable and Robust Communication Paradigm for Sensor Networks. *Proceedings of the ACM Mobi- Com'00*, pp. 56–67. Boston.

Isermann, R. (2005). Model-based fault-detection and diagnosis – status and applications. *Annual Reviews in Control*, Vol. 29, No. 1, pp. 71–85. doi:10.1016/j.arcontrol.2004.12.002.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM Computing Surveys*, Vol. 31, No. 3, pp. 264–323. doi:10.1145/331499.331504.

Jansen, A., & Niyogi, P. (2005). *A Geometric Perspective on Speech Sounds*, pp. 1–50.

Jardine, A. K. S., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based

maintenance. *Mechanical Systems and Signal Processing*, Vol. 20, No. 7, pp. 1483–1510. doi:10.1016/j.ymssp.2005.09.012.

Jolliffe, I. T. (2002). *Principal Component Analysis*, 1st ed. New York: Springer Series in Statistics.

Kankar, P. K., Sharma, S. C., & Harsha, S. P. (2011). Rolling element bearing fault diagnosis using wavelet transform. *Neurocomputing*, Vol. 74, No. 10, pp. 1638–1645. doi:10.1016/j.neucom.2011.01.021.

Kannan, S., Slochanal, S. M. R., Subbaraj, P., & Padhy, N. P. (2004). Application of particle swarm optimization technique and its variants to generation expansion planning problem. *Electric Power Systems Research*, Vol. 70, No. 3, pp. 203–210. doi:10.1016/j.epsr.2003.12.009.

Kantardzic, M. (2003). *Data Mining: Concepts, Models, Methods, and Algorithm*. New York: John Wiley & Sons, Inc.

Karaboga, D. (2005). *An idea based on honey bee swarm for numerical optimization*.

Karaboga, D., & Akay, B. (2009). A comparative study of Artificial Bee Colony algorithm. *Applied Mathematics and Computation*, Vol. 214, No. 1, pp. 108–132. doi:10.1016/j.amc.2009.03.090.

Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, Vol. 39, No. 3, pp. 459–471. doi:10.1007/s10898-007-9149-x.

Kasabov, N. (2001). Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, Vol. 31, No. 6, pp. 902–18. doi:10.1109/3477.969494.

Kegg, R. . (1984). On-line machine and process diagnostics. *Annals of the CIRP*, Vol. 32, No. 2, pp. 469–573.

Kennedy, J., & Eberhart, R. (2001). *Swarm Intelligence*. San Francisco: Morgan Kaufmann Publishers, Inc.

Kennedy, J., & Eberhart, R. C. (1997). A discrete binary version of the particle swarm algorithm. *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, Vol. 5, pp. 4104–4108. IEEE. doi:10.1109/ICSMC.1997.637339.

Kermani, B. G., Schiffman, S. S., & Nagle, H. T. (1998). A novel method for reducing the dimensionality in a sensor array. *IEEE Transactions*

on Instrumentation and Measurement, Vol. 47, No. 3, pp. 728–741. doi:10.1109/19.744338.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, Vol. 97, No. 1-2, pp. 273–324. doi:10.1016/S0004-3702(97)00043-X.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, Vol. 43, pp. 59–66.

Krink, T., Vesterstrom, J. S., & Riget, J. (2002). Particle swarm optimisation with spatial particle extension. *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, Vol. 2, pp. 1474–1479. IEEE. doi:10.1109/CEC.2002.1004460.

Kriofsky, T. ., & Kaplan, L. M. (1975). Inductively Coupled Transmitter-Responder Arrangement. US.

Krishnan, S., & Rangayyan, R. M. (2000). Denoising knee joint vibration signals using adaptive time-frequency representations. *Engineering Solutions for the Next Millennium. 1999 IEEE Canadian Conference on Electrical and Computer Engineering (Cat. No.99TH8411)*, Vol. 3, pp. 1495–1500. IEEE. doi:10.1109/CCECE.1999.804930.

Krohling, R. A. (2004). Gaussian swarm: a novel particle swarm optimization algorithm. *IEEE Conference on Cybernetics and Intelligent Systems, 2004.*, Vol. 1, pp. 372–376. IEEE. doi:10.1109/ICCIS.2004.1460443.

Krohling, R. A. (2005). Gaussian Particle Swarm with Jumps. *2005 IEEE Congress on Evolutionary Computation*, Vol. 2, pp. 1226–1231. IEEE. doi:10.1109/CEC.2005.1554830.

Krohn, S., Morthorst, P.-E., & Awerbuch, S. (2007). *The economics of wind energy*. *Renewable and Sustainable Energy Reviews*, Vol. 13, pp. 1–156. Retrieved from http://linkinghub.elsevier.com/retrieve/pii/S1364032108001299.

Kudo, T., & Matsumoto, Y. (2004). A Boosting Algorithm for Classification of Semi-Structured Text. *EMNLP*, pp. 301–308.

Kusiak, A., & Verma, A. (2011). Prediction of Status Patterns of Wind Turbines: A Data-Mining Approach. *Journal of Solar Energy Engineering*, Vol. 133, No. 1, pp. 1–10. doi:10.1115/1.4003188.

Kusiak, A., & Zhang, Z. (2010). Analysis of Wind Turbine Vibrations Based on SCADA Data. *Journal of Solar Energy Engineering*, Vol. 132, No. 3, pp. 1–12. doi:10.1115/1.4001461.

Reference

Lai, L. L. (1998). *Intelligent system application in power engineering: evolutionary programming and neural networks*. New York: Wiley, John & Sons.

Landt, J. (2001). *Shrouds of Time The history of RFID*, pp. 1–11. Pittsburgh, USA.

Landt, J. (2005). The history of RFID. *IEEE Potentials*, Vol. 24, No. 4, pp. 8–11. doi:10.1109/MP.2005.1549751.

Lee, I. (2011). Fault Diagnosis of Induction Motors Using Discrete Wavelet Transform and Artificial Neural Network, pp. 510–514.

Lee, J., Ni, J., Djurdjanovic, D., Qiu, H., & Liao, H. (2006). Intelligent prognostics tools and e-maintenance. *Computers in Industry*, Vol. 57, No. 6, pp. 476–489. doi:10.1016/j.compind.2006.02.014.

Lee, K. Y., & Yang, F. F. (1998). Optimal reactive power planning using evolutionary algorithms: a comparative study for evolutionary programming, evolutionary strategy, genetic algorithm, and linear programming. *IEEE Transactions on Power Systems*, Vol. 13, No. 1, pp. 101–108. doi:10.1109/59.651620.

Lewis, F. L. (2004). Wireless Sensor Networks. In D. J. Cook & S. K. Das (Eds.), *Smart Environments: Technologies, Protocols, and Applications*, pp. 1–18. New York: John Wiley. doi:10.1007/b117506.

Lewis, S. (2004). *A Basic Introduction to RFID technology and Its Use in the Supply Chain*, pp. 1–30.

Li, D., Wang, W., & Ismail, F. (2013). Enhanced fuzzy-filtered neural networks for material fatigue prognosis. *Applied Soft Computing*, Vol. 13, No. 1, pp. 283–291. doi:10.1016/j.asoc.2012.08.031.

Li, G., Qin, Q., & Dong, C. (2000). Monitor the Optimal Placement Position of the Sensors in System by Genetic Algorithm. *Engineering Mechanics*, Vol. 17, No. 1, pp. 25–34.

Li, R., Sopon, P., & He, D. (2009). Fault features extraction for bearing prognostics. *Journal of Intelligent Manufacturing*, Vol. 23, No. 2, pp. 313–321. doi:10.1007/s10845-009-0353-z.

Li, X., Ge, S. S., Pan, Y., Hong, K., Zhang, Z., & Hu, X. (2011). Feature extraction based on common spatial analysis for time domain parameters. *2011 8th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pp. 377–382. IEEE. doi:10.1109/URAI.2011.6145846.

Li, X., Qu, L., Wen, G., & Li, C. (2003). Application of wavelet packet analysis for fault detection in electro-mechanical systems based on

torsional vibration measurement. *Mechanical Systems and Signal Processing*, Vol. 17, No. 6, pp. 1219–1235. doi:10.1006/mssp.2002.1517.

Li, Z., Wu, Z., He, Y., & Fulei, C. (2005). Hidden Markov model-based fault diagnostics method in speed-up and speed-down process for rotating machinery. *Mechanical Systems and Signal Processing*, Vol. 19, No. 2, pp. 329–339. doi:10.1016/j.ymssp.2004.01.001.

Liang, J. J., Qin, A. K., Suganthan, P. N., & Baskar, S. (2006). Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Transactions on Evolutionary Computation*, Vol. 10, No. 3, pp. 281–295. doi:10.1109/TEVC.2005.857610.

Lienhart, W., & Brunner, F. K. (2003). MONITORING OF BRIDGE DEFORMATIONS USING EMBEDDED FIBER OPTICAL SENSORS. *Proceedings of11th FIG Symposium on Deformation Measurements*, pp. 1–7. Santorini.

Lin, J., & Qu, L. (2000). Feature extraction based on Morlet wavelet and its application for mechanical diagnosis. *Journal of Sound and Vibration*, Vol. 234, No. 1, pp. 135–148. doi:10.1006/jsvi.2000.2864.

Liu, G. P. (2001). *Nonlinear identification and control: a neural network approach. International Journal of Adaptive Control and Signal Processing*, Vol. 21. London: Springer. doi:10.1002/acs.953.

Liu, H., & Abraham, A. (2005). Fuzzy adaptive turbulent particle swarm optimization. *Fifth International Conference on Hybrid Intelligent Systems (HIS'05)*, p. 6 pp. IEEE. doi:10.1109/ICHIS.2005.49.

Liu, W., Gao, W., Sun, Y., & Xu, M. (2008). Optimal sensor placement for spatial lattice structure based on genetic algorithms. *Journal of Sound and Vibration*, Vol. 317, No. 1-2, pp. 175–189. doi:10.1016/j.jsv.2008.03.026.

Liu, Y., Guo, L., Wang, Q., An, G., Guo, M., & Lian, H. (2010). Application to induction motor faults diagnosis of the amplitude recovery method combined with FFT. *Mechanical Systems and Signal Processing*, Vol. 24, No. 8, pp. 2961–2971. doi:10.1016/j.ymssp.2010.03.008.

Llobet, E., Brezmes, J., Vilanova, X., Sueiras, J. E., & Correig, X. (1997). Qualitative and quantitative analysis of volatile organic compounds using transient and steady-state responses of a thick-film tin oxide gas sensor array. *Sensors and Actuators B: Chemical*, Vol. 41, No. 1-3, pp. 13–21. doi:10.1016/S0925-4005(97)80272-9.

Reference

Loucks, D. P., van Beek, E., Stedinger, J. R., Dijkman, J. P. M., & Villars, M. T. (2005). *Water Resources Systems Planning and Management: An Introduction to Methods, Models and Applications*, p. 690. Paris: UNESCO.

Lovbjerg, M., & Krink, T. (2002). Extending particle swarm optimisers with self-organized criticality. *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, Vol. 2, pp. 1588–1593. IEEE. doi:10.1109/CEC.2002.1004479.

Mallat, S. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 7, pp. 674–693. doi:10.1109/34.192463.

Markard, J., & Petersen, R. (2009). The offshore trend: Structural changes in the wind power sector. *Energy Policy*, Vol. 37, No. 9, pp. 3545–3556. doi:10.1016/j.enpol.2009.04.015.

Markou, M., & Singh, S. (2003). Novelty detection: a review–part 2: neural network based approaches. *Signal Processing*, Vol. 83, No. 12, pp. 2499–2521. doi:10.1016/j.sigpro.2003.07.019.

Marquez, A. C. (2007). *The Maintenance Management Framework*, p. 340.

Marseguerra, M., Zio, E., & Podofillini, L. (2002). Condition-based maintenance optimization by means of genetic algorithms and Monte Carlo simulation. *Reliability Engineering & System Safety*, Vol. 77, No. 2, pp. 151–165. doi:10.1016/S0951-8320(02)00043-1.

Marwala, T. (2012). *Condition Monitoring Using Computational Intelligence Methods*. London: Springer London.

Marwali, M. K. C., & Shahidehpour, S. M. (2000). Coordination between long-term and short-term generation scheduling with network constraints. *IEEE Transactions on Power Systems*, Vol. 15, No. 3, pp. 1161–1167. doi:10.1109/59.871749.

Marzi, H. (2004). Real-time fault detection and isolation in industrial machines using learning vector quantization. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, Vol. 218, No. 8, pp. 949–959. doi:10.1243/0954405041486109.

May, R., Dandy, G., & Maier, H. (2011). Review of Input Variable Selection Methods for Artificial Neural Networks. In K. Suzuki (Ed.), *Artificial Neural Networks - Methodological Advances and Biomedical Applications*. InTech. doi:10.5772/16004.

Reference

Mazomenos, E. B., Chen, T., Acharyya, A., Bhattacharya, A., Rosengarten, J., & Maharatna, K. (2012). A Time-Domain Morphology and Gradient based algorithm for ECG feature extraction. *2012 IEEE International Conference on Industrial Technology*, pp. 117–122. IEEE. doi:10.1109/ICIT.2012.6209924.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, Vol. 5, No. 4, pp. 115–133. doi:10.1007/BF02478259.

Mendonqa, L. F. (2006). Fault Isolation Using Fuzzy Model-based Observers. *IFAC Fault Detection, Supervision and Safety of technical processes*, pp. 735–740.

Meng, X., & Meng, X. (2010). Nonlinear System Simulation Based on the BP Neural Network. *2010 Third International Conference on Intelligent Networks and Intelligent Systems*, pp. 334–337. IEEE. doi:10.1109/ICINIS.2010.159.

Menon, S., Schoess, J. N., Hamza, R., & Busch, D. (2000). Wavelet-Based Acoustic Emission Detection Method with Adaptive Thresholding. In V. V. Varadan, N. M. Wereley, R. O. Claus, Y. Bar-Cohen, S.-C. Liu, T. T. Hyde, V. K. Varadan, et al. (Eds.), *Proc. SPIE 3986, Smart Structures and Materials 2000: Sensory Phenomena and Measurement Instrumentation for Smart Structures and Materials*, Vol. 3986, pp. 71–77. doi:10.1117/12.388093.

Miranda, A. A., Borgne, Y.-A., & Bontempi, G. (2008). New Routes from Minimal Approximation Error to Principal Components. *Neural Processing Letters*, Vol. 27, No. 3, pp. 197–207.

Miranda, V., & Fonseca, N. (2002). New evolutionary particle swarm algorithm (EPSO) applied to voltage/VAR control. *Proc. 14th Power Syst. Comput. Conf*.

Miranda, V., Srinivasan, D., & Proença, L. (1998). Evolutionary computation in power systems. *International Journal of Electrical Power & Energy Systems*, Vol. 20, No. 2, pp. 89–98. doi:10.1016/S0142-0615(97)00040-9.

Mobley, R. K. (1990). *An Introduction to Predictive Maintenance*. New York: Van Nostrand Reinhold.

Mobley, R. K. (2002). *An introduction to predictive maintenance*, Second., pp. 1–438. New York: Elsevier Science.

Molter, A., da Silveira, O. a. A., Fonseca, J. S. O., & Bottega, V. (2010). Simultaneous Piezoelectric Actuator and Sensor Placement Optimization and Control Design of Manipulators with Flexible Links

Using SDRE Method. *Mathematical Problems in Engineering*, Vol. 2010, pp. 1–23. doi:10.1155/2010/362437.

Momoh, J., & Button, R. (2003). Design and analysis of aerospace DC arcing faults using fast fourier transformation and artificial neural network. *2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No.03CH37491)*, pp. 788–793. IEEE. doi:10.1109/PES.2003.1270407.

Naimimohasses, R., Barnett, D. M., Green, D. a, & Smith, P. R. (1995). Sensor optimization using neural network sensitivity measures. *Measurement Science and Technology*, Vol. 6, No. 9, pp. 1291–1300. doi:10.1088/0957-0233/6/9/008.

Nakajima, S. (1988). *Introduction to TPM: Total Productive Maintenance*, pp. 1–129. Portland: Productivity Press.

Nakamoto, T., Iguchi, A., & Moriizumi, T. (2000). Vapor supply method in odor sensing system and analysis of transient sensor responses. *Sensors and Actuators B: Chemical*, Vol. 71, No. 3, pp. 155–160. doi:10.1016/S0925-4005(99)00186-0.

Nassim, L. (2011). Support Vector Machines for Fault Detection in Wind Turbines. In B. Sergio (Ed.), *Proceedings of the 18th IFAC World Congress, 2011*, pp. 7067–7072. Milano, Italy. doi:10.3182/20110828-6-IT-1002.02560.

Negnevitsky, M., & Kelareva, G. (1999). Application of genetic algorithms for maintenance scheduling in power systems. *ICONIP'99. ANZIIS'99 & ANNES'99 & ACNN'99. 6th International Conference on Neural Information Processing. Proceedings (Cat. No.99EX378)*, Vol. 2, pp. 447–452. IEEE. doi:10.1109/ICONIP.1999.845636.

Niu, G., Yang, B.-S., & Pecht, M. (2010). Development of an optimized condition-based maintenance system by data fusion and reliability-centered maintenance. *Reliability Engineering & System Safety*, Vol. 95, No. 7, pp. 786–796. doi:10.1016/j.ress.2010.02.016.

Obermaier, B., Guger, C., Neuper, C., & Pfurtscheller, G. (2001). Hidden Markov models for online classification of single trial EEG data. *Pattern Recognition Letters*, Vol. 22, No. 12, pp. 1299–1309. doi:10.1016/S0167-8655(01)00075-7.

Oja, E. (2002). Unsupervised Learning in Neural Computation. *Theoretical Computer Science*, Vol. 287, No. 1, pp. 187–207.

Okamura, S. (2011). *The Short Time Fourier Transform and Local Signals*. Carnegie Mellon University.

Reference

Orosz, J. E., & Jacobson, S. H. (2002). Analysis of Static Simulated Annealing Algorithms 1. *Journal of Optimization theory and Applications*, Vol. 115, No. 1, pp. 165–182. doi:10.1023/A:1019633214895.

Pan, H., Wei, X., & Xu, X. (2010). Research of optimal placement of gearbox sensor based on particle swarm optimization. *2010 8th IEEE International Conference on Industrial Informatics*, pp. 108–113. IEEE. doi:10.1109/INDIN.2010.5549454.

Pan, J. J., Yang, Q., Chang, H., & Yeung, D. (2001). A Manifold Regularization Approach to Calibration Reduction for Sensor-Network Based Tracking. *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, pp. 988–993. Boston.

Papadimitriou, C. (2004). Optimal sensor placement methodology for parametric identification of structural systems. *Journal of Sound and Vibration*, Vol. 278, No. 4-5, pp. 923–947. doi:10.1016/j.jsv.2003.10.063.

Parsopoulos, K. E., & Vrahatis, M. N. (2002). Recent approaches to global optimization problems through Particle Swarm Optimization. *Natural Computing*, Vol. 1, pp. 235–306. doi:10.1023/A:1016568309421.

Parsopoulos, K. E., & Vrahatis, M. N. (2004). On the Computation of All Global Minimizers Through Particle Swarm Optimization. *IEEE Transactions on Evolutionary Computation*, Vol. 8, No. 3, pp. 211–224. doi:10.1109/TEVC.2004.826076.

Patil, P. B., & Chavan, M. S. (2012). A wavelet based method for denoising of biomedical signal. *International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME-2012)*, pp. 278–283. IEEE. doi:10.1109/ICPRIME.2012.6208358.

Patterson, D. W. R., & Hughes, J. G. (1997). Case-based reasoning for fault diagnosis. *New Review of Applied Expert Systems and Emerging Technologies*, Vol. 3.

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, Vol. 2, No. 6, pp. 559–572.

Peng, Z., He, Y., Chen, Z., & Chu, F. (2002). Identification of the Shaft Orbit for Rotating Machines Using Wavelet Modulus Maxima. *Mechanical Systems and Signal Processing*, Vol. 16, No. 4, pp. 623–635. doi:10.1006/mssp.2002.1494.

Pennacchi, P., & Vania, A. (2008). Diagnostics of a crack in a load coupling of a gas turbine using the machine model and the analysis of the shaft

vibrations. *Mechanical Systems and Signal Processing*, Vol. 22, No. 5, pp. 1157–1178. doi:10.1016/j.ymssp.2007.10.005.

Pereira, C. M. N. a., Lapa, C. M. F., Mol, A. C. a., & da Luz, A. F. (2010). A Particle Swarm Optimization (PSO) approach for non-periodic preventive maintenance scheduling programming. *Progress in Nuclear Energy*, Vol. 52, No. 8, pp. 710–714. doi:10.1016/j.pnucene.2010.04.009.

Perla, H. F. et al. (1984). *A Guide for Developing Preventive Maintenance Programs in Electric Power Plants, EPRI NP-3416*.

Pham, D. T., Ghanbarzadeh, A., Koç, E., Otri, S., Rahim, S., & Zaidi, M. (2006). The Bees Algorithm – A Novel Tool for Complex Optimisation Problems. *Proceeding of 2nd International Virtual Conference on Intelligent Production Machines and Systems*, pp. 454–459.

Pinar Pérez, J. M., García Márquez, F. P., Tobias, A., & Papaelias, M. (2013). Wind turbine reliability analysis. *Renewable and Sustainable Energy Reviews*, Vol. 23, pp. 463–472. doi:10.1016/j.rser.2013.03.018.

Piñeyro, J., Klempnow, A., & Lescano, V. (2000). Effectiveness of new spectral tools in the anomaly detection of rolling element bearings. *Journal of Alloys and Compounds*, Vol. 310, No. 1-2, pp. 276–279. doi:10.1016/S0925-8388(00)00964-6.

Portnoff, M. R. (1980). Time-Frequency Representation of . Digital Signals. *IEEE Transactions on Acoustics, Speech and Signal Processing ASSP*, Vol. 28, No. 1, pp. 55–69.

Prabhakar, S., Mohanty, a. R., & Sekhar, a. . (2002). Application of discrete wavelet transform for detection of ball bearing race faults. *Tribology International*, Vol. 35, No. 12, pp. 793–800. doi:10.1016/S0301-679X(02)00063-4.

Rai, V. K., & Mohanty, A. R. (2007). Bearing fault diagnosis using FFT of intrinsic mode functions in Hilbert–Huang transform. *Mechanical Systems and Signal Processing*, Vol. 21, No. 6, pp. 2607–2615. doi:10.1016/j.ymssp.2006.12.004.

Ramani, S., Blu, T., & Unser, M. (2008). Blind optimization of algorithm parameters for signal denoising by Monte-Carlo Sure. *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 905–908. IEEE. doi:10.1109/ICASSP.2008.4517757.

Reisbec, C. K., & Schank, R. C. (1989). *Inside Case-Based Reasoning*. Hillsdale, NJ: Lawrence Earlbaum Associates.

Reference

Roussel, S., Forsberg, G., Steinmetz, V., Grenier, P., & Bellon-Maurel, V. (1998). Optimisation of electronic nose measurements. Part I: Methodology of output feature selection. *Journal of Food Engineering*, Vol. 37, No. 2, pp. 207–222. doi:10.1016/S0260-8774(98)00081-8.

Ruhanne, A., Hanhikorpi, M., Bertuccelli, F., Colonna, A., Malik, W., Ranasinghe, D., López, T. S., et al. (2008). *Sensor-enabled RFID Tag Handbook*, pp. 1–47.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning Internal Representations by Error Propagation. In D. E. Rumenhart & J. L. McCelland (Eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, pp. 318–362. Cambridge: MIT Press.

Saravanan, N., Kumar Siddabattuni, V. N. S., & Ramachandran, K. I. (2008). A comparative study on classification of features by SVM and PSVM extracted using Morlet wavelet for fault diagnosis of spur bevel gear box. *Expert Systems with Applications*, Vol. 35, No. 3, pp. 1351–1366. doi:10.1016/j.eswa.2007.08.026.

Saravanan, N., & Ramachandran, K. I. (2010). Incipient gear box fault diagnosis using discrete wavelet transform (DWT) for feature extraction and classification using artificial neural network (ANN). *Expert Systems With Applications*, Vol. 37, No. 6, pp. 4168–4181. doi:10.1016/j.eswa.2009.11.006.

Satoh, T., & Nara, K. (1991). Maintenance scheduling by using simulated annealing method (for power plants). *IEEE Transactions on Power Systems*, Vol. 6, No. 2, pp. 850–857. doi:10.1109/59.76735.

Saxena, a., & Vachtsevanos, G. (2005). A methodology for analyzing vibration data from planetary gear systems using complex morlet wavelets. *Proceedings of the 2005, American Control Conference, 2005.*, Vol. 2, pp. 4730–4735. IEEE. doi:10.1109/ACC.2005.1470743.

Schölkopf, B., Burges, C. J. C., & Smola, A. J. (1999). *Advances in Kernel Methods: Support Vector Learning*. Cambridge, MA: The MIT Press.

Schubert, U., Kruger, U., Arellano-Garcia, H., de Sá Feital, T., & Wozny, G. (2011). Unified model-based fault diagnosis for three industrial application studies. *Control Engineering Practice*, Vol. 19, No. 5, pp. 479–490. doi:10.1016/j.conengprac.2011.01.009.

Secrest, B. R., & Lamont, G. B. (2002). Visualizing particle swarm optimization - Gaussian particle swarm optimization. *Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No.03EX706)*, pp. 198–204. IEEE. doi:10.1109/SIS.2003.1202268.

Reference

Seker, S., & Ayaz, E. (2003). Feature extraction related to bearing damage in electric motors by wavelet analysis. *Journal of the Franklin Institute*, Vol. 340, No. 2, pp. 125–134. doi:10.1016/S0016-0032(03)00015-2.

Shakhnarovich, G., Darrell, T., & Indyk, P. (2005). *Nearest-Neighbor Methods in Learning and Vision : Theory and Practice*. Cambridge, MA: MIT Press.

Shenoy, D. B., & Bhadbury, B. (1998). *Maintenance resources management: Adapting MRP*. London: Taylor & Francis.

Shi, Y., & Eberhart, R. C. (2001). Fuzzy adaptive particle swarm optimization. *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, Vol. 1, pp. 101–106. IEEE. doi:10.1109/CEC.2001.934377.

Shibata, K., Takahashi, A., & Shirai, T. (2000). Fault Diagnostics of Rotating Machinery Through Visualization of Sound Signals. *Mechanical Systems and Signal Processing*, Vol. 14, No. 2, pp. 229–241. doi:10.1006/mssp.1999.1255.

Shinde, A. D. (2004). *A Wavelet Packet Based Sifting Process and Its Application for Structural Health Monitoring*. Worcester Polytechnic Institute.

Si, X., Wang, W., Hu, C., & Zhou, D. (2011). Remaining useful life estimation – A review on the statistical data driven approaches. *European Journal of Operational Research*, Vol. 213, No. 1, pp. 1–14. doi:10.1016/j.ejor.2010.11.018.

Siegelmann, H. T., & Sontag, E. D. (1994). Analog computation via neural networks. *Theoretical Computer Science*, Vol. 131, No. 2, pp. 331–360. doi:10.1016/0304-3975(94)90178-3.

Silva, C. W. De. (1989). *Control sensors and actuators*. Englwood Cliff, NJ: Prentice Hall.

Sindhwani, V., Niyogi, P., & Belkin, M. (2005). Beyond the point cloud: from transductive to semi-supervised learning. *Proceeding of ICML '05*, pp. 824–831. New York.

Smola, A. J., & Scholkopf, B. (2003). A Tutorial on Support Vector Regression. *Statistics and Computing*, Vol. 14, No. 3, pp. 199–222.

Soman, K. P., & Ramachandran, K. I. (2005). *Insight into Wavelets from Theory to Practice*, 2nd ed., p. 404. India: PHI Learning Pvt. Ltd.

Soman, R. R., Davidson, E. M., McArthur, S. D. J., Fletcher, J. E., & Ericsen, T. (2012). Model-based methodology using modified sneak

circuit analysis for power electronic converter fault diagnosis. *IET Power Electronics*, Vol. 5, No. 6, pp. 813.

Speybroeck, N. (2012). Classification and regression trees. *International journal of public health*, Vol. 57, No. 1, pp. 243–6. doi:10.1007/s00038-011-0315-z.

Spiewak, S. a., Duggirala, R., & Barnett, K. (2000). Predictive Monitoring and Control of the Cold Extrusion Process. *CIRP Annals - Manufacturing Technology*, Vol. 49, No. 1, pp. 383–386. doi:10.1016/S0007-8506(07)62970-9.

SS-EN 13306 Maintenance Terminology (2001). Swedish Standards Institute.

Staszewski, W. J. (1997). Vibration data compression with optimal wavelet coefficients. *Second International Conference on Genetic Algorithms in Engineering Systems*, Vol. 1997, pp. 186–190. IEE. doi:10.1049/cp:19971178.

Staszewski, W. J. (2002). Intelligent signal processing for damage detection in composite materials. *Composites Science and Technology*, Vol. 62, No. 7-8, pp. 941–950. doi:10.1016/S0266-3538(02)00008-8.

Staszewski, W. J., Worden, K., & Tomlinson, G. R. (1997). Time-frequency Analysis Gearbox Fault Detection Using the Wigner Ville Distribution and Pattern Recognition. *Mechanical Systems and Signal Processing*, Vol. 11, No. 5, pp. 673–692. doi:10.1006/mssp.1997.0102.

Steinwart, I., & Christmann, A. (2008). *Support Vector Machines*. New York: Springer-Verlag.

Stockman, H. (1948). Communication by Means of Reflected Power. *Proceedings of the IRE*, Vol. 36, No. 10, pp. 1196–1204. doi:10.1109/JRPROC.1948.226245.

Stuart, A., & Allocca, A. (1984). *Transducers: theory and Applications*. Englwood Cliff, NJ: Prentice-Hall.

Sullivan, G. P., Pugh, R., Melendez, A. P., & Hunt, W. D. (2010). *Operations & Maintenance Best Practices Release 3.0: A Guide to Achieving Operational Efficiency*, pp. 1–321.

Sun, H.-C., Huang, Y.-C., & Huang, C.-M. (2012). Fault Diagnosis of Power Transformers Using Computational Intelligence: A Review. *Energy Procedia*, Vol. 14, pp. 1226–1231. doi:10.1016/j.egypro.2011.12.1080.

Sun, X., Li, H., & Ou, J. (2008). Improved Genetic Algorithm of Dynamic Detection Measurement Point Optimization of Large Bridge and its

Application. *Architectural and Science Journal of Xian*, Vol. 38, No. 5, pp. 624–628.

Sutoh, T., Suzuki, H., & Nagai, N. (1994). Large scale generator maintenance scheduling using simulated evolution. *Proceedings of the International Conference on Intelligent System Application to Power Systems*. Montpellier.

Tan, P.-N., Steinbach, M., & Kumar, V. (2006). *Introduction to Data Mining*. Boston, MA: Pearson Education/Addison Wesley.

Teodorovic, D., & Dell'Orco, M. (2005). Bee colony optimization – a cooperative learning approach to complex transportation problems. *Proceeding of 10th EWGT Meeting and 16th Mini-EURO Conference*, pp. 51–60.

Tian, D., & Li, N. (2009). Fuzzy Particle Swarm Optimization Algorithm. *2009 International Joint Conference on Artificial Intelligence*, pp. 263–267. doi:10.1109/JCAI.2009.50.

Tian, X., Cao, Y. P., & Chen, S. (2011). Process fault prognosis using a fuzzy-adaptive unscented Kalman predictor. *International Journal of Adaptive Control and Signal Processing*, Vol. 25, No. 9, pp. 813–830. doi:10.1002/acs.1243.

Triki, E., Collette, Y., & Siarry, P. (2005). A theoretical study on the behavior of simulated annealing leading to a new cooling schedule. *European Journal of Operational Research*, Vol. 166, No. 1, pp. 77–92. doi:10.1016/j.ejor.2004.03.035.

Tsai, Y. T. (2009). Applying a case-based reasoning method for fault diagnosis during maintenance. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, Vol. 223, No. 10, pp. 2431–2441. doi:10.1243/09544062JMES1588.

Tse, P. W., Yang, W., & Tam, H. Y. (2004). Machine fault diagnosis through an effective exact wavelet analysis. *Journal of Sound and Vibration*, Vol. 277, No. 4-5, pp. 1005–1024. doi:10.1016/j.jsv.2003.09.031.

Vachtsevanos, G., Lewis, F., Roemer, M., Hess, A., & Wu, B. (2006). *Intelligent Fault Diagnosis and Prognosis for Engineering System*, p. 454. New Jersey: John Wiley & Sons, Inc.

Vafaei, S., & Rahnejat, H. (2003). Indicated repeatable runout with wavelet decomposition (IRR-WD) for effective determination of bearing-induced vibration. *Journal of Sound and Vibration*, Vol. 260, No. 1, pp. 67–82. doi:10.1016/S0022-460X(02)00900-8.

Reference

Valle, Y., Member, S., Venayagamoorthy, G. K., Member, S., & Harley, R. G. (2008). Particle Swarm Optimization : Basic Concepts , Variants and Applications in Power Systems. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, Vol. 12, No. 2, pp. 171–195.

Van den Kerkhof, P., Gins, G., Vanlaer, J., & Van Impe, J. F. M. (2012). Dynamic model-based fault diagnosis for (bio)chemical batch processes. *Computers & Chemical Engineering*, Vol. 40, pp. 12–21. doi:10.1016/j.compchemeng.2012.01.013.

Vasudevan, R. (1985). *Application of reliability-centered maintenance to component cooling-water system at Turkey Point Units 3 and 4, EPRI NP-4271*.

Verma, A., & Kusiak, A. (2012). Fault Monitoring of Wind Turbine Generator Brushes: A Data-Mining Approach. *Journal of Solar Energy Engineering*, Vol. 134, No. 2, pp. 021001. doi:10.1115/1.4005624.

Walker, I. (1987). Development of a maintenance program. *Proceedings of the 14thInte-ram Conference*. Toronto.

Wang, C., Kang, Y., Shen, P., Chang, Y., & Chung, Y. (2010). Applications of fault diagnosis in rotating machinery by using time series analysis with neural network. *Expert Systems with Applications*, Vol. 37, No. 2, pp. 1696–1702. doi:10.1016/j.eswa.2009.06.089.

Wang, C., Zhang, Y., & Zhong, Z. (2008). Fault Diagnosis for Diesel Valve Trains based on Time–frequency Images. *Mechanical Systems and Signal Processing*, Vol. 22, No. 8, pp. 1981–1993. doi:10.1016/j.ymssp.2008.01.016.

Wang, D. D. (1996). Computational intelligence based machine fault diagnosis. *Proceedings of the IEEE International Conference on Industrial Technology (ICIT'96)*, pp. 465–469. IEEE. doi:10.1109/ICIT.1996.601632.

Wang, H., Song, Z., & Wang, H. (2002). Statistical process monitoring using improved PCA with optimized sensor locations. *Journal of Process Control*, Vol. 12, pp. 735–744. doi:http://dx.doi.org/10.1016/S0959-1524(01)00048-8.

Wang, K. (2002). *Intelligent Condition Monitoring and Diagnosis Systems*. Amsterdam: IOS Press.

Wang, K. (2005). *Applied Computational Intelligence in Intelligent Manufacturing Systems*. Australia: Advanced Knowledge International Pty Ltd.

Reference

Wang, K., & Zhang, Z. (2010). *Intelligent Fault Diagnosis and Prognosis systems (IFDPS) for Condition-based Maintenance*, pp. 1–21. Trondheim.

Wang, K., & Zhang, Z. (2012). *Application of Radio Frequency Identification (RFID) to Manufacturing*, pp. 1–24. Trondheim.

Wang, Y., & Handschin, E. (2000). A new genetic algorithm for preventive unit maintenance scheduling of power systems. *International Journal of Electrical Power & Energy Systems*, Vol. 22, No. 5, pp. 343–348. doi:10.1016/S0142-0615(99)00062-9.

Watson, I., & Marir, F. (2009). Case-based reasoning: A review. *The Knowledge Engineering Review*, Vol. 9, No. 04, pp. 327. doi:10.1017/S0269888900007098.

Wei, L., & Keogh, E. (2006). Semi-supervised time series classification. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '06*, p. 748. New York, New York, USA: ACM Press. doi:10.1145/1150402.1150498.

Wei, X., & Pan, H. (2010). *Particle Swarm Optimization and Intelligent Fault Diagnosis*. Beijing: National Defence Industry Press.

White, J., Kauer, J. S., Dickinson, T. a, & Walt, D. R. (1996). Rapid analyte recognition in a device based on optical sensors and the olfactory system. *Analytical chemistry*, Vol. 68, No. 13, pp. 2191–202. doi:10.1021/ac9511197.

White, R. M. (1987). A sensor classification scheme. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, Vol. 34, No. 2, pp. 124–6.

Wilson, A. (2002). *Asset Maintenance Management: A Guide to Developing Strategy and Improving Performance*. New York: Industrial Press, Inc.

Wilson, D. M., & DeWeerth, S. P. (1995). Odor discrimination using steady-state and transient characteristics of tin-oxide sensors. *Sensors and Actuators B: Chemical*, Vol. 28, No. 2, pp. 123–128. doi:10.1016/0925-4005(95)80036-0.

Wireman, T. (1990). *World Class Maintenance Management*. New York: Industrial Press.

Worden, K., & Burrows, A. P. (2001). Optimal sensor placement for fault detection. *Engineering Structures*, Vol. 23, pp. 885–901. doi:http://dx.doi.org/10.1016/j.bbr.2011.03.031.

Wu, J., & Chen, J.-C. (2006). Continuous wavelet transform technique for fault signal diagnosis of internal combustion engines. *NDT & E*

International, Vol. 39, No. 4, pp. 304–311. doi:10.1016/j.ndteint.2005.09.002.

Wu, J., & Kuo, J.-M. (2009). An automotive generator fault diagnosis system using discrete wavelet transform and artificial neural network. *Expert Systems with Applications*, Vol. 36, No. 6, pp. 9776–9783. doi:10.1016/j.eswa.2009.02.027.

Wu, J., & Liu, C.-H. (2009). An expert system for fault diagnosis in internal combustion engines using wavelet packet transform and neural network. *Expert Systems with Applications*, Vol. 36, No. 3, pp. 4278–4286. doi:10.1016/j.eswa.2008.03.008.

Wu, S., & Chow, T. W. S. (2004). Induction Machine Fault Detection Using SOM-Based RBF Neural Networks. *IEEE Transactions on Industrial Electronics*, Vol. 51, No. 1, pp. 183–194. doi:10.1109/TIE.2003.821897.

Xie, X., Zhang, W., & Yang, Z. (2002). Dissipative particle swarm optimization. *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, Vol. 2, pp. 1456–1461. IEEE. doi:10.1109/CEC.2002.1004457.

Xin, W., Xiao-yun, C., Ke-ju, X., Rong-min, Z., Qing-jun, P., & Lin, L. (2012). Ice-coated AC transmission lines tension prognosis with autoregressive model. *2012 3rd International Conference on System Science, Engineering Design and Manufacturing Informatization*, Vol. 2, pp. 168–170. IEEE. doi:10.1109/ICSSEM.2012.6340835.

Yang, H., & Liao, C. (2001). A de-noising scheme for enhancing wavelet-based power quality monitoring system. *IEEE Transactions on Power Delivery*, Vol. 16, No. 3, pp. 353–360. doi:10.1109/61.924810.

Yare, Y., & Venayagamoorthy, G. K. (2010). Optimal maintenance scheduling of generators using multiple swarms-MDPSO framework. *Engineering Applications of Artificial Intelligence*, Vol. 23, No. 6, pp. 895–910. doi:10.1016/j.engappai.2010.05.006.

Yare, Y., Venayagamoorthy, G. K., & Aliyu, U. O. (2008). Optimal generator maintenance scheduling using a modified discrete PSO. *IET Generation, Transmission & Distribution*, Vol. 2, No. 6, pp. 834. doi:10.1049/iet-gtd:20080030.

Yellen, J., Al-Khamis, T. M., Vemuri, S., & Lemonidis, L. (1992). A decomposition approach to unit maintenance scheduling. *IEEE Transactions on Power Systems*, Vol. 7, No. 2, pp. 726–733. doi:10.1109/59.141779.

Reference

Yen, G. G., & Lin, K. (1999). Conditional health monitoring using vibration signatures. *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No.99CH36304)*, Vol. 5, pp. 4493–4498. IEEE. doi:10.1109/CDC.1999.833249.

Yen, G. G., & Lin, K. (2000). Wavelet Packet Feature Extraction for Vibration Monitoring, Vol. 47, No. 3, pp. 650–667.

Yoshimoto, K., Yasuda, K., Yokoyama, R., & Cory, B. J. (1993). Decentralized Hopfield neural network applied to maintenance scheduling of generating units in power systems. *Third International Conference on Artificial Neural Networks, 1993.,*, pp. 277–281. Brighton.

Yu, D., Yang, Y., & Cheng, J. (2007). Application of Time–frequency Entropy Method based on Hilbert–Huang Transform to Gear Fault Diagnosis. *Measurement*, Vol. 40, No. 9-10, pp. 823–830. doi:10.1016/j.measurement.2007.03.004.

Yuan, J. (2012). Manifold Assumption and Semi-Supervised Learning for Fault Diagnosis. *Data Mining for Zero-Defect Manufacturing*, pp. 133–148. Trondheim: Tapir Academic Press.

Zaher, A., McArthur, S. D. J., Infield, D. G., & Patel, Y. (2009). Online wind turbine fault detection through automated SCADA data analysis. *Wind Energy*, Vol. 12, No. 6, pp. 574–593. doi:10.1002/we.319.

Zhang, Z., & Kusiak, A. (2012). Monitoring Wind Turbine Vibration Based on SCADA Data. *Journal of Solar Energy Engineering*, Vol. 134, No. 2, pp. 021004. doi:10.1115/1.4005753.

Zhang, Z., & Wang, K. (2010). Application of Improved Discrete Particle Swarm Optimization (IDPSO) in Generating Unit Maintenance Scheduling. In K. Wang, O. Myklebust, & D. Tu (Eds.), *International Workshop of Advanced Manufacturing and Automation (IWAMA2010)*, pp. 79–86. Shanghai: Tapir Academic Press.

Zhang, Z., & Wang, K. (2011). Fault Isolation Using Self-organizing Map (SOM) ANNs. *IET International Conference of Wireless Mobile & Computing*, pp. 425–431. Shanghai: Institute Engineering and Technology.

Zhang, Z., & Wang, K. (2013). Dynamic Condition-Based Maintenance Scheduling Using Bee Colony Algorithm (BCA). In E. Qi, J. Shen, & R. Dou (Eds.), *Proceedings of International Asia Conference on Industrial Engineering and Management Innovation (IEMI2012)*, pp. 1607–1618. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-38445-5_169.

Reference

Zhang, Z., Wang, Y., & Wang, K. (2012). Fault diagnosis and prognosis using wavelet packet decomposition, Fourier transform and artificial neural network. *Journal of Intelligent Manufacturing*. doi:10.1007/s10845-012-0657-2.

Zhang, Z., Wang, Y., & Wang, K. (2013). Intelligent fault diagnosis and prognosis approach for rotating machinery integrating wavelet transform, principal component analysis, and artificial neural networks. *The International Journal of Advanced Manufacturing Technology*, Vol. 68, No. 1-4, pp. 763–773. doi:10.1007/s00170-013-4797-0.

Zheng, H., Li, Z., & Chen, X. (2002). Gear Fault Diagnosis Based on Continuous Wavelet Transform. *Mechanical Systems and Signal Processing*, Vol. 16, No. 2-3, pp. 447–457. doi:10.1006/mssp.2002.1482.

Zheng, Y., Tay, D. B. H., & Li, L. (2000). Signal extraction and power spectrum estimation using wavelet transform scale space filtering and Bayes shrinkage. *Signal Processing*, Vol. 80, No. 8, pp. 1535–1549. doi:10.1016/S0165-1684(00)00054-2.

Zou, M., Dayan, J., & Green, I. (2000). Dynamic simulation and monitoring of a non-contacting flexibly mounted rotor mechanical face seal. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, Vol. 214, No. 9, pp. 1195–1206. doi:10.1243/0954406001523632.