University of Alberta

Design of Knowledge-based Fault Detection and Identification for Dynamical Systems

by

Zhihan Xu 

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the
requirements for the degree of Master of Science

Department of Electrical and Computer Engineering

Edmonton, Alberta
Fall 2002

University of Alberta

Library Release Form

Name of Author: Zhihan Xu

Title of Thesis: Design of Knowledge-based Fault Detection and Identification for Dynamical Systems

Degree: Master of Science

Year this Degree Granted: 2002

Xu Zhihan

Zhihan Xu
Department of Electrical and Computer Engineering
Edmonton, Alberta
Canada T6G 2V4

Date: Sept. 3. 2002

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Design of Knowledge-based Fault Detection and Identification for Dynamical Systems** submitted by Zhihan Xu in partial fulfillment of the requirements for the degree of **Master of Science**.

Dr. Qing Zhao

Dr. Horacio J. Marquez

Dr. Biao Huang

Date: Sept. 3, 2002

Abstract

An effort to design a universal knowledge-based FDI scheme with generalized framework is made in this thesis. It relies on both signal processing techniques and qualitative methods by combining signal processing, statistical analysis, and neural network.

First, some model-based and knowledge-based FDI methods published in journals and proceedings are reviewed and compared. Second, the basic knowledge of wavelet analysis and improved regional self-organizing map neural network are introduced.

In order to detect the faults that reflect themselves as fault-induced time-frequency changes at certain time instants in the measured signal, wavelet analysis is applied to capture such changes and extract fault features. By introducing the concept of hierarchy training, coarse and delicacy training, and regional recognition, a so-called Regional SOM (RSOM) neural network proposed in this thesis has achieved higher clustering and matching-up precision compared to the conventional SOM network. Therefore, the proposed FDI scheme has a lower missing detection rate, and especially robust to the uncertainties and small variations in the systems.

Finally, case studies based on a nonlinear tank system and a DC motor system will be performed to demonstrate the effectiveness of this FDI scheme. Also, a model-based method, extended Kalman filter, is carried out and results are compared. Furthermore, the design is applied to a real process training system, and achieved satisfactory performance.

Keywords: Fault Detection and Identification, Wavelet Analysis, RSOM Neural Network

Acknowledgements

I would first like to thank my advisor, Dr. Qing Zhao, for directing the research and making this thesis possible. I am deeply grateful for her constant encouragement and enthusiasm.

I also thank committee members, Dr. Horacio J. Marquez and Dr. Biao Huang, for their careful reading of the thesis and many valuable suggestions and corrections.

I would like to thank all the fine people at the advanced control system laboratory who have made my time here enjoyable.

Finally, I would like to thank my parents for their love, support, encouragement, understanding, and patience. They are the foundation for who I am, and anything I have been able to accomplish is a tribute to them.

To My Parents

Contents

List of Figures

List of Tables

Chapter I

Introduction

1.1 What is FDI?

A typical control system consists of four basic elements: the dynamic plant, controllers, actuators and sensors, which work in a closed-loop configuration. Any kind of malfunction in these components can result in unacceptable anomaly in overall system performance. They are referred to as faults in a control system. According to their physical locations, they can be classified as dynamic faults, controller faults, actuator faults and sensor faults. The objective of Fault Detection and Identification (FDI) schemes is to detect, isolate, and identify these faults so that the system performance can be recovered.

A basic FDI scheme can be regarded as a two-step algorithm:

(1) To generate fault residual signals (or fault symptoms) by processing actual system measurement and nominal data from normal system models.

(2) To evaluate the residual signals/fault symptoms so that the time instant of the fault occurrence can be determined, and its location can be identified.

By achieving the above tasks, FDI can help human operator locate the faults in time and prevent the system from imminent severe damage. However, in order for the FDI scheme to be integrated with the reconfigurable control in a real-time environment (i.e. to achieve automatic *fault tolerant control*), more advanced FDI algorithms with higher intelligence need to be developed. They should include additional tasks as follows:

- Estimate the size of the fault. The nature and the cause of the fault is studied by analyzing the relations between the symptoms and their physical causes.
- Maintain the functionality (even at a degraded level) of the system by collaborating with reconfigurable control scheme.

By incorporating the above tasks in a control system, the structure of the closed-loop system is shown in Figure 1-1.

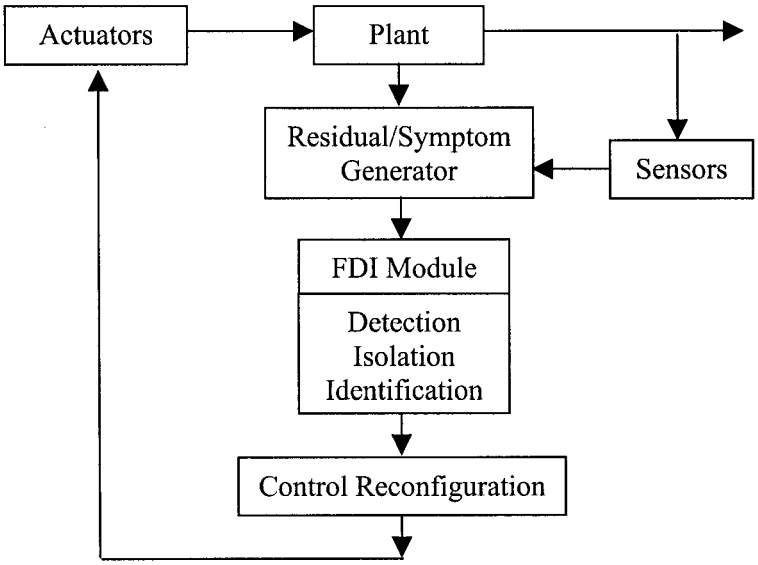


Figure 1-1. Structure of Integrated FDI and Reconfigurable Control Scheme

The research on FDI and fault tolerant control is originally motivated by the control problems in aircraft design. Many designs have been successfully applied to the flight control problems. Along with the development of the computer and information technologies, automatic control systems become more complex in structure, with more sophisticated control algorithms. There is a growing demand for fault tolerance built in these systems, which can be achieved not only by improving the individual reliabilities of the functional units but also by an effective fault-detection-identification scheme. With the FDI in the system, the overall reliability will be greatly improved and the operating expense will be reduced correspondingly. For this reason, the FDI technique has found its way to a wide variety of applications in addition to aircraft design, they are, just to name a few, process control, automobile design, power systems, aerospace, and even manufacturing industries.

For a well designed FDI scheme, the decision on faults should be independent of the magnitude, time duration and direction. Several important performance criteria include, but not limited to, the short detection time delay, the low rate of false alarm, the low rate of missing detection, and the low rate of incorrect identification.

In the last two decades, there have been extensive research efforts on developing model-based FDI methods [1]-[5]. However, in most practical systems, the mathematical models are nonlinear and/or not precisely known. In this case, the performance of the model-based schemes is jeopardized. To deal with this problem, knowledge-based (also called model-free or qualitative in some literatures) methods have been studied and developed in these years [6]-[8]. Stimulated by the quick development of artificial intelligence technologies and their successful applications, research in this field has become more vigorous recently [9]-[10]. This thesis will focus on the study of knowledge-based FDI methods, with comparison to the classical model-based methods.

1.2 Model-based FDI Methods

Most existing model-based FDI methods developed in the last two decades were based on the concept of *analytical redundancy*. The idea behind it is the use of analytical relationships of the dynamic interconnections between different system components [2], [4], [11], [12]. Basically, a *residual* signal is generated based on the differences of the actual system measurements and the estimated ones from a nominal model (i.e. the analytical redundancy). After being processed, it can be used as the indicator of normal/abnormal behavior for the purpose of fault detection. Whenever this indicator deviates from its theoretical value, one can assume that a fault has happened.

Therefore, for a model-based FDI method, the key is the validity of the analytical system models and statistical procedures to generate residuals or errors that are used to determine the probability of faults. Figure 1-2 describes a general structure of the model based FDI.

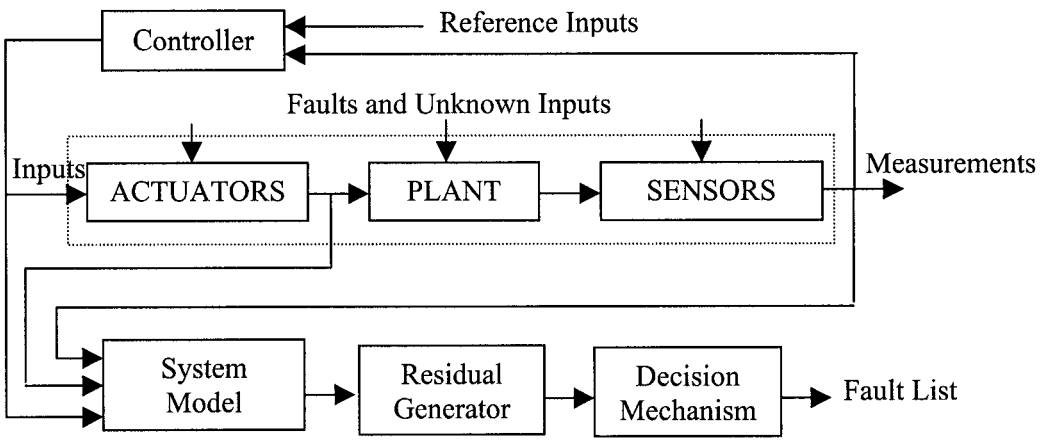


Figure 1-2. Conceptual Structure of FDI using Model-based Method

The systems models can be either in state space or input-output representations of dynamic systems. Accordingly, three important FDI methods have been developed based on state estimation (observer/Kalman filters), parameter estimation, and parity equations. The residuals are generated from the continuous comparison of the system under analysis with a nominal model, and are defined as differences between measured and expected quantities. The decision mechanism generates a decision rule to detect the fault based on the residual signal. It may contain simple threshold testing, or algorithms on statistical decision (hypothesis testing), such as generalized likelihood ratio testing, χ^2 testing, etc. [14].

Most Model-based approaches are based on the advanced information processing techniques such as state/parameter estimation, adaptive filtering, variable threshold logic, statistical decision theory, pattern recognition, etc. For model-based methods to be highly effective, it requires that the system model must be known explicitly. The sensitivity of the detection system to modeling errors degrades its performance and reduces its reliability. It is therefore a source of missing detections, false alarms or wrong diagnosis. To tackle such a problem, robust FDI methods have been developed, which, ideally, are insensitive or even invariant to modeling errors and uncertain factors [15]-[17]. In these methods, some careful treatments are proposed. However, the performance is still sacrificed due to the lack of fidelity of the model-based methods in highly uncertain and/or nonlinear systems.

1.3 Knowledge-based FDI Methods

In many practical applications, the systems are rather complex. By experiencing rapidly changing environments, and encountering various unexpected component faults, it is impossible to obtain the precise mathematical models. Often the traditional model-based FDI methods cannot guarantee satisfactory performance. For this reason, knowledge-based methods have been developed.

Knowledge-based methods are sometimes called model-free or qualitative methods. The 'model', herein, refers to the mathematical representation of the dynamical system. In the absence of complete analytical information on the system, these methods incorporate experience knowledge from 'engineers' or training process. The rationale behind this approach is that many experienced engineers can sense and locate faults even though they do not have extensive knowledge of the mechanical or electrical dynamics of the system. Indeed, one could make use of *a priori* knowledge of the system and the sensor measurements to diagnose faults. The methods rely on the data-driven and knowledge-based techniques to estimate the system dynamic, and determine the occurrence of the faults. The following figure shows the conceptual structure of a general knowledge-based FDI scheme.

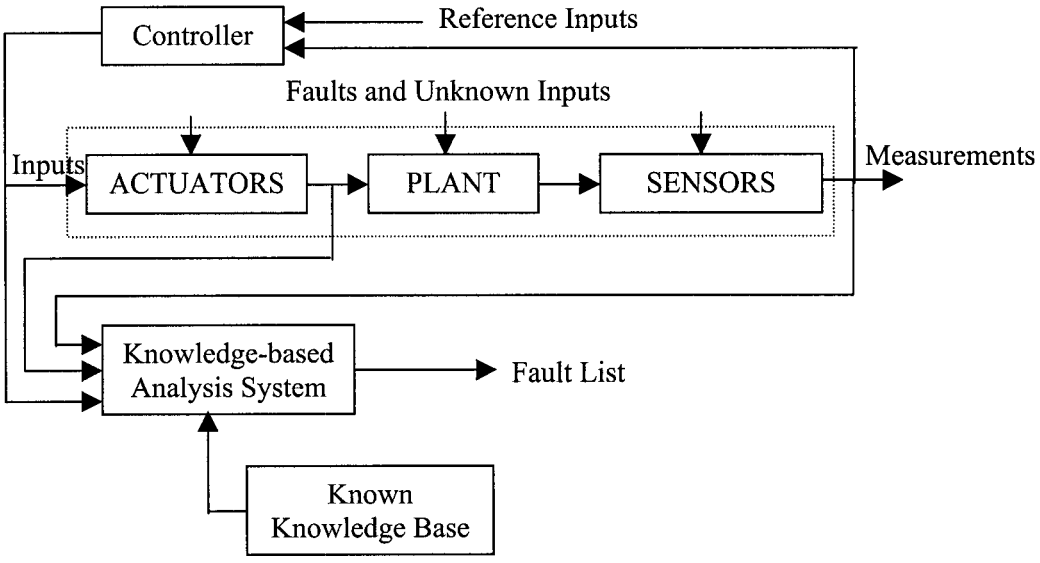


Figure 1-3. Conceptual Structure of FDI using Model-free Method

Design of knowledge-based FDI involves real-time signal processing, neural network, fuzzy and expert system, statistical decision theory, and pattern recognition, etc. There exist mainly four types of design algorithms, and they are, namely, Artificial Intelligence (AI) based direct methods, AI modeling based methods, signal processing (SP) based method, and AI & SP based hybrid method. In the following sections, these methods are reviewed with details in existing literatures.

1.3.1 AI Based Direct Methods

This method employs the artificial intelligence techniques, such as Artificial Neural Network (ANN), Fuzzy Inference System (FIS), and Expert System (ES), to diagnose the faults directly by mapping different operating status. Which AI system is selected depends on the special application case and the characteristics of the AI algorithm. The system measurements are taken as the input data to AI system. The output data of the AI system are post-processed by the decision function mechanism to generate the indicators of the operating situations, including normal and faulty ones. Figure 1-4 describes the concept of this method.

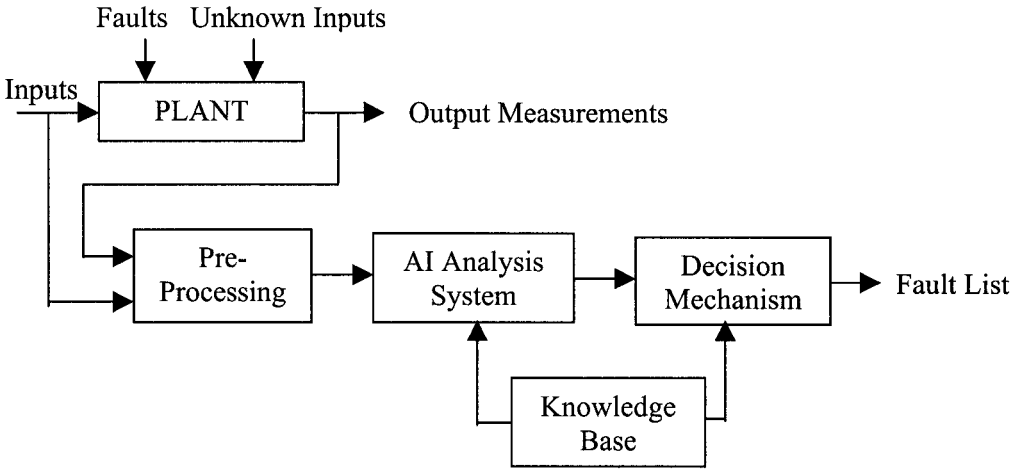


Figure 1-4. FDI Scheme with AI Based Direct Method

The structure of this FDI system contains the following three blocks:

The pre-processing layer is optional. It pre-processes the raw input data and output measurements, so that they satisfy the condition for effective AI analysis. Some

useful techniques include pre-filtering, normalization, and extraction of useful indices, etc.

The AI analysis system, in this case, is capable of generating and matching features based on the processed system measurements and the heuristic information from the knowledge base, thus provides the fault information to decision mechanism. According to the different AI methods applied, the structure of AI analysis system is also different. The choice depends on the special application case and the characteristics of the AI algorithm.

The decision mechanism incorporates suitable rules, which interprets the output data from AI system to generate the diagnostic information.

In the following sections, different AI methods are introduced:

- **ANN Analysis System**

An artificial neural network (ANN) is a computational structure that is "trained" to produce outputs corresponding to given inputs, on the basis of known input-output association [18]-[25]. It is known that ANNs are general nonlinear function approximations. The following figure describes the ANN analysis system.

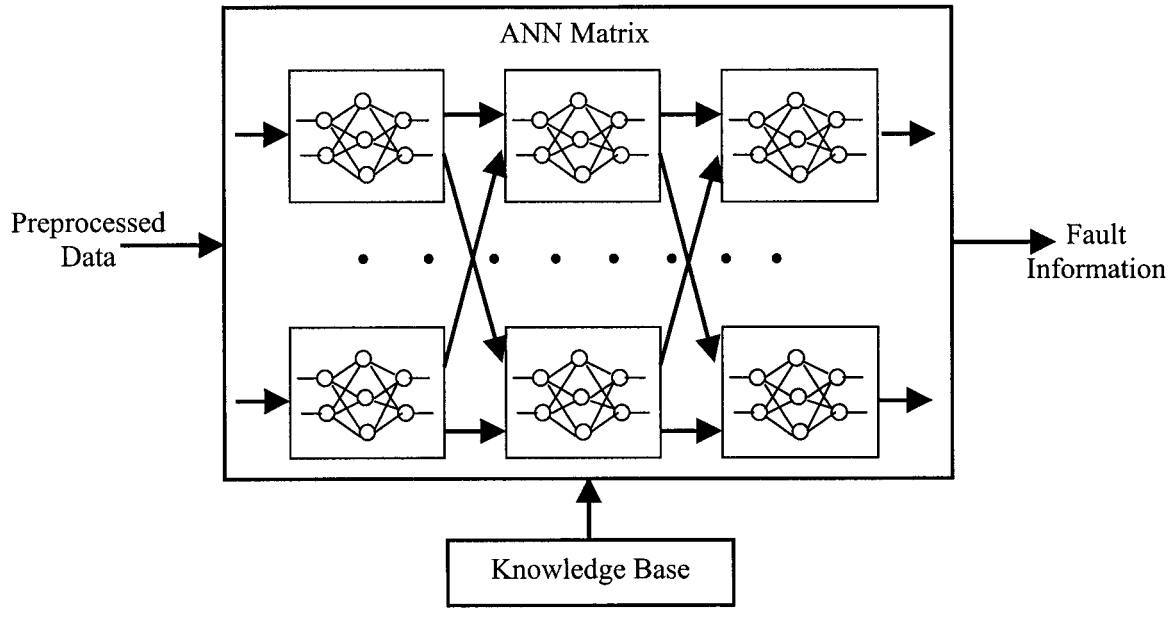


Figure 1-5. General Structure of ANN Analysis System

Different ANN structure have been proposed [26][27], it can be a single ANN, a vertical ANN vector (parallel ANN array), a horizontal ANN vector (serial ANN array), or an ANN matrix (multistage ANNs). The back propagation (BP) neural net, which has the excellent ability to map the nonlinear relationship and memorize the trained data sequences, is a common structure to be used in these references [18]-[27].

To design the detection system shown in Figure 1-5, one has to: (1) define the scheme architecture (definition of the ANN matrix), (2) build input data sets, and output data sets for supervising ANN learning, (3) choose ANN learning rules. The learning phase is carried out off-line on sample sequences extracted from system input/output signals, performed both in the absence of faults and in the presence of the possible faults. After learning, the neural net stores all possible operating information of the system in a knowledge base, which will be used for diagnosing a fault on line. While operating on line, ANN analyzes the data and provides the system information to the decision mechanism to identify whether the system belongs to the normal status or to one of the abnormal ones associated with the faults.

- **ES Analysis System**

An expert system (ES) is mainly based both on "passive knowledge", composed of facts and data that are known prior to the ES operating, and on "active knowledge", composed of knowledge rules in the form of IF - Then logic [28]. The following figure shows a general structure of ES analysis system.

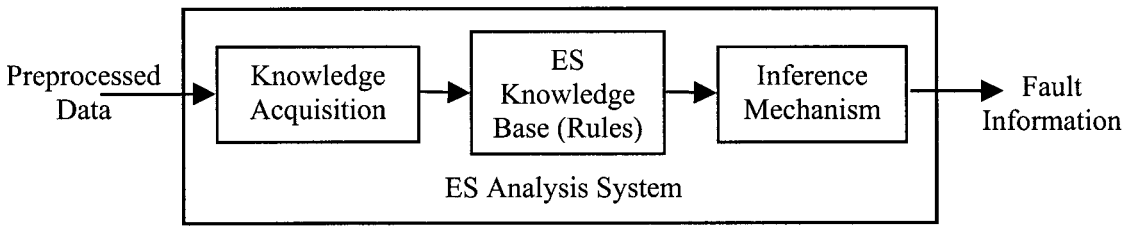


Figure 1-6. A General ES Analysis System

It is composed of two procedures: generation of residuals and thresholds, and generation of knowledge rules. The thresholds are derived based on the system

knowledge obtained by monitoring the system off-line and the residuals are generated on-line. It requires a specific expertise by the designer in the behavior of all operating devices. The knowledge rules build up the relationship among the residuals, thresholds, and fault information.

- **FIS Analysis System**

A fuzzy inference system (FIS) [29]-[33] is aimed at the development of a set of concepts and techniques for dealing with sources of uncertainty, imprecision, or incompleteness. The nature of fuzzy rules and the relationship between fuzzy sets of differing shapes are the basis of its powerful capability for incrementally modeling a system whose complexity makes traditional expert system, mathematical, and statistical approaches very difficult. FIS can also be considered as a general nonlinear function approximator. The following figure is an FIS analysis system that can be embedded in AI analysis system in Figure 1-4.

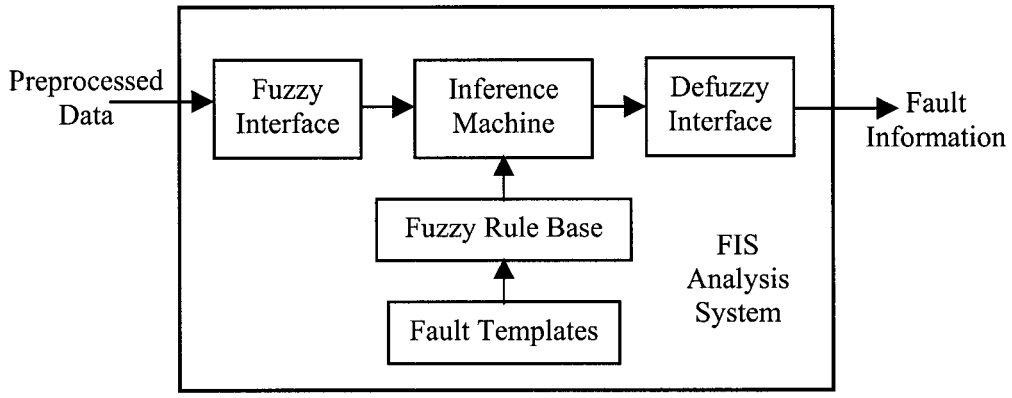


Figure 1-7. General AI Analysis System Structure with FIS

Design of FIS contains four steps: (1) Choose system structure, (2) Define membership function for input and output data, (3) Determine fuzzy control rules, and (4) Determine inference mechanism. In Figure 1-7, the fuzzy interface converts the data into the fuzzy values that is in the range of [0, 1], which is used for fuzzy inference and solution of fuzzy output. The control rules, i.e. a sets of logic inference rules, represent the human empirical and heuristic knowledge in the natural language expressions of the *if-then* type, which are derived from the experiences of experts and technicians in relative

field. They map the input data to the output data in a qualitative format. Then the output data are generated using some inference method implemented in the inference mechanism. The de-fuzzy interface converts the fuzzy output data back into precise values such to generate information on system operating conditions.

The three AI analysis systems introduced above can also be combined with each other horizontally and vertically and used for FDI, such as fuzzy expert system [34], fuzzy neural network [35]-[40], etc.

1.3.2 Qualitative Modeling Based Methods

In this approach, the analytical redundancy is introduced using the qualitative model instead of mathematical representations such as state space equations and differential equations etc. The qualitative model with AI method has excellent memory ability and is useful, especially for nonlinear systems or systems with high uncertainties, where the precise mathematical model is difficulty to obtain. Without caring about the accurate inner structures of the plant, AI architecture map the input and output data to construct a redundant relationship, and memorize such mapping relationship by adjusting the suitable coefficients of its structure.

To build an AI model, it includes three steps: (1) definition of the AI method and architecture (ANN or FIS), (2) AI learning (for ANN) or tuning (for FIS), and (3) generation of residuals.

- **ANN Based Modeling**

Employing inputs and measurements of the process, a suitable ANN can be trained to adapt the input and output behavior of the process [41]-[46]. After training, a "neural model" can be adopted in place with the mathematical or heuristic model in the residual generation.

The following structure shows the ANN model learning structure.

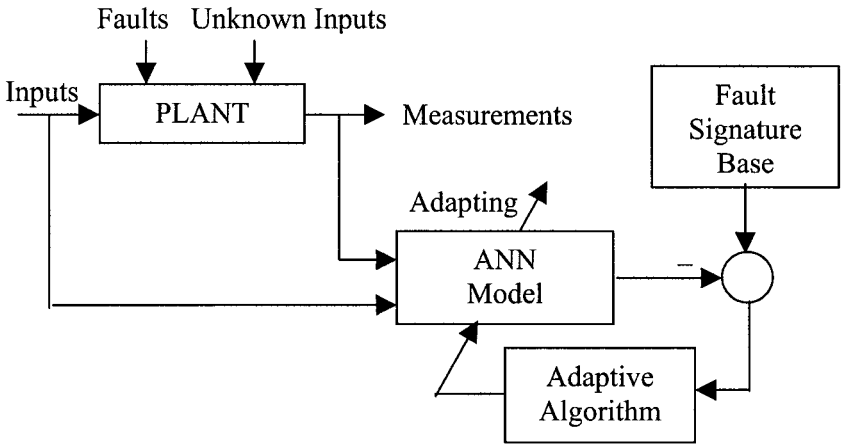


Figure 1-8. Scheme for ANN Model Learning

Driven by the learning algorithm, the ANN is trained with the inputs and measurements of the plant, and adjusting the coefficients (weights among neurons, thresholds etc.) in the ANN structure according to the known fault signature. Finally, a neural model is built. When a particular input pattern is presented to ANN, a desired output sequence (residual sets) is generated, which is used for fault diagnosis. The following structure shows the FDI with ANN model.

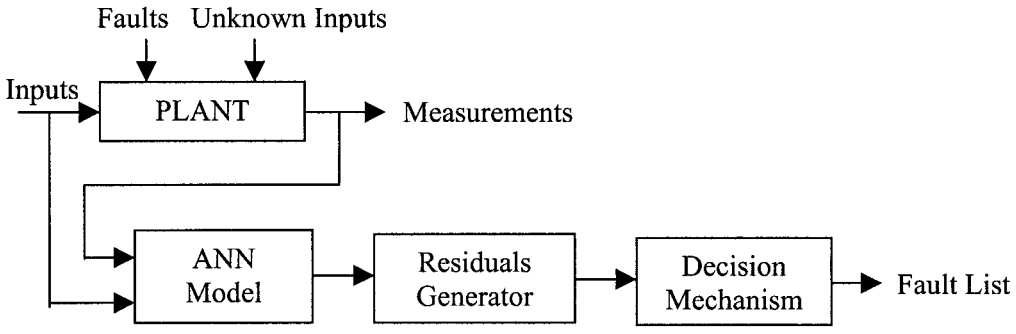


Figure 1-9. Scheme for FDI with ANN Model

When operating on line, the ANN model will estimate the outputs of the plant at the normal condition, therefore the residual signal can be generated by comparing the actual outputs and estimated outputs. Similarly as the other methods, the residuals are used for generating fault information by decision mechanism.

- **FIS Based Modeling**

The procedure using FIS based modeling [47]-[49] is very similar with the one of ANN based modeling. The tuning process of fuzzy inference system is corresponding to the training process of ANN and both assume the same function. The difference lies in that the ANN is learning to adjust the coefficients in ANN, but the FIS is tuning to adjust the membership functions and fuzzy control rules. The following figure shows the procedure of FIS tuning.

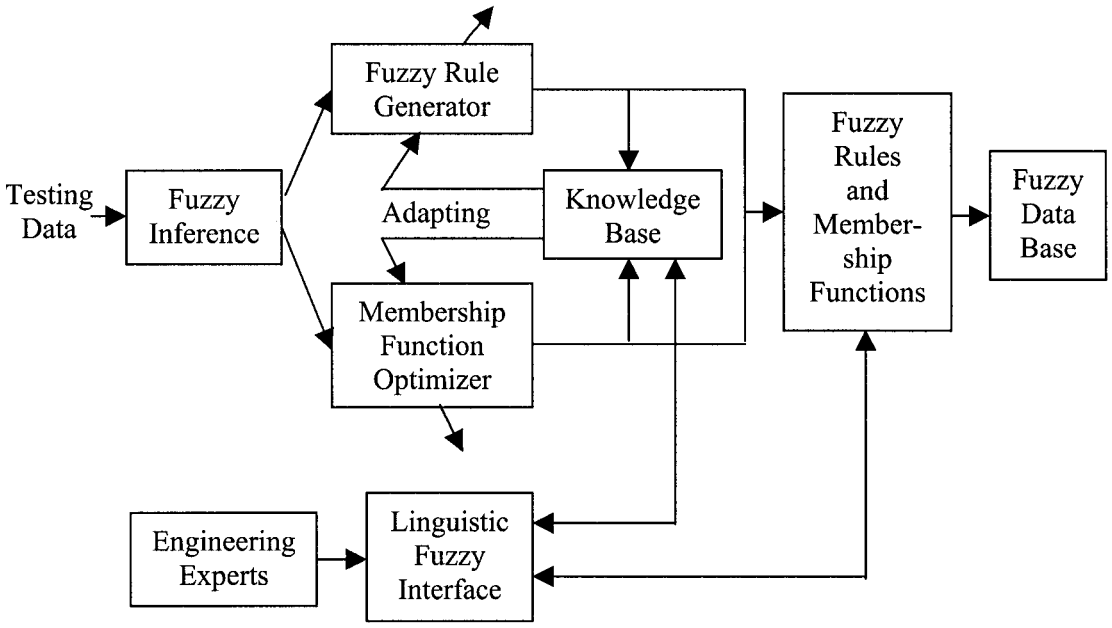


Figure 1-10. Scheme for FIS Model Tuning

The online FDI processes are the same for both ANN and FIS methods, shown in Figure 1-9.

The AI based modeling methods can achieve good performance in detecting and identifying the faults, when prior knowledge of the faults is available.

1.3.3 Signal Processing Based Methods

In many cases, faults can cause sudden changes in the measured signals resulting in characteristic changes in both time and frequency domains. The signal processing

techniques, such as Fast Fourier Transform (FFT), Short Time Fourier Transform (STFT), Wavelet Transform (WT), Wiener Filter, and Evolutionary Spectrum analysis, can capture the time instant and certain properties of the changes. Ideally, fault can be detected and isolated according to the concept that the different fault would result in changes of the different characteristics in time domain and/or frequency domain [50]-[58]. The following figure shows the conceptual scheme of the FDI system with SP method.

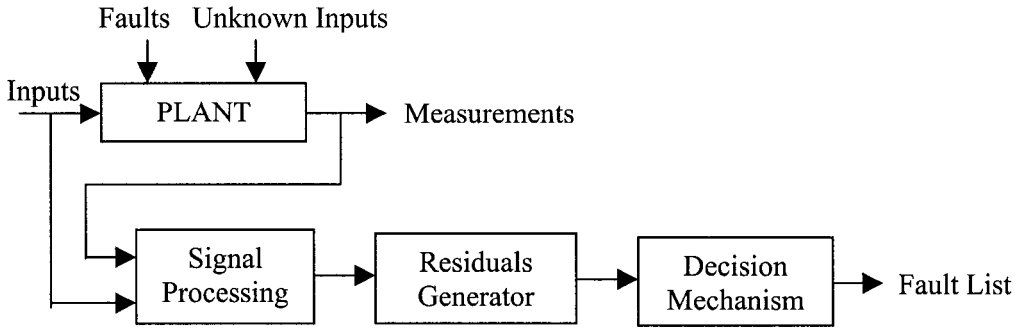


Figure 1-11. Scheme for FDI with SP

The setting up of the SP method is composed of two activities: definition of SP method, and generation of fault information. The different SP approaches will analyze the signals in different domains (time and/or frequency domain) and get the different results to describe the character of the signals, which will be used for generating the system information.

Solely with signal processing, even though different faults result in different changes in terms of the time domain and frequency domain responses, it is difficult to characterize and categorize such differences. It means that the fault identification is difficult. Moreover, normal system operations such as changing reference inputs naturally result in changes in the controlled signals. Such normal system changes should not be 'misunderstood' as some kind of faults.

1.3.4 AI and Signal Processing Based Hybrid Methods

These methods combine the AI based methods with signal processing, hence given the name 'hybrid'. The general procedure can be summarized in the following steps: 1) Pre-

process the measured signal to obtain a figure of merit; 2) Design a feature extraction algorithm to reduce its dimensionality while keeping most of its information intact, if the figure of merit is high dimensional; 3) Train the neural net such that it would store the feature of the all possible kinds of operating situations [59]-[62].

This method involves design procedures of both AI modeling and SP: Choice of SP method; Design of feature extraction; Choice of the AI method and architecture (ANN or FIS); AI tuning (for FIS) or learning (for ANN); Generation of decision information. The following figure is conceptual scheme FDI system with AI&SP method.

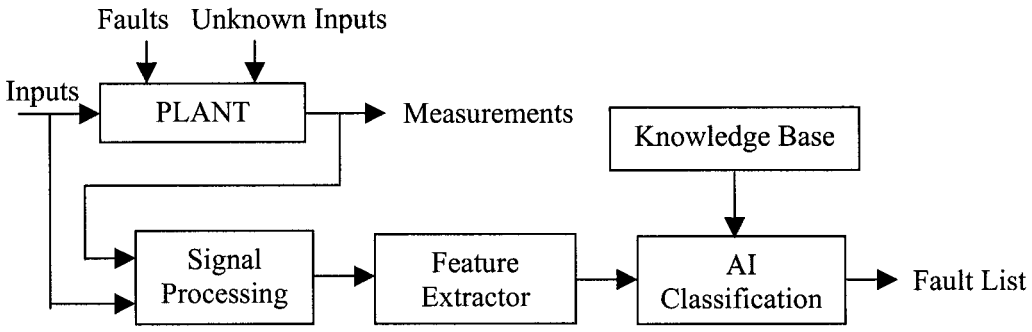


Figure 1-12. Scheme for FDI with AI&SP

The block for signal processing can reduce complex raw data (measurements and inputs) to feature vectors that can be used to classify the signals.

To further improve the accuracy of the detection and identification, a suitable feature extractor with capability to distinguish the characteristics of the signal is employed for the task of providing the patterns to the AI for training and decision making. Feature extractor intends to extract the most important characteristics from the processed data, such as spike, slopes, levels, relevant frequencies, etc., to a format suitable for the fuzzy logic or neural network to analyze. After feature extraction process, the feature or signature of the signal can provide discrimination between pattern classes.

The AI used here also includes ANN and FIS. However, a difference is that the function of the ANN here is not its capability of mapping, but its capability of classification. So the ANN architecture is different from the one in the ANN direct method. The FIS processing procedure is similar with the FIS direct method. However,

the difference is that its input data are features or some symptoms, but not preprocessed data in direct method.

When running on line, FDI system processes digital signals, generates the feature, and diagnoses the feature to make a decision.

1.4 Contribution of this Thesis

The following design objectives are desired for an effective FDI scheme with good performance.

(1) High detection and identification accuracy. Faults can be detected and isolated. Unknown operating conditions can be detected with low rate of missing detection;

(2) Transient states can be detected dynamically. Transient states resulted from faults can be identified from certain noise environment, and the ones caused by normal regulation can be isolated;

(3) Short detection time delay. That means detection and isolation in real-time;

(4) Low rate of false alarm, missing detection and incorrect identification;

(5) Robustness to noise, disturbance, and uncertainties in systems;

(6) Universal system. The system is not application specified. It can be applied to various industrial processes with little adjustments.

The knowledge-based methods reviewed above can achieve some desired goals for the specific applications, however, there are some disadvantages and open problems of these existing methods, which are listed as follows:

(1) Knowledge-based methods are relatively new trend for FDI design, and they lack general framework. Many design methods are application specified, they are only efficient for certain applications, and are hard to generalize.

(2) Fault detection is considered, but fault isolation and identification is not even mentioned in some literatures.

(3) Sensitivity and robustness to noise and disturbance is not properly discussed in many existing literature.

(4) The case studies of the proposed FDI systems in some papers are implemented in open loop instead of closed-loop, which is less practical since most control systems used on site are closed-loop.

In this thesis, an effort to design a universal FDI scheme with generalized framework is made. This approach relies on both signal processing techniques and qualitative methods by combining signal processing, statistic analysis, and neural network to attain the six design goals mentioned above.

In order to detect faults that reflect themselves as fault-induced sharp and abnormal changes/singularities at certain time instants in the measured signals, wavelet analysis, a powerful time-frequency analysis tool, is applied to capture such changes. Combined with wavelet analysis, the dynamic change can be detected in real-time and it can also help reveal the hidden fault-induced change buried in the signal by its 'zooming-in' function.

Some statistical analysis methods are used to extract features and to provide more distinct signal for the purpose of fault detection and identification.

A so-called regional self-organizing feature map (RSOM) neural network is developed together with wavelet analysis and statistical analysis to detect and identify the fault on-line in real time. By introducing the concept of hierarchy training, coarse and delicacy training, and regional recognition, the RSOM neural net proposed in this paper has achieved higher clustering and matching-up precision compared to the conventional SOM network. Also, it reduces sensitivity to the minor differences appeared in features of the same event due to the disturbance or noise of the operating environment. Furthermore, because of the fast recognition process of the SOM neural network, the speed of FDI can be guaranteed for real-time applications.

Case studies based on a nonlinear tank system and a DC motor system will be performed to demonstrate the effectiveness of this FDI scheme. Furthermore, the comparison is carried out between the proposed one and one of the model-based methods, EKF based. Finally, the design is applied to a real process training system, and achieved satisfactory performance. The proposed design can be easily modified for many practical situations, such as systems with different time delay, model known or unknown, linear or

nonlinear system, etc., which also means the design is easy to be generalized to different applications.

1.5 Scope of the Thesis

Chapter 2 provides the fundamental concepts of wavelet analysis, conventional SOM neural network, and RSOM neural network. Chapter 3 is the core of this thesis and introduces the structure of FDI scheme proposed, the development of algorithms used in FDI, and advantages of the algorithms. The proposed FDI scheme is applied to two systems, one nonlinear and one linear, in Chapter 4. The comparison with a FDI method based on extended Kalman filter, is simulated in Chapter 5. In Chapter 6, the design is applied to a real process training system, and the results are discussed. Finally, the conclusion and future work are included in Chapter 7.

Chapter II

Fundamentals of Wavelet Analysis and SOM Neural Network

The proposed FDI scheme contains two important modules: wavelet analysis module to pre-filter the data for further detection and identification, and RSOM neural network module for constructing the qualitative model. Therefore, in this chapter, the basic knowledge related to these two modules is provided.

2.1 Introduction to Wavelet Analysis

Unlike the conventional Fourier-type transforms, the Wavelet Transform (WT) is capable of providing time and frequency information simultaneously, hence giving a time-frequency representation of the signal. More specifically, wavelets are mathematical functions that decompose data into different frequency components, and then study each component with a resolution matched to its scale. They have advantages over traditional Fourier transformation methods in analyzing physical situations where the signal contains time-varying characteristics, such as discontinuities, abrupt changes, and sharp spikes [63]-[64].

2.1.1 Continuous Wavelet Transform

A wavelet transform can measure the time-frequency variations of spectral components as in the short time Fourier transform (STFT),

$$STFT(t,f) = \int_{-\infty}^{\infty} [f(t)w^*(\tau - t)]e^{-j2\pi f\tau} d\tau \tag{2-1}$$

But it has a different time-frequency resolution. Generally speaking, a wavelet transform is defined as follows,

$$Wf(time,scale) = \int_{-\infty}^{\infty} f(t)\Psi(time,scale,t)dt \tag{2-2}$$

The signal $f(t)$ is multiplied to a window function of time and scale. The parameter '*time*' is used in the same sense as used in the STFT and can also be considered as '*translation*', which is related to the location of the window, when shifted through the signal. It, obviously, corresponds to time information of the transform. The parameter '*scale*' here is similar to the one used in maps, where the high scale

corresponds to a coarse global view of the region, and the low scale corresponds to a detailed local view. In signal analysis, the scale is related to the frequency, where low frequencies (high scales) correspond to the global information of a signal, while high frequencies (low scales) correspond to a detailed information of a hidden pattern in the signal that usually lasts a relatively short time.

More specifically, the wavelet transform can be written as the following equation,

$$Wf(u,s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} f(t)\Psi^*(\frac{t-u}{s})dt \tag{2-3}$$

Where, $\Psi_{u,s}$ is called the wavelet function. With different time u and scale s, the wavelet function is the shifted and scaled version of function $\Psi(\bullet)$, a function with zero average, i.e.

$$\int_{-\infty}^{\infty} \Psi(t)dt = 0 \tag{2-4}$$

$\Psi(\bullet)$ is also called the mother wavelet. As seen in the Eq.(2-3), the wavelet transform is based on a set of functions derived from a mother wavelet by adjusting the time-shifting and time-scale parameters.

The name mother wavelet comes from two important factors: the term '*wavelet*' means a function with a small oscillatory waveform. The smallness refers to the condition that this window function is of finite length (compactly supported). The term '*mother*' implies that it is the prototype for generating other window functions with different regions of support that are used in the transformation process. The wavelet coefficient $Wf(u, s)$ depends on the values $f(t)$ in the time-frequency domain where the energy of mother wavelet $\Psi_{u,s}$ is concentrated. Time varying harmonics are detected from the position and scale of high amplitude wavelet coefficients.

There is no direct frequency parameter for WT as for STFT and FFT. However, the scale parameter is associated with the pseudo-frequency by:

$$F_a = \frac{\Delta \bullet F_c}{s} \tag{2-5}$$

where, s is a scale, Δ is the sampling period, F_c is the center frequency of a wavelet (note that different wavelet has different center frequency), F_a is the pseudo-frequency corresponding to the scale.

Wavelet analysis allows the use of long time intervals when one needs more precise low-frequency information, and short time intervals when one needs high-frequency information. Therefore, it has good time resolution but poor frequency resolution at high frequencies (low scales), and good frequency resolution but poor time resolution at low frequencies (high scales). This can be seen from the Figure 2-1.

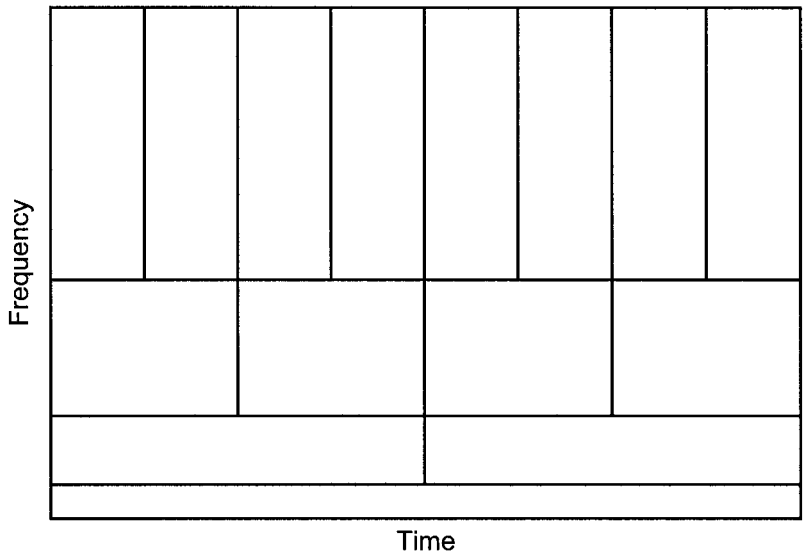


Figure 2-1. Time-Frequency Resolution of Wavelet Transform

The width or height means the resolution of time or frequency. The shorter or narrower the box, the better resolution it has. Since the product of time and frequency, i.e., the area of box in the above figure, is constant, the time resolution and frequency resolution are inversely proportional. That is, when frequency resolution gets better, the time resolution becomes worse, and vice versa. In Figure 2-1, at higher frequencies (lower scales), the width of the boxes decreases, that is, the time resolution gets better; and when the height of the boxes increases, the frequency resolution gets worse. Similarly, lower frequencies (higher scales) have better frequency resolution.

Therefore, the following relationship can be found:

Low scale → *Compressed* wavelet → *Rapidly* changing *details* → *High* frequency domain

High scale → *Stretched* wavelet → *Slowly* changing *approximations* → *Low* frequency domain

2.1.2 Discrete Wavelet Transform

In the case of discrete wavelet transform, the filters with different cutoff frequencies are used to analyze the signal at different levels. The signal passes through a series of high pass filters and a series of low pass filters so that the high frequency components and low frequency ones are analyzed separately, and detail and approximation coefficients are generated respectively. The following figure illustrates a procedure of discrete wavelet transform, with only 2 levels are shown in figure.

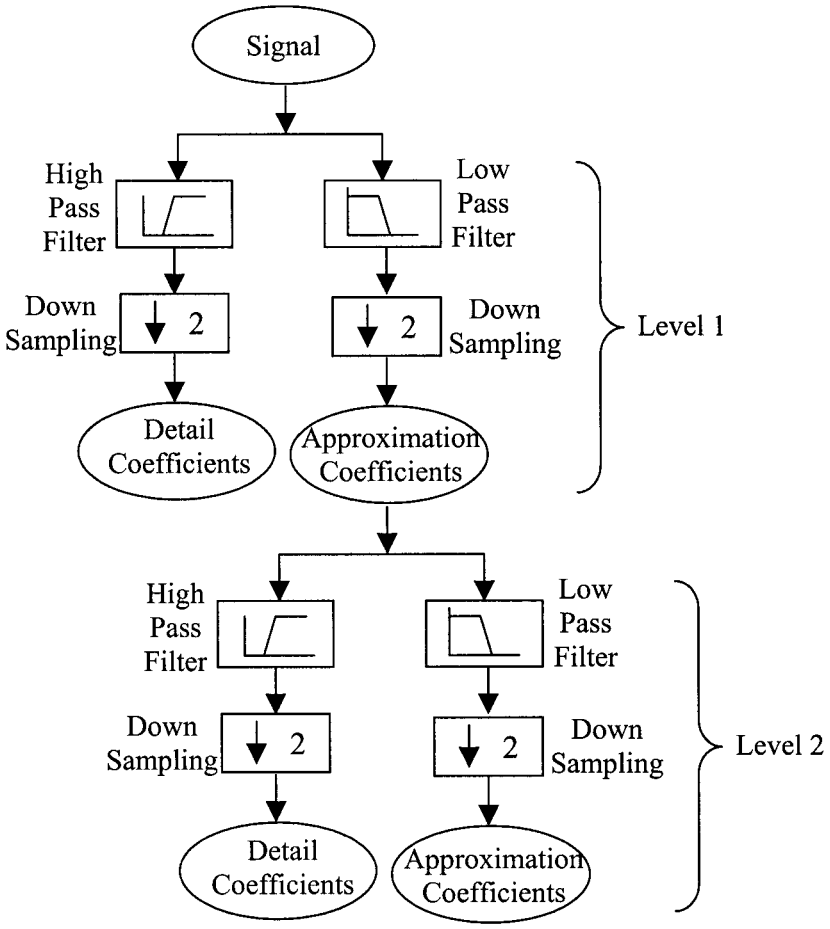


Figure 2-2. Procedure of Discrete Wavelet Transform (DWT)

First of all, the signal passes through two complementary filters, one high pass filter and one low pass filter, such that half amount of information in frequency domain is removed at each output of the filter. This filtering process doubles frequency resolution since the frequency band of the signal now spans only half of the previous frequency

band, effectively reducing the uncertainty in the frequency by 50%, but leaves time resolution unchanged.

After filtering, half of the samples, which are redundant, can be eliminated according to the Nyquist's rule. That is, the filtered signals can be downsampled by 2, simply by discarding every other sample data. This downsampling process leaves frequency resolution unchanged, and halves time resolution since only half the number of samples now characterizes the entire signal.

Then, the approximation coefficients (low frequency domain signal) repeat the above processes successively to decompose the signal into different frequency bands.

At this point, we know that the signal decomposed by discrete wavelet transform by passing through a series of low/high pass filters can be divided into two parts, details and approximations, in different scales. Details and approximations in different scales are associated with the information in high and low frequency domains respectively. By analyzing the information in various frequency domains, the characters or features of the analyzed signal can be extracted easily.

2.1.3 Detection of Singularity by Wavelet Transform

The fault in the system will result in certain changes in the measured signals, such as discontinuities, abrupt changes, and sharp spikes etc. These changes have some special characteristics with singularities and irregular structures, which often carry essential information in signal. Wavelet transformation can detect such special characteristics.

It is known that the wavelet transform *Wf(u,s)*,

$$Wf(u,s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} f(t) \Psi^*(\frac{t-u}{s}) dt \qquad (2\text{-}6)$$

has a function of zooming into the signal structures with a scale $s \to 0$ because the mother wavelet can be compressed enough to zoom in the signal and then analyze the signal signature. This property is very useful in detecting signal singularities/discontinuities or sharp transients, more specifically, by locating the converging local *modulus maxima* of the wavelet transform at fine scales [65]-[66].

The term, modulus maxima, describes any point (u_0, s_0) at which $|Wf(u,s_0)|$ is locally maximum, i.e., it satisfies that,

$$\frac{\partial}{\partial u} Wf(u, s_0)|_{u=u_0} = 0 \tag{2-7}$$

This is also true for discrete wavelet transform when the smallest scale is limited by the sampling rate of the data. For example, for fast calculation algorithm with s chosen at a dyadic scale, $\alpha f_s^{-1} \le s \le 2^j \; j \in Z$. The α is large enough to avoid sampling coarsely the wavelets at the finest scale. The largest scale 2^j should be smaller than the distance between two consecutive singularities to avoid having other singularities influence the value of Wf. It has been proven that wavelet maxima carry important information on signal variations, and it can even reconstruct signals with small approximation errors.

2.2 Introduction to Self-Organizing Feature Map Neural Network

With the capability of adapting and learning, an artificial neural network becomes one of the best candidates for eliminating the complex modeling process and therefore has received substantial attention in the intelligent control community.

The self-organizing feature map neural network, which is a type of Kohonen neural network is an excellent tool in exploratory phase of data mining. It has the capability of learning and adapting without supervision, and has the good performance in cluster and vector quantification. It projects high dimensional input space on prototypes of a low dimensional regular grid that can be effectively utilized to visualize and explore properties of the data [67]-[68].

The Kohonen neural network is capable of pattern classification and has the good performance in cluster and vector quantification. Given a set of input patterns, it can distribute the input sets into a two-dimensional Euclidean space and build a pattern table at the output of the self-organizing neural net, in which every pattern element is relative to an input model and neighbor neurons are developed into detectors of specific categories of patterns.

The unsupervised learning, which is always used for describing data rather than predicting them, seeks to find pattern or regularity in the input data, that is, make the network "self-organizing". Therefore, it can be used to construct categories without guidance in SOM neural network.

The following sections will introduce the basic SOM algorithm and improved RSOM algorithm used in this work.

2.2.1 Basic Self-Organizing Feature Map Neural Network

The following figure is the structure of the basic SOM neural network.

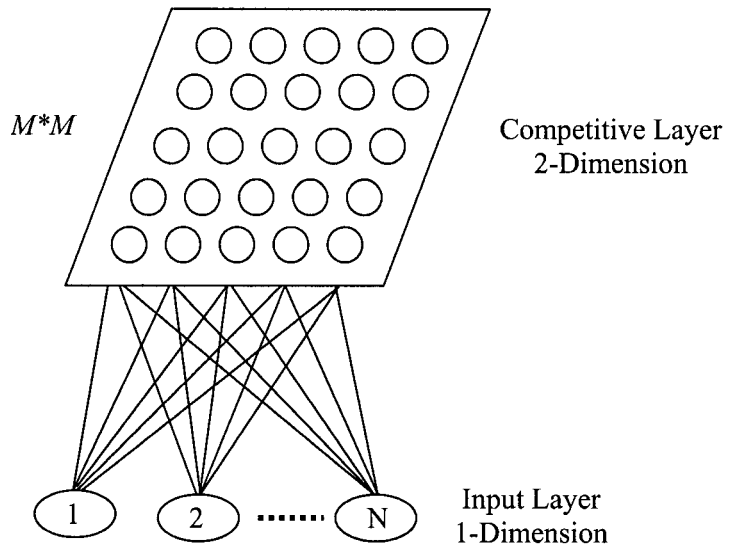


Figure 2-3. Topologic Structure of SOM

The SOM neural network includes three parts, input layer, weight coefficient matrix and competitive layer. The input layer is a one-dimensional array and composed of *N* neurons, which are used for inputting a set of data into network. The competitive layer is a two-dimensional Euclidean space and composed of *M*M* neurons, which are used for saving the categories of input data patterns. The weight coefficients matrix links the input layer and competitive layer. The learnt elements in matrix store the information that maps the non-linear relationship of input data models in input layer and known pattern neurons in competitive layer.

The SOM algorithm includes the unsupervised learning process for a series of known data models and recognizing process for an unknown data model. Using the unsupervised learning approach, a series of known data models is input into neural network and learnt to adjust the elements in weight coefficients matrix successively.

After numerous learning steps, the following characters are found and memorized in the neural network.

- Pattern table associated with the input models, which is stored in neurons in the competitive layer;
- Non-linear relationship between input neurons and competitive neurons, which is stored in weight coefficients matrix.

For an unknown input model, the recognizing process tries to find the matching pattern in the pattern table by using the known weight coefficients matrix and constructed pattern table. The matching between any input data and known pattern in table means that the resemble likelihood between input data and known pattern is strong, that is, the unknown input data has the strong similarity with the known pattern. Therefore, the nature of unknown data model can be clustered and classified [69]-[71].

The following procedure is the learning procedure of the basic SOM.

Step 1. Initialize the elements in the weight coefficients matrix W, as a set of random constant in range [0,1], the total number of learning T, initial learning ratio $\eta(0)$ $(0<\eta(0)<1)$, initial size of neighbor zone $N_z(0)$.

Here, defining the neighbor zone as the zone around the winning neuron in competitive layer, shown in gray area in Figure 2-4. The shape of zone can be square, circle and rectangle etc. The winning neuron will be defined in Step 4.

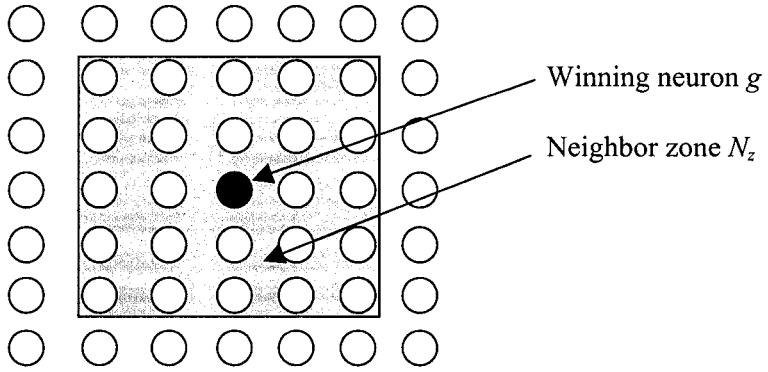


Figure 2-4. Definition of Neighbor Zone

Step 2. Input the p input vectors, $A_k = (a_{1k}, a_{2k}, ..., a_{Nk})$. Where N is the number of input neurons, k is the kth input vector and $k=1,2,...,p$.

Step 3. Compute the Euclidean distance d_j between the weight vector W_j = $(w_{j1}, w_{j2}, ..., w_{jN})$ and input vector $A_k = (a_{1k}, a_{2k}, ..., a_{Nk})$,

$$d_j = \| A_k - W_j \|_2 = [\sum_{i=1}^{N} (a_{ik} - w_{ji})^2]^{1/2} \qquad j = 1,2,...,M*M \tag{2-8}$$

Where, $M*M$ is the size of the competitive layer.

Step 4. Find the minimum distance, and the corresponding neuron g associated with this minimum distance in competitive layer is the winning neuron.

$$d_g = MIN[d_j] \qquad j = 1,2,...,M*M \tag{2-9}$$

Step 5. Adjust the weights in neighbor zone N_z, i.e, all weights linked to the neurons in the neighbor zone could be adjusted by the following equation,

$$\begin{aligned} & w_{ji}(t+1) = w_{ji}(t) + \eta(t) \bullet [a_{ik} - w_{ji}(t)] \\ & j \in N_z(t) \qquad i = 1,2,...,N \qquad 0 < \eta(t) < 1 \end{aligned} \tag{2-10}$$

Where, $\eta(t)$ is the learning ratio at t^{th} learning step.

Step 6. Repeat step 3~5 until all p input data are trained.

Step 7. Modify the learning ratio $\eta(t)$, size of neighbor zone $N_z(t)$.

$$\eta(t) = \eta(0) \bullet (1 - t/T) \tag{2-11}$$

Where, t is the current learning step number, T is the total learning number.

Assuming the coordinate of one neuron z is (x_z, y_z), then the size of neighbor zone is changed to the new square zone, which vertexes are $(x_z + N_z(t), y_z + N_z(t))$, $(x_z + N_z(t), y_z - N_z(t))$, $(x_z - N_z(t), y_z + N_z(t))$, $(x_z - N_z(t), y_z - N_z(t))$, where,

$$N_z(t) = INT[N_z(0) \bullet (1 - t/T)] \tag{2-12}$$

Where, $N_z(0)$ is initial value, $INT[x]$ is function to round the x to the nearest integers towards zero.

Step 8. $t=t+1$, repeat from step 2 until $t=T$.

After learning, p input models are distributed in the $M*M$ competitive layer. The recognizing algorithm is shown as the follows:

Step 1. Input an unknown model $A = (a_1, a_2, ..., a_N)$.

Step 2. Compute the distance and find the winning neuron g.

$$d_g = MIN(d_j) = MIN([\sum_{i=1}^{N} (a_i - w_{ji})^2]^{1/2}) \qquad j = 1,2,...,M*M \tag{2-13}$$

Step 3. Find if the winning neuron has the corresponding neuron in the known pattern

table. If yes, the unknown input is classified, otherwise, the input is still unknown.

2.2.2 Regional Self-Organizing Feature Map Neural Network

The concept of RSOM neural network is an original idea proposed by the author. Based on the basic SOM algorithm, the regional algorithm introduces some new improvements.

- Hierarchy learning (two regions learning),
- Coarse and delicacy learning,
- Neighbor region recognition.

This enhancement will achieve the higher clustering and matching-up precision, decrease the sensitivity of the small difference in features of the same event and improve the accuracy of the recognition.

First of all, two neighbor regions around the winning neuron are introduced. The *strong neighbor region* is defined as the zone adjacent to the winning neuron, and the *weak neighbor region* as the zone just around the strong neighbor region. They are depicted in Figure 2-5, where, the circles are neurons in competitive layer, the black one is winning neuron, the neurons in gray shape are the strong neighbor region, and the neurons in the loop between dash and solid line are the weak neighbor region.

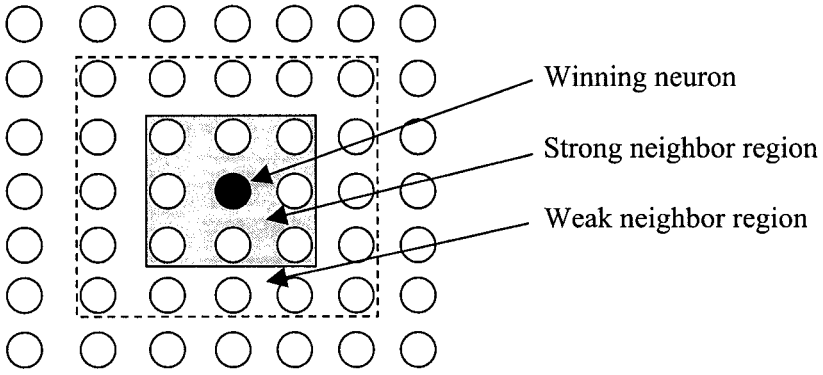


Figure 2-5. Definition of Strong and Weak Neighbor Regions

The following procedure is the learning algorithm.

Step 1. Initialize the elements in the weights coefficients matrix W, as a set of random constant in range [0,1], the total number of learning T, initial learning ratio $\eta(0)$ ($0<\eta(0)<1$), initial size of strong neighbor region $N_{zs}(0)$, and the size of weak neighbor region $N_{zw}(0)$.

Step 2. Input the p input vectors, $A_k = (a_{1k}, a_{2k}, ..., a_{Nk})$. Where N is the number of input neurons, k represents the kth input model with $k=1,2,...,p$.

Step 3. Compute the Euclidean distance d_j between the weight vector $W_j = (w_{j1}, w_{j2}, ..., w_{jN})$ and input model $A_k = (a_{1k}, a_{2k}, ..., a_{Nk})$,

$$d_j = \| A_k - W_j \|_2 = [\sum_{i=1}^{N} (a_{ik} - w_{ji})^2]^{1/2} \qquad j = 1,2,...,M*M \tag{2-14}$$

Where, $M*M$ is the size of the competitive layer.

Step 4. Find the minimum distance, and the corresponding neuron g associated with this minimum distance in competitive layer is the winning neuron.

$$d_g = MIN[d_j] \qquad j = 1,2,...,M*M \tag{2-15}$$

Step 5. Adjust the weights in the strong neighbor region N_{zs} by the following equation,

$$\begin{aligned} &w_{ji}(t+1) = w_{ji}(t) + \eta(t) \bullet [a_{ik} - w_{ji}(t)] \\ &j \in N_{zs}(t) \qquad i = 1,2,...,N \qquad 0 < \eta(t) < 1 \end{aligned} \tag{2-16}$$

Where, $\eta(t)$ is the training ratio in t-th step.

Step 6. Adjust the weights in the weak neighbor region N_{zw} by the following equation,

$$\begin{aligned} &w_{ji}(t+1) = w_{ji}(t) + u(t) \bullet \eta(t) \bullet [a_{ik} - w_{ji}(t)] \\ &j \in N_{zw}(t) \qquad i = 1,2,...,N \qquad 0 < \eta(t) < 1 \qquad 0 < u(t) < 1 \end{aligned} \tag{2-17}$$

Where, $u(t)$ is the member factor of the weak neighbor region at t-th step. With this factor, the links of the input model to the neurons in the weak neighbor region are weakened.

Step 7. Repeat step 3~6 until all p input model data are trained.

Step 8. Modify the training ratio $\eta(t)$, size of the strong neighbor region $N_{zs}(t)$, size of the strong neighbor region $N_{zw}(t)$, and the member factor $u(t)$.

$$\begin{aligned} \eta(t) &= \eta(0) \bullet e^{-\alpha \cdot t/T} \\ N_{zs}(t) &= INT[N_{zs}(0) \bullet e^{-\alpha \cdot t/T}] \\ u(t) &= REM[N_{zs}(0) \bullet e^{-\alpha \cdot t/T}] \end{aligned} \tag{2-18}$$

Here, t is the current training step number, T is the total learning number, $N_{zs}(0)$ is initial dimension of the strong neighbor region, $INT[x]$ is the function to round x to the nearest integers towards zero, $REM[x]$ is function to get the reminder of x. α is the adjusting degree factor and $\alpha>0$. Greater α results in the shorter coarse training time and the longer delicacy learning time. According to the definition,

the weak neighbor zone is adjusted with the adjustment of the strong neighbor zone.

Step 9. $t=t+1$, repeat from step 2 until $t=T$, the total training number.

After learning, p input models are distributed in the $M*M$ competitive layer evenly. The recognizing procedure is shown as the follows:

Step 1. Input an unknown model $A = (a_1,a_2,...,a_N)$.

Step 2. Compute the distance and finding the winning neuron g.

$$d_g = MIN(d_j) = MIN([\sum_{i=1}^{N}(a_i - w_{ji})^2]^{1/2}) \qquad j = 1,2,...,M*M \qquad (2\text{-}19)$$

Step 3. Find if the winning neuron has the corresponding matched neuron in the known pattern table. If yes, the unknown input is classified.

Step 4. If no, find if there is the corresponding neuron(s) in neighbor zone in the known pattern table. If so, selecting the neuron with the minimal distance as classifying result. If no, the input is unknown.

The differences between the basic and improved algorithms are marked with bold and italic font in context. The advantages of the improved algorithm over the basic one will be discussed in Chapter 3.

Chapter III

Design of Fault Detection and Identification Scheme

The core of the proposed FDI scheme is introduced in this chapter in details. The structure of FDI scheme, the development of algorithms used in FDI, and the advantages of the algorithms are discussed.

3.1 Problem Formulation

The FDI scheme in this thesis integrates signal processing techniques, statistical analysis and neural network. It belongs to the categories of data-driven and knowledge-based FDI design techniques. The research objective of this proposed FDI system is nonlinear system and/or model not precisely known system. The particular goals of this system are:

(1) Detect and isolate faults at high accuracy. Pre-simulated faults can be detected and isolated. Unknown operating condition can be detected with low rate of missing detection;

(2) Transient states can be detected dynamically. Transient states resulted from fault can be identified from certain noise environment, and the ones from normal regulation can be isolated;

(3) Detection and isolation with short detection time delay (real time);

(4) Maintain the low rate of false alarm, missing detection and incorrect identification;

(5) Robustness to noise, disturbance, and uncertainties in systems;

(6) Universal system. The system is not application specified. It can be applied to various industrial processes with little adjustments.

So three problems arise and need to be emphasized in order to realize the above design goals.

(1) How can fault signals be captured as fast as possible in different noise-level environment?

(2) How can significant features/signatures be constructed that contain as much information about faults as possible?

(3) How can the reliable decision be concluded that reflects the system operating conditions and gives the type and origin of the faults?

The discussion about FDI system in this chapter will focus on these problems.

The architecture and algorithms are composed of fault detection and identification system. The architecture focuses on the structure of FDI system and the algorithms concentrates on describing the approaches used in this FDI system and their advantages.

3.2 System Architecture Description

For complex systems, where the model is non-linear or not precisely known, model-based methods would not give the ideal result and maybe results in warning the incorrect fault or missing the fatal fault. Compared with model-based methods, knowledge-based FDI can be used in broader applications in such systems.

The knowledge-based FDI system acquires the necessary information from the measured signals and makes the decision according to the changes of the signal by capturing the abrupt deviations of the signals in real time when the fault occurs. These deviations include spikes, steps and so on in low and high frequency domains. So far, the faults are detected. Then, the detected changes in signals can be clustered to different features and the corresponding pattern recognition method is used to classify features. Therefore, the faults are identified.

In this work, a knowledge-based FDI algorithm integrated with signal processing using wavelet analysis, feature extraction using statistical analysis, and pattern recognition using neural network is proposed. The system architecture is clearly depicted in the following figure.

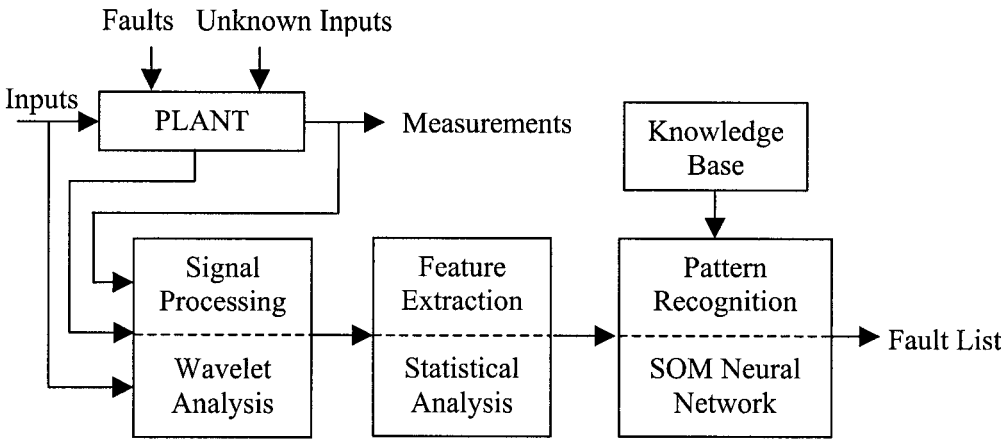


Figure 3-1. Architecture of the Proposed FDI System

In order to detect faults that reflect themselves as fault-induced frequency and time changes at certain time instances in the measured signals, wavelet analysis is applied to seize such changes. A few of statistical methods are used to extract fault features from the decomposed signals. A Regional Self-Organizing feature Map (RSOM) neural network is utilized to detect and isolate the fault on-line. According to the characteristics of RSOM neural network, network will learn some known data models off-line and recognize the unknown data model on-line. So the FDI system can be divided into two modules, off-line training module and on-line recognizing module shown in Figure 3-2.

The off-line training module and the on-line recognizing module are not independent or exclusive each other. They have a tight relationship according to the consistency of the FDI algorithm. It is shown in the following figure, that both of the off-line and on-line modules have the same components for pre-processing the raw measurements: the wavelet analysis and feature extraction.

In the off-line training module, the faults are simulated in the system. Wavelet transform analyzes the measured signals and generates the decomposed signals. The features of these signals are extracted by using statistical approach. Here, the different operating status will have different features. The improved RSOM network clusters the normal feature, disturbance features, and the predicted fault features of the system and generates a feature pattern table by neural net learning procedure. This table is stored in the knowledge base and will be used for on-line recognizing process.

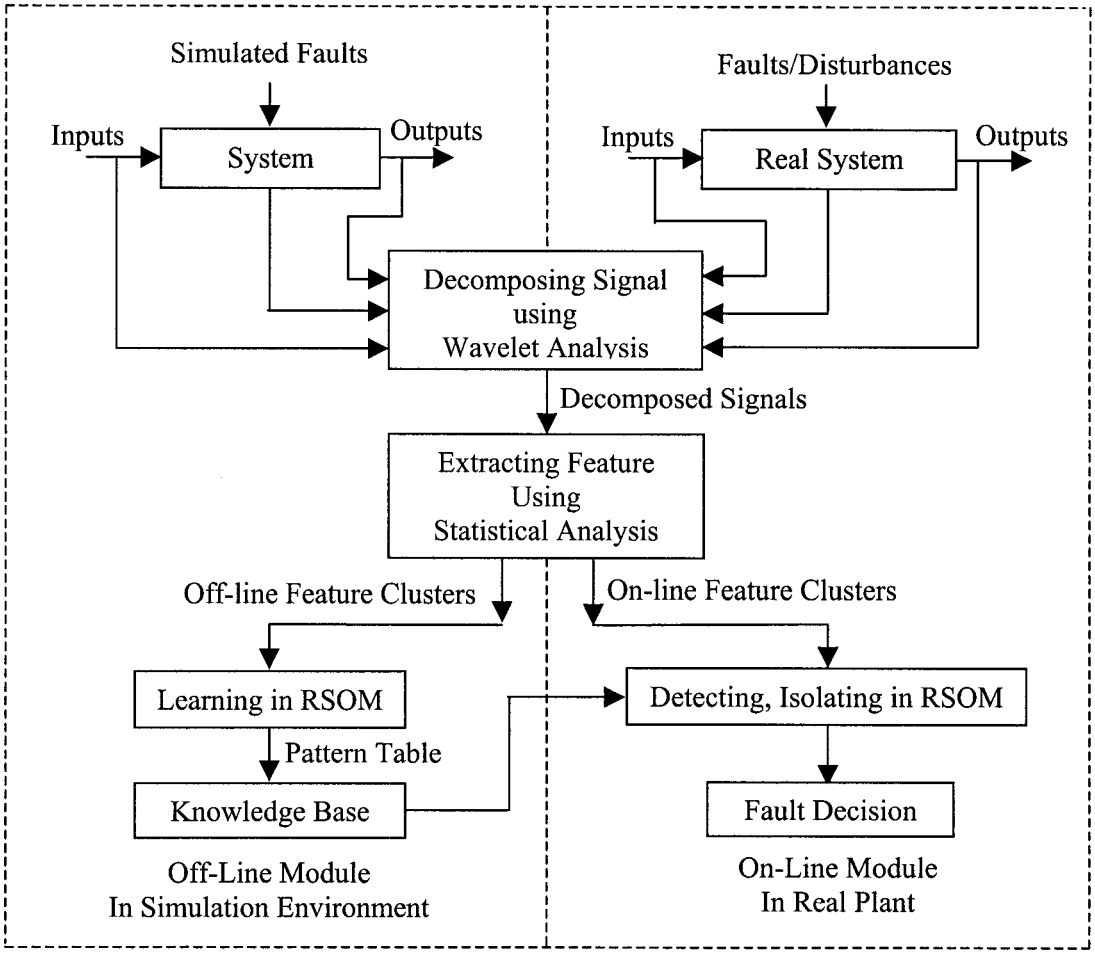


Figure 3-2. FDI System Structure

In the on-line recognizing module, the signals are monitored in real time. Using the same transforming and extracting algorithms, the features are generated for recognizing in RSOM neural network. However, in the off-line training module, the feature clusters are built based on the a *priori* knowledge about the system and all possible operating conditions including the faults, while the on-line feature clusters are built based on the real system measurements.

The neural network matches the real-time signal feature with the known patterns in the pattern table, which are memorized in the knowledge base. The operating status can be classified and a fault is detected and isolated as the feature is matched with the pattern table in real time.

In summary, the differences between the off-line training and on-line recognizing modules are illustrated in the following table.

Table 3-1. Differences in On-line and Off-line Processing

	Off-line Processing	**On-line Processing**
Operating Environment	Simulating environment and off-line state	Real environment and on-line state
System Operating Status	Simulated and known	Unknown
Input Signals Requirement	Simulated signals including all possible faulty and normal status	Real signals measured from the real system
Processed Data Section	Data section at predicted faulty instance	All data section varying with time. When and where fault occurs is unknown
RSOM Neural Network	Learning all known features and clustering them	Recognizing the unknown feature
Goal	Feature pattern table	System operating situation

3.3 Development of Algorithms

Three techniques are implemented in FDI system, in name, wavelet analysis, statistical analysis, and RSOM neural network. The detailed algorithms and their advantage will be discussed in this section. An example, which is depicted along with the algorithm discussion, is based on the three-tank system that is shown in Chapter 4.

3.3.1 Wavelet Analysis

Faults in the system usually manifest themselves as certain changes in the response of measured signals, e.g. changes in time response and in frequency response. These changes would result in transient behavior of system variables, and transient analysis becomes critical for fast and accurate fault detection and isolation. In order to achieve on-line and real-time fault detection, new analysis tools that are sensitive to the transient

phenomena caused by the faults are needed. For this reason, the wavelet analysis method is chosen to detect the possible fault induced transients in the first stage.

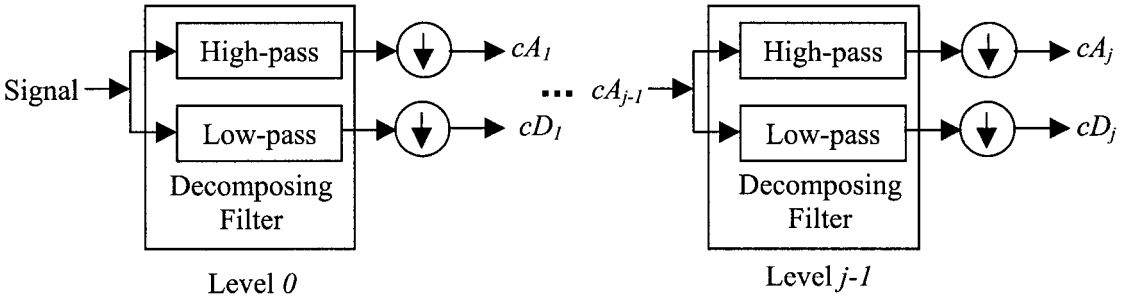
The reason for using wavelet analysis is that it is capable of detecting the change/transition in the signal almost in real-time. Such a novel property can be used for fault detection where the change is due to the occurrence of a fault in the system. For example, by incorporating the wavelet maxima in the feature construction, it can give good indication to abnormal system behavior in the existence of component faults.

The discrete wavelet transform is implemented in this work, which is introduced in section 2.1. It contains two steps:

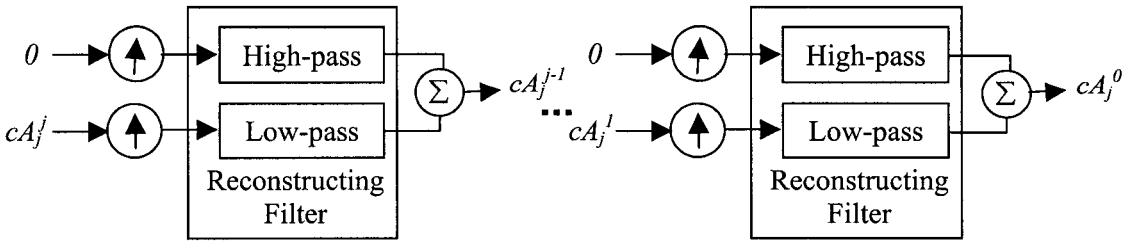
- Decomposition: Signal is decomposed into detail and approximation coefficient vectors in successive level by wavelet analysis;
- Reconstruction: Detail and approximation coefficient vectors in certain level, which can reflect the nature of signal changing, are reconstructed.

The reconstructed coefficients will be processed in the next procedure, feature extraction.

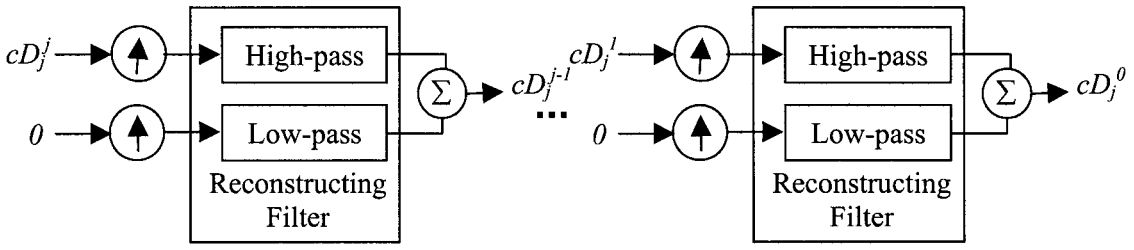
The following figure shows the discrete wavelet analysis in FDI system.



(a) Detail and Approximation Coefficients Decomposition Step



(b) Approximation Coefficients Reconstruction Step



(c) Detail Coefficients Reconstruction Step

Figure 3-3. Wavelet Analysis Processing

Firstly, shown in figure (a), the signal passes a high-pass filter and a low-pass filter, and the filtered signal is then down-sampled. The detail and approximation coefficients can be obtained respectively, which are corresponding to the high frequency domain and low frequency domain. The low frequency signal can be processed again in the same way until the decomposed coefficients can manifest the low frequency and high frequency nature of raw signal.

Because of down-sampling, the coefficients would only contain one of 2^j sampling data of the raw signal at the decomposed level j, which is needed to reconstruct the original number of the signal shown in figure (b) and (c).

The approximation coefficient vector cA_j^j in level j is up-sampled by a factor of 2, passing through low-pass filter, and finally generates the reconstructed approximation coefficient vector cA_j^{j-1}. When reconstructed j times, the decomposed coefficient vector cA_j^j can transform into the reconstructed coefficient vector cA_j^0, which has the same length as the original signal. Similarly, the detail coefficient vector can be reconstructed using the similar process.

In the decomposing and reconstructing process, it is necessary to construct 4 filters, namely, high-pass decomposing filter, low-pass decomposing filter, high-pass reconstructing filter, and low-pass reconstructing filter. The construction of filters is related to the wavelet scaling function.

Assuming the wavelet scaling function (discrete) is a set of data, *ws* with the length of *2N*. We have four impulse response functions of filters.

Low-pass reconstructing filter:

$$h_{L\_R}[n] = \frac{ws}{norm(ws)} \qquad n = 1,2,....,2N \tag{3-1}$$

where, *norm(ws)* is the largest singular value of vector *ws*.

High-pass reconstructing filter:

$$h_{H\_R}[n] = (-1)^n h_{L\_R}[2N - n + 1] \qquad n = 1,2,....,2N \tag{3-2}$$

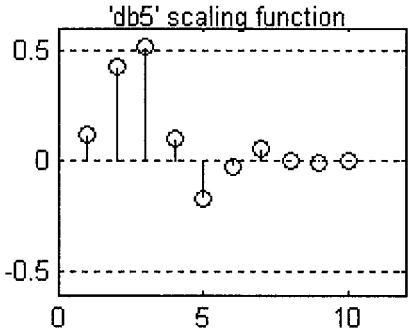
Low-pass decomposing filter:

$$h_{L\_D}[n] = h_{L\_R}[2N - n + 1] \qquad n = 1,2,....,2N \tag{3-3}$$

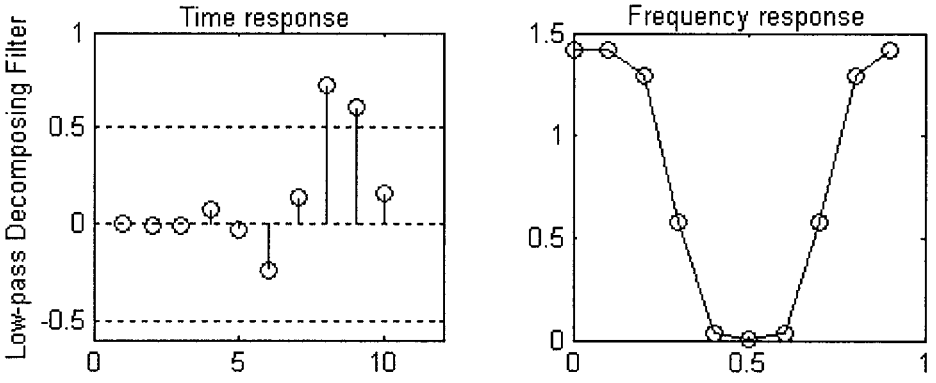
High-pass decomposing filter:

$$h_{H\_D}[n] = h_{H\_R}[2N - n + 1] \qquad n = 1,2,....,2N \tag{3-4}$$

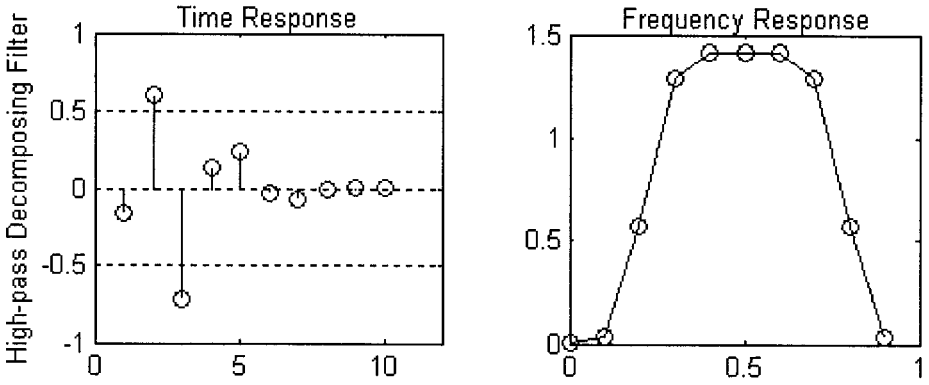
For example, we consider the wavelet function, '*db5*'. The following figures show the time domain and frequency domain of four filters.



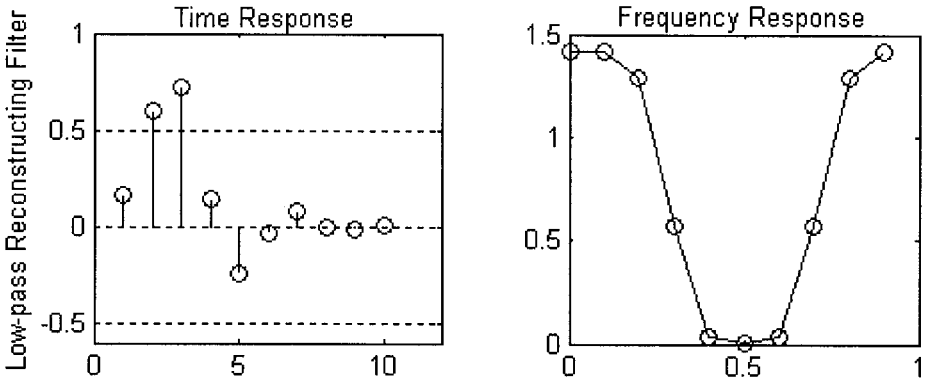
(a) 'db5' scaling function



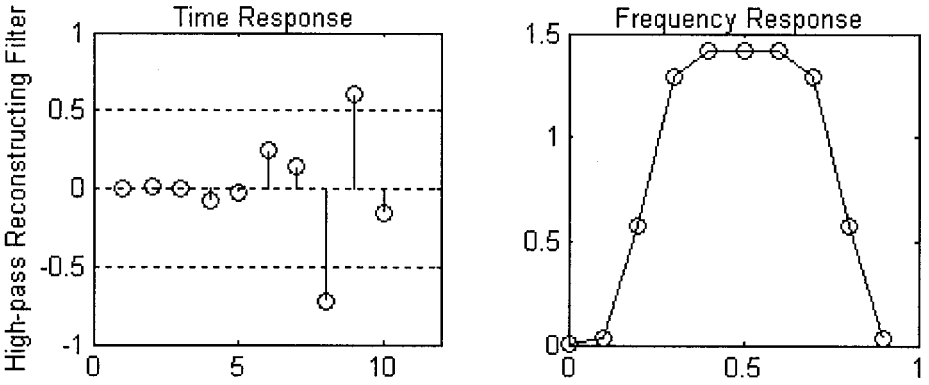
(b) Time and Frequency Response of Low-pass Decomposing Filter



(c) Time and Frequency Response of Low-pass Decomposing Filter



(d) Time and Frequency Response of Low-pass Reconstructing Filter



(e) Time and Frequency Response of High-pass Reconstructing Filter

Figure 3-4. Filters for Wavelet Function 'db5'

Here, considering an example to explain the discrete wavelet analysis in FDI system.

For the tank system shown in Chapter 4, Figure 3-5 shows the system response in the case of occurrence of leakage fault in tank 2. In the normal system operating condition, the level for tank 2 is maintained at 0.3m. A fault occurs at the 10th second.

Due to the adjustment of PID controller, the pumps in tank1 and tank 3 will automatically increase the water flow to keep the level of tank 2 at a desired height when tank 2 leaks. As a result, the heights of tank1&3 will increase. However, the height of tank 2 will remain unchanged in spite of leaking.

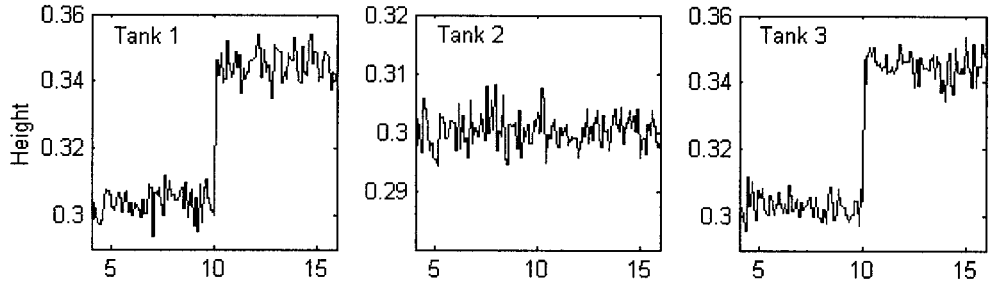
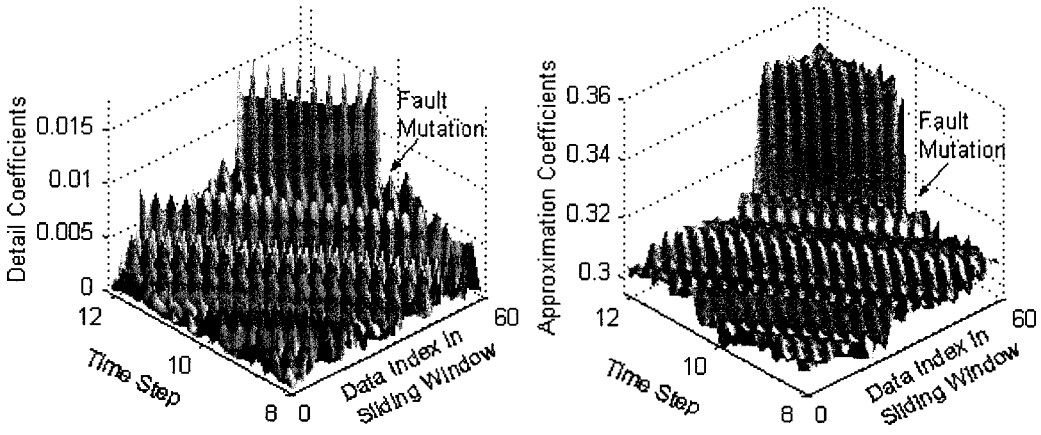
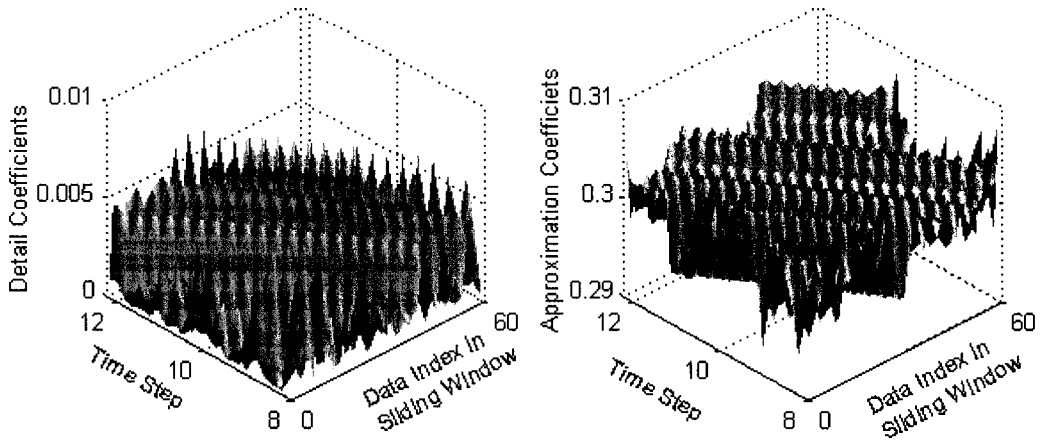


Figure 3-5. Tank Levels with Leakage Fault in Tank 2

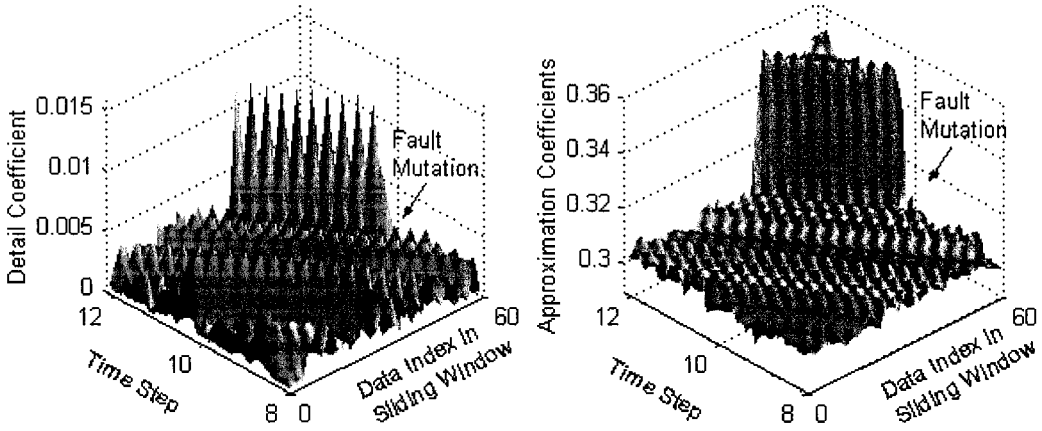
A sliding window of 60 samples is used for the wavelet analysis (the sampling rate is 10Hz). The data in the sliding window are analyzed by discrete wavelet transform using method of 'db5' in scale 1. In the sliding window, larger data index means more updated samples. Detail and approximation coefficients at the first scale in the different time instance are displayed in the following figures.



(a) Detail and Approximation Coefficients of Level of Tank 1



(b) Detail and Approximation Coefficients of Level of Tank 2



(c) Detail and Approximation Coefficients of Level of Tank 3

Figure 3-6. 3-Dimensional Graph of Coefficients

From the figures, it can be seen that, when the fault occurs, both detail coefficients and approximation coefficients of level 1&3 show abrupt changes in the latest samples. Such changes continuously shift to the left along with increased time steps and as more updated samples move into the sliding window and more obsolete samples move out of the window, thus build an evident 'wall'. However, the measurement in tank 2 does not show apparent changes. This is the effect of the closed-loop control. The influence of the fault on the level of tank 2(the system output) is greatly weakened by the compensation of the PID control. (In deed, in a closed-loop configuration, it is a well-known fact that the more robust the control, the poorer

performance the fault detection system, and vice versa.) Therefore, one needs to obtain more information on other system states, in our case, the levels in tank 1 and 3. By observing these signals, we can detect the changes and further identify the system operating status.

Also, comparing the Fast Fourier Transform in the same sliding window, we can find that there are the tiny changes in frequency domain due to leakage fault in tank 2 shown in Figure 3-7, which cannot be used to detect fault effectively.

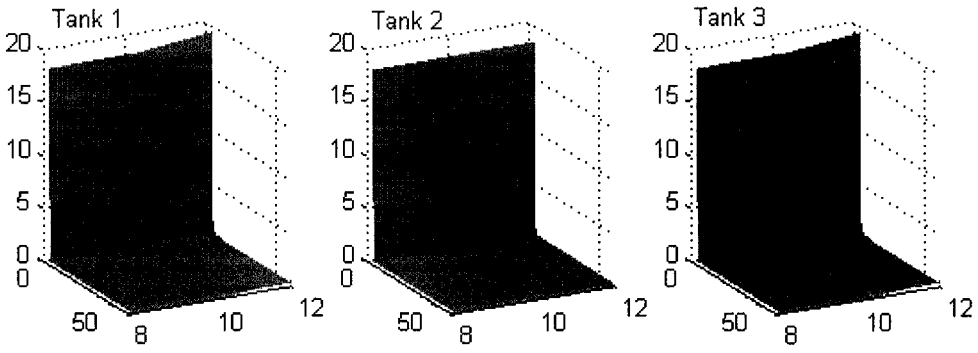


Figure 3-7. FFT Analysis Results

As seen from Figure 3-6, the changes in signal can be detected instantly when fault occurs. However, when used for fault detection and isolation, this condition only cannot provide a reliable indicator for faults. Therefore, an improved self-organizing feature mapping neural network is synthesized, to further analyze the features built from wavelet analysis, to finally detect and identify the faults. The bridge to connect the wavelet analysis and neural network is feature, which is extracted form analyzed signals by wavelet analysis and is processed by neural network. The extracted feature would be any property of the signal that is useful depicting normal or abnormal signal behavior and provide the greatest discrimination among the analyzed signals. So the validity of feature extraction is crucial to the accuracy of the fault detection and identification.

However in many cases, by checking the wavelet maxima only may not guarantee that there must exist singularities or sharp transitions in the signal, i.e. the condition is not sufficient for detecting general singularities. Furthermore, some normal system operations such as input or set-point changes result in similar wavelet maxima. Therefore,

when used for fault detection, this condition only cannot provide a reliable indication for faults. In this work, the feature extraction method is developed.

3.3.2 Feature Extraction Module

With the decomposition process of the raw system measurements using the discrete wavelet transform described in the previous section, the signal is separated into two parts: approximations and details, corresponding to the low and high frequency components respectively. The stochastic characteristics of the signal in low and high frequency would not change abruptly in normal operating environment, but change abruptly when the faults occur. So the statistical analysis method is used to find the deviation of the stochastic characteristics [72].

The following feature types can be useful to detect the bias of the signal, which can be raw signal, decomposed signal, or reconstructed signal.

- *Signal Segment Length L.* Signal behavior within a segment is dependent on the length of the segment. For example, the low-frequency signal has good resolution in longer segment, but high-frequency signal in shorter segment.
- *Signal Range* Δs. The difference of the maximum value and the minimum value of the signal in a segment.

$$\Delta s = s_{\max} - s_{\min} \tag{3-5}$$

- *Signal Max or Min Value* $s_{\max}$ *or* $s_{\min}$. The maximum or minimum values of the signal within the same segment are useful detecting the out of bounds condition.
- *Signal Energy E.* The energy of the signal within a segment.

$$E = \frac{\sum_i s_i^2}{L} \tag{3-6}$$

Where, L is signal segment length.

The above feature types place more emphasis on the static property of the signal. Considering that data slide in and out segment when time varying and amplitude changing in real-time, the dynamic property of data stream should be considered when

constructing feature type. So we need to divide a data segment into several sections that contain the data in different time intervals. We have the following feature types.

- *The Maximum or Minimum Absolute Value Changing Ratio* $\nabla s_{\max}$ *or* $\nabla s_{\min}$.

  The changing ratio of maximum or minimum values of the signal in different sections, e.g., past data section and current data section, within the same segment.

$$\nabla s_{\max} = \frac{|S|_{\max\_now} - |S|_{\max\_past}}{|S|_{\max\_past}}, \nabla s_{\min} = \frac{|S|_{\min\_now} - |S|_{\min\_past}}{|S|_{\min\_past}} \tag{3-7}$$

Where, $|S|_{\max\_now}$ and $|S|_{\min\_now}$ are maximum and minimum absolute value of the data for current data, $|S|_{\max\_past}$ and $|S|_{\min\_past}$ for the data in past time.

- *The Variance Changing Ratio* ∇V. The changing ratio of the variance of the signal in the different sections within the same segment. The variance reflects the scatter or deviation or fluctuation from the mean value.

$$\nabla V = \frac{VAR_{now} - VAR_{past}}{VAR_{past}} \tag{3-8}$$

Where, *VAR* represents variance value of data in present and past data sections. The definitions of these sections are introduced in the preceding context.

- *The Energy Changing Ratio* ∇E. The changing ratio of the energy of the signal in different sections within the same segment.

$$\nabla E = \frac{E_{now} - E_{past}}{E_{past}} \tag{3-9}$$

Where, E represents average energy of data in different data segments.

From the above calculation, it is clear that different changing ratio has different sensitivity to the static and dynamic properties of the signal. By choosing proper changing ratio for all system measurements and incorporating them in a feature vector, a feature table can be constructed for all possible system operating status. One of the main concerns in choosing the different changing ratio is the real-timeliness of the detection system, it also depends on the dynamic properties of the signal. Considering the properties of the signal in FDI system, two feature methods are selected, *Absolute Max Value Changing Ratio* and *Variance Changing Ratio*. As discussed in section 2.1.3, we

know that the wavelet maxima can be located at the fine scales to detect the discontinuities in the signal. Also the wavelet maxima can be manifested as the higher magnitude in the decomposed coefficient. In feature extraction, to find the absolute max value in coefficient also is to find the wavelet maxima so that the discontinuity and singularity can be detected.

Also, for processing in real time, the sliding window technique is used to track dynamic and detect the transient state of system. Usually, in a sliding window, the latest several samples carry the most up-to-date information on any changes in the signal. These data should be taken better use of to capture the changes in time. For this purpose, we divide the data window into several segments that contain the sample data in different time intervals. Concepts of sub-windows and data zones are defined as shown in Figure 3-8.

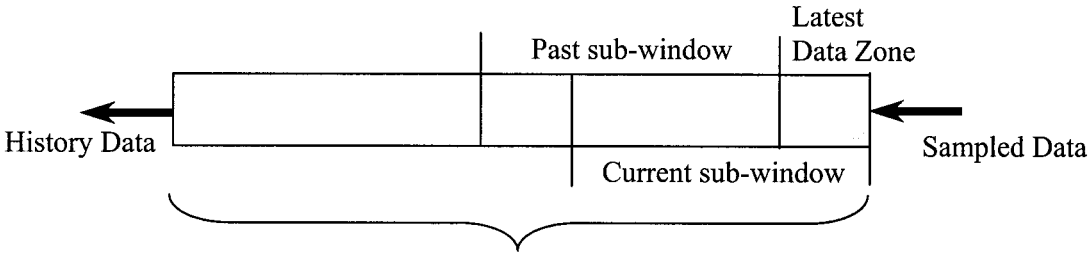


Figure 3-8. Definition of Sliding Window

The current sub-window (*CW*) contains the data sampled at recent time steps while the past one (*PW*) contains the data sampled in the past time. The data in *CW* contain the most updated operating information, including information on fault induced transients. More specifically, such important information is hidden in the latest several samples contained in the so called latest data zone (*LZ*). Clearly, the difference between *CW* and *PW* is that *CW* contains *LZ*, while *PW* does not.

By performing decomposition of the signal using the discrete wavelet transform, the signal is separated into two parts: approximations and details, corresponding to the low-frequency and high-frequency components respectively. When the signal changes abruptly, it results in the wavelet maxima. For this reason, an abrupt change is also shown in the detail and approximation parts. Specifically, the approximation part contains the changing tendency of the signal and the detail part contains the changing instance of the

signal. In the presence of system noise, the wavelet coefficients will subject to the variations due to the noise. The variance changes in this case can also be observed when faults occur. Thus, by calculating the changing ratio of the data in sub-windows defined as below, fault induced changes can be captured at the time instant of occurrence. Meanwhile, the effect of noise is reduced. The rules, r_1 and r_2, are used to represent these ratios based on the current data and past data in the sub-windows and data zone defined as above, and will be used in the case study.

$$r_1 = \frac{VAR(CWdata) - VAR(PWdata)}{VAR(PWdata)} \tag{3-10}$$

$$r_2 = \frac{MAX(|\, LZdata\,|) - MAX(|\, PWdata\,|)}{MAX(|\, PWdata\,|)} \tag{3-11}$$

It needs to point out that a feature vector contains the changing ratio values for more than one signal. In order to make the feature as distinctive as possible, it is desired to incorporate as more information from the system as possible, and combine more features from other signals, and build a more complete feature vector. Here, two methods are introduced to extend the feature vector and increase the information capacity.

- *Space Expansion*. By including more measured signals from the system to make the feature more distinctive. The space expansion has limitations due to the number of sensors in the system.
- *Time Extension*. By incorporating successive samples in the feature vector. This extension is especially useful when signals have the irregular time-shifting property varying with the different faults.

The following simple equation calculates the number of elements in the feature vector in the space/time extension methods.

$$F = Nf \bullet Ns \bullet Nt \tag{3-12}$$

Where, F is the number of elements in a feature vector, Nf is the number of feature for one signal, Ns is the number of the measured signals related to the space extension, Nt is the number of the samples related to the time extension.

The feature vector is obtained by incorporating these changing ratios based on the reconstructed signals. The typical ways to incorporate the features into a vector are listed as follows.

- *Copy*. Copy features of each signal into vector without any change.
- *Arithmetic Calculation*. Move features of each signal into vector after the simple arithmetic calculating, e.g., addition, subtraction, multiplication and division.
- *Ratio*. Move features of each signal into vector after computing the ratio of two relative features.

Usually, the value of the feature vector is greater than one, so we need to normalize the vector to train in the neural network. The following *sigma* function is used for the normalization.

$$Sigma(x) = \frac{1 - e^{-0.5x}}{1 + e^{-0.5x}} \tag{3-13}$$

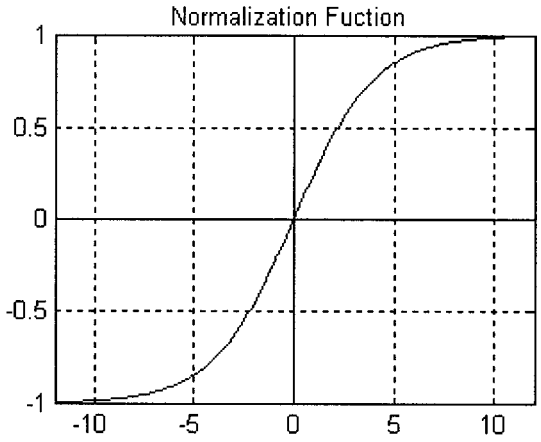


Figure 3-9. Normalization Function

For example, the feature vector of the leakage fault in tank 2 will be built by using *space expansion* method and *copy* incorporating method, which is shown in the following figure. The each element in the feature vector will be discussed in next chapter.

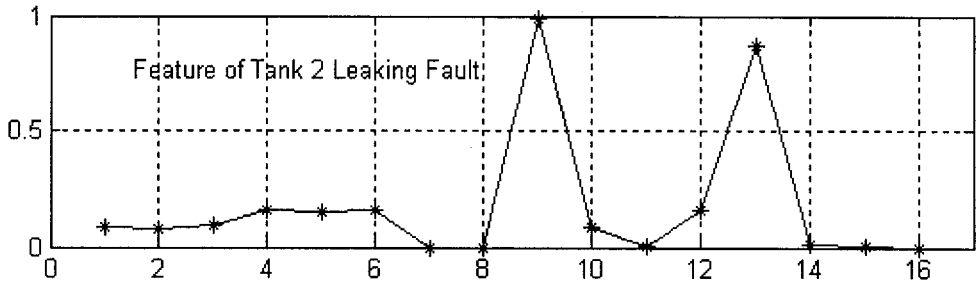


Figure 3-10. Feature Vector of Tank 2 Leakage Fault

3.3.3 RSOM Neural Network

The inputs of the RSOM neural network are on-line feature clusters in on-line and off-line feature clusters in off-line. The unexpected yet commonly existed external noises/disturbances and system variations may result in differences in the off-line feature cluster and the on-line feature cluster, even for the same system operating status. A nice pattern recognizing mechanism should be robust enough to these differences and correctly identify the operating status, to avoid false alarm, missing detection or incorrect decision.

As discussed in Chapter 2, the improved RSOM neural network in the FDI system introduces the concepts of hierarchy learning, coarse and delicacy learning, and neighbor region recognition into the conventional SOM neural network algorithm, therefore it would achieve the higher clustering and matching-up precision, and reduces sensitivity to the minor differences appeared in features of the same event due to the disturbance or noise of the operating environment. Furthermore, because of the fast recognizing process of the SOM neural network, the speed of FDI can be guaranteed for real-time applications.

In order to not only detect the faults but also locate and identify them, all possible fault scenarios should be considered a *priori*. In the process of neural network learning, the features are obtained for all fault scenarios.

The advantages of the improved SOM neural network can be embodied in three aspects:

- Continuous and gradually-decaying weights adjustment instead of stepwise and discontinuous adjustment in basic algorithm;
- Full-evenly-distributed pattern table instead of partly-distributed pattern table;
- Recognized plane instead of recognized point.

Recalling procedure of learning process discussed in Section 2.2.2. In basic SOM algorithm, there is only one adjusting zone and the adjusting zone will decrease with the increment of learning number. So the process of weight adjustment is discontinuous, stepwise and suddenly-stopping. The adjusting intensity is same for all neurons in the adjusting zone. Figure 3-11 shows this adjustment procedure.

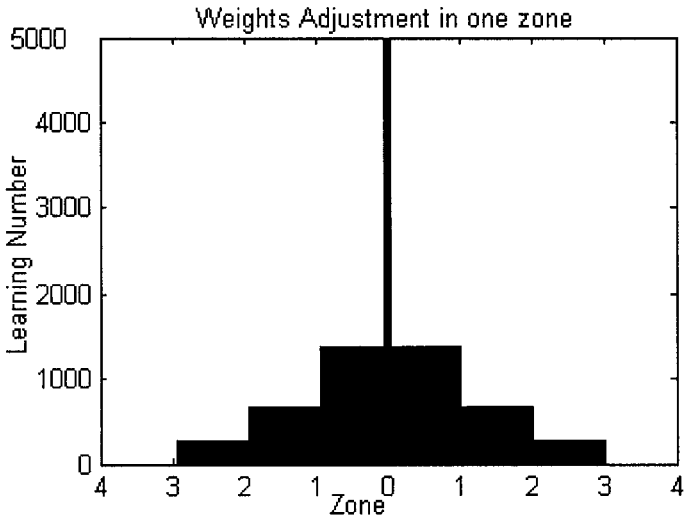


Figure 3-11 Weights Adjustment in Basic Algorithm

It is observed in the above figure, the weights adjustment in one zone is step and suddenly-stopping when the zone decreases with the learning number increasing. The gray-scale means the strong degree of adjustment, darker color, stronger adjustment. It also can be observed that all color is black, which means all adjusting intensity is the same.

The concept of hierarchy learning is introduced in improved algorithm. The weights adjustment is processed in two regions, strong and weak regions, with the different adjusting intensity. Figure 3-12 shows the adjustment procedure of improved algorithm.

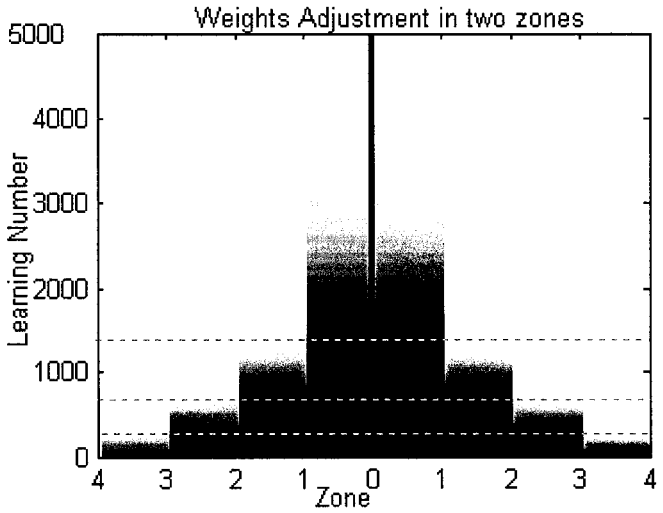
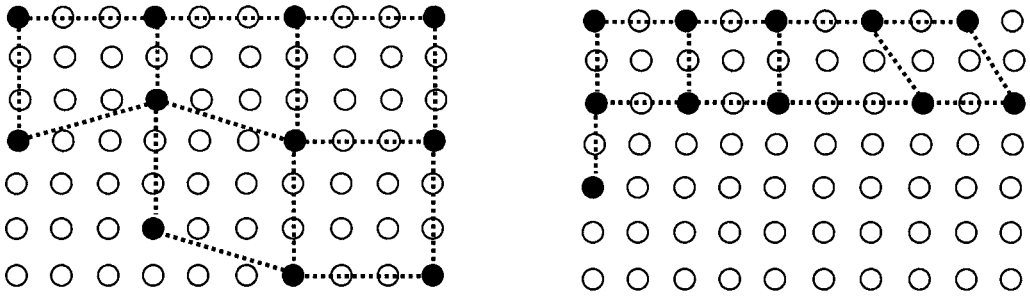


Figure 3-12 Weights Adjustment in Improved Algorithm

Seen the above figure, the weights adjustment in two regions is continuous and gradually-decaying when the regions decrease with the learning number increasing. The gray-scale changes form dark black to white gradually, which means the adjusting intensity for different regions is different at the different learning number.

After learning, a pattern table will be generated in the competitive layer, which looks like a mesh. The improved algorithm will distribute the input models into a full-evenly-distributed mesh which extends in full output layer as regular and big as possible shown in Figure 3-13(a). The figure (b) shows the learning results of the basic algorithm.



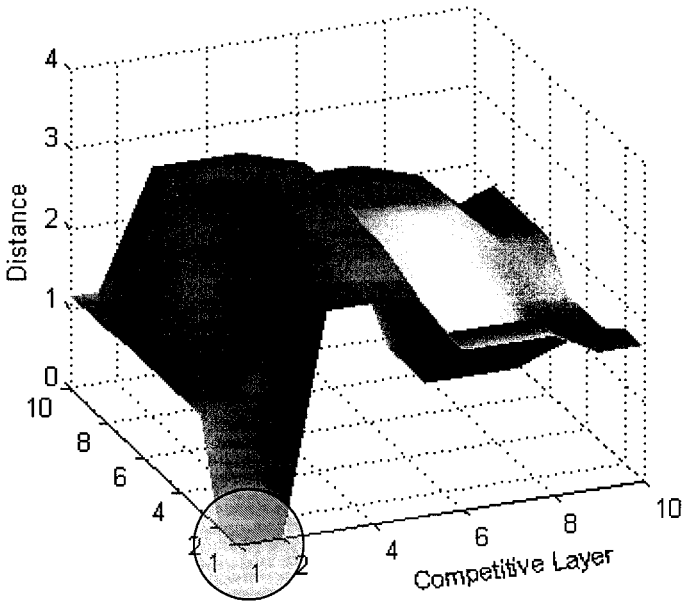
(a) Improved Algorithm (b) Basic Algorithm

Figure 3-13. Pattern Tables Generated by Two Algorithms

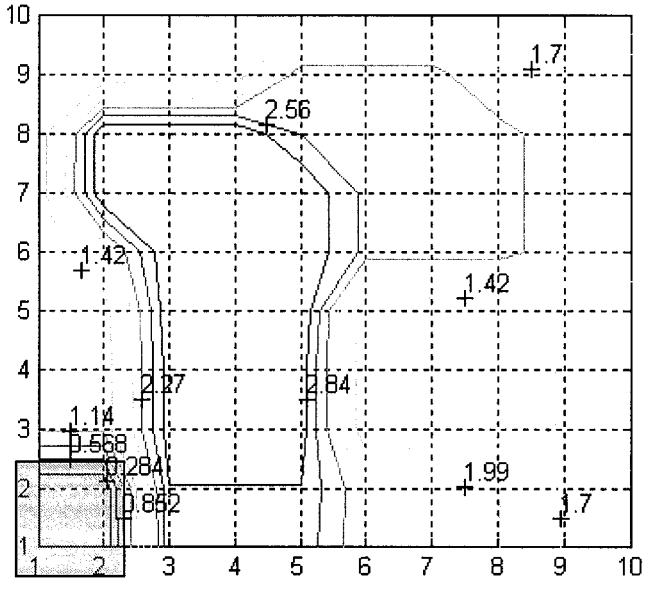
The circles in the above figures are the neurons in the competitive layer and the black ones are the learnt winning neurons, which represent the known patterns in the pattern table. Obviously, the mesh in the improved one is full-evenly-distributed, extended in full layer regularly and decentralized, however, the mesh in the basic one is partly-evenly-distributed but extended in a small region and centralized. This property is very useful for classifying the unknown input data with tolerance of minor difference with the known feature for the corresponding operating status, which is resulted from the small variations, or the disturbances in the operating environment.

When recognizing, the improved algorithm will use neighbor region (plane) recognition method, which means that the process of the pattern recognition will check the neighbor zone of the winning neuron, which contains several neurons, to decide the matched pattern. But in the basic one, the single point recognition is used to only check the winning neuron to find the matched pattern.

For example, the following 3-dimensional figures show the distance of the recognition when inputting the feature data of drifting and bias fault of sensor 1 into the trained neural network. The winning neuron for the improved algorithm is (1,1).



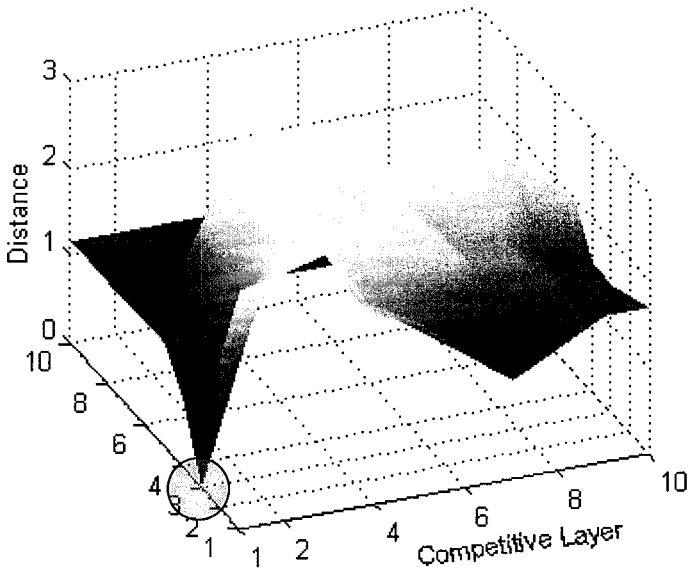
(a) Distance when recognizing



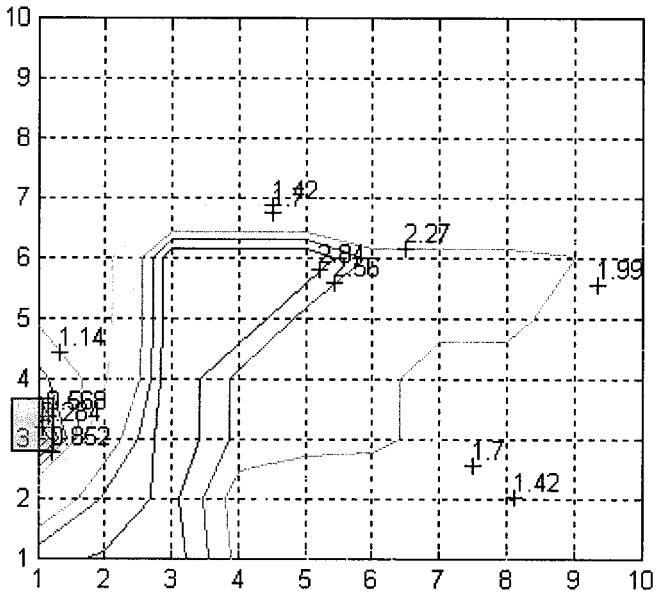
(b) Contour of Figure (a)

Figure 3-14. Recognizing Distance for Improved RSOM

Also, same figures for basic algorithm are shown in Figure 3-15. The winning neuron is (1,3).



(a) Distance when recognizing



(b) Contour of Figure (a)

Figure 3-15. Recognizing Distance for Basic SOM

By the observation from the above figures, we can know that a recognized plane is generated in the distance matrix by the improved algorithm shown in Figure 3-14(a). The distances fluctuate with small amplitude in the neighbor region around the winning neuron, and the contour is sparse in this area. But for neurons far from the winning neuron, the distance will increase abruptly. That is, the improved algorithm is robust to the small difference of the input features, and these analogous features will be regarded as the same pattern. Therefore, the effect of the difference in feature, which is generated from the changing environment or noise, will be eliminated in the recognizing process.

However, a recognized point is generated by the basic approach given in Figure 3-15(a). The distances change with gradual average grade without any abrupt varying so the contour is varying gradually. Therefore, a group of features resulted from the same event with small difference cannot be clustered as the same pattern because the distance is very sensitive to the differences of features.

By these improvements, the RSOM algorithm is more robust to the small differences of the input features for the same event. That is, it will still recognize the input feature for the same operating condition even if it has small uncertainties due to the system uncertainties and disturbances noises, while the conventional will wrongfully identify it as a different status. By using such an algorithm, the robustness and the accuracy of the fault detection and identification system are enhanced.

Chapter IV

Case Studies

In this chapter, simulation studies are performed on a cascaded three-tank system and a lab-scale DC motor system in order to demonstrate the performance of the FDI design. The applied techniques and simulated results are discussed in details.

4.1 Nonlinear Tank System

In this case study, the system is a three-tank system with nonlinear dynamics. The system description is shown in Figure 1. The fault detection and identification system is implemented in a closed-loop configuration. For the purpose of simplicity and without the loss of generality, the tank system is configured as a BISO system (two inputs and one output). The inlet flow driven by two pumps to tank 1 and 3 acts as two inputs while the level in tank 2 is considered as the output variable. A PID controller can be implemented to maintain the liquid in tank 2 at the desired level. Ideally, even when faults occur, the closed-loop system should be capable of maintaining its present operation to a certain degree. The faults, disturbances, or manipulative adjustment occurring in the system will be detected and isolated in real time.

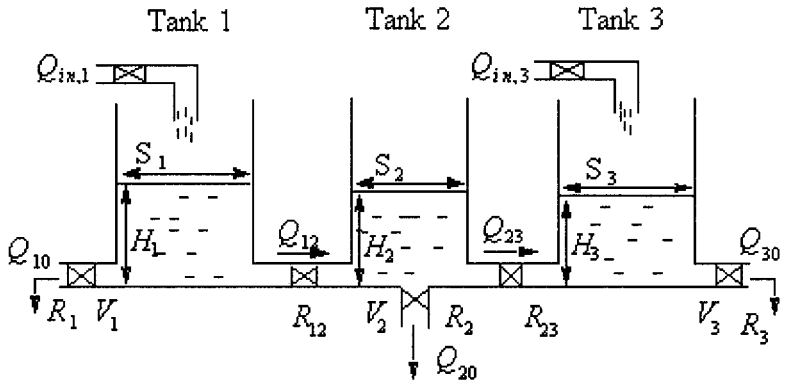


Figure 4-1. Nonlinear Three-tank System

The parameters and the variables of the system are given as follows:

S_1: Cross-sectional area of the tank 1. (8.107×10^{-3} m^2)

S_2: Cross-sectional area of the tank 2. (4.560×10^{-3} m^2)

S_3: Cross-sectional area of the tank 3. (8.107×10^{-3} m^2)

ρ: Density of the liquid (water, 1.0×10^3 kg/m^3)

H_i: Liquid level of the tank i, i=1,2,3 (m)

$Q_{in,i}$: Liquid inlet flow rates from the pump i, i=1,3 (m^3/s)

Q_{i0}: Liquid drainage rates from the tank i, i=1,2,3 (m^3/s)

Q_{ij}: Liquid flow rates between the tanks i and j, i,j=1,2,3 (m^3/s)

R_i: Drainage resistance of the valve i, i=1,2,3 (kg/m^5)

R_{ij}: Flow resistance between the tanks i and j, i,j=1,2,3 (kg/m^5)

The dynamic relationships between the height of the liquid levels and the inlet and outlet flow rates are shown as follows:

$$\begin{aligned}
\dot{H}_1(t) &= -\frac{k_1}{S_1}\sqrt{g\cdot H_1(t)} - \frac{k_{12}}{S_1}\mathrm{sgn}(H_1(t)-H_2(t))\sqrt{g\cdot|H_1(t)-H_2(t)|} + \frac{Q_{in,1}}{S_1} \\
\dot{H}_2(t) &= -\frac{k_2}{S_2}\sqrt{g\cdot H_2(t)} + \frac{k_{12}}{S_2}\mathrm{sgn}(H_1(t)-H_2(t))\sqrt{g\cdot|H_1(t)-H_2(t)|} \\
&\quad -\frac{k_{23}}{S_2}\mathrm{sgn}(H_2(t)-H_3(t))\sqrt{g\cdot|H_2(t)-H_3(t)|} \\
\dot{H}_3(t) &= -\frac{k_3}{S_3}\sqrt{g\cdot H_3(t)} + \frac{k_{23}}{S_3}\mathrm{sgn}(H_2(t)-H_3(t))\sqrt{g\cdot|H_2(t)-H_3(t)|} + \frac{Q_{in,3}}{S_3}
\end{aligned} \tag{4-1}$$

Where,

$$k_1 = \frac{\rho}{R_1} \quad k_2 = \frac{\rho}{R_2} \quad k_3 = \frac{\rho}{R_3} \quad k_{12} = \frac{\rho}{R_{12}} \quad k_{23} = \frac{\rho}{R_{23}}$$

This system is a symmetric system, that is, the inlet flow rates of the tank 1&3 are same. A PID controller is used to maintain the liquid level in tank 2 at a desired level. The structure of the closed-loop system is shown in the following figure.

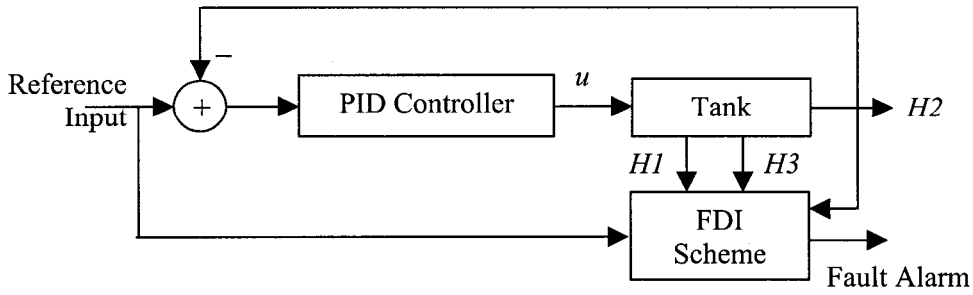


Figure 4-2. Closed-loop System of Tank System

The transfer function of the PID controller is

$$C(s) = 2 + \frac{0.2}{s} + 0.5s \tag{4-2}$$

Also, the simulated model in MATLAB is shown in Figure 4-3.

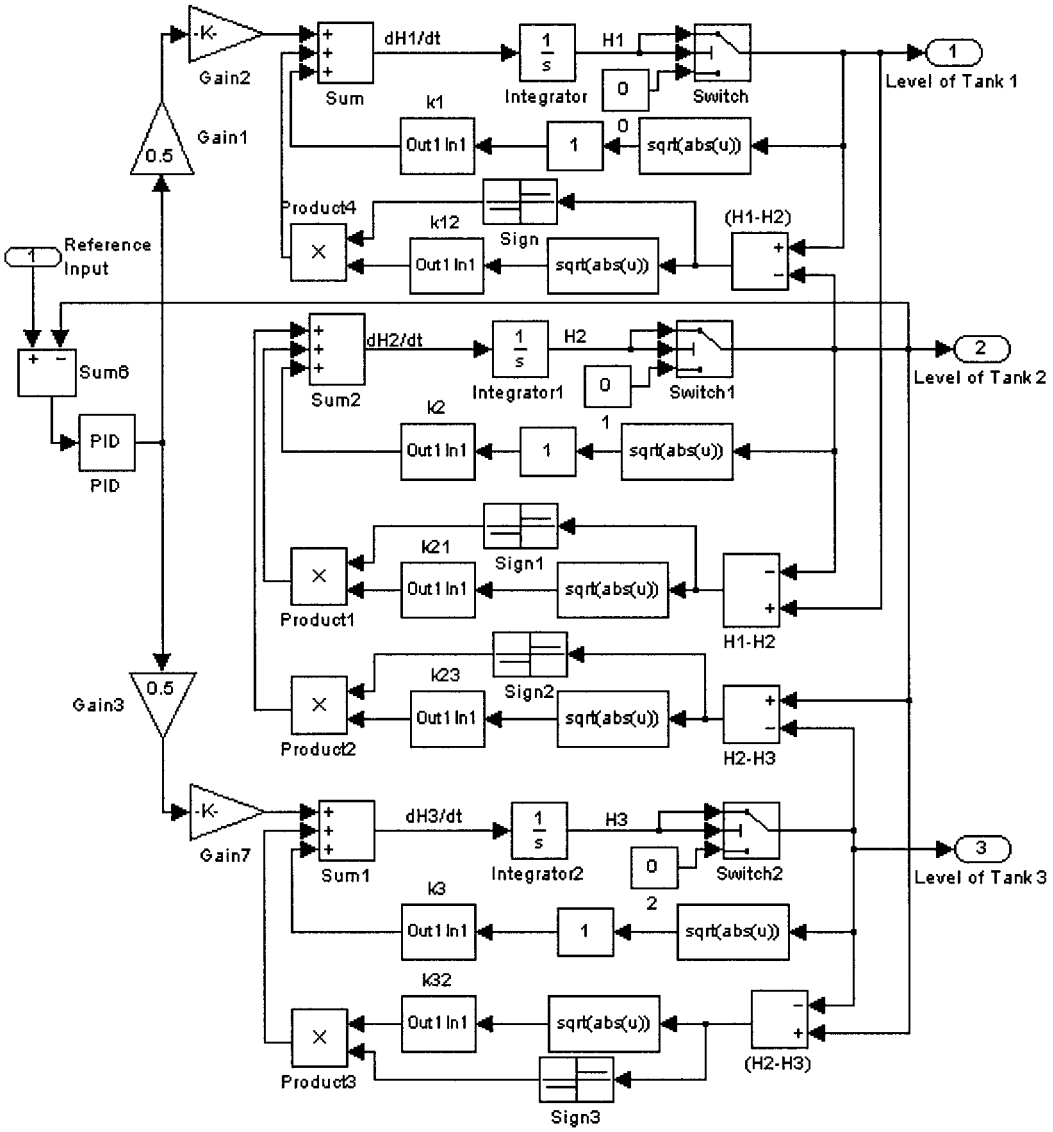


Figure 4-3. Simulated Model of Tank System in MATLAB

For this system, 11 operating conditions are simulated, including sensor faults, leakage type of faults, and disturbance. The input regulation (i.e. normal changes of operating points) is also simulated. Table 4-1 illustrates the details. Some technical details in the process of FDI are listed in Table 4-2.

Table 4-1. Operating/Faulty Status of Tank System

Status ID	Status
0	Normal operating condition (steady state)
1	Leakage in valve 1
2	Leakage in valve 2
3	Sensor 1: drifting/bias
4	Sensor 1: dead signal (stuck type)
5	Sensor 2: drifting/bias
6	Sensor 2: dead signal (stuck type)
7	Sensor 3: drifting/bias
8	Sensor 3: dead signal (stuck type)
9	Input regulation
10	Disturbance in pump

Table 4-2. Techniques Used in Tank System

Techniques	**Sub-items**	**Details**
Sliding Window	Window Size	60
	Latest Data Zone Size	5
	Current Sub-window Size	20
	Past Sub-window Size	20
	Sampling Period	0.1 Sec
Wavelet Analysis	Mother Wavelet	*DB5*
	Decompoed Level	1
	DWT extension mode	*Smooth-padding of order 1*
	Reconstructed Signal	*App. & Detail*
Feature Extraction	Extraction Rule	*R1 & R2*
	Combination Rule	*Space Expansion*
	Incorporation Rule	*Copy*
	Normalization	*Sigmod*
Regional SOM	Number of Input Neurons	16
	Size of Ouput Neurons	10*10
	Number of Training Loop	5000

Associated with the sliding window explained in Figure 3-8, the sizes of sliding window, latest data zone, past sub-window, and current sub-window are defined. The parameters in discrete wavelet analysis discussed in section 2.1.2 and 3.3.1, such as mother wavelet type, decomposed level and extension method, are defined. Also, the coefficients in feature extraction are defined. Finally, the architecture of RSOM neural network is defined.

The above parameters are specific for this system, and can be adjusted for designing a different system.

Corresponding to the 11 operating conditions listed in Table 4-1, the features are constructed based on Rule 1 and Rule 2 defined in Eq.(3-10) and Eq.(3-11). Each feature vector, $X_j = \{x_i\}_j \quad i = 1,2,...,16 \quad j = 1,2,...,11$, contains 16 points:

x_1: Changing ratio r_1 for the wavelet *details* of the tank 1 level

x_2: Changing ratio r_2 for the wavelet *details* of the tank 1 level

x_3, x_4: Changing ratio r_1 and r_2 respectively, for the wavelet *details* of the tank 2 level

x_5, x_6: Changing ratio r_1 and r_2 respectively, for the wavelet *details* of the tank 3 level

x_7, x_8: Changing ratio r_1 and r_2 respectively, for the wavelet *details* of the input signal

$x_9 - x_{16}$: Same as $x_1 - x_8$, for the wavelet *approximations* of the above measured signals.

The following figure shows the feature vectors of the 11 simulated operating conditions.

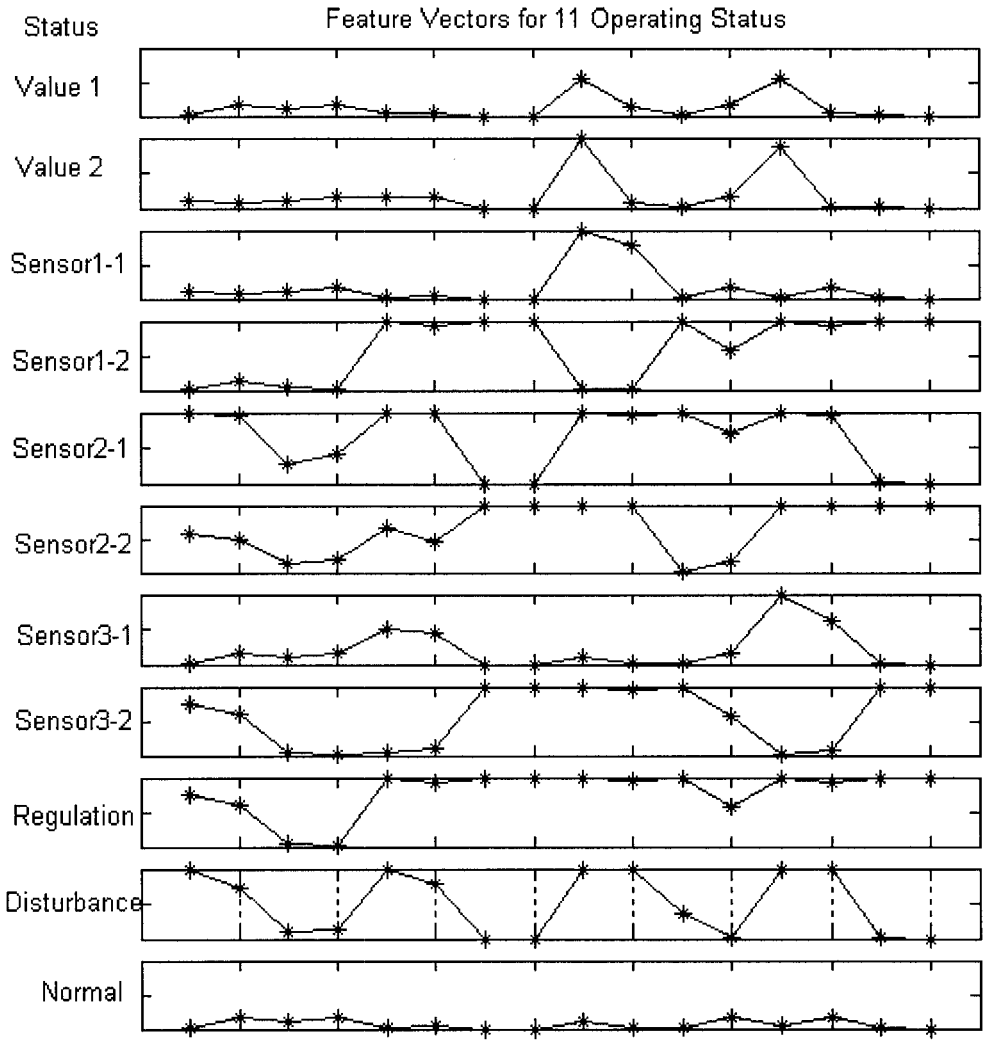


Figure 4-4. Feature Vectors of 11 Simulated Operating Conditions

It can be seen that the above features are distinctive. Being trained in the RSOM neural network proposed in section 2.2.2, these features are easily classified to generate a feature pattern table shown in Figure 3-13(a).

Simulating in different noise levels and fault sizes, the FDI results are given in Table 4-3.

Table 4-3. On-Line Simulating Results of Tank System

Source	Type	Results
Sensor 1,2&3	Dead Signal, Drifting, Offset	Detected and isolated when signal changing ratio > 5%; Dead signal can be detected and isolated when tuning input. All three types are classified as sensor faults, however, which type is not identified.
Valve 1&3	Leaking	Detected and isolated when signal changing ratio > 5%
Valve 2	Leaking	Detected and isolated when signal changing ratio > 5%
Input	Normal regulation	Detected when signal changing ratio > 5%
Disturbance	Pump disturbance	Identified when signal changing ratio > 5%
Normal	Normal operating steady state	Identified with or without noise

At the testing simulation phase, all fault/disturbance conditions with different noise levels and fault sizes are tested to verify the proposed FDI system. All fault time instances are detected and fault locations are classified only when the changing ratio of the raw signal is great than 5%, which is determined by the noise level. In this case, the signal-to-noise ratio is 60db. When SNR decreases, the percentage will increase due to the fault-induced change will be buried by the noise, which will result in missing detection and incorrect identification. It needs to point out that the normal operation such as input regulation is not misinterpreted as 'fault', although it would result in the fault-like feature in outputs. The effect of disturbance can also be distinguished from the system component faults.

4.2 DC Motor System

In this case study, the system is a lab-scale DC motor. The system description is shown in Figure 4-5.

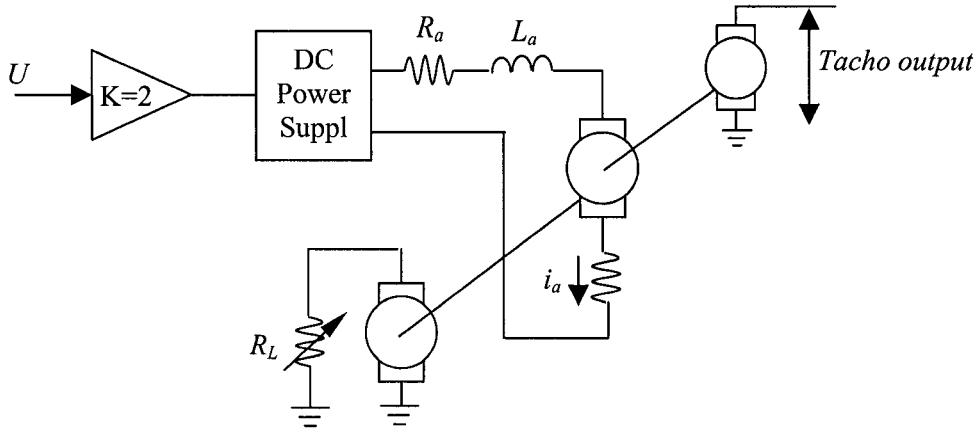


Figure 4-5. DC Motor System

The structure of the above open-loop system is illustrated as diagram as follows.

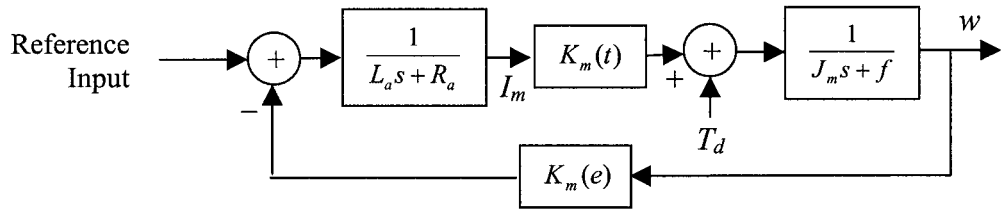


Figure 4-6. Structure of DC Motor System

The related parameters of the system are given as follows:

L_a: The motor armature inductance (0.08 H)

R_a: The motor armature resistance (4.0 Ω)

J_M: The moment of inertia of the whole system (0.012 $kg \bullet m^2$)

f: The motor friction coefficient (0.002 $N \bullet m \bullet s/rad$)

$K_m(e)$: The motor voltage constant (0.46 $V \bullet s/rad$)

$K_m(t)$: The motor mechanical constant (0.46 $V \bullet m/A$)

K_s: The tacho speed-voltage constant (0.0215 V/rpm)

The fault detection and identification system is implemented in a closed-loop configuration. A PID controller is implemented to maintain the desired angle speed of

motor, which is measured by a tachometer. The faults, disturbances, or manipulative adjustment occurring in the system will be detected and isolated in real time. The simulated model with a PID controller in SIMULINK is shown in Figure 4-7.

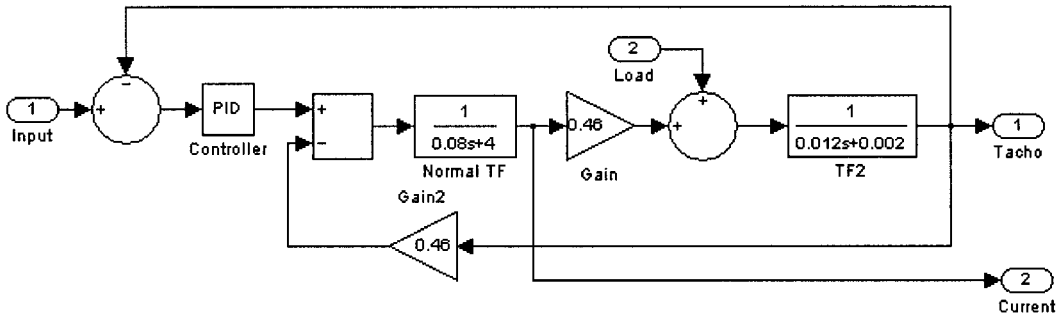


Figure 4-7. Simulated Model of Tank System in MATLAB

For this system, 8 operating conditions are simulated, including normal input regulation, sensor faults, leakage type of faults, and disturbance. Table 4-4 illustrates the details.

Table 4-4. Operating/Faulty Status of Motor System

Status ID	Status
0	Normal operating condition (steady state)
1	Sensor 1: drifting/bias
2	Sensor 1: dead signal (stuck type)
3	Sensor 2: drifting/bias
4	Sensor 2: dead signal (stuck type)
5	Model Fault
6	Input regulation
7	Disturbance in Load

Similarly as Table 4-2, the following techniques are used in processing in FDI analysis for motor system.

Table 4-5. Techniques used in Motor System

Techniques	Sub-items	Details
Sliding Window	Window Size	*60*
	Latest Data Zone Size	*5*
	Current Sub-window Size	*20*
	Past Sub-window Size	*20*
	Sampling Period	*0.1 Sec*
Wavelet Analysis	Mother Wavelet	*DB5*
	Decompoed Level	*1*
	DWT extension mode	*Smooth-padding of order 1*
	Reconstructed Signal	*App. & Detail*
Feature Extraction	Extraction Rule	*R1*
	Combination Rule	*Space/Time*
	Incorporation Rule	*Copy*
	Normalization	*Sigmod*
Regional-SOM	Number of Input Neurons	*12*
	Size of Ouput Neurons	*7*7*
	Number of Training Loop	*5000*

Corresponding to the 8 operating conditions listed in Table 4-4, the features are constructed based on Rule 1. From the motor dynamics shown in Figure 4-7, one can see that there is a time delay between the armature current I_m and the motor speed w when sampling them. In order to capture the possible dynamic changes more rapidly, one has to extend the feature space in order to include more information. Two samples at adjacent sampling steps are processed simultaneously and included in the feature vector. This is called the time extension, which is one of the two methods that make the proposed FDI design suitable for many applications. The feature vector then has 12 samples, $X_j = \{x_i\}_j \;\; i = 1,2,...,12 \;\; j = 1,2,...,8$.

x_1: Changing ratio r_1 for the wavelet *details* of the speed

x_2: Changing ratio r_1 for the wavelet *details* of the current

x_3: Changing ratio r_1 for the wavelet *details* of the input signal

$x_4 - x_6$: Same as $x_1 - x_3$, for the wavelet *approximations* of the above measured signals

$x_7 - x_{12}$: Same as $x_1 - x_6$, for the same coefficients as above but sampled at the next sampling period.

The following figure shows the feature vectors of the 8 simulated operating status.

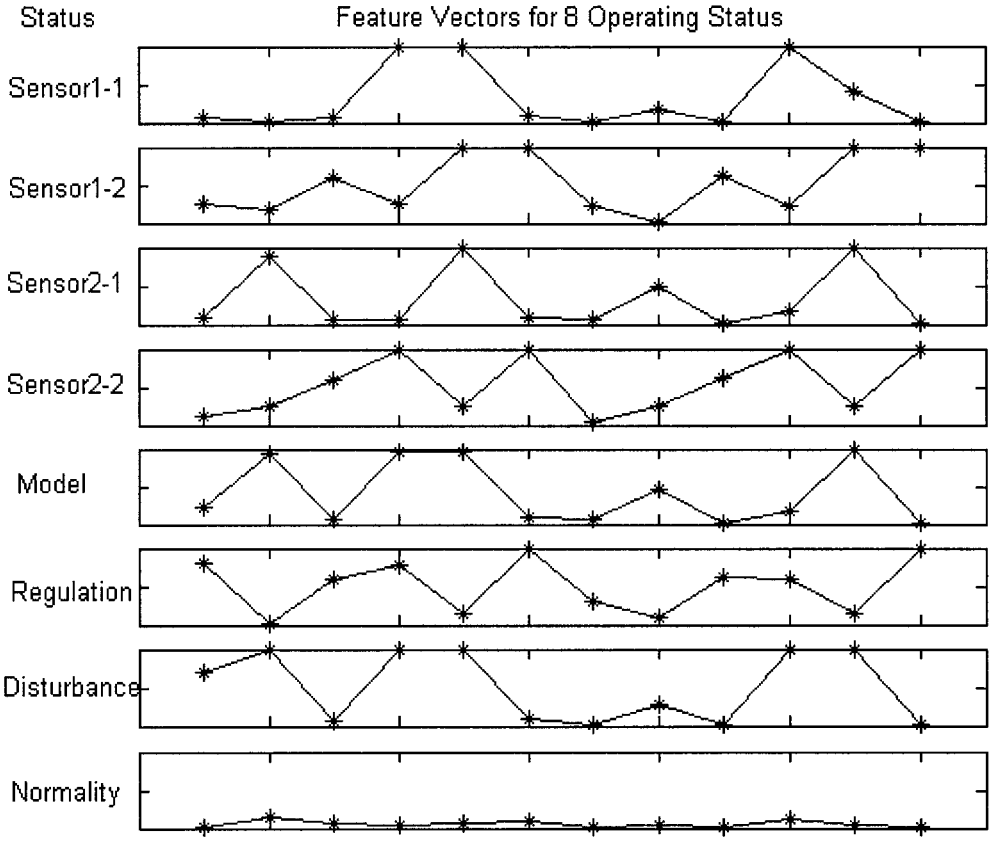


Figure 4-8. Feature Vectors of 8 Simulated Operating Status

It can be seen that the above features are distinctive. Being trained in the RSOM neural net proposed in section 2.2.2, these features are easily classified to generate a feature pattern table shown in Figure 4-9.

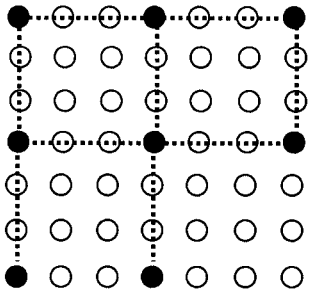


Figure 4-9. Pattern Table

By performing simulation with different noise levels and fault sizes, the results are shown in Table 4-6.

Table 4-6. On-Line Simulating Results of Motor System

Source	Type	Results
Sensor 1,2	Dead Signal, Drifting, Offset	Detected and isolated when signal changing ratio > 5%; Dead signal can be detected and isolated when tuning input. All three types are classified as sensor faults, however, which type is not identified.
Model	Parameters jumping	Detected and isolated when signal changing ratio > 5%
Input	Normal regulation	Detected when signal changing ratio > 5%
Disturbance	Pump disturbance	Identified when signal changing ratio > 5%
Normal	Normal operating steady state	Identified with or without noise

With the cases studied above, one can conclude that the design goals mentioned in Chapter 3 are attained, and at the same time, the performance and effectiveness of the proposed FDI system are also guaranteed.

Chapter V

Comparison Study with Model-based Method

In this chapter, a comparison with model-based method, extended Kalman filter, is studied and performed on the same three-tank system. The results will show the advantages and disadvantages of the model-based method. The advantages of the method in this thesis over the model-based one are also discussed.

5.1 Introduction to Extended Kalman Filter

The extended Kalman filter (EKF) approach is to apply the standard Kalman filter (for linear systems) to nonlinear systems with additive process and measurement white noise by continuously updating a linearization around the previous state estimate, starting with an initial guess. In other words, it is considered that a linear Taylor approximation of the system function at the previous state estimate and that of the observation function at the corresponding predicted position [73]-[77].

For a nonlinear discrete system with process and measurement white noise,

$$\begin{aligned} x(k+1) &= f(k,u(k),x(k)) + \Gamma(k)v(k) \qquad R^n \times R^q \to R^n \\ y(k+1) &= h(k+1,x(k+1)) + e(k+1) \qquad R^n \to R^m \end{aligned} \tag{5-1}$$

Where, $x \in R^n$ is n-dimensional state vector of nonlinear system, $u \in R^q$ is q-dimensional manipulative input variable(s), and $y \in R^m$ is m-dimensional output vector. Nonlinear functions $f(\cdot), h(\cdot)$ are differentiable with respect to state vector x. $v \in R^p$ is p-dimensional process noise and $e \in R^m$ is m-dimensional measurement noise with the following statistical characteristics,

$$\begin{aligned} &E[v(k)] = E[e(k)] = 0 \\ &E[v(k)v^T(j)] = Q_1(k)\delta_{k,j} \\ &E[e(k)e^T(j)] = Q_2(k)\delta_{k,j} \\ &E[v(k)e^T(j)] = 0 \end{aligned} \tag{5-2}$$

Where, $\delta_{k,j}$ is Dirac function,

$$\delta_{k,j} = \begin{cases} 1 & k = j \\ 0 & k \neq j \end{cases} \tag{5-3}$$

And Q_1 is symmetric positive-definite matrix, and Q_2 is symmetric nonnegative-definite matrix.

The EKF algorithm is given as follows:

The state estimation:

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + K(k+1) \cdot \gamma(k+1) \tag{5-4}$$

where,

$$\hat{x}(k+1|k) = f(k, u(k), \hat{x}(k|k)) \tag{5-5}$$

K is gain matrix,

$$\begin{aligned} K(k+1) = {} & P(k+1|k) H^T(k+1, \hat{x}(k+1|k)) \\ & [H(k+1, \hat{x}(k+1|k) P(k+1|k) H^T(k+1, \hat{x}(k+1|k) + Q_2(k+1)]^{-1} \end{aligned} \tag{5-6}$$

$P(k+1|k)$ is *a priori* estimate error covariance matrix,

$$P(k+1|k) = F(k, u(k), \hat{x}(k|k)) P(k|k) F^T(k, u(k), \hat{x}(k|k)) + \Gamma(k) Q_1(k) \Gamma^T(k) \tag{5-7}$$

$P(k+1|k+1)$ is *a posteriori* estimate error covariance matrix,

$$P(k+1|k+1) = [I - K(k+1) H(k+1, \hat{x}(k+1|k))] P(k+1|k) \tag{5-8}$$

γ is residual sequence,

$$\begin{aligned} \gamma(k+1) &= y(k+1) - \hat{y}(k+1) \\ &= y(k+1) - h(k+1, \hat{x}(k+1|k)) \end{aligned} \tag{5-9}$$

F is the Jacobian matrix of f with respect to state variables x,

$$F(k, u(k), \hat{x}(k|k)) = \frac{\partial f(k, u(k), x(k))}{\partial x} \Big|_{x(k)=\hat{x}(k|k)} \tag{5-10}$$

H is the Jacobian matrix of h with respect to state variables x,

$$H(k+1, \hat{x}(k+1|k)) = \frac{\partial h(k+1, x(k+1))}{\partial x} \Big|_{x(k+1)=\hat{x}(k+1|k)} \tag{5-11}$$

5.2 EKF Based Fault Detection for Tank System

The closed-loop tank system is shown in Figure 4.2. The structure of the implementation of the extended Kalman Filter is shown in the following figure.

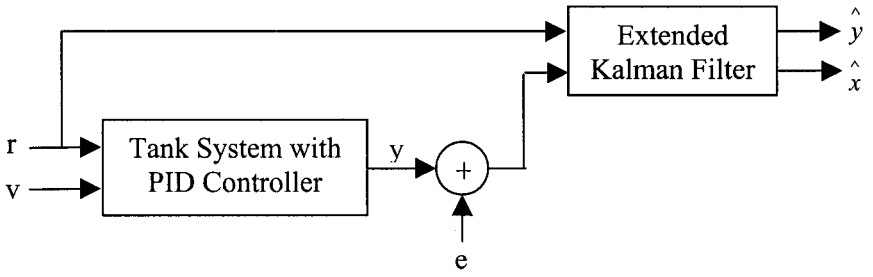


Figure 5-1. Implementation of Extended Kalman Filter

The equations of the nonlinear system are shown in Eq.(4-1) in section 4.1. The transfer function of the PID controller is

$$C(s) = 2 + \frac{0.2}{s} + 0.5s \tag{5-12}$$

For the practical consideration of implementing the derivative term in the above equation, one can use the following form:

$$C(s) = 2 + \frac{0.2}{s} + \frac{0.5s}{0.005s + 1} \tag{5-13}$$

Then, the equations of the closed-loop system in time domain become,

$$\begin{aligned}
\dot{H}_1(t) &= -\frac{k_1}{S_1}\sqrt{g \cdot H_1(t)} - \frac{k_{12}}{S_1}\operatorname{sgn}(H_1(t) - H_2(t))\sqrt{g \cdot |H_1(t) - H_2(t)|} \\
&\quad + \frac{-102H_2(t) - 156.25H_4(t) + 0.625H_5(t)}{2S_1} + \frac{102r(t)}{2S_1} \\
\dot{H}_2(t) &= -\frac{k_2}{S_2}\sqrt{g \cdot H_2(t)} + \frac{k_{12}}{S_2}\operatorname{sgn}(H_1(t) - H_2(t))\sqrt{g \cdot |H_1(t) - H_2(t)|} \\
&\quad - \frac{k_{23}}{S_2}\operatorname{sgn}(H_2(t) - H_3(t))\sqrt{g \cdot |H_2(t) - H_3(t)|} \\
\dot{H}_3(t) &= -\frac{k_3}{S_3}\sqrt{g \cdot H_3(t)} + \frac{k_{23}}{S_3}\operatorname{sgn}(H_2(t) - H_3(t))\sqrt{g \cdot |H_2(t) - H_3(t)|} \\
&\quad + \frac{-102H_2(t) - 156.25H_4(t) + 0.625H_5(t)}{2S_3} + \frac{102r(t)}{2S_3}
\end{aligned} \tag{5-14}$$

$$\dot{H}_4(t) = -128H_2(t) - 200H_4(t) + 128r(t)$$

$$\dot{H}_5(t) = 0.5H_4(t)$$

Where, H_1, H_2, H_3 are measurable state variables but H_4, H_5 are not physical variables and cannot be measured. The output equations are,

$$\begin{aligned} y_1(t) &= H_1(t) \\ y_2(t) &= H_2(t) \\ y_3(t) &= H_3(t) \end{aligned} \tag{5-15}$$

By discretizing the above continuous system, and with the process and measurement noises modeled as white noises, one gets the following discrete system,

$$\begin{aligned} H_1(k+1) = H_1(k) + T \bullet (&-\frac{k_1}{S_1}\sqrt{g \cdot H_1(k)} - \frac{k_{12}}{S_1}\mathrm{sgn}(H_1(k) - H_2(k))\sqrt{g \cdot |H_1(k) - H_2(k)|} \\ &+ \frac{-102H_2(k) - 156.25H_4(k) + 0.625H_5(k)}{2S_1} + \frac{102r(k)}{2S_1}) + v_1(k) \end{aligned}$$

$$\begin{aligned} H_2(k+1) = H_2(k) + T \bullet (&-\frac{k_2}{S_2}\sqrt{g \cdot H_2(k)} + \frac{k_{12}}{S_2}\mathrm{sgn}(H_1(k) - H_2(k))\sqrt{g \cdot |H_1(k) - H_2(k)|} \\ &- \frac{k_{23}}{S_2}\mathrm{sgn}(H_2(k) - H_3(k))\sqrt{g \cdot |H_2(k) - H_3(k)|}) + v_2(k) \end{aligned} \tag{5-16}$$

$$\begin{aligned} H_3(k+1) = H_3(k) + T \bullet (&-\frac{k_3}{S_3}\sqrt{g \cdot H_3(k)} + \frac{k_{23}}{S_3}\mathrm{sgn}(H_2(k) - H_3(k))\sqrt{g \cdot |H_2(k) - H_3(k)|} \\ &+ \frac{-102H_2(k) - 156.25H_4(k) + 0.625H_5(k)}{2S_3} + \frac{102r(k)}{2S_3}) + v_3(k) \end{aligned}$$

$$H_4(k+1) = H_4(k) + T \bullet (-128H_2(k) - 200H_4(k) + 128r(k)) + v_4(k)$$

$$H_5(k+1) = H_5(k) + T \bullet (0.5H_4(k)) + v_5(k)$$

Where, T is sampling period. The corresponding output equations are shown as follows,

$$y(k+1) = \begin{bmatrix} y_1(k+1) \\ y_2(k+1) \\ y_3(k+1) \end{bmatrix} = \begin{bmatrix} H_1(k+1) \\ H_2(k+1) \\ H_3(k+1) \end{bmatrix} + \begin{bmatrix} e_1(k+1) \\ e_2(k+1) \\ e_3(k+1) \end{bmatrix} \tag{5-17}$$

To implement the extended Kalman Filter, we need to compute the Jacobian matrices F and H. The matrix F is,

$$F(1,1)=\left[1+T\cdot(-\frac{k_1}{2S_1}\sqrt{\frac{g}{H_1(k)}}-\frac{k_{12}}{2S_1}\sqrt{\frac{g}{|H_1(k)-H_2(k)|}})\right]$$

$$F(1,2)=\left[T\cdot(\frac{k_{12}}{2S_1}\sqrt{\frac{g}{|H_1(k)-H_2(k)|}}-\frac{102}{2S_1})\right]$$

$$F(1,4)=\left[T\cdot(-\frac{156\ .25}{2S_1})\right]$$

$$F(1,5)=\left[T\cdot(\frac{0.625}{2S_1})\right]$$

$$F(2,1)=\left[T\cdot(\frac{k_{12}}{2S_2}\sqrt{\frac{g}{|H_1(k)-H_2(k)|}})\right]$$

$$F(2,2)=\left[1+T\cdot(-\frac{k_2}{2S_2}\sqrt{\frac{g}{H_2(k)}}-\frac{k_{12}}{2S_2}\sqrt{\frac{g}{|H_1(k)-H_2(k)|}}-\frac{k_{23}}{2S_2}\sqrt{\frac{g}{|H_2(k)-H_3(k)|}})\right]$$

$$F(2,3)=\left[T\cdot(\frac{k_{23}}{2S_2}\sqrt{\frac{g}{|H_2(k)-H_3(k)|}})\right]$$

$$F(3,2)=\left[T\cdot(\frac{k_{23}}{2S_3}\sqrt{\frac{g}{|H_2(k)-H_3(k)|}}-\frac{102}{2S_3})\right] \qquad (5\text{-}18)$$

$$F(3,3)=\left[1+T\cdot(-\frac{k_3}{2S_3}\sqrt{\frac{g}{H_3(k)}}-\frac{k_{23}}{2S_3}\sqrt{\frac{g}{|H_2(k)-H_3(k)|}})\right]$$

$$F(3,4)=\left[T\cdot(-\frac{156.25}{2S_3})\right]$$

$$F(3,5)=\left[T\cdot(\frac{0.625}{2S_3})\right]$$

$$F(4,2)=T\cdot(-128)$$

$$F(4,4)=1+T\cdot(-200)$$

$$F(5,4)=T\cdot(0.5)$$

$$F(5,5)=1$$

Other elements in F matrix are zero.

The Jacobian matrix H is,

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{5-19}$$

When the system operates normally, the residual signal $\gamma(k)$ generated from Eq. (5-9), the difference between the estimated signals and the actual sensor signals, is approximate to a Gaussian white noise [78], whose covariance matrix is,

$$\begin{aligned} V(k) &= E[\gamma(k)\gamma^T(k)] \\ &\approx H(k,\hat{x}(k \mid k-1))P(k \mid k-1)H^T(k,\hat{x}(k \mid k-1)) + Q_2(k) \end{aligned} \tag{5-20}$$

Where, $V(k)$ is time-varying and its stochastic characteristics is changing with k. Therefore, define another stochastic variable $\xi(k)$ which is a Gaussian white noise,

$$\xi(k) = V^{-1/2}(k)\gamma(k) \tag{5-21}$$

Then, define $\xi^T(k)\xi(k)$ to make calculation easy.

$$\xi^T(k)\xi(k) = \gamma^T(k)V^{-1}(k)\gamma(k) \tag{5-22}$$

The sequence $\xi^T(k)\xi(k)$ can be approximated to have a χ^2_{m-1} distribution, by calculating the index $d(k)$ defined as follows, one can use a standard hypothesis test to detect the faults,

$$\begin{aligned} d(k) &= \frac{1}{N}\sum_{j=k-N+1}^{k}\xi^T(k)\xi(k) \\ &= \frac{1}{N}\sum_{j=k-N+1}^{k}\gamma^T(k)V^{-1}(k)\gamma(k) \end{aligned} \tag{5-23}$$

Where, N is the size of the data in window.

When the system operates normally, $d(k)$ has very small value. When fault occurs abruptly, $\xi(k)$ does not satisfy the characteristics of the white noise and $d(k)$ will change.

Assume a confidence measure P_0 and a threshold β, we have

$$P(d(k) > \beta) = P_0 \tag{5-24}$$

The following hypothesis test can be performed to detect the fault,

$$\begin{array}{lll} H_0 & d(k) < \beta & normal \\ H_1 & d(k) > \beta & faulty \end{array} \tag{5-25}$$

How to choosing β is due to the consideration of FDI performance, such as to obtain the balanced false alarm and missing detection rates.

5.3 Simulation Result

Giving the following parameters

$$Q_1 = \begin{bmatrix} 1.5\times10^{-4} & 0 & 0 \\ 0 & 1.0\times10^{-4} & 0 \\ 0 & 0 & 1.3\times10^{-4} \end{bmatrix} \tag{5-26}$$

$$Q_2 = 10^{-4} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5-27}$$

The following figure is the actual and estimated variables at the normal regulating conditions, the water level adjusting from 0.3m to 0.4m.

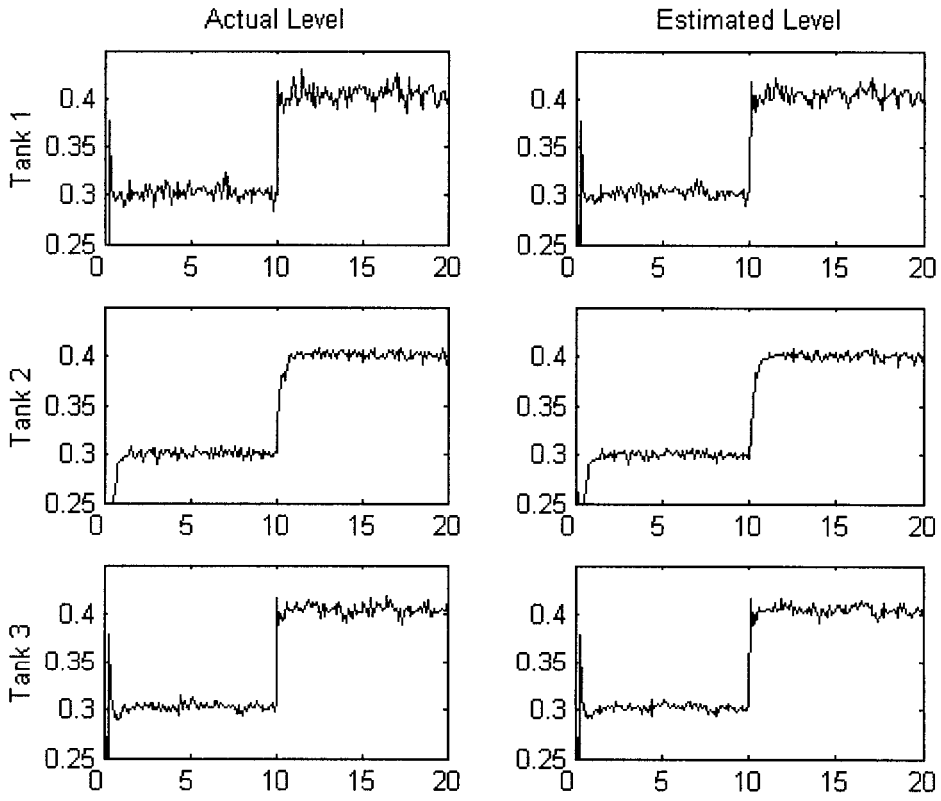


Figure 5-2. Actual and Estimated Levels when Regulating

The residual sequence is shown in the following figure. The residual has no any change when regulating and it is close to zero under normal and regulating conditions.

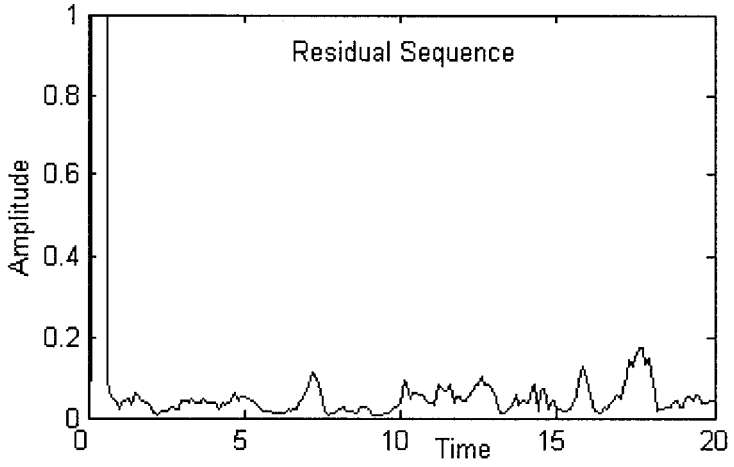


Figure 5-3. Residual Sequence

When leakage fault occurs in valve 1, the levels of tank 1,3 will decrease and the level of tank 2 will keep same because of the PID controller.

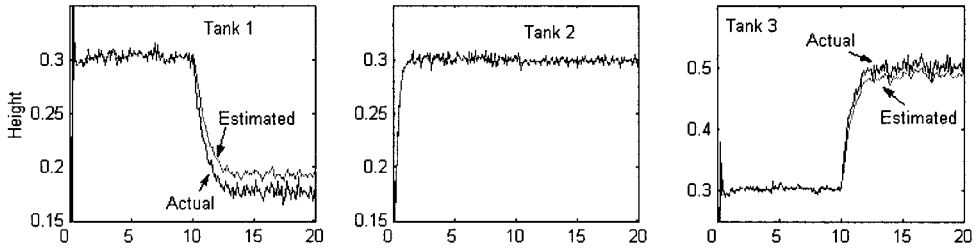


Figure 5-4. Actual and Estimated Values

Observing from the above figure, the estimated value will follow the actual value at the normal environment. When fault occurs, the estimated value will also follow the actual one but a small bias will be generated after fault. The residual sequence will increase from zero abruptly when fault occurs, shown in the following figure.

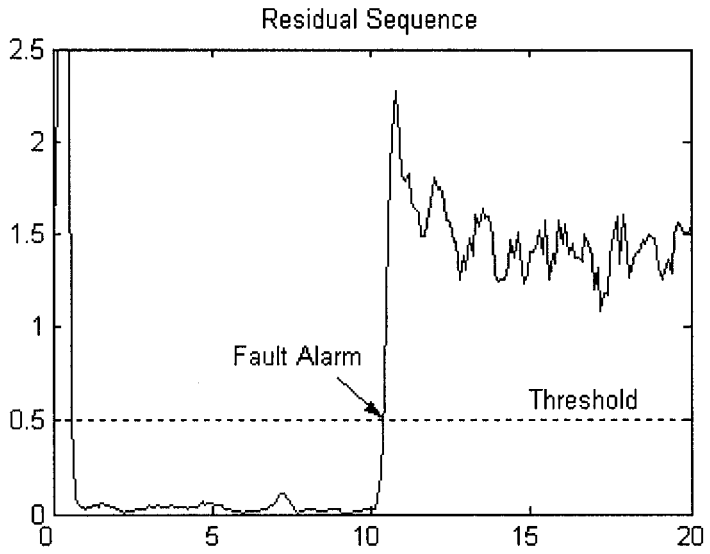


Figure 5-5. Residual Sequence when Faulting

The above figure gives the generated residual signal based on the estimation bias. If the residual is greater than this threshold, the fault alarm is given. The threshold can be chosen properly with considerations of detection delay, rate of missing detection and false alarm.

5.4 Comparison

Based on the simulation under the all different operating conditions, one can compare the proposed method (knowledge based) to the method based on EKF (model based). The following table shows the comparison results.

Table 5-1. Comparison of EKF-based and Knowledge-based Method

Characteritics	EKF	Knowledge-based
Requirement on Models	Obtain precise system models	Known or Unknown
Pre-works	Build precise model	Simulating all possible operating statuses
Effect of Threshold	Important. Deciding efficiency and effectiveness of detection	No threshold set
Effect of Input Regulation	No effect on residual.	Affected and should be detected
Detection Delay	Detected after 1~5 sampling periods. Also depending on value of threshold	Fault Detected after 1~3 sampling periods
Sensitivity to fault-size	Detected when signal changing ratio > 10%. Effective detection depends on the value of threshold	Detected and isolated when signal changing ratio > 5%
Disturbance in pump	False alarm in real-time, can be detected off-line	Detected on-line
Effects of Noise	Result in missing detection in big noise environment. Validity of noise statistical characteristic is important in detecting.	Fault can not be detected if fault-induced change is buried by noise
Isolation	No isolation	High accuracy

Observed from the above table, it can be seen that the proposed FDI design has achieved better performance than the model-based method. The precise model is not required in the knowledge-based method, but required in EKF-based method. The design in this thesis is much more robust to the noise than EKF-based does. Most importance of all is EKF-based could not isolate the fault and other method is required to identify the faults, however, the proposed design can isolate the faults with high accuracy.

Chapter VI

Implementation and Experiment Results

In this chapter, the design is applied to a pilot-scale tank system. The experiment results are discussed.

6.1 System Description

The proposed FDI method is implemented on a pilot-scale stirred tank system. Schematic diagram is shown in Figure 6-1.

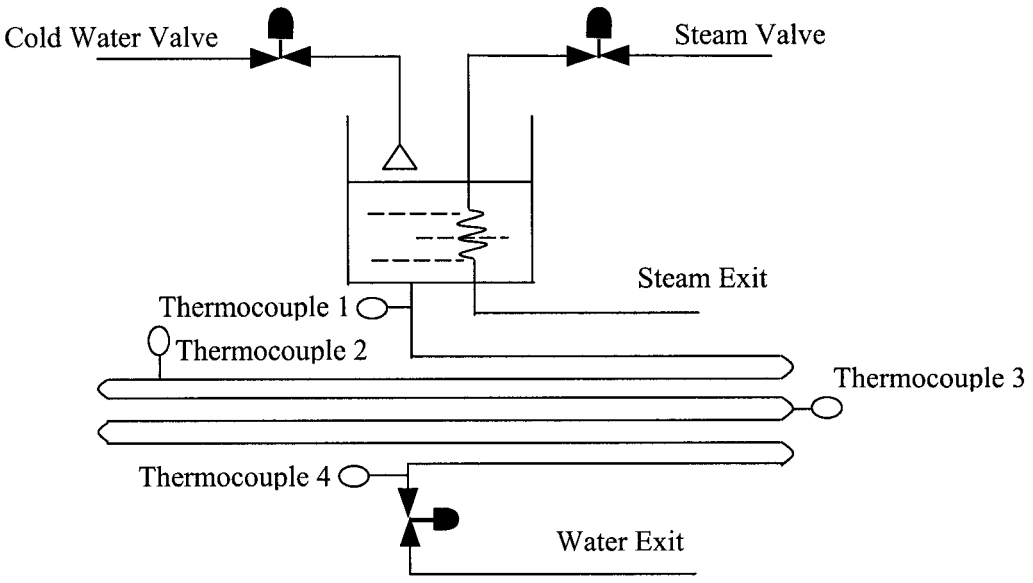


Figure 6-1. Schematic Diagram of Tank

It consists of a cylindrical tank with a long exit pipe and a stream coil running through the tank. There are two valves to manipulate the inlet cold water and steam flow rates. Another valve at the end of the exit pipe can be adjusted manually. A number of thermocouples are placed all along the exit pipe. It is possible to measure the temperature from any of these thermocouples. A DP cell measures the level and an orifice meter on the inlet cold water measures the flow rate.

The FDI scheme is applied in the following closed-loop system configuration.

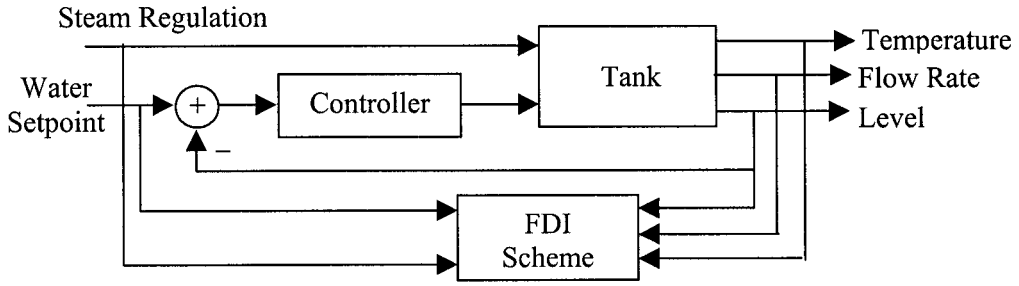


Figure 6-2. FDI Scheme in Closed-loop System

6.2 Experiment Results

For this system, 10 system operating conditions are simulated, including normal input regulation, sensor faults, and leakage fault. Table 6-1 illustrates the details.

Table 6-1. Operating/Faulty Status

Status ID	Status
0	Normal operating condition (steady state)
1	Cold water regulation
2	Steam regulation
3	Leaking/blocking fault
4	Temperature sensor: drifting/bias
5	Temperature sensor: dead signal (stuck type)
6	Flow rate sensor: drifting/bias
7	Flow rate sendor: dead signal (stuck type)
8	Level sensor: drifting/bias
9	Level sendor: dead signal (stuck type)

One set of signals generated under all operating conditions listed in the above table, are collected (sampling rate is 2 second) and processed as the training data. Then the on-line test is performed. In this case, the fault sizes are usually different from the ones simulated in the training stage. The experiment is performed with 21 events (labeled as A~U in Figure 6-2) sequentially schedule on the system in 122 minutes (3680

*2 seconds). Figure 6-2 shows the system responses in the testing phase. Figure 6-3 shows the testing results. Table 6-2 explains 21 testing events.

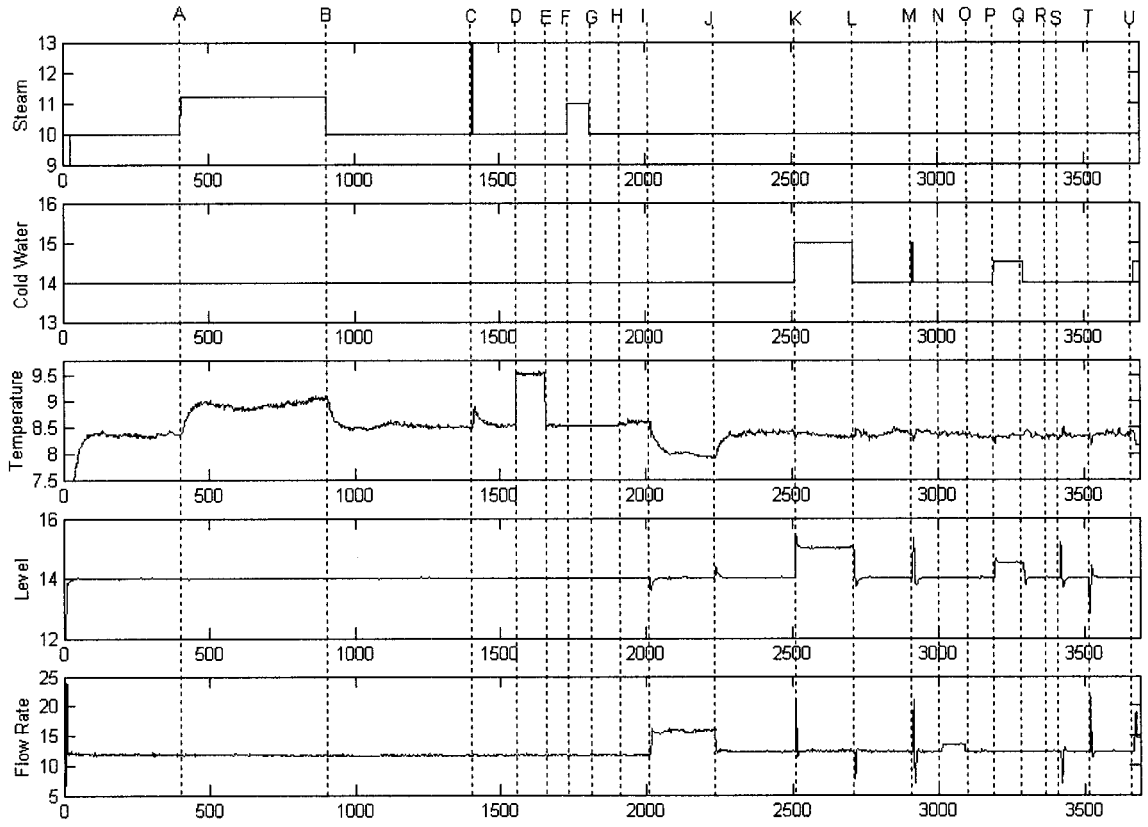


Figure 6-2. Data Sequence in Test

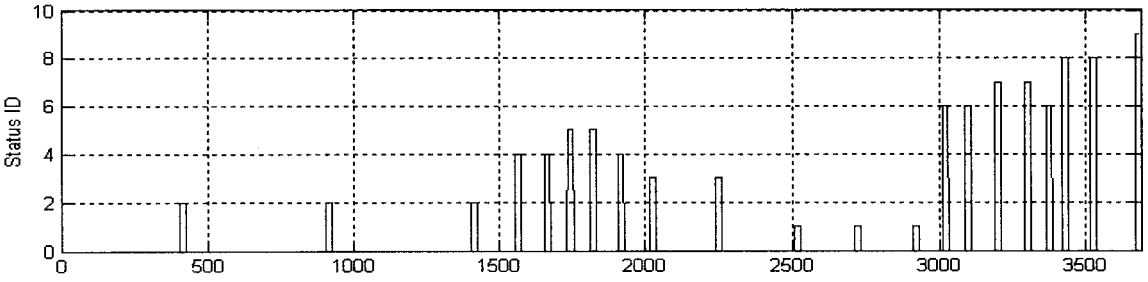


Figure 6-3. Recognized Results

Table 6-2. Testing Events

Event No	Status ID	Conditions
A	2	Steam flow rising regulation
B	2	Steam flow dropping regulation

C	2	Steam flow short time regulation
D	4	Temperature sensor drifting
E	4	Temperature sensor drifting
F	5	Temperature sensor dead
G	5	Temperature sensor dead
H	4	Temperature sensor from dead to normal, like sensor drifting
I	3	Leaking fault
J	3	Clogging fault
K	1	Water flow rising regulation
L	1	Water flow dropping regulation
M	1	Water flow short time regulation
N	6	Water flow sensor drifting
O	6	Water flow sensor drifting
P	7	Water flow sensor dead
Q	7	Water flow sensor dead
R	6	Flow sensor from dead to normal, like sensor drifting
S	8	Level sensor drifting
T	8	Level sensor drifting
U	9	Level sensor dead

It is known form Figure 6-2, Figure 6-3, and Table 6-2 that all 21 events are recognized correctly in real time. The proposed FDI scheme achieves the desired performance.

Chapter VII

Conclusion and Future Works

In this thesis, a novel fault detection and identification scheme, the knowledge-based method integrated with wavelet analysis, statistic analysis, and improved RSOM neural network, is developed, based on the desired design objectives. Case studies based on the tank and motor systems demonstrate the effectiveness of this FDI scheme. Also, a model-based method, extended Kalman filter based, is performed in the tank system to compare the proposed approach in this thesis. Furthermore, the design is applied to a real process training system.

It has been shown that the proposed FDI scheme has achieved satisfactory performance.

The knowledge-based FDI scheme in this thesis can achieve high accuracy of detection and isolation, also can maintain low rate of missing detection, false alarm, and incorrect identification. It needs to point out that this FDI system combines the signal processing with the AI technique, so it is impossible to analytically evaluate the rates of missing detection and false alarm. The noise, disturbance and uncertainties in the system can affect the results of detection and isolation and thus the rate of missing detection and false alarm. The proposed FDI system uses wavelet analysis and improved RSOM neural network to reduce the effect of the uncertain factors in systems. First of all, if a fault occurs, the signals may encounter some abrupt changes and enter the transient states. Wavelet analysis can capture the abrupt change and analyze the transients in the system. Therefore, wavelet analysis is able to guarantee the real time detection and extract the signature of signal from a noisy environment. Secondly, the RSOM neural network is robust to the small difference of the input features resulted from the same operating condition. So a group of input features resulted from the same operating condition with small difference can be identified as the same pattern by RSOM. Therefore, the rate of false alarm and missing detection is lowered. Also, by case studies, experiment, and comparison with EKF-based method, the same conclusion is made.

The time delay of fault detection is 1~3 sampling period(s). During this period, the fault feature can be captured and extracted by the wavelet analysis. In the off-line

module, the features of the operating conditions are trained in the neural network, thus provide the corresponding system statuses. When a fault occurs on-line in real time, its feature can be recognized instantly by matching it with the ones in knowledge base. This way, the real-time detection and isolation can be guaranteed.

This FDI can be generalized as a universal framework for the different applications by adjusting the parameters in the three techniques.

However, there are some limitations of this FDI scheme, which appear to be common in most knowledge-based FDI methods. First of all, all possible faults need to be simulated off-line. In particular, changes in the environmental conditions and long time changes due to the aging condition are hardly captured in a limited time frame, which will result in longer time delay or even missing detection. In addition, this FDI will result in missing detection when the fault-induced change in signal is buried by the noise because this FDI approach is basically data-driven, which heavily depends on the signal profile from the system.

When there is an unknown fault in the system, it can be detected because the wavelet analysis is still capable of capturing the fault-induced change. However the fault may not be correctly isolated due to the fact that there is no matched pattern in the current knowledge base, which will result in incorrect identification. In this case, RSOM should be re-trained with this new fault type added and rebuild the knowledge base.

Following the work of this thesis, the future works include,

(1) Design more effective feature extraction method in order to reflect the system operating conditions more precisely and therefore to reduce the effect of noise.

(2) Possible combination of the model based method and the proposed knowledge design. For example, with the assumption of the known noise profile in the system, how to use such an important knowledge in the offline module to build the knowledge base is an interesting yet challenging problem.

(3) Use the wavelet analysis to help model the unknown system is another interesting problem, which appears in many other research topics. It is known that by using the wavelet analysis, the residual signal can be generated and processed for the fault scenario in the dynamical system. How to combine this technique with the proposed method is under investigation.

Bibliography

[1] A.S. Willsky, "A survey of design methods for failure detection in dynamic systems", *Automatica*, Vol. 12, No. 6, 1976, pp. 601-611

[2] R.J. Patton, P.M. Frank, R.N. Clark, *Fault Diagnosis in Dynamic Systems. Theory and Application*, Prentice Hall, New York, 1989

[3] R. Isermann, "Process fault detection based on modeling and estimation methods – A survey", *Automatica*, Vol. 20, No. 4, 1984, pp. 387-404

[4] P.M. Frank, "Survey of robust residual generation and evaluation methods in observer-based fault detection systems", *Journal of Process Control*, Vol.7, No.6, 1997, pp.403-424

[5] R. Isermann, "On the applicability of model based fault detection for technical processes", *Control-Engineering-Practice*, Vol.2, No.3, 1994, pp. 439-450

[6] X.C. Lou, A.S. Willsky, and G.C. Verqhess, "Failure Detection with uncertain models", *Proc. Of the 1983 American Control Conference*, pp.956-959

[7] P.M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy – A survey and some new results", *Automatica*, Vol. 26, No.2, 1990, pp. 459-474

[8] G. Betta, A. Pietrosanto, "Instrument fault detection and isolation: state of the art and new research trend", *Instrumentation and Measurement Technology Conference, 1998. IMTC/98. Conference Proceedings. IEEE*, Vol. 1, 1998, pp. 483 -489

[9] F. Filippetti, G. Franceschini, C. Tassoni, P. Vas, "Recent developments of induction motor drives fault diagnosis using AI techniques", *IEEE Transactions on Industrial Electronics*, Vol. 47, No. 5, Oct. 2000, pp. 994 -1004

[10] H.E. Rauch, "Intelligent fault diagnosis and control reconfiguration", *IEEE Control Systems Magazine*, Vol. 14, No. 3, June 1994, pp. 6 -12

[11] J. Gertler, "Analytical redundancy methods in fault detection and isolation", *IFAC-Symposia-Series*, No.6, 1992, pp. 9-21

[12] R.J. Patton, "Design of fault detection and isolation observers: A matrix Pencil approach", *Automatica*, Vol. 34, No. 9, 1998, pp. 1135-1140

[13] Jose Ragot, Didier Maquin, "An algorithm for obtaining the redundancy equations of LTI systems", *Automatica*, Vol. 30, No. 3, 1994, pp. 537-542

[14] A. S. Willsky, and H. L. Jones, "A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems," *IEEE Transaction on Automatic Control AC-21*, 1976, pp.108-121

[15] J. Chen, H. Zhang, "Robust detection of faulty actuators via unknown input observers", *INTJ of System Science*, Vol.22, No.10, 1991, pp.1829-1839

[16] Chia-Chi Tsui, "A general failure detection, isolation and accommodation system with model uncertainty and measurement noise", *IEEE transactions on Automatic Control*, Vol. 39, Issue 11, Nov. 1994, pp. 2318 –2321

[17] D.N. Shields, "Robust fault detection for generalized state space systems", *Proceedings of the International Conference on CONTROL '94*, Vol.2, 1994, pp. 1335-1339

[18] Y. Maki, K.A. Loparo, "A neural-network approach to fault detection and diagnosis in industrial processes", *IEEE Transactions on Control Systems Technology*, Vol. 5, No. 6, Nov. 1997, pp. 529 -541

[19] Xiaoqin Pei, F.N. Chowdhury, "Unsupervised neural network for fault detection and classification in dynamic systems", *Proceedings of the 1999 IEEE International Conference on Control Applications*, 1999, Vol. 1, 1999, pp. 640 -645

[20] D. Ruiz, J.M. Nougues, L. Pulgjaner, "Artificial neural networks applied to online fault diagnosis in chemical plants", *the 7th IEEE International Conference on Emerging Technologies and Factory Automation, 1999. Proceedings. ETFA '99*, Vol. 2, 1999, pp. 977 -986

[21] A. Bernieri, G. Betta, C. Liguori, "On-line fault detection and diagnosis obtained by implementing neural algorithms on a digital signal processor", *IEEE Transactions on Instrumentation and Measurement*, Vol. 45, No. 5, Oct. 1996, pp. 894 -899

[22] S.L. Ho, K.M. Lau, "Detection of faults in induction motors using artificial neural networks", *Seventh International Conference on Electrical Machines and Drives*, 1995, pp. 176 –181

[23] M.J. de la Fuente, P. Vega, "A neural networks based approach for fault detection and diagnosis: application to a real process", *Proceedings of the 4th IEEE Conference on Control Applications*, 1995, pp. 188 -193

[24] Timo Sorsa, Heibki N. Koivo, "Application of artificial neural networks in process fault diagnosis", *Automatica*, Vol. 29, No. 4, 1993, pp. 843-849

[25] Andrea Bernieri, Giovanni Betta, Antonio Pietrosanto, Carlo Sansone, "A neural network approach to instrument fault detection and isolation", *IEEE Transaction on Instrumentation and Measurement*, Vol. 44, No 3, Jun. 1995, pp. 747 –750

[26] K. Madani, "A survey of artificial neural networks based fault detection and fault diagnosis techniques", *International Joint Conference on Neural Networks, IJCNN '99*, Vol. 5, 1999, pp. 3442 -3446

[27] G. Betta, C. Liguori, A. Pietrosanto, "An advanced neural-network-based instrument fault detection and isolation scheme", *IEEE Transactions on Instrumentation and Measurement*, Vol. 47, No. 2, April 1998, pp. 507 –512

[28] G. Betta, C. Liguori, A. Pietrosanto, "Comparison between IFDI schemes based on expert systems and neural networks", *IEEE Transactions on Instrumentation and Measurement*, Vol. 47, No. 5, Oct. 1998, pp. 1106 -1111

[29] R. Isermann, "On fuzzy logic applications for automatic control, supervision, and fault diagnosis", *IEEE Transactions on Systems, Man and Cybernetics*, Part A, Vol. 28, No. 2, March 1998, pp. 221 -235

[30] D. Fuessel, R. Isermann, "Hierarchical motor diagnosis utilizing structural knowledge and a self-learning neuro-fuzzy scheme", *IEEE Transactions on Industrial Electronics*, Vol. 47, No. 5, Oct. 2000, pp. 1070 –1077

[31] E.G. Laukonen, K.M. Passino, V. Krishnaswami, G.C. Luh, G. Rizzoni, "Fault detection and isolation for an experimental internal combustion engine via fuzzy identification", *IEEE Transactions on Control Systems Technology*, Vol. 3, No. 3, Sept. 1995, pp. 347 -355

[32] Wen-Hui Chen, Chih-Wen Liu, Men-Shen Tsai, "On-line fault diagnosis of distribution substations using hybrid cause-effect network and fuzzy rule-based method", *IEEE Transactions on Power Delivery*, Vol. 15, No. 2, April 2000, pp. 710 -717

[33] S. Mofizul Islam, T. Wu, G. Ledwich, "A novel fuzzy logic approach to transformer fault diagnosis", *IEEE Transactions on Dielectrics and Electrical Insulation*, Vol. 7, No. 2, April 2000, pp. 177 -186

[34] S.M. El-Shal, A.S. Morris, "A fuzzy expert system for fault detection in statistical process control of industrial processes", *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Vol. 30, No. 2, May 2000, pp. 281 -289

[35] N.M. Roehl, C.E. Pedreira, H.R. Teles De Azevedo, "Fuzzy ART neural network approach for incipient fault detection and isolation in rotating machines", *IEEE International Conference on Neural Networks, 1995. Proceedings*, Vol. 1, 1995, pp. 538 –542

[36] S. Altug, Mo-Yuen Chow, "Comparative analysis of fuzzy inference systems implemented on neural structures", *International Conference on Neural Networks*, Vol. 1, 1997, pp. 426 - 431

[37] B.N. Huallpa, E. Nobrega, F.J. Von Zuben, "Fault detection in dynamic systems based on fuzzy diagnosis", *IEEE World Congress on Computational Intelligence, The 1998 IEEE International Conference on Fuzzy Systems*, Vol. 2, 1998, pp. 1482 -1487

[38] P.V. Goode, Mo-yuen Chow, "Using a neural/fuzzy system to extract heuristic knowledge of incipient faults in induction motors. Part I-Methodology", *IEEE Transactions on Industrial Electronics*, Vol. 42, No. 2, April 1995, pp. 131 -138

[39] P.V. Goode, Mo-yuen Chow, "Using a neural/fuzzy system to extract heuristic knowledge of incipient faults in induction motors: Part II-Application", *IEEE Transactions on Industrial Electronics*, Vol. 42, No. 2, April 1995, pp. 139 -146

[40] P. Wang, N. Propes, N. Khiripet, Y. Li, G. Vachtsevanos, "An integrated approach to machine fault diagnosis", *Textile, Fiber and Film Industry Technical Conference, 1999 IEEE Annual*, 1999, pp. 7-13

[41] Hong Wang, "Fault detection and diagnosis for unknown nonlinear systems: a generalized framework via neural networks", *IEEE International Conference on Intelligent Processing Systems ICIPS '97*, Vol. 2, 1997, pp. 1506 -1510

[42] A. Srinivasan, C. Batur, "Hopfield/ART-1 neural network-based fault detection and isolation", *IEEE Transactions on Neural Networks*, Vol. 5, No. 6, Nov. 1994, pp. 890 -899

[43] M. Rezai, P.D. Lawrence, M.R. Ito, "Analysis of faults in hybrid systems by global Petri nets", *IEEE International Conference on Systems, Man and Cybernetics*, 1995. Intelligent Systems for the 21st Century, Vol. 3, 1995, pp. 2251 -2256

[44] M. Borairi, H. Wang, "Actuator and sensor fault diagnosis of nonlinear dynamic systems via genetic neural networks and adaptive parameter estimation technique", *Proceedings of the 1998 IEEE International Conference on Control Applications*, Vol. 1, 1998, pp. 278 -282

[45] H. Wang, Y. Wang, "Neural-network-based fault-tolerant control of unknown nonlinear systems" *IEE Proceedings Control Theory and Applications*, Vol. 146, No. 5, Sept. 1999, pp. 389 -398

[46] B. Koppen-Seliger, P.M. Frank, "Fault detection and isolation in technical processes with neural networks", *Proceedings of the 34th IEEE Conference on Decision and Control*, Vol. 3, 1995, pp. 2414 -2419

[47] Yi Lu, Tie Qi Chen, B. Hamilton, "A fuzzy system for automotive fault diagnosis: fast rule generation and self-tuning", *IEEE Transactions on Vehicular Technology*, Vol. 49, No. 2, March 2000, pp. 651-660

[48] P. Balle, R. Isermann, "Fault detection and isolation for nonlinear processes based on local linear fuzzy models and parameter estimation", *Proceedings of the American Control Conference*, Vol. 3, 1998, pp. 1605 -1609

[49] N. Maruyama, M. Benouarets, A.L. Dexter, "Detecting faults in nonlinear dynamic systems using static neuro-fuzzy models", *IEE Colloquium on Qualitative and Quantitative Modeling Methods for Fault Diagnosis*, 1995, pp. 8/1-8/10

[50] F.N. Chowdhury, J.L. Aravena, "A modular methodology for fast fault detection and classification in power systems", *IEEE Transactions on Control Systems Technology*, Vol. 6, No. 5, Sept. 1998, pp. 623-634

[51] R. Burnett, J.F. Watson, S. Elder, "The application of modern signal processing techniques to rotor fault detection and location within three phase induction motors", *Instrumentation and Measurement Technology Conference, 1995. IMTC/95. Proceedings. Integrating Intelligent Instrumentation and Control, IEEE*, 1995, pp. 426-431

[52] J.Y. Keller, M. Darouach, "Fault isolation filter design for linear time-invariant systems", *Proceedings of the 37th IEEE Conference on Decision and Control*, Vol. 1, 1998, pp. 598 – 603

[53] J. Momoh, T. Rizy, "Application of wavelet theory to power distribution systems for fault detection", *International Conference on Intelligent Systems Applications to Power Systems, ISAP '96*, 1996, pp. 345 -350

[54] S.K. Pandey, L. Satish, "Multiresolution signal decomposition: a new tool for fault detection in power transformers during impulse tests", *IEEE Transactions on Power Delivery*, Vol. 13, No. 4, Oct. 1998, pp. 1194 -1200

[55] S. Fornero, N. Kehtarnavaz, M. Swaminadham, D.A. Phillips, "Fourier and wavelet transform features for whirl tower diagnostics", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 4, 1999, pp. 2267 -2270

[56] Shyh-Jier Huang, Cheng-Tao Hsieh, "High-impedance fault detection utilizing a Morlet wavelet transform approach", *IEEE Transactions on Power Delivery*, Vol. 14, No. 4, Oct. 1999, pp. 1401 -1410

[57] L. Satish, "Short-time Fourier and wavelet transforms for fault detection in power transformers during impulse tests", *IEE Proceedings Science, Measurement and Technology*, Vol. 145, No. 2, March 1998, pp. 77 -84

[58] M.E.H. Benbouzid, H. Nejjari, R. Beguenane, M. Vieira, "Induction motor asymmetrical faults detection using advanced signal processing techniques", *IEEE Transaction on Energy Conversion*, Vol. 14, No. 2, June 1999, pp. 147 -152

[59] Hong Guo, J.A. Crossman, Y.L. Murphey, M. Coleman, "Automotive signal diagnostics using wavelets and machine learning", *IEEE Transactions on Vehicular Technology*, Vol. 49, No. 5, Sept. 2000, pp. 1650 -1662

[60] D.R. Hush, C.T. Abdallah, G.L. Heileman, D. Docampo, "Neural networks in fault detection: a case study", *Proceedings of the 1997 American Control Conference*, Vol. 2, 1997, pp. 918 -921

[61] M. Mufti, G. Vachtsevanos, "Automated fault detection and identification using a fuzzy-wavelet analysis technique", *AUTOTESTCON '95. Systems Readiness: Test Technology for the 21st Century. Conference Record*, 1995, pp. 169 -175

[62] J.L. Aravena, F.N. Chowdhury, "A new approach to fast fault detection in power systems", *International Conference on Intelligent Systems Applications to Power Systems, ISAP '96.*, 1996, pp. 328 -332

[63] C.K. Chui, *An introduction to wavelets*, Boston, Academic Press, 1992

[64] L. Debnath, *Wavelet transforms & time-frequency signal analysis*, Boston, Birkhauser, 2000

[65] S. Mallat, *A wavelet tour of signal processing*, 2nd Edition, Academic Press, London, 1999

[66] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis", *IEEE Transaction on Information Theory*, Vol.36, No.5, 1990, pp. 961 - 1005

[67] T. Kohonen, *Self-Organizing Maps*, New York, Springer, 1995

[68] J. Vesanto, E. Alhoniemi, "Clustering of the self-organizing map", *IEEE Transactions on Neural Networks*, Vol. 11, No. 3, May 2000, pp. 586 –600

[69] Mu-Chun Su, Hsiao-Te Chang, "Fast self-organizing feature map algorithm", *IEEE Transactions on Neural Networks*, Vol. 11, No. 3, May 2000, pp. 721 –733

[70] N.B. Karayiannis, "Soft learning vector quantization and clustering algorithms based on ordered weighted aggregation operators", *IEEE Transactions on Neural Networks*, Vol. 11, No. 5, Sept. 2000, pp. 1093 –1105

[71] T. Sabisch, A. Ferguson, H. Bolouri, "Identification of complex shapes using a self organizing neural system", *IEEE Transactions on Neural Networks*, Vol. 11, No. 4, July 2000, pp. 921 –934

[72] V.P. Tuzlukov, *Signal Detection Theory*, Boston, Birkhauser, 2000

[73] A.H. Jazwinski, *Stochastic Processes and Filtering Theory*, New York, Academic, 1970

[74] C.K. Chui, and G. Chen, *Kalman Filtering*, New York, Springer, 1991

[75] D.S. Wall, and F.M.F. Gaston, "Modified extended Kalman filtering", *13th International Conference on Digital Signal Processing DSP 97*, Vol.2, 1997, pp.703-706

[76] K. Reif, and R. Unbehauen, "The extended Kalman filter as an exponential observer for nonlinear systems", *IEEE Transactions on Signal Processing*, Vol. 47 No.8, 1999, pp.2324 - 2328

[77] D.P. Bertsekas, "Incremental least squares methods and the extended Kalman filter", *Proceedings of the 33rd IEEE Conference on Decision and Control*, Vol.2, 1994, pp.1211 - 1214

[78] A.S. Willsky, "Adaptive Filtering and Self-Test Methods for Failure Detection and Compensation", *Proc. of Joint Automatic Control Conference*, pp.637-645, 1974