# INTELLIGENT FAULT DETECTION AND ISOLATION FOR PROTON EXCHANGE MEMBRANE FUEL CELL SYSTEMS

**MAHANIJAH MD KAMAL**

A thesis submitted in partial fulfilment of the requirements of the Liverpool John Moores University for the degree of Doctor of Philosophy

MAY 2014

# ABSTRACT

This work presents a new approach for detecting and isolating faults in nonlinear processes using independent neural network models. In this approach, an independent neural network is used to model the proton exchange membrane fuel cell nonlinear systems using a multi-input multi-output structure. This research proposed the use of radial basis function network and multilayer perceptron network to perform fault detection. After training, the neural network models can give accurate prediction of the system outputs, based on the system inputs. Using the residual generation concept developed in the model-based diagnosis, the difference between the actual and estimated outputs are used as residuals to detect faults. When the magnitude of these residuals exceed a predefined threshold, it is likely that the system is faulty. In order to isolate faults in the system, a second neural network is used to examine features in the residual. A specific feature would correspond to a specific fault. Based on features extracted and classification principles, the second neural network can isolate faults reliably and correctly. The developed method is applied to a benchmark simulation model of the proton exchange membrane fuel cell stacks developed at Michigan University. One component fault, one actuator fault and three sensor faults were simulated on the benchmark model. The simulation results show that the developed approach is able to detect and isolate the faults to a fault size of ±10% of nominal values. These results are promising and indicate the potential of the method to be applied to the real world of fuel cell stacks for dynamic monitoring and reliable operations.

# ACKNOWLEDGEMENT

It is a great privilege and honour to be associated with **Prof. Dingli Yu** for carrying out the research towards my Ph.D degree. His professional guidance, invaluable advice, patience, constant support and encouragement have made this thesis possible. Therefore, I would like to take this opportunity to express my sincere gratitude and thanks to him.

I am also grateful to **Kementerian Sains, Teknologi dan Inovasi** for funding my Ph.D studies through their **LDP scholarship scheme**. My sincere gratitude and thanks also goes to **Universiti Teknology MARA** especially to **School of Electrical Engineering** for their assistance in making this thesis a success.

A special thanks to my **father Md Kamal Haron**, my **mother Wan Rohani Wan Mohamed**, my dearest **husband Aqsa Mohd Othman**, my **son Al-Aqmar Haziq Aqsa**, my **daughters Nur Maisara Aqsa**, **Nur Marissa Aqsa** and **Nur Damia Madihah Aqsa** for their endless support and motivation throughout my Ph.D studies. Without their constant assurance and assistance, completion of this research would have not been possible. And as a sign of my love, gratitude and affection I dedicate this thesis to them appreciating their love and support towards the success of my Ph.D thesis for these four years.

Thank you all.

**Mahanijah Md Kamal**

# TABLE OF CONTENTS

## LIST OF FIGURES

# LIST OF TABLES

# LISTS OF ABBREVIATIONS

| | |
|---|---|
| ANN | Artificial Neural Network |
| ARX | Auto Regressive Exogeneous |
| BP | Back Propagation |
| CM | Compressor motor |
| CV | Compressor motor voltage |
| CVA | Canonical Variate Analysis |
| EIS | Electochemical Impedance Spectroscopy |
| ENN | Elman Neural Network |
| FC | Fuel cell |
| FDD | Fault Detection and Diagnosis |
| FDI | Fault Detection and Isolation |
| FDIR | Fault Detection Isolation and Reconfiguration |
| IFAC | International Federal of Automatic Control |
| IM | Impedance Spectroscopy |
| MIMO | Multi Input Multi Output |
| K2 | Bayesian score |
| LPV | Linear Parameter Varying |
| MCMC | Markov Chain Monte Carlo |
| MIMO | Multi Input Multi Output |
| MLP | Multi Layer Perceptron |
| MSE | Mean Squared Error |
| NARMAX | Nonlinear Auto Regressive Moving Average with Exogeneous Input |

| | |
|---|---|
| NARX | Nonlinear Auto Regressive Exogeneous |
| NP | Net Power |
| OCV | Open Current Voltage |
| PID | Proportional Integral Derivative |
| PEM | Proton Exchange Membrane |
| SC | Stack Current |
| SV | Stack Voltage |
| R&D | Research & Development |
| RAS | Random Amplitude Signals |
| RBF | Radial Basis Function |
| RLS | Recursive Least Squares |
| ROLS | Recursive Orthogonal Least Squares |

# CHAPTER 1

# INTRODUCTION

## 1.1 Background

Early detection and isolation of faults are critical tasks in modern process industries. Many research works have been made during last three decades to improve fault detection and isolation (FDI) methods in terms of preventing and monitoring process. Several papers have been published considering the fuel cell (FC) operations in normal conditions but few of them addressed the FC under fault conditions. Faults are events that cannot be ignored in any real machine and their consideration is essential to improving the operability, flexibility and autonomy of automatic equipment (Riascos *et al*. 2008). Nowadays, there is a great demand and interest in the renewable energy technology which has provided motivation for researchers to conduct research in these technologies. Among these technologies, proton exchange membrane (PEM) FC has received a lot of attention because of its importance in global energy conversion and also the positive impact towards the environment. FCs are rapidly becoming a potential source of power in the future due to high electrical efficiency, flexibility and capacity, long lifetime and zero pollutions. Proton exchange membrane fuel cell (PEMFC) is based on hydrogen technology and operates at low temperatures between the ranges of 60°C-100°C. Therefore, these characteristic allow the use of PEMFC in many applications such as transportation, telecommunication, portable utilities and power generator (Kamal & Yu, 2012).

If faults occur in a process plant, they will affect the productivity, quality, safety and performance of the control systems of the plant. Therefore, early detection of possible faults would minimize the downtime, increase the safety of plant operations, prevent damage to the equipment, minimize the operation cost and also the maintenance. The PEMFC system model is used to implement FDI approach develop in this work. The simulator model of PEMFC built under MATLAB/Simulink environment was proposed in Pukrushpan *et al*. (2004a) where it is widely accepted nowadays in the control community as a good representation of the behaviour of an actual FC for control purposes. The main components considered in the system as shown in Fig. 1.1 are the air compressor, the air manifold, the FC stack, the anode manifold and the return manifold. A control system is needed to ensure that the temperature, pressure and humidification are within prescribed limits during their operations. Also, the flow of air should be control via air compressor to follow the flow of hydrogen to match the vary load current.

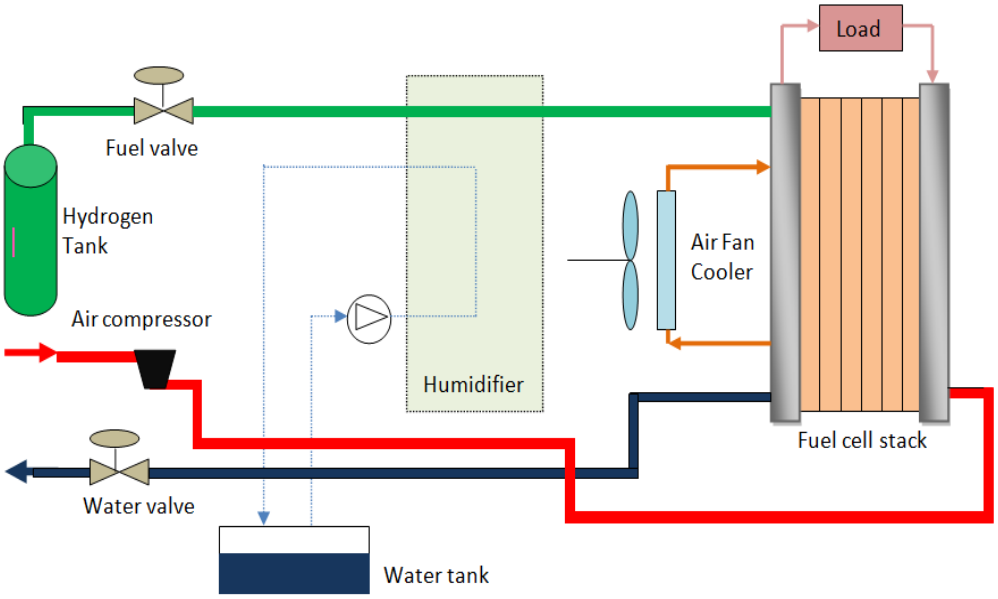


**Fig. 1.1** The diagram of FC system proposed by Pukrushpan (2004a)

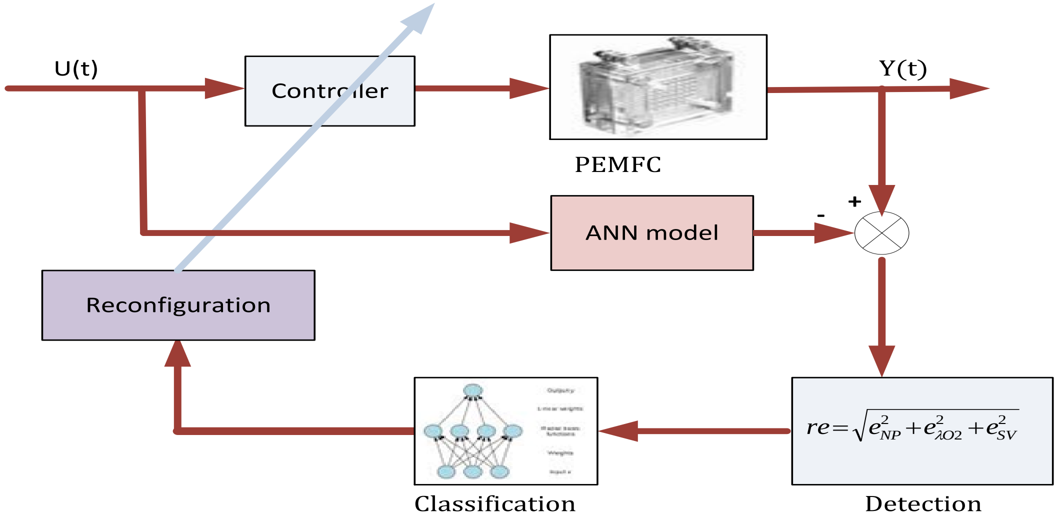


**Fig. 1.2** Fault detection, diagnostics and reconfiguration control system structure

In some cases, if a fault can be quickly detected and identified, appropriate reconfiguration control actions may be taken. Fault detection isolation and reconfiguration (FDIR) is a control methodology which ensures continual safe and acceptable operation of the system when a fault occurs through FDI. Many devices depend on automatic control for satisfactory operation, while assuring stability and performance with all components functioning properly. If the control system's structure or parameters can be altered in response to system failure, it is said to be reconfigurable (Hwang *et al*., 2010).

The purpose of a control system is to maintain a satisfactory operation with respect to disturbances and changes occurring in the process. FDI in control engineering concerns with a systems, identifying when a fault has occurred and then pinpoint the types of fault and its location. To achieve this a robust controller should be used to handle this kind of situation where it should be intelligent enough to respond and react to faults when they happen. With this intelligent FDI approach, the

PEMFC nonlinear dynamic system is intelligent to detect the faults quickly, isolate them and appropriate control action can be taken immediately. Fig. 1.2 shows the proposed intelligent FDI scheme in order to reconfigure with faults during the operation of the process plant.

## 1.2 Research motivations

PEMFC have attracted great attention in recent years as a promising replacement for traditional stationary and mobile power sources, especially due to their high power density and low greenhouse gas emissions. However, a number of fundamental problems must be overcome to improve their performance and to reduce their cost (Riascos *et al.*, 2008). The critical issue with the FDI for dynamic control systems is the nonlinear plants behaviour which plays a very important and challenging task in many engineering applications and attracts a lot of research opportunity in this area. The purpose of control systems is to maintain a satisfactory operation with respect to disturbances and changes occurring in the process. However, before control action can take place, appropriate fault diagnostic should tackle the faults happened during the operation of process plant. FDI in a control engineering concerns with a changing in the systems outputs, identifying when a fault has occurred and then pinpoint the type of fault and its location. By doing this, the information fed into the controller is more reliable and believable.

Another factor which inspired this research is the environment issue which have increased the demand for less polluting energy generation technologies. Major efforts to reduce greenhouse gas emissions have increased the demand for pollution-free energy sources, in the last few years. Governmental and private sector investment in R&D to support for clean energy generation including hydrogen based which now actively being conducted.

## 1.3 Aims and objectives of the research

The aim of this research is to develop an intelligent FDI scheme for PEMFC nonlinear dynamic systems using artificial neural network (ANN) modelling approach. The objectives of this research are:

i) To propose a new intelligent FDI approach for PEMFC nonlinear dynamic systems.

ii) To develop the FDI using independent neural network model.

iii) To investigate the effectiveness of RBF algorithm and MLP algorithm under healthy and unhealthy scenarios.

iv) To investigate the performance of RBF network and MLP network under feed-forward and feedback control.

v) To develop RBF classifier to perform fault isolation for the PEMFC.

## 1.4 Contributions to knowledge

The fundamental contribution of this work is to develop independent neural networks for RBF network and MLP network model. An algorithm based on RBF network and MLP network has been developed and the FDI techniques are used to perform fault diagnosis. These independent models are used to evaluate five faults simulated in the benchmark model of the PEMFC systems under open-loop and closed-loop condition. The main contributions of this research is firstly, the way of the independent neural network model dedicated to fault detection which is new in this research area.

Secondly, the developed independent neural network model-based FDI is applied to the PEMFC stack system. All the five simulated faults including one actuator fault, one component fault and three sensor faults have been successfully detected and isolated. The residual is designed sensitive to the fault amplitude as low as ±10% of their nominal values.

## 1.5 Thesis Organisation

In this thesis, the structure of work to complete in achieving the above objectives is outlined as follows:

Chapter 1: Introduction to the study. The background, motivations, objectives, contribution of knowledge and achievements are presented.

Chapter 2: The literature studies relating to this work and the selection of the methods used is briefly explained. Definition and techniques of model-based FDI were discussed and compared.

Chapter 3: The PEMFC nonlinear dynamic systems and the modelling are described here. In this chapter, the block diagram and the operation of FC stack is explained. The PEMFC systems consist of several subsystems in order to operate.

Chapter 4: The theory of ANN involving the RBF network and MLP network models are explained. In this chapter, the structures, the training algorithm used and the data selection have been described for both methods.

Chapter 5: The strategy of FDI and system configuration is discussed in this chapter. The design of feedforward and feedback control is explained here. The independent neural network, residual generation and fault isolation are clearly explained in here.

Chapter 6: This chapter tells the approach of simulating faults in open-loop and closed-loop control scheme. It also explained the type of inputs used during the simulation. RAS and step signals are fed as inputs while collecting three outputs from the process plant.

Chapter 7: Definition of fault and simulation results is presented in this chapter. Discussion of results obtained during the simulation for open-loop and closed-loop control were discussed.

Chapter 8: The theory of fault isolation based on RBF classifier is explained in details under this chapter. The structured residuals are trained using the RBF networks before fault isolation can be performed.

Chapter 9: Provides the conclusions of this thesis and a proposal of future work to continue the research in the subject.

# CHAPTER 2

# LITERATURE REVIEW

## 2.1 Introduction

Fault detection, isolation and reconfiguration (FDIR) is an important and challenging task in many engineering applications and continues to be an active area of research in the control environment (Hwang *et al*., 2010). In some cases, if a fault can be quickly detected and identified, appropriate reconfiguration control action may be taken. FDIR is a control methodology which ensures continual safe and acceptable operation of the system when a fault occurs through FDI. Many devices depend on automatic control for satisfactory operation, and while assuring stability and performance with all components functioning properly. If the control system's structure or parameters can be altered in response to system failure, it is said to be reconfigurable (Stengel, 1990).

Fault diagnosis is one of the important tasks in safety-critical and intelligent control systems (Patton *et al*., 1994). Fault diagnosis in technical systems has received a lot of theoretical and practical attention over the last few decades. Diagnosis is a complex reasoning activity, which is currently one of the domains where artificial intelligence techniques have been successfully applied as these techniques use association rule, reasoning and decision making processes as would the human brain in solving diagnostic problems. The problem of fault detection and diagnosis (FDD) in dynamic systems has received considerable attention in last decades due to the growing complexity of modern engineering systems and ever increasing demand for fault tolerance, cost efficiency, and reliability (Willsky, 1976; Basseville, 1988). Existing FDD approaches can be divided into two categories including model-based and knowledge-based approaches (Venkatasubramanian *et al.*, 2003a; Venkatasubramanian *et al.*, 2003b). Model-based approaches make use of the quantitative analytical model of the physical system. Knowledge-based approaches do not need full analytical modelling and allow one to use qualitative models based on the available information and knowledge of a physical system. Whenever the mathematical models describing the systems are available, analytical model-based methods are preferred because they are more amenable to perform analysis (Zhou *et al.*, 2011).

## 2.2 FDI methods and approaches

Mostly in fault diagnosis, many researchers are interested in model-based approach. The basic idea behind model-based fault diagnosis is the generation of residuals, consisting of the difference between the process plant output and the estimated model output. Residuals for diagnosis may be generated based on estimated outputs, inputs and parameter of the plant. It is well known that the core element of model-based fault detection in control systems is the generation of residual signals which act as an indicator of faults. Various methods have been proposed to carry out such as estimation problems, including observers, parity equations and various parameter estimation schemes. The model-based approach to FDI in automated processes has received considerable attention during the last three decades, both in research and application (Willsky, 1976; Isermann 1984; Gertler 1988; Patton *et al.*, 1989; Frank, 1990; Patton and Chen, 1991b; Patton, 1994). In model-based fault detection a model (mathematical or heuristic) is employed to describe the nominal behaviour of the monitored system. The generated residual signals that indicate differences between the model's output and measured process output are interpreted and evaluated to isolate faults (Angeli, 2010).

### 2.2.1 Observer-based approach

The observer-based method requires a model of the investigated process. Observer-based approaches have been the most widely considered (Chen *et al*., 1996; Ding & Frank, 1990; Ding *et al*., 1994; Duan, Patton *et al*., 1997; Frank, 1990; 1994; Ge & Fang, 1988; Gertler, 1989, 1991; Hou & Müller, 1992, 1994; Patton & Chen, 1991, 1997). Patton et al. (1989) and Frank (1987) used the observer-based to generate diagnostic signals which are residuals. In the framework of FDI, faults are detected by setting a fixed or variable threshold on each residual signal. A number of residuals can be designed, each having special sensitivity to individual faults occurring in different locations in the systems.

The basic idea behind the diagnostic observer approaches is to estimate the outputs of the systems from the measurements (or a subset of measurements) by using either Luenberger observers in a deterministic setting Beard (1971), Jones (1973) or Kalman filters is a stochastic setting (Mehra & Peschon, 1971; Willsky & Jones, 1976; Basseville & Benvenista, 1986). Then the weighted output estimation error is used as a residual. Thereby, while flexibility in selecting the observer gains is used to minimize the noise effect on the FDI properties in the Luenberger approach. As a result, the dynamics of the fault response can be controlled, within certain limits, by placing the poles of the observers. This trend was followed by a long line of researchers, including Frank & Keller (1980); Viswanadham & Srichander (1987), Patton et al., (1989), Patton and Chen (1999) and Puig *et al*., (2003).

### 2.2.2 Kalman Filter approach

The Kalman filter is a set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of the process, in a way that minimizes the mean of squared error. The filter is very powerful in several aspects: it supports estimations of past, present and even future states (Welch and Bishop, 1995). Kalman filters have been developed in order to detect, isolate and estimate an accurate state of a system in presence of faults/failures when a model is defined around operating point (Li & BarSholom, 1993; Maybeck, 1999). In Diao and Passino (2002), a multi-model strategy is developed where each model represents a particular fault in the system. More recently, effectiveness of a multi-model approach for real industrial systems for fault diagnosis (Bhagwat *et al*., 2003; Gatzke & Doyle, 2002) and for control purposes (Athans *et al*., 2005; Porfirio *et al*., 2003) have been demonstrated under the assumptions that weighing functions of models are not affected by faults. Rodrigues et al., (2008) developed a multi-model of a dynamic hydraulic system for fault diagnosis according to a faulty multi-model representation based on robust weighing functions generation through decoupled Kalman filters developed by Keller (1999) in linear case.

### 2.2.3 Parity space approach

Several survey papers have been published on the parity relation (or parity equation) based fault detection methods of the 1990's (Gertler, 1997; Patton, 1988; Patton & Chen, 1991, 1994; Patton *et al*., 1989). Parity relations are rearranged direct input-output model equations, subjected to a linear dynamic transformation. The transformed residuals serve for detection and isolation. The parity relation approach generate the residual based upon consistency checking on system input-output data was originally proposed by Mironovski (1979,1980).The parity space approach is based on the test (parity check) of the consistency of parity equations by using the measured signals of the actual process. The modification of the system equations aims at the decoupling among different faults to enhance their diagnose ability. From the inconsistency (residuals) of the parity equations one can detect the faults (Frank, 1996). Chow and Willsky (1984) derived the parity equations from the state space model of the system. Further contributions focusing on transfer function relations are due to Gertler and his coworkers (1990; 1991; 1995), Delmaire *et al*., (1994) and Staroswiecki *et al*., (1993). More development regarding parity relation approaches can be found in Patton and Chen (1999) and Ploix & Adrot (2006). Yu *et al*. (1995); Yu & Shields (1997) used the parity equations of FDI for bilinear systems with unknown inputs where a wide selection of faults are detected and isolated.

### 2.2.4 Parameter estimation approach

Another FDI approach is the use of parameter estimation which is based directly on system identification technique. In 1984, Isermann illustrated that process FDI can be achieved using the estimation of non-measurable process parameters and/or state variables in his survey paper. Parameter estimation is a natural approach to the detection and isolation of parametric (multiplicative) faults. A reference model is obtained by first identifying the plant in fault-free situation. Then, the parameters are repeatedly re-identified on-line. Deviations from the reference model serve as a basis for detection and isolation. The parameter estimation method requires knowledge of the model structure of the investigated model structure and actual process measurements (Sneidar and Frank, 1996). This approach based on the assumption that the faults are reflected in the physical system parameters. A most common approach to parameter estimation is that of using least squares methods. In most practical cases the process parameters are not known at all, or are not known exactly enough. They can be determined with parameter estimation methods measuring input and output signals if the basic structure of the model is known. The parameter estimation includes equation error methods. The latest development and applications can be found in Isermann (1997), Ingimundarson et al. (2005a, 2005b).

## 2.3 Intelligent methods

Since the late 1980s, artificial neural networks (ANNs) have been widely discussed for model-based FDI. ANN is a mathematical tool, which tries to represent low-level intelligence in natural organisms and it is a flexible structure, capable of making a nonlinear mapping between input and output spaces (Rumelhart *et al.*, 1986). A well-designed ANN model provides useful and reasonably accurate input-output relations because of its excellent multi-dimensional mapping capability. ANNs are computational paradigms made up of massively interconnected adaptive processing units, known as neurons (Ou & Achenie, 2005). They have been extensively employed in various areas of science and technology, such as pattern recognition, signal processing and process control in engineering.

### 2.3.1 FDI with neural networks

The potential of neural networks for FDI in nonlinear systems has been demonstrated in recent years. Neural networks provide an excellent mathematical tool for dealing with nonlinear problems (Narendra and Parthasarthy, 1990), as they have the important property that any nonlinear function can approximated by a neural network given suitable weighting factors and architecture (Patton *et al.*, 1990). A neural network has the ability to learn from the training history using the inputs of the systems where the learning algorithm is set without the knowledge of the process. This provides a great flexibility for modelling nonlinear systems whilst the learning can be carried on-line which is very useful for on-line fault diagnosis (Patton *et al.*, 1990). Neural networks have been proposed for classification and function approximation problems. In general, neural networks having been used for fault diagnosis can be classified into two: (i) the architecture of the network such as sigmoid, radial basis and so on; and (ii) supervised and unsupervised learning (Venkatasubramanian *et al.*, 2003b).

#### 2.3.1.1 Radial basis function neural network

RBF networks have a very strong mathematical foundation rooted in regularization theory for solving ill-conditions problems. Such networks, consists of three layers: an input layer, a hidden layer and an output layer. As its name implies, radially symmetric basis function is used as activation functions of hidden nodes. The transformation from the input nodes to the hidden node is nonlinear while the hidden node to the output is a linear transformation (Kashaninejad *et al*., 2009). Fault diagnosis for chemical reactor was investigated by Yu *et al*. (1999) where the RBF networks is used to generate residuals for diagnose the sensor faults in the reactor uses the output prediction of the dependent and semi-independent neural network model and a nonlinear dynamic process. The RBF network has also been used as a classifier which shows a satisfactory fault analysis. The continuous stirred-tank reactor (CSTR) process with multiple-inputs multiple-outputs (MIMO) has also been investigated by Yu *et al*. (2005) using the RBF networks to model the nonlinear CSTR system which shows the effectiveness of the method. The RBF network has been used as the estimator in their work and levenberg-marquart algorithm by Chong and Zak (2013) as updating algorithm. This method has been widely recommended for further research, but its drawback is that the difficulty of developing an accurate nonlinear state space model for the plant makes it not easy for real applications.

#### 2.3.1.2 Multilayer perceptron neural network

There are a number of papers that address the problem of FDD using back-propagation (BP) neural networks. The MLP network sometimes known as BP network and quite popular in engineering problems due to its nonlinear mapping. It consists of an input layer, a hidden layer and an output layer. Each one collects the input from all input nodes after multiplying each input value by weight, attaches a bias to this sum, and passes on the results through the nonlinear transformation like the sigmoid transfer function (Kashaninejad *et al*., 2009). In chemical process, Gomm *et al*. (1996) used the MLP model and simulated the nonlinear dynamic system which shows a significant improvement in the performance. Patton et al. (1994b) proposed an approach for detecting and isolating faults in a nonlinear dynamic process where the MLP network was trained to predict the future system states and again used a neural network as a classifier to isolate faults from these state prediction errors.

### 2.3.2 FDI with fuzzy logic

The theory of fuzzy logic is aimed at the development of a set of concepts and techniques for dealing with sources of uncertainty, imprecision, or incompleteness (Zadeh, 1971; Yager, 1987 and Zimmermann, 1992). Fuzzy systems have been successful in many applications including control system when gradual adjustments are necessary (Ayoubi, 1995; Nomura et al., 1992; Rhee & Krishnapuram, 1993; Takagi & Sugeno, 1985 and Kang, 1993). Fuzzy logic systems handle the imprecision of input and output variables directly by defining them with fuzzy memberships and sets that can be expressed in linguistic terms (Uraikul *et al*., 2007). Recently, fuzzy theory is used in many technical disciplines taking care of vague descriptions. The fuzzy approach is used to build an adaptive fuzzy threshold which take cares of modelling errors, so that no increased threshold is necessary and even small faults can quickly be detected (Schneidar and Frank, 1994). Schneidar and

Frank (1994) applied an observer-based fault detection using a dynamic robot model and used fuzzy logic to generate residual. Frank and Seliger (1997) studied fuzzy logic and neural network applications for fault diagnosis. They used a dependent neural network for residual generation and fuzzy logic for residual evaluation.

### 2.3.3 FDI with expert systems

An expert system is a software system that captures human expertise for supporting decision-making; this is useful for dealing with problems involving incomplete information or large amounts of complex knowledge. Expert systems are particularly useful for on-line operations in the control field because they incorporate symbolic and rule-based knowledge that relate situation and action(s), and they also have the ability to explain and justify a line of reasoning (Chiang *et al*., 2001). A common application of expert system technology in process control is for fault diagnosis. Typically, the basic components of an expert system include a knowledge base, an inference engine and user interface. The knowledge base contains either shallow knowledge based on heuristics, or deep knowledge based on structural, behavioral or mathematical models (Chiang *et al*., 2001). Various types of knowledge representation schemes can be used, including production rules, frames, and semantic networks (Xia & Rao, 1999). Since performance of the expert system is highly dependent on the correctness and completeness of the information stored in the knowledge base, updates to the knowledge base is necessary should the industrial process changes. The inference engine provides inference mechanisms to direct use of the knowledge, and the mechanisms typically include backward and forward chaining, hypothesis testing, heuristic search methods, and meta-rules (Prasad *et al*., 1998; Norvilas *et al*., 2000; Rao *et al*., 2000). Finally, the user interface translates user input into a computer understandable language and presents conclusions and explanations to the user. Early applications of expert systems primarily focused on medical diagnosis (Clancey and Shortliffe, 1984). Currently, expert systems have been adopted in many industrial applications, including equipment maintenance, diagnosis and control, plant safety, and other areas in engineering.

## 2.4 Fault detection and isolation for FC systems

There are a large number of publications on FC studies, but studies on FDI of FC systems are still few. Model-based FDI methods for PEMFC become more and more important because it involves the comparison between the observed behavior of the process with a reference model. Model-based approach gives the insight analysis of the subsystem interactions and also provides guidelines during the conduction of the experiment. The system behavior can been analyzed in deep understanding and later this information can be used for future design and development.

### 2.4.1 Model-based failure monitoring and diagnosis

Most approaches for FDI in some sense involve the comparison of the observed behavior of the process to a reference model. In the aspect of failure monitoring and diagnosis for PEMFC, some papers have been published. Ingimundarson *et. al*. (2005) considered about the safety for FC systems and presented an approach to detect hydrogen leaks. The method is applicable during setup and shutdown as well as normal conditions. The method relies on simple mass equations and takes into account the natural leak of the stack and humidity. Two approaches have been presented to detect this problem, one based on decoupling effect by measuring vapor pressure directly with relative humidity sensors. The other is to use the vapor pressure above the saturation pressure in order to create an adaptive alarm threshold. Xue *et al*. (2006) proposed a model-based condition monitoring scheme that employs the Hotelling $T^2$ statistical analysis for fault detection of PEMFC. This model-based robust condition monitoring scheme can deal with the operating condition variation, various uncertainty in a FC system, and measurement of noise.

A long life time, durability and safety are important issues when PEMFC is used as power supply. Tian *et al*. (2008) presents some test data and descriptions related with two FC stacks that failed due to internal gas leakages. Too low humidification levels and too high stack temperatures, insufficient reactant gas flows and bad control of pressure gaps between anode and cathode can lead to severe and irreversible damages in PEMFC and break in the membrane. Some electrical signals which reflect the faults are generated. The open current voltage (OCV) of the cells are monitored in different operating conditions linked with the gas supply (flow rate and pressure). The procedures allow the detection of the failed cells, which show abnormal performance and voltage pattern when compared to the other cells in the stack. However this technique is only applicable to low power PEM three-cell stack with power levels of 100W test and 5kW test bench. While, Buchholz *et al*.

(2008) shown that linear canonical variate analysis (CVA) which is a state-space models with a relatively low order are able to present the non-linear behavior of a PEMFC stack well in the regarded operating range for load cycles similar to driving cycles of vehicles. Additionally, two concept, a Kalman filter and inverse model, were introduced which show how CVA state-space models can be used for diagnosis for non-measurable inputs. Both concepts were far studied for the oxygen stoichiometry. The Kalman filter has been shown to work as FDI for lower values than normal, the inverse model approach showed good diagnosis result for all failure models of the oxygen stoichiometry.

In order to improve the efficiency of PEMFC, some investigation has been done. Lebbal and Lecoeuche (2009) developed a PEM electrolyser model, an identification approach and a diagnosis algorithm. It consists of a steady-state electrical model and a linear dynamic thermal model. The identification algorithm uses a nonlinear least squares method for the estimation of the electrical model and the thermal model parameters are estimated using first order response properties. This diagnosis approach can be used to avoid electrode destruction, membrane melting, membrane drying, electrode pressure argumentation, membrane hot-point and membrane tear. Ibrir and Cheddie (2009) proposed a model-based observer, which allows for the monitoring of the partial pressures of hydrogen, oxygen assuming that the water vapor during transient FC operation is known. It is possible to estimate the vapor pressure in the anode and cathode compartments from the temperature measurements. The humidity or partial vapor pressures of the inlet hydrogen and oxygen gas streams need to be carefully monitored and managed in order to maintain water balance especially during transient operation where the load changes continually. Recently, fault detection systems based on residual generation have been designed using the dynamic neural network. Escobet *et al*. (2009) presented and tested a model-based fault diagnosis methodology based on the relative residual fault sensitivity. The principle of model-based fault detection is to check the consistency of observed behaviour while fault isolation tries to isolate the component that is in fault. Furthermore, it allows faults to be isolated although all the considered faults affect all the residuals whenever the sensitivities were different. De Lira *et al*. (2009) proposed a robust model-based fault diagnosis method for nonlinear system that can be approximated using a linear parameter varying (LPV) model. Robust fault detection is based on the use of an internal LPV observer that bounds the effect of parametric uncertainty in the residual using output zonotopes. Fault isolation is based on set of structured residuals that are analyzed using a relative fault sensitivity approach.

### 2.4.2 On-line diagnosis and monitoring

Several papers have been published considering the FC operation in normal conditions, but few of them addressed the FC operation under fault conditions. Riascos *et al*. (2006) considered the effects of different types of fault on a model of PEMFC such as  faults in the air reaction fan, faults in the cooling system, growth of the fuel crossover and internal loss current, and faults in the hydrogen feed line. Here, Bayesian network algorithms are applied to construct a graphical-probabilistic model for fault diagnosis in PEMFCs from databases of fault records design by a supervisory system. For the construction of a network structure, the probabilistic approaches such as the Bayesian score (K2) and markov chain monte carlo (MCMC) algorithms have been implemented. Later, Morales *et al*. (2008) introduced a supervisor system which is able to diagnose different types of faults during the online operation of PEMFC. The execution of the diagnosis was based on Bayesian network, which qualifies and quantifies the cause-effect relationship within the variables. By doing this, an early alert of an incipient faults allow decisions to be made to avoid degradation of other components and catastrophic faults in the equipment.

The other researchers focused on on-line detection of FC dysfunction in embedded applications using impedance spectroscopy (IS). Again, Quan *et al*. (2009) proposed a methodology based on electrochemical impedance spectroscopy (EIS) to guide the hardware and software design of on-line monitoring system of PEMFC internal resistance. The ohmic resistance of PEMFC is mostly decided by PEM's water content, which can directly affect the efficiency of electricity generation, so it is important for the measurement and control of PEMFC's humidity by means of monitoring its internal resistance.  Bethoux *et al*. (2009) presented a new approach to obtain actual FC system an on-line diagnosis in order to allow a robust control strategy for well monitoring operating cells conditions. The on-line monitoring is based on the impedance spectroscopy using Randles circuit consist of a couple of resistors and a capacitor for validation. Rosich *et al*. (2013) focus on the design of on-line FDI based on residual generators to tackle four sensors obtained during the simulation in Simulink environment.

### 2.4.3 Artificial intelligence

ANNs have been suggested as a possible data-based technique to cope with the robustness problem in FDI systems, designed especially for complex processes where analytical models are not or rarely available (Vaidyanathan and Venkatasubranian, 1992; Sorsa and Koivo, 1993; Frank, 1996, Isermann and Bellè, 1997). ANNs have been used as both predictors of dynamic nonlinear models for residual generation and pattern classifiers for symptom evaluation.

In order to tackle the nonlinear systems behavior, neural network has been used widely in the fault detection for FC systems. Bayesian network is used as an early alert to diagnose faults in the air reaction fan, faults in the cooling system, growth of the fuel crossover and internal loss current, and faults in the hydrogen feed line (Riascos *et al*., 2008). Alternatively, to improve reliability and durability of PEMFC systems, Steiner *et al*. (2009, 2010, 2011) presented a flooding diagnosis based on black-box model. Here, a model-based diagnosis procedure using the comparison between measured and calculated pressure drops is introduced. The model based on elman neural network (ENN) is trained with data recorded in flooding-free condition and the difference between calculated and experimental pressure drop is residual.

## 2.5 Summary

FDI for the PEMFC systems is challenging due to its nonlinear nature. Thus, a method or approach needs to be developed which can tackle the problems in a simple and effective way. This is the motive of this research. Based on the literature reviews above, the most common and popular method used by researchers in order to do FDI are the model-based method. Therefore, considering about the cost, safety, efficiency and reliability of PEMFC performance, the best way to implement FDI is by using neural network to model the FC stacks in independent mode, and generate the residual. Model-based approach gives the insight analysis of the subsystem interactions and provides guidelines during system. The system behavior can be analyzed for deep understanding and this information can be used for future design and development. The use of neural network will overcome the nonlinear behavior of PEMFC dynamic systems. Therefore, in this PhD thesis, the contribution is to develop an intelligent FDI scheme for PEMFC dynamic systems using a neural network. By acquiring process data with different input signals when the systems is in its using healthy and faulty conditions, the neural network models will be trained and tested, to detect and isolate all possible faults in the PEMFC dynamic systems.

# CHAPTER 3

# PROTON EXCHANGE MEMBRANE FUEL CELL DYNAMIC SYSTEMS

## 3.1 Introduction

FCs are prime candidates for power generation devices in the future. Their potential applications include stationary power generation, distributed power generation, transport applications, portable applications and also telecommunications. A well-known simulator bench of PEMFC developed by University of Michigan (Pukrushpan *et. al*, 2004a) is adopted in this research to study and analyse the behaviour of PEMFC. This simulator is constructed using MATLAB/Simulink model which is widely used by the researchers especially in the area of control engineering. A FC consists of two electrodes; a negative electrode (anode) and a positive electrode (cathode) separated by an electrolyte. FCs convert the chemical energy of the hydrogen fuel (on the anode side) into electric energy while through a chemical reaction with oxygen (on the cathode side) produced water and heat as end product. Hydrogen atoms separate into protons and electrons once the chemical reaction happens. The electrons go through the load which contains a flow of electricity while the protons migrate through the electrolyte to the cathode side, where they reunite with oxygen to produce water and heat as shown in Fig. 3.1.

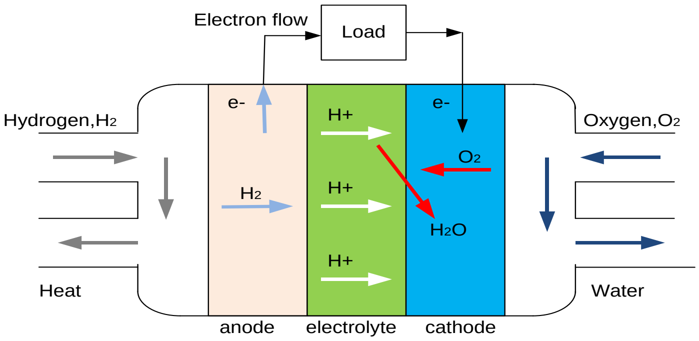


**Fig. 3.1** PEMFC chemical reaction

## 3.2 PEMFC dynamic model

The PEMFC systems offer high efficiency, low emission of carbon monoxide and has been become popular as an alternate power source for various application such as transportation, telecommunication, portable utilities, stationary and power generation. The FC stack needs to be integrated with other components to form a FC stack system. The diagram in Fig. 3.2 illustrates the components required to form a FC stack which consists of four main subsystems (Pukrushpan, 2003). The subsystems are:

i) Hydrogen supply system.

ii) Air supply systems.

iii) Cooling system.

iv) Humidification system.

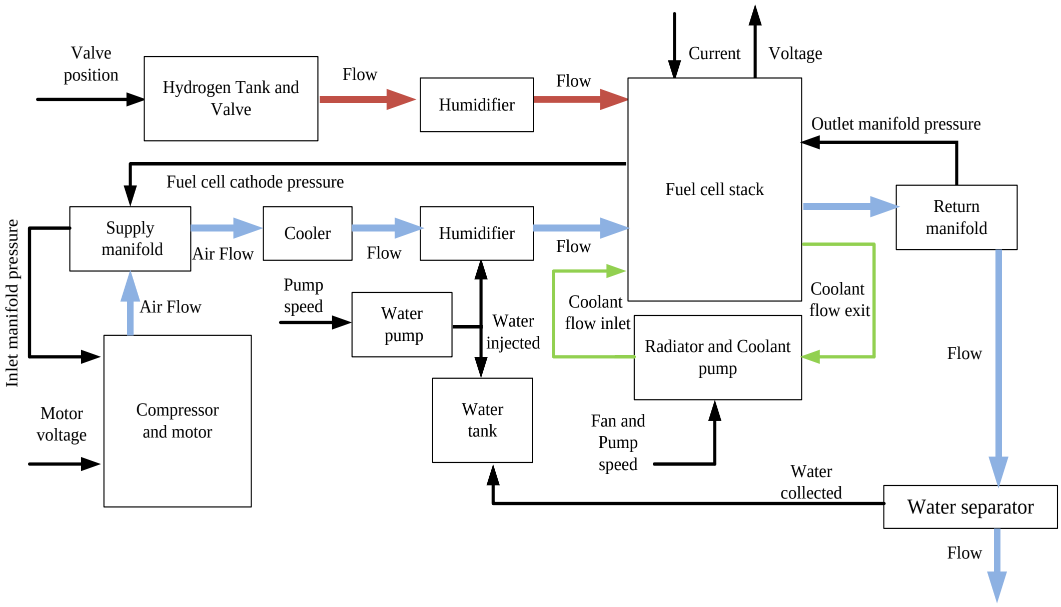


**Fig. 3.2** The PEMFC system block diagram (Pukrushpan, 2003)

The power of the FC stack depends on current drawn from the stack and stack voltage. The FC voltage is the function of the current, reactant partial pressure, temperature and membrane humidity. As the current is drawn from the FC, oxygen, $O_2$ and hydrogen, $H_2O$ are used in the reaction. Water and heat also generated. To maintain the desired $H_2O$ partial pressure, the $H_2O$ needs to be replenished by $H_2O$ supply system, which includes $H_2O$ pressurized tank and $H_2O$ control valve. Similarly, the air supply system needs to replenish the $O_2$ to maintain the $O_2$ partial pressure. The air supply is composed of compressor, electric motor and pipe between the components. The compressor compress air flow to high pressure which significantly improves the reaction rate, and thus FC efficiency and power density. Since the high pressure air flow leaving the compressor has high temperature, air cooler is needed to reduced air temperature entering the stack. Humidifier is also used to humidify the air and $H_2O$ flow in order to prevent dehydration of the FC membrane. The water used in the humidifier is supplied from the water tank. Water level in the water tank needs to be maintained by collecting water generated in the stack, which is carried out with the air flow. The excessive heat released in the FC reaction also needs to be removed by the cooling system, which re-circulates de-ionized water through the FC stack (Pukrushpan, 2003).

**Table 3.1** Model parameters used in PEMFC dynamic systems

| Symbol | Variable | Value |
|---|---|---|
| $\rho_{m,dry}$ | Membrane dry density | 0.002 kg/cm$^3$ |
| $M_{m,dry}$ | Membrane dry equivalent weight | 1.1 kg/mol |
| $t_m$ | Membrane thickness | 0.01275 |
| n | Number of cells in stack | 381 |
| $A_{fc}$ | Fuel cell active area | 280 cm$^2$ |
| $d_c$ | Compressor diameter | 0.2286 |
| $J_{cp}$ | Compressor and motor inertia | 5 x 10$^{-5}$ kg.m$^2$ |
| $V_{an}$ | Anode volume | 0.005 m$^3$ |
| $V_{ca}$ | Cathode volume | 0.01 m$^3$ |
| $V_{sm}$ | Supply manifold volume | 0.02 m$^3$ |
| $V_{rm}$ | Return manifold volume | 0.005 m$^3$ |
| $C_{D,rm}$ | Return manifold throttle discharge coefficient | 0.0124 |
| $A_{T,rm}$ | Return manifold throttle area | 0.002 m$^2$ |
| $k_{sm,out}$ | Supply manifold outlet orifice constant | 0.3629 x 10$^{-5}$ kg/(s.Pa) |
|  | Anode outlet flow constant | 3.9320 x 10$^{-9}$ kg(s.Pa) |
| $k_{ca,out}$ | Cathode outlet orifice constant | 0.2177 x 10$^{-5}$ kg(s.Pa) |
| $k_v$ | Motor electric constant | 0.0153 V/(rad/s) |
| $k_t$ | Motor electric torque | 0.0153 N-m/A |
| $R_{cm}$ | Compressor motor circuit resistance | 0.816 Ω |
| R | Air gas constant | 286.9 J/(kg*K) |
| T | Temperature | 353Kelvin |

The PEMFC stack is made up of 381 cells with an active area of 280 cm$^2$ and the stack operating temperature is at 80°C developed by University Michigan is used as a benchmark model. The parameters used in the model are given in Table 3.1. Most of the parameters are based on the 75 kW stacks used in the FORD P2000 FC prototype vehicle (Grujicic *et al.*, 2004).

### 3.2.1 Compressor model

The compressor model is separated into two parts as shown in Fig. 3.3. The first part is a static compressor map which determines the air flow rate through the compressor. Thermodynamic equations are used to calculate the exit air temperature and the required compressor power. The second part represents the compressor and motor inertia and defines the compressor speed. The speed is consequently used in the compressor map to find the air mass flow rate.

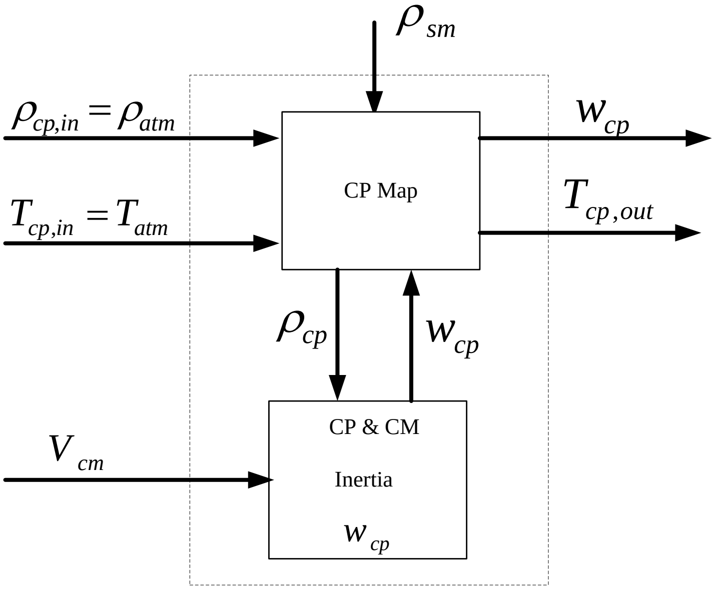


**Fig. 3.3** Compressor block diagram (Pukrushpan, 2003)

The only dynamic state in the model is the compressor speed, $\omega_{cp}$. The inputs to the model include air inlet pressure, $p_{cp,in}$, its temperature, $T_{cp,in}$, voltage command to compressor motor, $V_{cm}$ and down stream pressure, which is the supply manifold pressure, $P_{cp,out}$=psm. The inlet air is typically atmospheric and its pressure and temperature are assumed to be fixed at $P_{atm}$=1 atm and $T_{atm}$=25°C, respectively. The motor commond is one of the inputs to the FC system. The down stream pressure is determined by the supply manifold model.

The flow and temperature out of the compressor ($W_{cp}$ and $T_{cp}$) depend on the compressor rotational speed $\omega_{cp}$. A lumped rotational model is used to represent the dynamic behavior of the compressor (Pukrushpan *et al.*, 2004a):

$$J_{cp}\frac{d\omega_{cp}(t)}{dt} = \tau_{cm}(t) - \tau_{cp}(t) \tag{1}$$

where $\tau_{cm}(v_{cm}, \omega_{cp})$ is the compressor motor (CM) torque and $\tau_{cp}$ is the load torque. The compressor motor torque is calculated using a static motor equation:

$$\tau_{cm} = \eta_{cm}\frac{k_t}{R_{cm}}\left[V_{cm}(t) - k_v\omega_{cp}(t)\right] \tag{2}$$

where $k_t$, $R_{cm}$ and $k_v$ are motor constants and $\eta_{cm}$ is the motor mechanical efficiency. The torque required to drive the compressor is calculated using the thermodynamic equation.

$$\tau_{cp}(t) = \frac{c_p T_{atm}(t)}{\eta_{cp}\omega_{cp}(t)}\left[\left(\frac{p_{sm}(t)}{p_{atm}}\right)^{\frac{(\gamma-1)}{\gamma}} - 1\right]W_{cp}(t) \tag{3}$$

where $\gamma$ is the ratio of the specific heats of air (=1.4), $c_p$ is the constant pressure specific heat capacity of air (=1004 $J.kg^{-1}.K^{-1}$), $\eta_{cp}$ is the motor compressor efficiency, $p_{sm}$ is the pressure inside the supply manifold and $p_{atm}$ and $T_{atm}$ are the atmospheric pressure and temperature, respectively.

### 3.2.2 Supply manifold model

The manifold model represents the lumped volume associated with pipes and connections between each device. The supply manifold volume includes the volume of the pipes between the compressor and the FC stack including the volume of the cooler and the humidifier. The return manifold represents the pipeline at the FC stack exhaust.

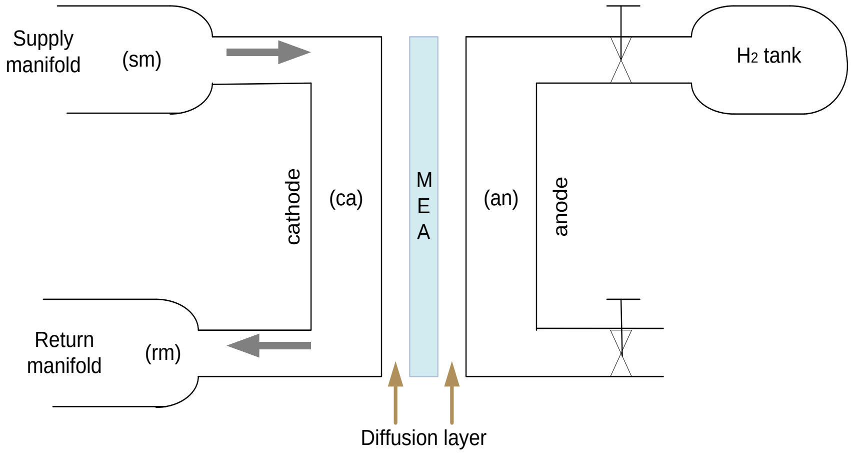


**Fig. 3.4** The FC supply system (Pukrushpan *et al*., 2004a)

The cathode supply manifold (sm) includes pipe and stack manifold volumes between the compressor and the FCs as shown in Fig. 3.4. The supply manifold pressure, $p_{sm}$, is governed by mass continuity and energy conservation equations (Pukrushpan *et al*., 2004b):

$$\frac{dm_{sm}(t)}{dt} = W_{cp}(t) - W_{sm,out}(t) \tag{4}$$

$$\frac{dp_{sm}(t)}{dt} = \frac{\gamma R}{M_a^{atm} V_{sm}} \left[ W_{cp}(t) T_{cp}(t) - W_{sm,out}(t) T_{sm}(t) \right] \tag{5}$$

where $R$ is the universal gas constant, γ ia the ratio of a specific heat capacities of air, $M_a^{atm}$ is the molar mass of atmospheric air at $\phi_{atm}$, $V_{sm}$ is the manifold volume and $T_{sm} = \frac{p_{sm} V_{sm} M_a^{atm}}{R m_{sm}}$ is the supply manifold gas temperature

### 3.2.3 FC stack model

The FC stack model contains four interacting sub-models which are the stack voltage, the anode flow, the cathode flow and membrane hydration model as in Fig. 3.5. The voltage model contains an equation used to calculate stack voltage that depends on varying FC variables: pressure, temperature, reactant gas partial pressures and membrane humidity. The dynamically varying pressure and relative humidity of the reactant gas flow inside the stack flow channels are calculated in the cathode and the anode flow model using mass conservation along with the thermodynamic properties. The process of water transfer across the membrane is represented by the membrane hydration model.

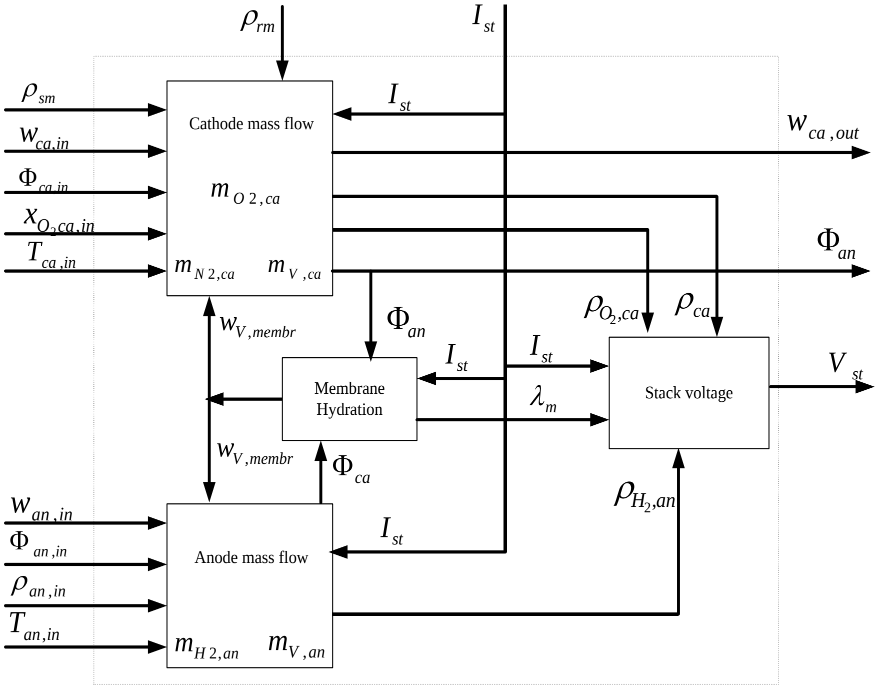


**Fig. 3.5** FC stack block diagram (Pukrushpan *et al*., 2002)

### 3.2.3.1 Stack voltage model

In the stack voltage model, the stack voltage is calculated as a function of stack current, cathode pressure, reactant partial pressure, FC temperature and membrane humidity. Since the FC stack comprises of multiple FC connected in series, the stack voltage, SV, is calculated by multiplying the cell voltage, $V_{FC}$, by the number of cell, n, in the stack.

$$SV = nV_{FC} \tag{6}$$

The FC losses related to stack voltage are the activation loss, the ohmis loss and the concentration loss. The stack current, SC, is equal to cell current. For a set of FC operating conditions (pressures, temperature, humidity) the characteristic of FC is typically given in the form of polarization curve, which is the plot of cell voltage, $V_{FC}$ and cell current density, $i_{FC}$. The current density is defined as cell current per unit of cell active area where:

$$i_{FC} = \frac{SC}{A_{FC}} \tag{7}$$

3.2.3.2 Cathode flow model

The cathode, ca mass flow model represents the air flow behaviour inside the cathode flow channel of the FC. The model is developed using the mass conservation principle and thermodynamic and psychrometric properties of air. The mass continuity is used to balance the mass of three elements, which are $O_2$, $N_2$ and $H_2O$, inside the cathode volume.

$$\frac{dm_{O2}}{dt} = W_{O2,in} - W_{O2,out} - W_{O2,reacted} \tag{8}$$

$$\frac{dm_{N2}}{dt} = W_{N2,in} - W_{N2,out} \tag{9}$$

$$\frac{dm_{w,ca}}{dt} = W_{v,ca,in} - W_{v,ca,out} + W_{v,gen} + W_{v,reacted} \tag{10}$$

Using the mass of $O_2$, $m_{O2}$, nitrogen, $m_{N2}$, water, $m_w$ and the stack temperature, $T_{st}$, $O_2$, $N_2$ and vapour partial pressure, $\rho_{O2}$, $\rho_{N2}$, $\rho_v$, cathode pressure, $\rho_{ca}= \rho_{O2}+ \rho_{N2}+\rho_v$, relative humidity, $\phi_{ca}$, and dry air $O_2$ mole fraction, $x_{O2,ca}$, of the gas inside the cathode channel can be calculated using the ideal gas law and thermodynamic properties. The flow rates are functions of the stack current, SC in ampere.

$$W_{O2,reacted} = M_{O2} \times \frac{nI_{SC}}{4F} \tag{11}$$

$$W_{v,ca,gen} = M_{v} \times \frac{nI_{SC}}{2F} \tag{12}$$

## 3.3 Summary

This chapter explained the overall operation systems of PEMFC including the sub-systems and the dynamic equations involved in designing the simulator model. This PEMFC simulator model was developed in MATLAB/Simulink environment by University of Michigan to tackle the problem in control engineering especially related to FC applications. Later, this simulator model is modified to introduce five types of faults in order to do fault detection and isolation.

# CHAPTER 4

# ARTIFICIAL NEURAL NETWORK MODELLING

## 4.1 Introduction

Artificial neural network (ANN) is a mathematical model designed to train, visualise, and validate neural network model (Nazari and Ersoy, 1992) and the ANN is a model-free estimator as it does not rely on an assumed form of the underlying data (Chang and Islam, 2000). The neural network model can be defined as a data structure that can be adjusted to produce a mapping from a given set of input data to features of or relationships among the data. The model is adjusted, or trained, using a collection of data from a given source as input, typically referred to as the training set. After successful training, the neural network will be able to perform classification, estimation, prediction, or simulation on new data from the same or similar sources (Moustafa, 2011). Due to the limits of using mathematical models in complex modelling and to make FDI algorithm practical for real systems, an approach to the simulation of the PEMFC dynamics was applied using neural network modelling techniques, such as radial basis function (RBF) and multilayer perceptron (MLP). A neural network provides a general way to model a nonlinear system with memory and it has been used by many researchers to describe the relationship between the input and output of monitored systems (Kamal and Yu, 2011). There are many different types of ANN and techniques for training them but here we are just going to focus on the RBF and MLP neural network. Fig. 4.1 shows the flow chart implemented for both networks during training session.

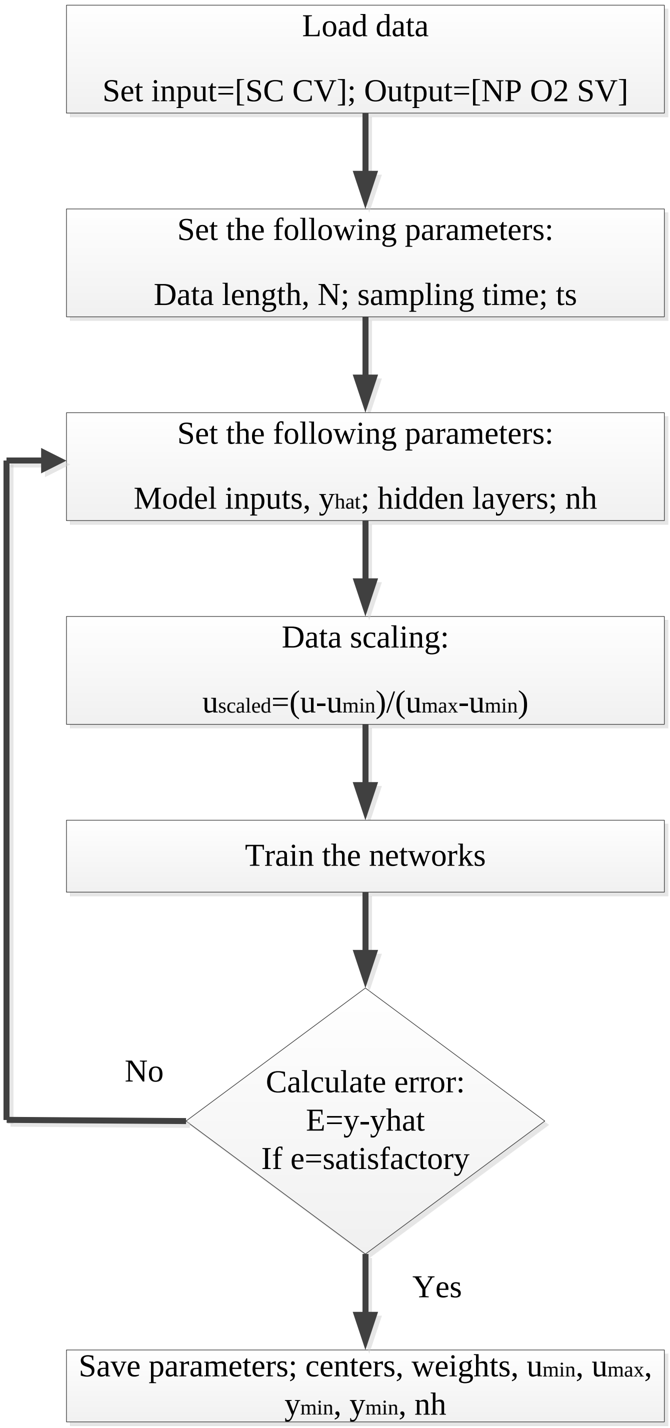

Load data
Set input=[SC CV]; Output=[NP O2 SV]
Set the following parameters:
Data length, N; sampling time; ts
Set the following parameters:
Model inputs, $y_{hat}$; hidden layers; nh
Data scaling:
$u_{scaled}$=(u-$u_{min}$)/($u_{max}$-$u_{min}$)
Train the networks
No
Calculate error:
E=y-yhat
If e=satisfactory
Yes
Save parameters; centers, weights, $u_{min}$, $u_{max}$, $y_{min}$, $y_{min}$, nh

**Fig. 4.1** The flow chart of training the neural networks

## 4.2 Radial basis function neural network

RBF neural network is chosen because of its characteristic which has the ability of approximation of a nonlinear input system to a linear output. Besides that the training process is faster and better compared with another neural network. The RBF network is capable of approximating any continuous function with certain precision level and therefore, can be used in dynamic system modelling and control (Li *et al.*, 2009).

### 4.2.1 RBF network structure

The RBF network has an input, a hidden layer and an output layer as shows in Fig. 4.2. The neurons in the hidden layer contain the RBF whose outputs inversely proportional to the distance from the centre of the neutron. The output units implement a weighted sum of outputs from the hidden unit to form their outputs.

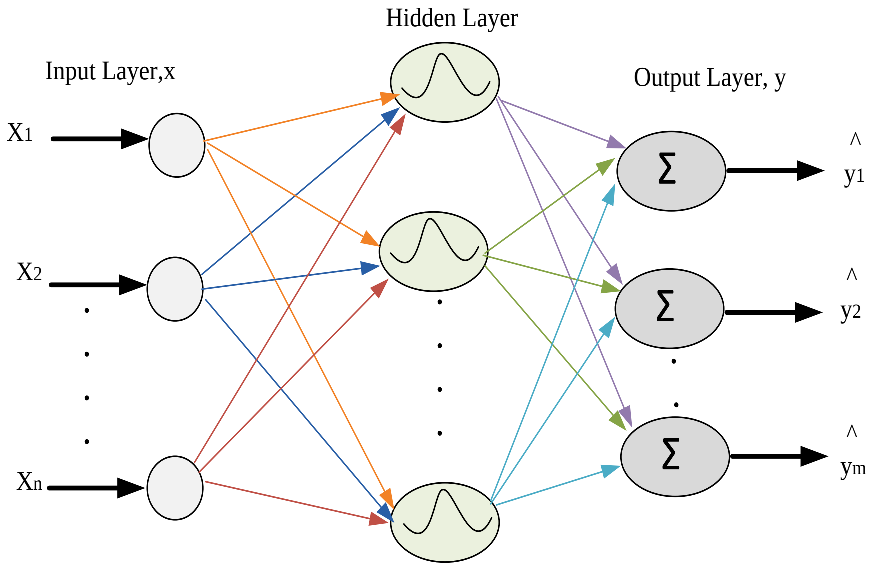


**Fig. 4.2** The structure of RBF networks

The nonlinear system can be modeled by the multivariable NARX model of the following form,

$$y(k) = f[y(k-1),\cdots,y(k-n_y),u(k-1-d),\cdots,u(k-n_u-d)]+e(k) \tag{13}$$

where $u \in \Re^m$, $y, e \in \Re^p$ are the process input, output and noise vectors respectively with $m$ and $p$ being the number of inputs and outputs, $n_y$ and $n_u$ are the maximum lags in the outputs and inputs respectively, $d$ is a dead-time vector representing delayed time to different control variables, $f(*)$ is a vector-valued nonlinear function. Suppose that the RBF network model precisely models the system; the model can then be represented by:

$$\hat{y}(k) = f[\hat{y}(k-1),\cdots,\hat{y}(k-n_y),u(k-1-d),\cdots,u(k-n_u-d)]+e(k) \tag{14}$$

RBF neural network is feedforward network consisting of three layers which are the input layer, hidden layer and output layer. Each hidden node contains a centre $c_j$, which is a cluster centre on the input vector space, and calculates the euclidean distance between the centre and the network input vector $x$ defined by $||x(t)-c_j(t)||$ with $x$ given as:

$$x(k) = [y(k-1),\cdots,y(k-n_y),u(k-1-d),\cdots,u(k-n_u-d)] \tag{15}$$

then the output of the hidden layer node is a nonlinear function of the euclidean distance. In this work the Gaussian function is chosen as the nonlinear function.

$$\varphi_i = \exp\left(-\frac{\|x-c_i\|^2}{{\sigma_i}^2}\right), \quad i = 1,\ldots,n_h \tag{16}$$

where $c_i, \sigma, x \in \Re^{n_h}$ are centre vector, width vector and input vector respectively. The network output is then sum the weighted output of all hidden nodes and bias. The input-output mapping for the RBF networks (Yu *et al.*, 1999) can then be described as:

$$\hat{y} = \phi^T W \tag{17}$$

where $\hat{y} \in \Re^p$, $W \in \Re^{p \times n_h}$, $\phi \in \Re^{n_h}$, are estimated output vector, weight matrix and hidden layer output vector respectively.

### 4.2.2 The training algorithm

Training an RBF network is optimizing parameters including the hidden layer centres and the widths in Gaussian functions and network weights, to achieve minimum model prediction error. In this work, the network centres are selected using the *K*-means clustering algorithm, so that the sum squared distance of each input data from the centre of the data group, to which it belongs, is minimized. The width of the Gaussian functions are chosen using the ρ-nearest centre algorithm (Yu *et al*., 1999), to achieve that any input data is properly sampled by a few near centres. The weights between the hidden layer and the output are trained using the recursive orthogonal least squares (ROLS) as it is a numerically robust algorithm (Kamal *et al*., 2014). The detailed description of the three algorithms for the RBF network training can be found in Wang *et al*. (2006).

#### 4.2.2.1 Recursive K-means algorithm

The centres are set by the K-means clustering method whose objective is to minimize the sum squared distances from each input data to its closest centre so that the data is adequately covered by the activation functions $\Phi_j(t)$.The K-means clustering method proceeds as follows:

i) Randomly choose some input data to be the initial centres. The number of the centres is equal to the number of hidden nodes, $n_h$.

ii) Let k(x) denote the index of the best-matching centre for the input vector x. Find k(x) at iteration t by minimizing the sum squared distances:

$$k(x) = \arg\min \left[ \frac{1}{2} \sum_{k=1}^{nh} \left( x(t) - c_k(t) \right)^2 \right] \tag{18}$$

where $c_k(t)$ is the centre of the kth activation function at iteration t.

iii) Update the centres of the activation functions by using the following rule:

$$c_k(t+1) = \begin{cases} c_k(t) + \alpha\left[x(t) - c_k(t)\right] & if \quad k = k(x) \\ c_k(t) & otherwise \end{cases} \tag{19}$$

iv) Increment t by 1 and go back to step 2. Continue the algorithm until no noticeable changes are observed in the centres $c_k$.

4.2.2.2 p-nearest neighbours method

The widths are computed by the ρ-nearest neighbours method. The excitation of each node should overlap with other nodes (usually closest) so that a smooth interpolation surface between nodes is obtained. In this method, the widths for each hidden node are set as the average distance from the centre to the ρ-nearest s as given by:

$$\sigma_i = \frac{1}{P} \sum_{d=1}^{P} \left\| c_i(t) - c_d(t) \right\| \qquad i = 1, \ldots, n_h \tag{20}$$

cj is the ρ-nearest neighbour of ci. For the non-linear function, the value of ρ depends on the type of problem encountered.

### 4.2.2.3 The recursive orthogonal least squares algorithm

Training of the RBF network weights with the ROLS algorithm is as follows. Considering (17) at sample interval *k* for a set of *N* samples of input-output training data from $k - N + 1$ to *k*, in other words a window going back in time N samples, we have

$$Y(k) = \hat{Y}(k) + E(k) = \Phi(k)W(k) + E(k) \tag{21}$$

where $Y \in \Re^{N \times p}$ is the desired output matrix, $\hat{\boldsymbol{Y}} \in \boldsymbol{\Re}^{N \times p}$ is the neural network output matrix, $\Phi \in \Re^{N \times n_h}$ is the hidden layer output matrix, $\boldsymbol{E} \in \boldsymbol{\Re}^{N \times p}$ is the error matrix and equation (21) can be solved for $W(k)$ using the recursive MIMO least squares algorithm to minimize the following time-varying cost function,

$$J(k) = \left\| \begin{bmatrix} \sqrt{\lambda} Y(k-1) \\ ---- \\ y^T(k) \end{bmatrix} - \begin{bmatrix} \sqrt{\lambda} \Phi(k-1) \\ ---- \\ \phi^T(k) \end{bmatrix} W(k) \right\|_F \tag{22}$$

where the *F*-norm of a matrix is defined as $\|A\|_F^2 = trace(A^T A)$ and $\lambda < 1$ is used to introduce exponential forgetting to the past data. It has been shown in Gomm and Yu (2000) that minimizing (22) is equivalent to minimizing the following cost function,

$$J(k) = \left\| \begin{bmatrix} \sqrt{\lambda} \widehat{Y}(k-1) \\ ---- \\ y^T(k) \end{bmatrix} - \begin{bmatrix} \sqrt{\lambda} R(k-1) \\ ---- \\ \phi^T(k) \end{bmatrix} W(k) \right\|_F \tag{23}$$

where *R* is an $n_h \times n_h$ upper triangular matrix, and $\widehat{Y}$ is computed by an orthogonal decomposition as follows,

$$\begin{bmatrix} \sqrt{\lambda}R(k-1) \\ ---- \\ \phi^T(k) \end{bmatrix} = Q(k)\begin{bmatrix} R(k) \\ --- \\ 0 \end{bmatrix}, \begin{bmatrix} \hat{Y}(k) \\ --- \\ \eta^T(k) \end{bmatrix} = Q^T(k)\begin{bmatrix} \sqrt{\lambda}\hat{Y}(k-1) \\ ---- \\ y^T(k) \end{bmatrix} \tag{24}$$

where $Q$ is an orthogonal matrix. Combining (22) and (23) and considering that the $F$-norm is preserved by orthogonal transformation, the following equivalent cost function is obtained,

$$J(k) = \left\| \begin{bmatrix} \hat{Y}(k) - R(k)W(k) \\ --------- \\ \eta^T(k) \end{bmatrix} \right\|_F \tag{25}$$

which allows the optimal solution of $W(k)$ to be solved straightforwardly from

$$R(k)W(k) = \hat{Y}(k) \tag{26}$$

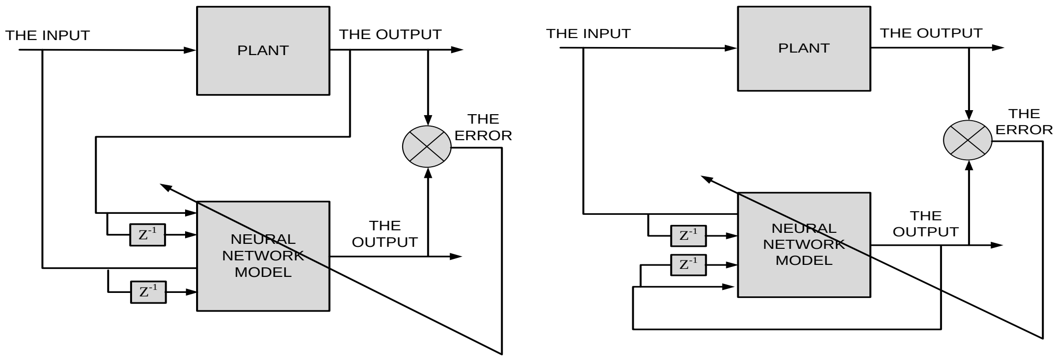
and leaves the residual at sample interval $k$ as $\left\|\eta^T(k)\right\|_F$. Since $R(k)$ is an upper triangular matrix, $W(k)$ can be easily solved from (26) by backward substitution. The decomposition in (24) can be achieved efficiently by applying givens rotations to an augmented matrix to obtain the following transformation by Gomm and Yu (2000):

$$\begin{bmatrix} \sqrt{\lambda}R(k-1) & \sqrt{\lambda}\hat{Y}(k-1) \\ \phi^T(k) & y^T(k) \end{bmatrix} \rightarrow \begin{bmatrix} R(k) & \hat{Y}(k) \\ 0 & \eta^T(k) \end{bmatrix} \tag{27}$$

The procedure of the ROLS algorithm is therefore the following: for on-line training, calculate $\phi(k)$ at each sampling period to update the augmented matrix and compute the givens rotations to realize the transformation in (27). Then solve $W(k)$ in (26) with $R(k)$ and $\hat{Y}(k)$ obtained in (27). In this case, $W(k)$ is needed at each sample instant for prediction. Also, $\lambda < 1$ is needed to follow time-varying dynamics at the current time. For use in off-line mode, the givens rotations can be computed to realize the transformation in (27) continuously to the end of training, then $W$ is solved finally from (26). In this case, $\lambda$ is set to 1. Initial values for $R(k)$ and $\widehat{Y}(k)$ in both cases can be assigned as $R(0) = \mu I$ and $\widehat{Y}(0) = 0$ , where $\mu$ is a small positive number and $I$ is a unity matrix with appropriate dimension.

### 4.2.3 Network modelling modes

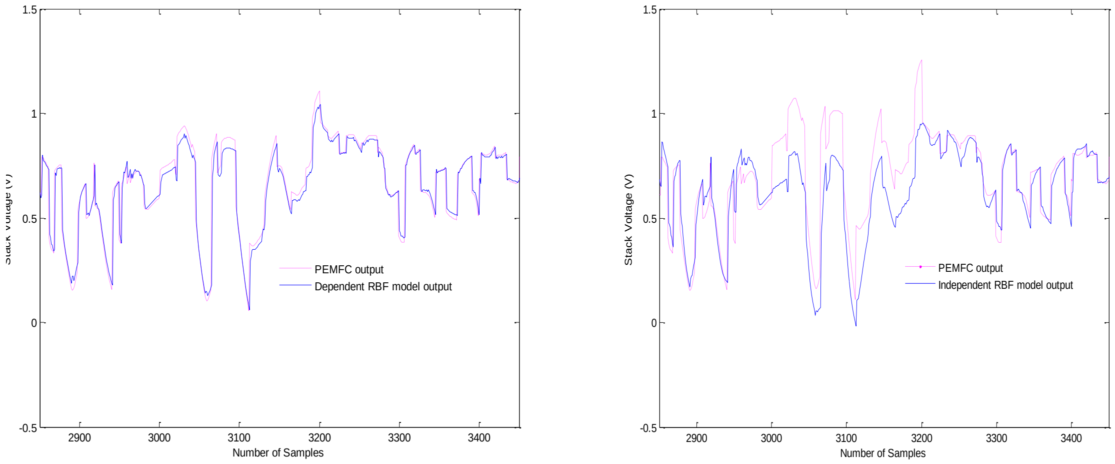
There are two different modes of modelling a dynamic system using neural networks, one is the dependent mode and the other is the independent mode as defined by Narendra and Parthasarathy (1990). The structures of the two modes are displayed in Fig. 4.3 where Fig. 4.3(a) shows the block diagram of dependent mode (the process output $y(k)$ in the input vector $x(k)$ of the network in (14)) while in Fig. 4.3(b) shows the block diagram of independent mode (the model output $\hat{y}(k)$ in the input vector $x(k)$ of the network in (15)). From Fig. 4.3(a), it can be seen that in the dependant mode, the output of the plant is used as part of the network inputs. Therefore, the neural network model will be dependent on the plant and cannot run independently. When the dependent model runs alone, after predicts for one-step-ahead the plant output would not be available. Therefore, this mode of the model cannot do multi-step-ahead prediction and cannot run independently. The dependent mode is suitable to be trained for accurate one-step-ahead prediction, but one-step-ahead prediction is very limited for applications.



(a) Dependent mode (b) Independent mode

**Fig. 4.3** The structure of the two modelling modes used in neural network modelling

On the contrast, the past model output is used, instead of process output, as part of model input in the independent mode. Since the independent model can be run independently of the process, it can be used to do multi-step-ahead prediction, and can also be used as simulation model. The above features have been experienced by the authors in the past research (Yu *et al*., 1999). The model of the dependent mode can predict the process output for one-step-ahead only, while the independent model can predict for unlimited steps as long as the input, N is available. When the two different modes of model are used for fault detection, the difference is significant. When a fault occurs to the plant and affect the plant output, the dependent model output will also be affected through the plant output being used as the model input; whilst the independent model will not be affected by the occurred fault as the model is independent from the plant. This can be clearly seen from the plant output and model prediction shown in Fig. 4.4(a) and (b) below.



(a) (b)

**Fig. 4.4** Stack voltage predictions by (a) dependent mode, (b) independent mode in presence of fault

Fig. 4.4 shows the results of RBF network when faults occurred in the plant for stack voltage output. Here, a fault has been introduced at k=3000-3200. The same signal inputs have been used for both modes to compare the different between these two. Fig. 4.4(a) shows the result of dependent mode when fault occurred at that particular times. As can be seen, when a fault occurs in the process, the process output is affected, at the same time due to that the process output is used as the model input, the model prediction is also affected. Consequently, the error between the process and model output as the residual will not be sensitive to the occurrence of the fault. This is displayed in Fig. 4.4(a) with a fault occurred during sample instant *k*=3000-3200. The same output with the prediction by an independent model is displayed in Fig. 4.4(b). From the result, it shows that when there is a fault in the plant, it will not be effected the prediction of the independent network. Therefore, the modelling error of the independent model is very sensitive to the fault and the residual is big to indicate that there is a fault occurred in the plant.

### 4.2.4 Data collection and scaling

Collecting and preparing sample data is the first step in designing neural network model. The inputs of the PEMFC plant are stack current, SC and compressor motor voltage, CV while the outputs are net power, NP, oxygen excess, $\lambda_{O2}$ and stack voltage, SV. For data collection, the PEMFC systems have been injected with two types of input signals comprises SC and CV. At first, a set of random amplitude signals (RAS) have been generated and used as the system inputs, where the signals ranging from 100 to 300 amperes for the SC and from 100 to 235 volts for CV. A few data samples have been collected consists of 1500, 3000, 5000 and 6000 samples have been acquired with a sample period set at 0.1. The RAS signals have been injected to the PEMFC systems as shown in the simulink model of Fig. 4.5.

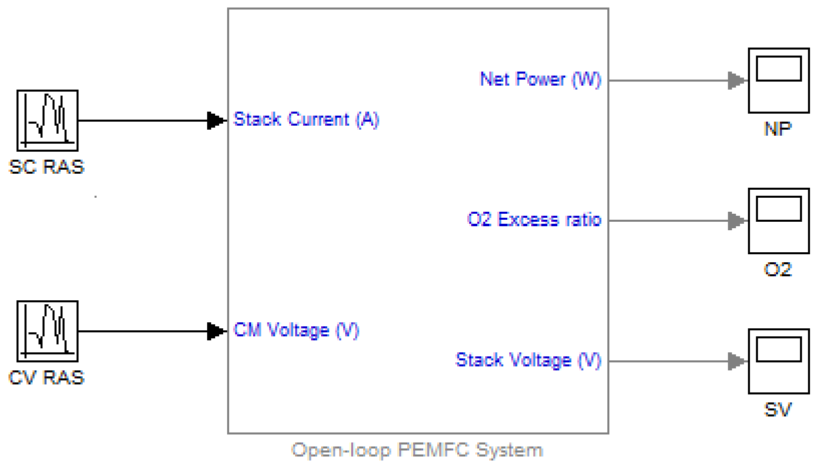
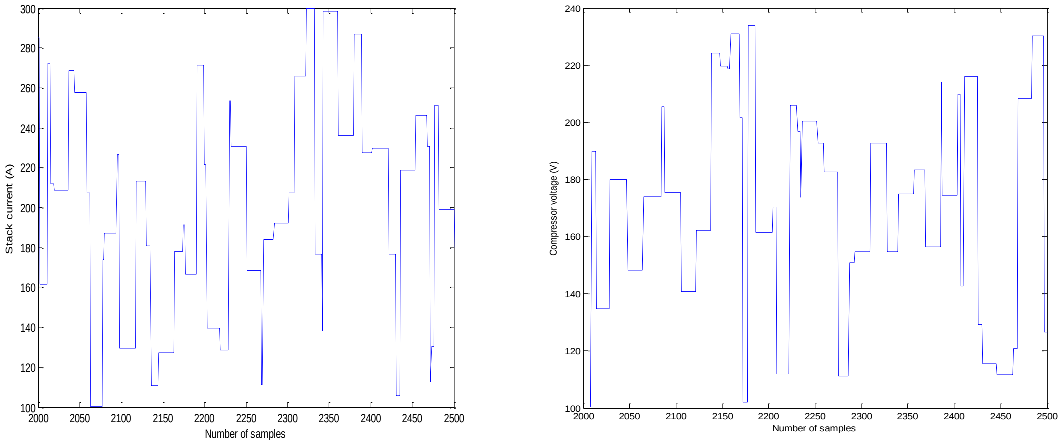


**Fig. 4.5** The Simulink model of RAS excitation signals

Fig. 4.6 shows the close-up of the RAS signals used as input in this work. The signals are generated randomly to cover the whole range of frequencies and entire operating space of amplitude in the PEMFC systems.



(a) (b)

**Fig. 4.6** The RAS excitation signals of SC and CV after zoom-in the samples

The second type of input signals used in this work is a step input. The collection of input-output samples data was generated using the Simulink model as in Fig. 4.7. Again, the range of input signals is between 100 to 300 amperes for the SC and from 100 to 235 volts for CV. The excitation inputs of SC and CV for 5000 data samples can be referred in Fig. 4.7 while Fig. 4.8 shows the input signals of SC and CV.

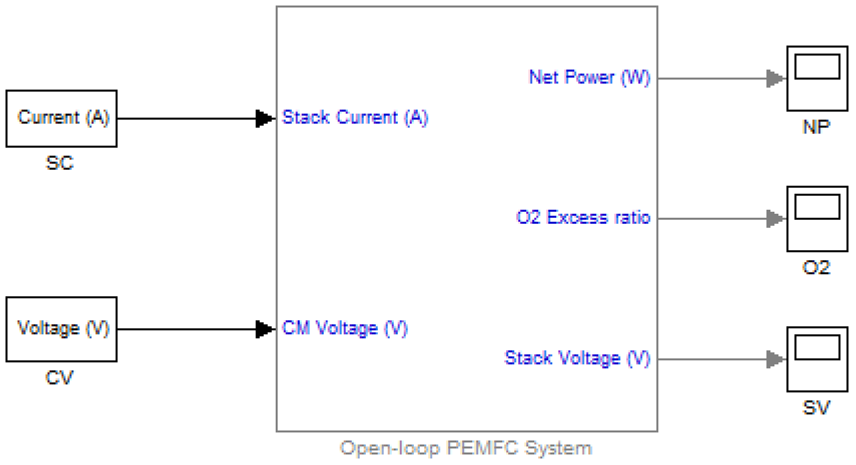
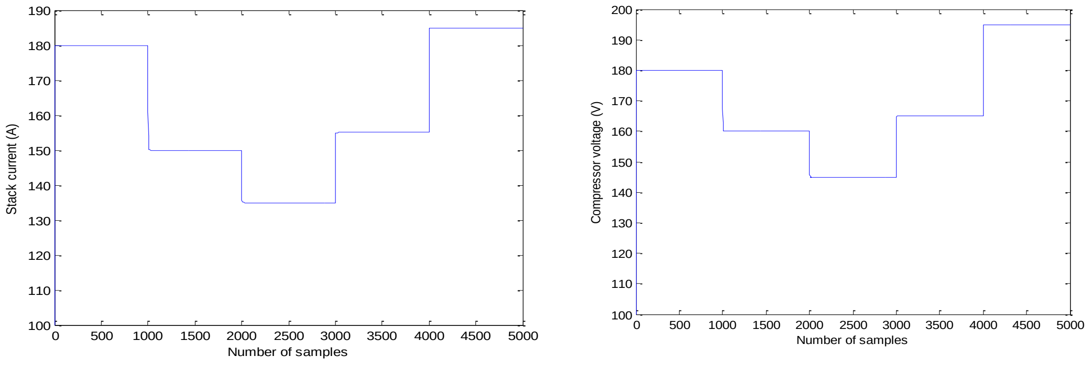


**Fig. 4.7** The Simulink model of step-input generation



(a) (b)

**Fig. 4.8** The step inputs used during the process

After data collection, all the data samples have been normalized before the training and testing procedure to ensure the accuracy of the modelling performance using the following equation:

$$u_{scale}(k) = \frac{u(k) - u_{\min}}{u_{\max} - u_{\min}} \tag{28}$$

$$y_{scale}(k) = \frac{y(k) - y_{\min}}{y_{\max} - y_{\min}} \tag{29}$$

where $u_{min}$, $u_{max}$, $y_{min}$ and $y_{max}$ are the minimum and maximum inputs-outputs of the data set, while $u_{scaled}$ and $y_{scaled}$ are the scaled input and output respectively.

The structure selection of RBF model is based on the smallest modelling error, mean square error (MSE) defined in the following equation:

$$e_{mse} = \frac{1}{N}\sum_{j=1}^{N}[y(j) - \hat{y}(j)]^2 \tag{30}$$

Where y and $\hat{y}$ is the PEMFC system output and the RBF model prediction respectively.

### 4.2.5 Model structure selection

The next step is to determine the input variables of the RBF model. The PEMFC system to be modelled has two inputs: the stack current, SC and the compressor motor voltage, CV with three outputs: net power, NP, oxygen excess, $\lambda_{O2}$ and stack voltage, SV. The selections of RBF model structure is based on modelling trials where the model structure with the smallest modelling errors were selected using the equation (30). Different orders and time delay of these variables have been tried in the model training and the following order is found to be most appropriate to give the minimum modelling error. The equation is:

$$\hat{y}(k) = f[\hat{y}(k-1), \hat{y}(k-2), \hat{y}(k-3), u(k-1), u(k-2)] + e(k) \tag{31}$$

Fig. 4.9 shows the structure of RBF network model implies the equation (31) consists of thirteen inputs and three outputs. The centre is calculated using the K-means clustering algorithm, and the width, σ was chosen using the p-nearest neighbours algorithm. During the training process, the hidden nodes have been selected at 22 hidden nodes while the weights were chosen using the p-nearest neighbours algorithm. The training of this algorithm were applied using the RLS algorithm developed by Zhai and Yu (2007) and the following initial values were used: μ=0.999, $w(0)=1.0\times10^{-6}\times U(nhx3)$, $P(0)=1.0\times10^{5}xI(_{nh})$, where μ is the forgetting factor, I is an identity matrix, U is the matrix with all element unity, and $n_h$ is the number of hidden layers.

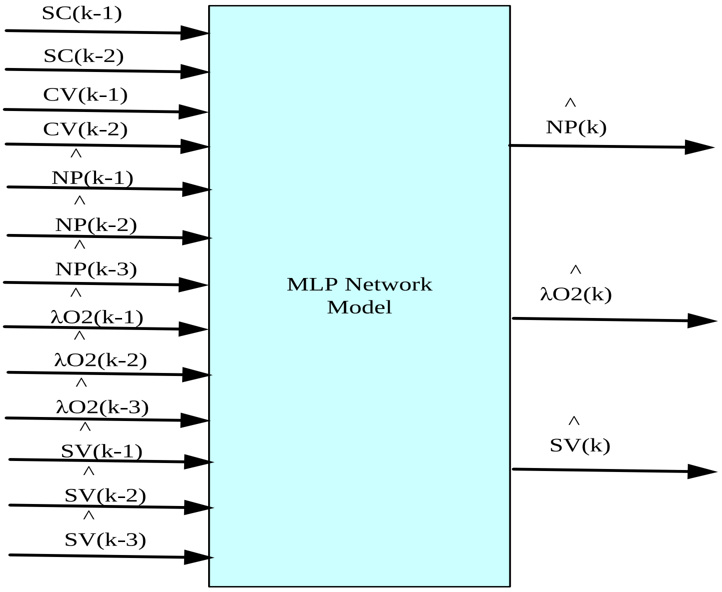


**Fig 4.9** The structure of RBF network model

4.2.6 Simulation results

In total there are five sets of samples collected from the FC stack. These data were divided into two sets, the first set of 4000 samples was used for RBF network training while the other 1000 samples was used for testing. The first 4000 samples have been used to determine centres using the K-means clustering method, and width using the ρ-nearest distance method. Then, the weights were trained using the RLS algorithm, where the input data to the RBF model was formed as shown in Fig. 4.9. After training, the model was tested using 1000 samples that have not been used in the training. The simulation results of the FC stack outputs and the RBF model predictions are shown in Fig. 4.10 and Fig. 4.11 for 1000 data samples. These results show the actual and estimated outputs during the testing process when there is no fault occurring in the process.

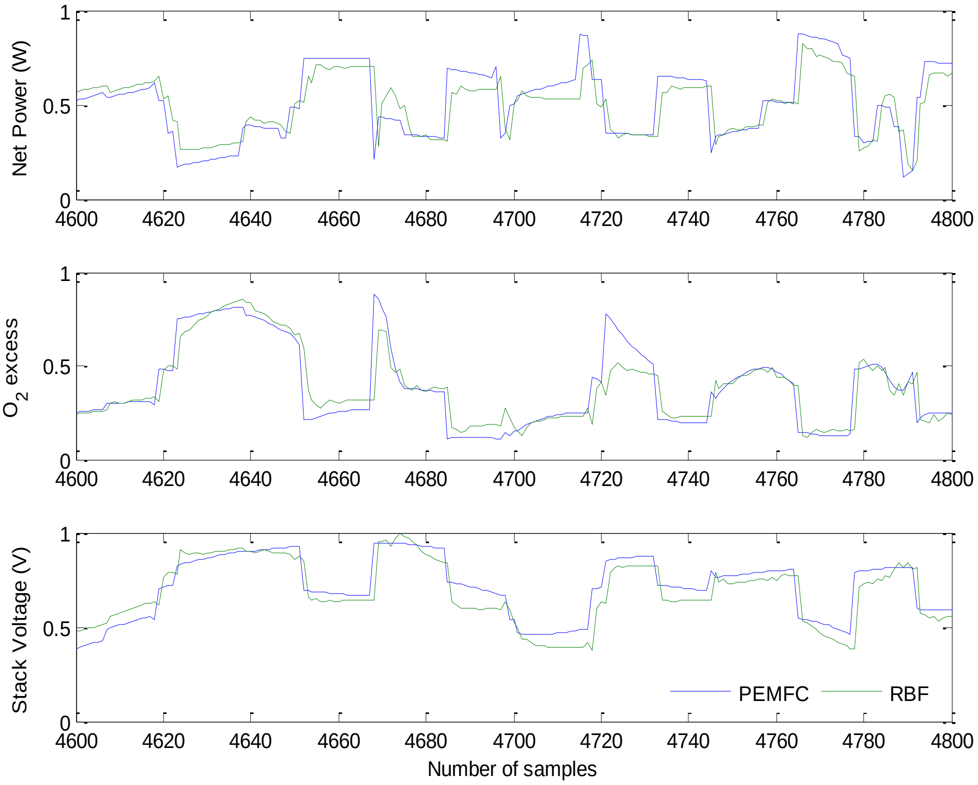


**Fig. 4.10** The actual and estimated outputs of RBF model during testing

From the result, Fig. 4.10 shows the simulation of the process plant and the estimated output of RBF model using the RAS as the input signals while Fig. 4.11 shows the simulation of the process plant with the RBF model using the step input. As can be seen here, the estimated outputs of the RBF model is closely near to the process plant output.

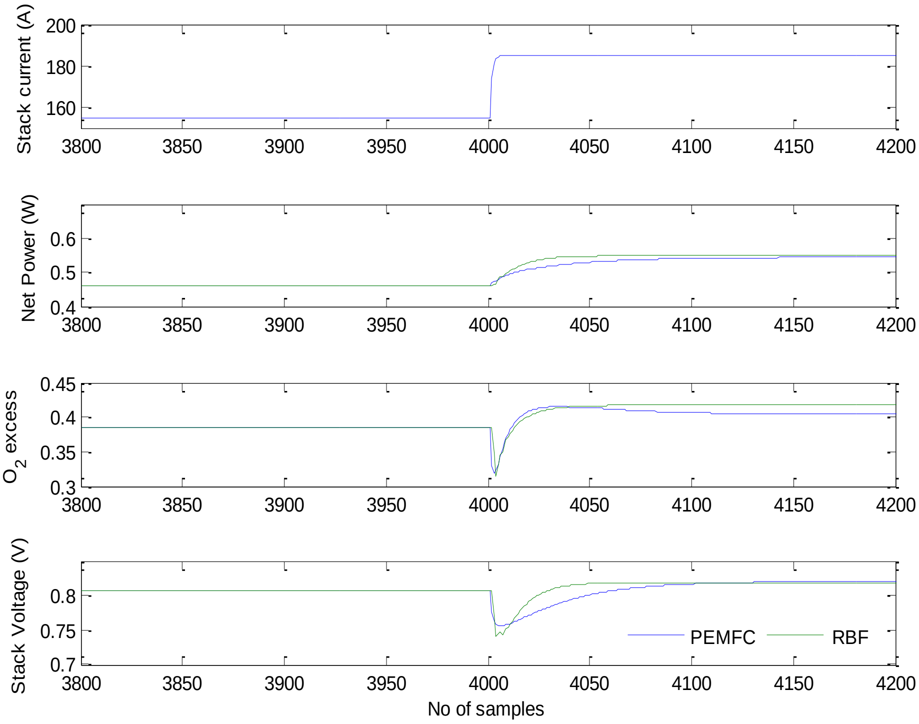


**Fig. 4.11** The actual and estimated outputs of RBF model during testing with step input

The model prediction error for the normalized test data was also measured with MSE and Table 4.1 shows the MSE errors for both types of signals used as input.

**Table 4.1** The MSE obtained during testing of RBF network model

| Types of input signals | NP | $\lambda_{O2}$ | SV |
|---|---|---|---|
| RAS | 0.0092 | 0.0071 | 0.0051 |
| Step input | 1.0 x10$^{-6}$ | 1.0 x10$^{-6}$ | 1.0 x10$^{-6}$ |

## 4.3 Multi-layer perceptron neural network

The multi-layer perceptron (MLP) networks with the back-propagation (BP) training algorithm (Rumelhart *et al*., 1986) are the most commonly used type of feed-forward neural MLP has three types of layers: an input layer, a hidden layer and an output layer. The number of hidden layers in a MLP, and the number of nodes in each layer can vary for a given problem. Here, three layers MLP has been used as the architecture based on justification made by Lippmann (1987) which stated that three layers of MLP is adequate on the basis. The structures of the multi-layer networks are able to generate the internal representations which enable to classify input regions that either intersect each other or are disjoint (Lippmann, 1987). By considering the layers individually, a simple concept model of the operation of the three layer network can be obtained (Woodland, 1989). The input layer distributes the inputs to subsequent layers. Inputs nodes have linear activation functions, while the hidden nodes have a nonlinear activation functions and the outputs have a linear activation function. Hence, each input signal feeding into the hidden layer nodes multiplied by a weight and then is passed through the activation function that may be linear or nonlinear (Delashmit and Manry, 2005).

The most widely used training algorithm for MLP network is the BP algorithm. The BP scheme consists of two major steps: the forward activation and the backward error flow. This algorithm is based on minimizing error of the neural network output compared to the required output. The feed-forward flow used to calculate the error of MLP output. Later, the error is passed backward to adjust the weights between the hidden nodes. The training is carried out by minimizing an error function which allows the output of MLP network to represent classification (Bishop, 1995). Fig. 4.12 shows the flow of BP algorithm used in MLP network during training.

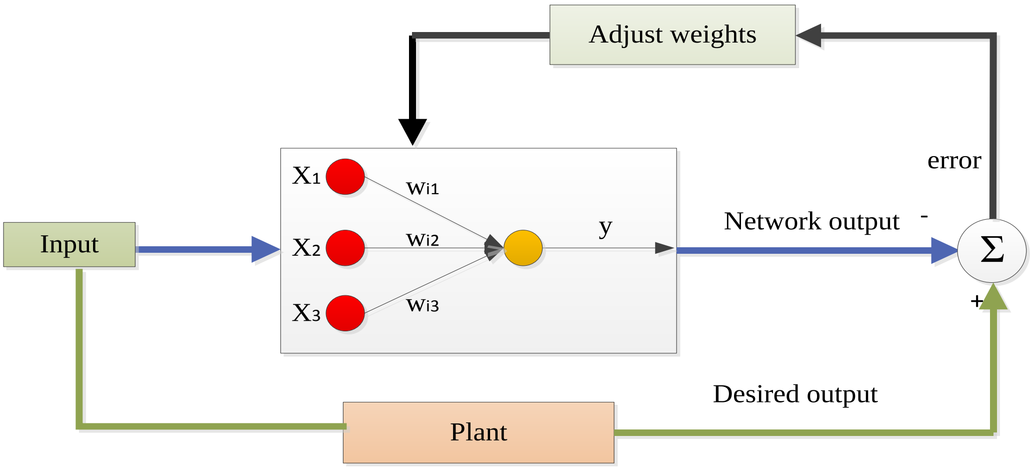


**Fig. 4.12** The schematic diagram of BP algorithm

4.3.1 MLP network structure

Accurate model is required in order to have a very satisfactory result. A NARMAX structure proposed by Leontaritis and Billings (1985) used as a general nonlinear dynamic system described by the following model:

$$\hat{y}_{mlp}(k) = f[\hat{y}_{mlp}(k-1), \hat{y}_{mlp}(k-2), \hat{y}_{mlp}(k-3), u(k-1), u(k-2)] + e(k) \qquad (32)$$

where u is the process input, $\hat{y}$ is the estimated plant output, k is the order of plant output, and f is the nonlinear function.

In this work, three layers of MLP network consists of input layer, hidden node and output layer are used for training. Besides that the input structure of the MLP model has been standardize as in the RBF model structure where the inputs of MLP has been set to thirteen with three outputs are also being analysed. This setting is chosen to make the study more accurate and easy to evaluate. Fig. 4.13 shows the structure of MLP model selection used in this study.

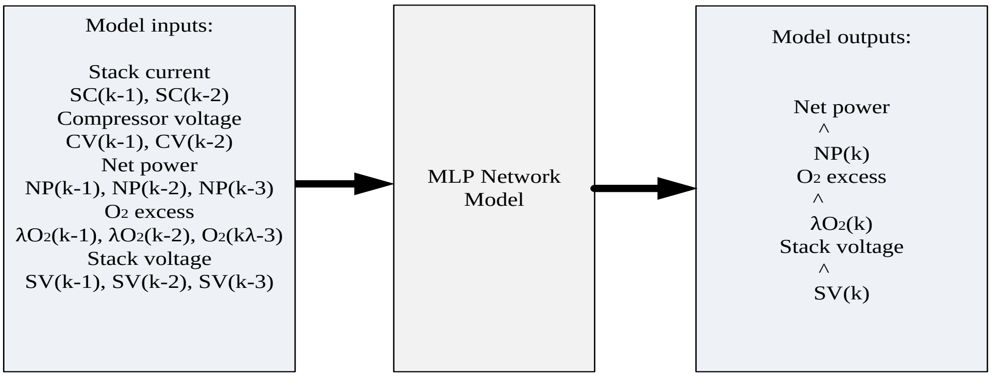


**Fig 4.13** The structure of MLP network model

4.3.2 The training algorithm

The training is carried out to minimize an error function which allows the output of the network to represent classification functions (Bishop, 1995). The steps involved are:

i) Initialize the network, with all weights set at random numbers between -1 and +1 and choose $\eta$, learning rate between $0 < \eta < 1$. Here, learning rate was chosen at 0.1.

ii) Perform feed-forward calculation through the network to produce outputs.

   a) Training set is applied to the input. For each training sample, input patterns and target outputs. Assuming J hidden neurons and N inputs for a three layer MLP.

   b) The input of hidden neuron hi is calculated by:

$$h_i = f\left(\sum_{i=0}^{N} w_{ij} x_i\right) \qquad (33)$$

where hi is output from each hidden neuron J and wij is the weight connecting the input, xi and hidden input and f is the tangent sigmoid activation function.

c) The output of hidden neuron, ho is calculated by:

$$h_0 = \left( \frac{2}{(1 + \exp(-h_i) - 1)} \right) \tag{34}$$

where the activation function take the inputs and squash the outputs into the range -1 and 1.

d) The output of network, $\hat{y}$mlp is given by:

$$\hat{y}_{mlp} = w_{jk} h_o \tag{35}$$

where wjk is the weight connecting the output layer and the output of hidden neuron.

iii)The BP algorithm is designed to reduce error between the actual output and the desired output of the network in a gradient descent manner given by :

$$J = \sum_{j=1}^{g} e_j^2 \tag{36}$$

a) For each output unit k, compute the error, e.

$$e = y - \hat{y}_{mlp} \tag{37}$$

b) The gradients for the weights between the output layer and output of hidden neuron are updated by:

$$w_{jk\_new} = w_{jk\_old} + (\alpha h_o e) \tag{38}$$

where α is the learning rate.

c) The gradients of the weights between the input hidden neuron and the input are updated:

$$w_{ij\_new} = w_{ij\_old} + \left( \beta \left( \frac{(1-h_o)(1+h_o)'}{2} \right) x_i \right) \qquad (39)$$

where β is the learning rate.

iv) Repeat step (ii) – (iii) until the minimum sum of squared error is achieved.

### 4.3.3 Data collection and scaling

To conduct a simulation using the MLP algorithm, the same data collection mentioned in section 3.2.4 was used. As well as for scaling, the same equation as in (28) and (29) is applied in the MLP algorithm. Based on the error obtained, the training process is evaluated using this equation:

$$e = y - \hat{y} \qquad (40)$$

where y and $\hat{y}$ is the PEMFC system output and the RBF model prediction respectively.

### 4.3.4 Simulation results

In the training process, 4000 samples were used to train the MLP network model. After training, the model was validated using 1000 samples that have not been used in the training. The simulation results of the FC stack outputs and the MLP model predictions are shown in Fig. 4.14 and Fig. 4.15 for 1000 data samples. These results show the actual and estimated outputs during the testing process when there is no fault occurring in the process.

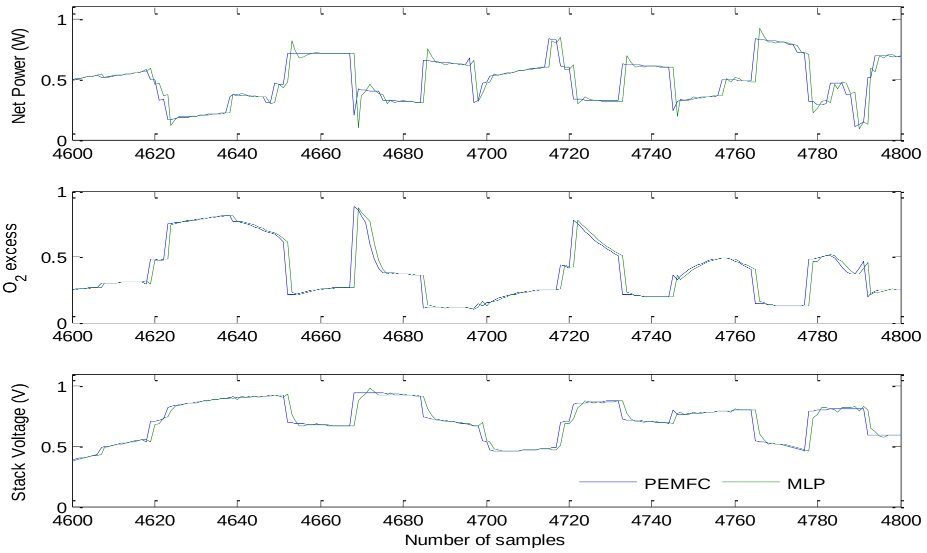


**Fig. 4.14** The actual and estimated outputs of MLP model during testing

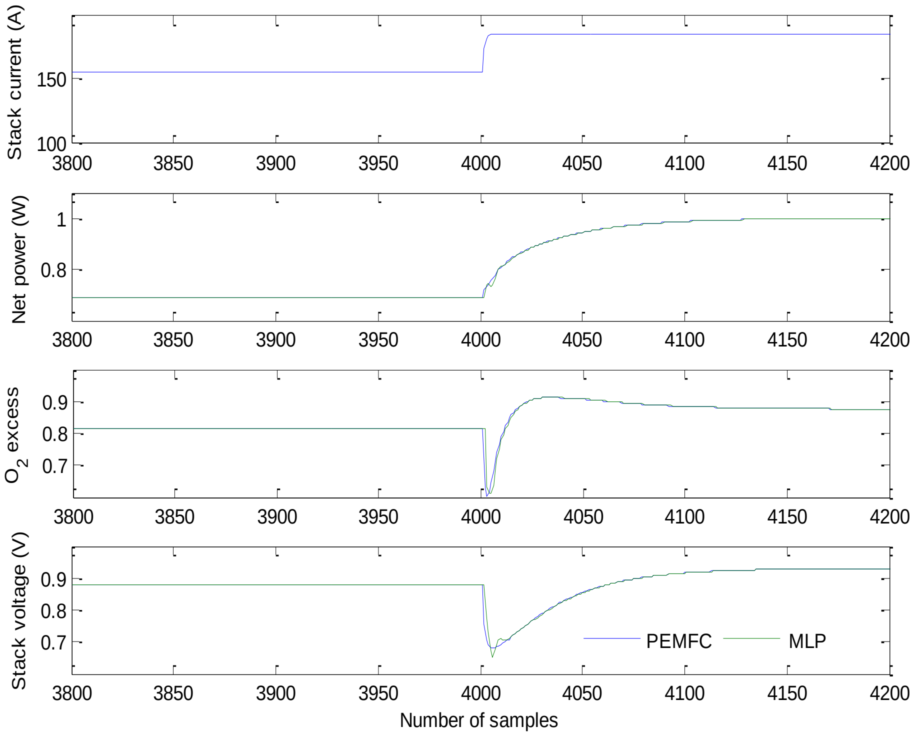


**Fig. 4.15** The actual and estimated outputs of MLP model during testing with step input

The model prediction error for the test data was also measured with MSE equation and Table 4.2 shows the errors for both types of signals used as input in the MLP network.

**Table 4.2** The MSE obtained during testing of MLP network model

| Types of input signals | NP | $\lambda_{O2}$ | SV |
|---|---|---|---|
| RAS | 0.0392 | 0.0083 | 0.0063 |
| Step input | $1.0 \times 10^{-11}$ | $1.0 \times 10^{-11}$ | $1.0 \times 10^{-11}$ |

## 4.4 Summary

The ANN is used to overcome and tackle the nonlinear behaviour of PEMFC dynamic systems. In this work, the RBF network model and the MLP network model has been chosen as the algorithm in order to investigate and perform FDI of PEMFC dynamic systems. The performance and the efficiency of the independent model of both structure is tested and evaluated. Here, two types of input signals have been used to the RBF network model and the MLP network model to train and test the model structure of both networks. This training and testing procedure is carried out to test the model structure using healthy data set collected from the FC stack. This process is very important in the FDI produce because later the modelling errors are used to do fault detection and fault isolation.

# CHAPTER 5

# FDI STRATEGY AND CONFIGURATION

## 5.1 FDI System Configuration

Process faults, if undetected, have a serious impact on process economy, product quality, safety, productivity and pollution level. In order to detect, diagnose and correct these abnormal process behaviors, efficient and advanced automated diagnostic systems are of great importance to modern industries (Albert *et al*, 2007). Once a fault has been detected and its evolution is monitored, the severity of that fault can be evaluated and a decision can be made on the course of action to be taken. Monitoring creates the opportunity to strategically plan and schedule outrages and to manage equipment utilization and availability (Jean-Pierre and Joseph, 1995). Fig. 5.1 shows the flow chart of model-based FDI in general.

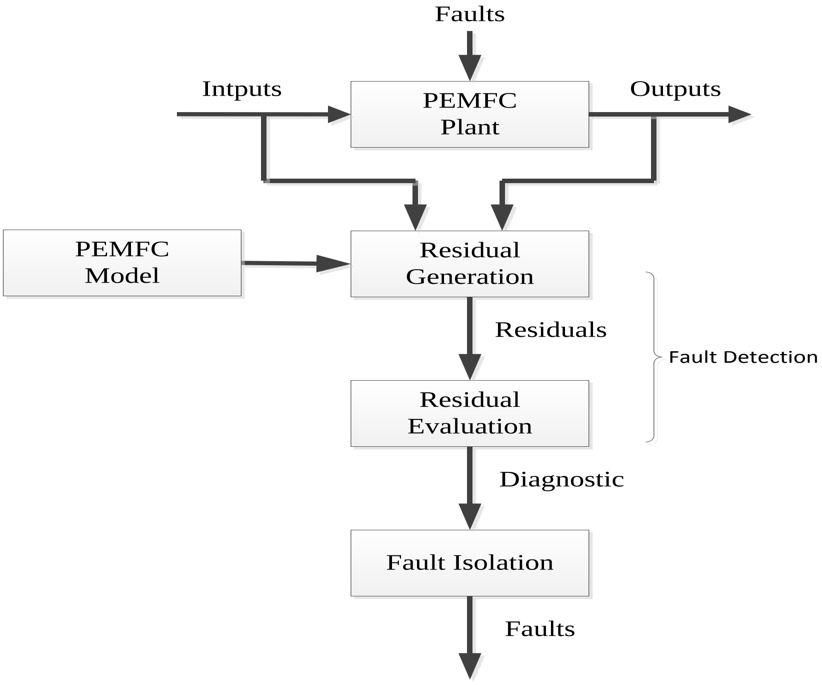


**Fig 5.1** Flow chart of model-based FDI (adopted Isermann, 1984)

In this work, based on flow chart in Fig. 5.1, the FDI configuration used in this work can be seen in detail as in Fig. 5.2 which can be divided into two categories. The first part of the FDI process is the fault detection process and the second part is the fault isolation process. In this FDI configuration, the modelling errors of the output measurements are used both in fault detection and isolation.

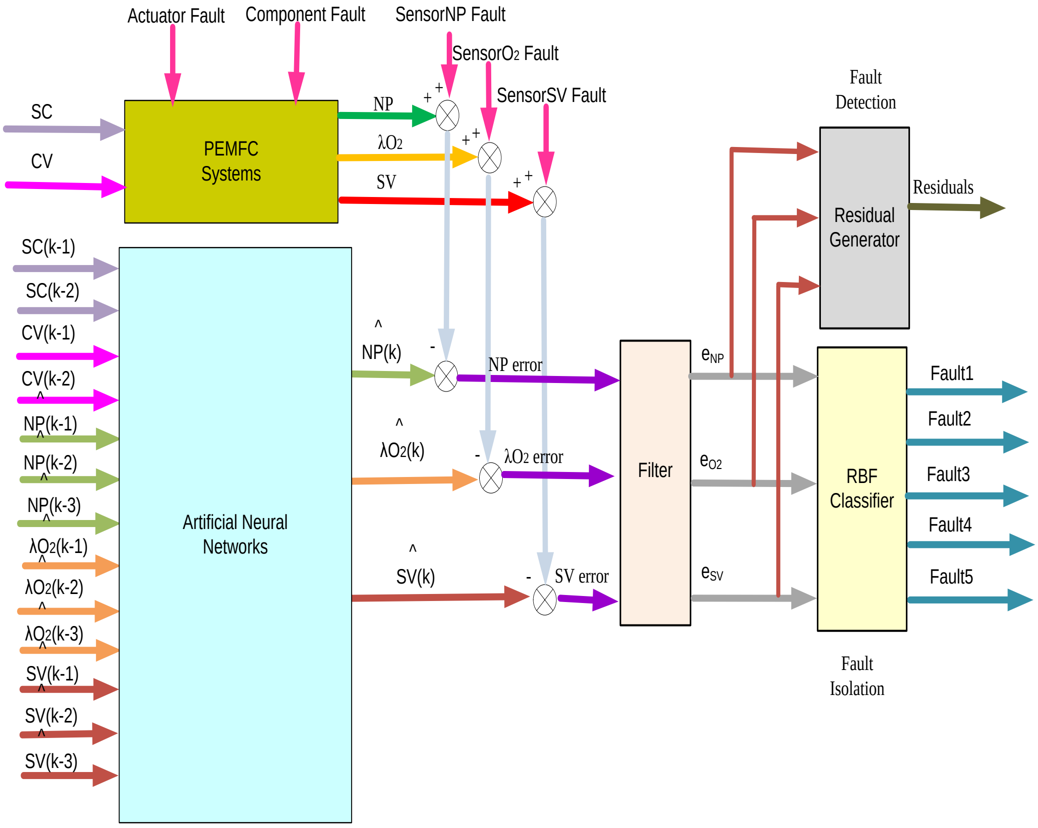


**Fig. 5.2** The configuration of FDI systems with independent neural network model

5.2 Independent neural network model

Normally for prediction, classification and identification, most of the researchers used the dependent mode to predict the future values of the process plant. The disadvantage of this dependent model is that, it is insensitive to faults because it used the output plant as part of the input to the neural network model. Else, for independent model, the prediction is based on the neural network model itself. This feature has been studied by Yu *et al*. (1999) which has been applied in this work. The explanation of these two networks has been explained in section 3.2.3. To perform FDI for PEMFC dynamic systems, the following two types of independent neural networks have been applied in this research. The RBF neural networks and the MLP neural networks are the most common ANN used by researchers.

### 5.2.1 RBF network model

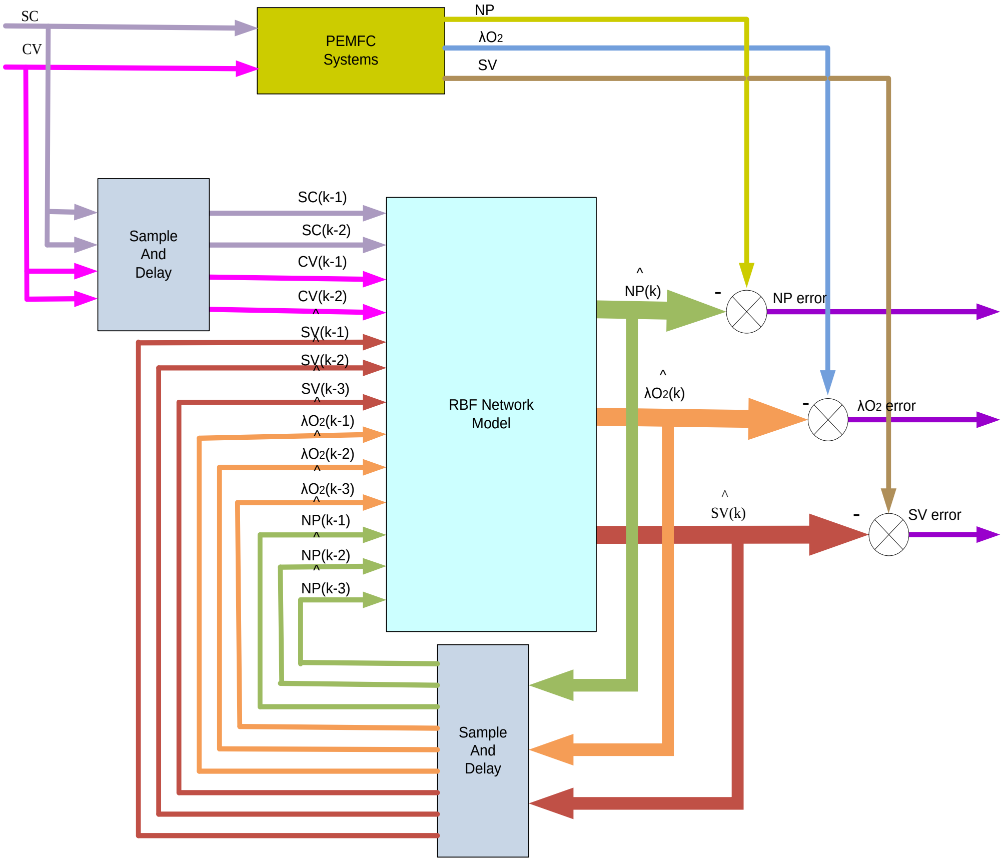


**Fig. 5.3** The structure of an independent RBF network model

The independent RBF network model for PEMFC dynamic systems proposed in this work can be referred to Fig. 5.3. The structure of RBF network consists of two inputs, SC and CV while the three estimated outputs, NP, $\lambda_{O2}$ and SV with their delayed values to form thirteen inputs can be seen in Fig. 5.3 (Kamal and Yu, 2011). The model orders and time delays of the FC stack were selected based on heuristic studies based on minimum squared error obtained during the training and testing process. The structure of this RBF network model is used for the whole proses beginning from training and testing the network then identify the types of faults until the process to classify the faults.

### 5.2.2 MLP network model

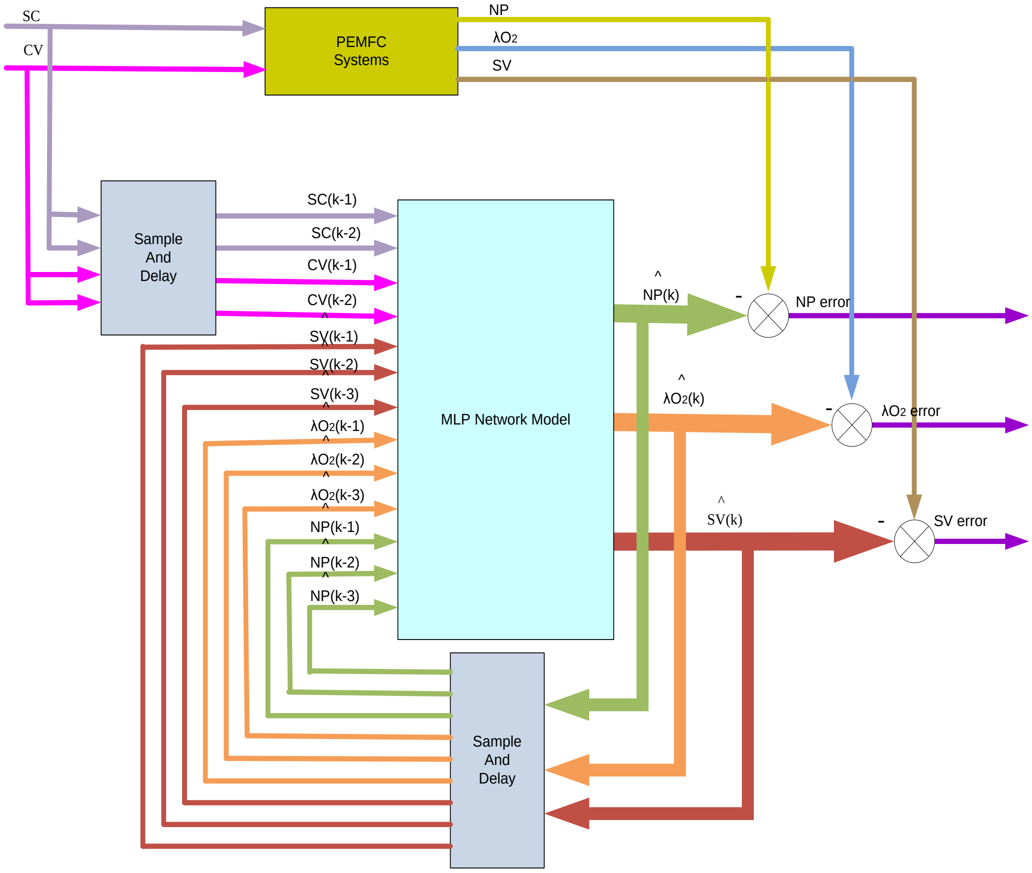


**Fig. 5.4** The structure of an independent MLP network model

To make the analysis accurate and comparable, the inputs of the independent MLP networks model for PEMFC dynamic systems proposed in this work has been set equally as the model structure used in the RBF networks model. Here, the inputs are also set as thirteen inputs with three estimated outputs, NP, $\lambda_{O2}$ and SC with their delayed values can be referred in Fig. 5.4.

## 5.3 Residual generation

Residual generation is the important component in fault detection process where the residuals are generated using the difference between the real system output, y(t) and the estimated outputs of the neural networks, $\hat{y}(t)$. Detectable deflections of the residuals yield to symptoms given by:

$$r(t) = y(t) - \hat{y}(t) \tag{41}$$

The faults are located and fault causes are determined (Kimmich *et al*., 2005). Fig. 5.5 illustrates the block diagram of fault detection with three inputs and one output which is the residual error signal.

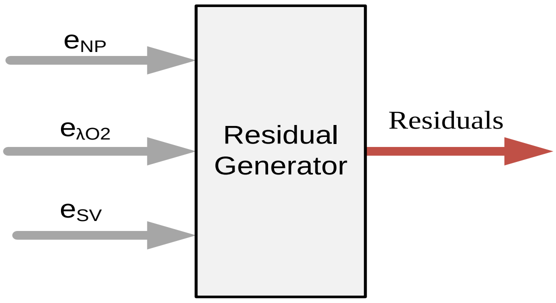


**Fig. 5.5** The block diagram of fault detection process

In order to do fault detection, the residuals are calculated by combining the three model prediction errors, so that the sensitivity of the residual to each fault can be significantly enhanced, and consequently the false alarm rate would be reduced. The residual in this work is calculated as:

$$re = \sqrt{e_{NP}^2 + e_{\lambda O2}^2 + e_{SV}^2} \tag{42}$$

where $e_{NP}$, $e_{\lambda O2}$ and $e_{SV}$ are the filtered modelling error of net power, $\lambda_{O2}$ and stack voltage, respectively. When faults are detected, the fault isolation procedure is started.

## 5.4 Fault isolation

Fault isolation step is important in the FDI process due to its ability to classify the type of faults at that particular time when faults occurred. The RBF network is well known for its powerful ability to classify components with different features from a mixed signal. Fault isolation in this study is implemented by adding another RBF network as a classifier (Kamal and Yu, 2012). The model prediction error vector obtained from the fault detection part is caused by faults, and is a nonlinear function of the faults. As this vector is multi-dimension, three dimensions in this study, it will have different structures for different faults. The residual vector feature has also been used in the neural network method by Patton *et al*. (1994) and Yu *et al*. (1996). The RBF classifier uses this feature to classify these faults. Based on this idea, the RBF classifier is designed with three inputs to receive the three elements of the filtered and squared model prediction error vector, and five outputs with each being dedicated to one fault. The general structure of the RBF classifier used is shown in Fig. 5.6, where $e^2_{NP}$, $e^2_{O2}$ and $e^2_{SV}$ are the input signals to the RBF classifier later classified faults according to the individual symptoms and the location of each faults.

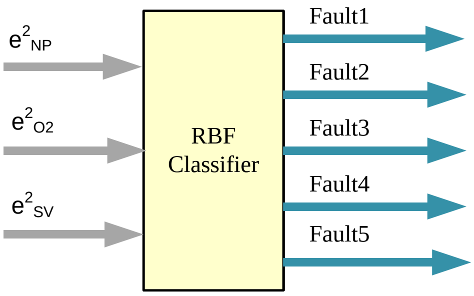


**Fig. 5.6** The block diagram of fault isolation

.

In this work, the RBF classifier employed the Gaussian basis function as defined in Chapter 4 under section 4.2.1 until 4.2.2.2 to do fault isolation.

## 5.5 Summary

In this chapter, the design structure of FDI process is presented. The process started from fault detection part where the residual signals are calculated by combining three modelling prediction errors as in equation (42) to do fault detection. Later, these modelling prediction errors are fed into the RBF classifier to conduct the fault isolation process. Here, the Gaussian basis function is used to train each fault according to individual fault before the classification can be carried out.

# CHAPTER 6

# SYSTEM CONTROL AND SIMULATING FAULTS

## 6.1 Closed-loop control system design

Feedback control systems are often referred to as closed-loop control systems. In practice, the terms feedback control and closed-loop control are used interchangeably. In a closed-loop control system the actuating error signal, which is the difference between the input signal and the feedback signal, is fed to the controller so as to reduce the error and bring the output of the system to a desired value. The term closed-loop control always implies the use of feedback control action in order to reduce system error. An advantage of the closed-loop control system is the fact that the use of feedback makes the system response relatively insensitive to external disturbances and internal variations in system parameters. It is thus possible to use relatively inaccurate and inexpensive components to obtain the accurate control of a given plant, whereas doing so is impossible in the open-loop case (Lecture notes, ME 475).

A FC should be controlled effectively to ensure (a) the system supply of required power in the represent of rapid variations in the external loads, (b) high efficiency of the system, and (c) long life of the system, among others. Another control problem in FC is the phenomena of oxygen starvation, which may occur when there is a sudden large increase in the load power (Bavarian *et al*., 2010). The accuracy and transient behaviour of sensors and actuators are of high importance in optimal control of FCs.

Actuators used in the control of FC stacks include valves, pumps, compressors motors, expander vanes, fan motors, humidifiers and condensers (Bavarian *et al*, 2010).

### 6.1.1 Feedforward control

The feedforward control is used to control compressor motor voltage, CV based on the current drawn from the FC stack. In this work, a look-up table acts as feedforward control as presented in Table 6.1 with respect to the signal range of stack current, SC ranging from 100 to 300 amperes. To design the feedforward controller, the SC signal is adjust at the value illustrated in Table 6.1 and fed to the FC stack while tuning the CV until $\lambda_{O2} = 2$. The nominal value for oxygen excess ratio is selected at $\lambda_{O2} = 2$, which correspond to the maximum FC net power for the nominal current (Pukrushpan et al., 2002).

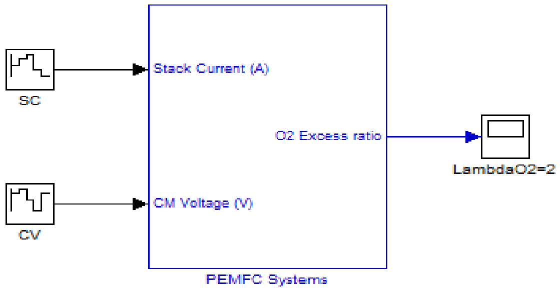
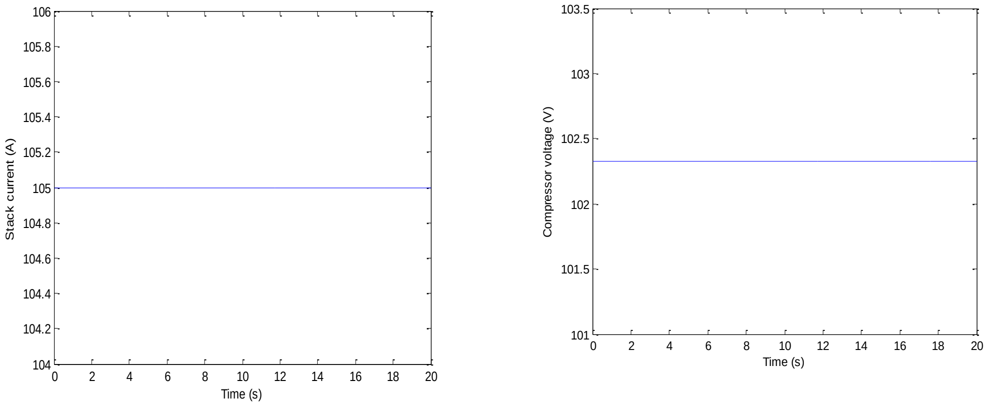
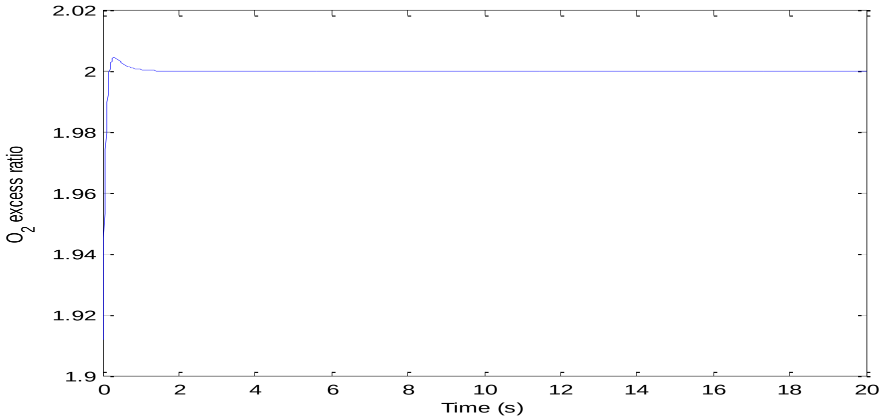


**Fig. 6.1** Simulink model to build look-up table values



(a) (b)



(c)

**Fig 6.2** The construction of look-up table based on SC-CV tune technique

**Table 6.1** The design of feedforward controller

| SC(A) | CV(A) | $\lambda_{O2}$ | SC(A) | CV(A) | $\lambda_{O2}$ |
|---|---|---|---|---|---|
| 105 | 102.33 | 2 | 190 | 163 | 2 |
| 110 | 105.98 | | 195 | 166.45 | |
| 115 | 109.98 | | 200 | 169.85 | |
| 120 | 113.15 | | 205 | 173.25 | |
| 125 | 116.77 | | 210 | 176.65 | |
| 130 | 120.43 | | 215 | 179.85 | |
| 135 | 124.05 | | 220 | 183.25 | |
| 140 | 127.68 | | 225 | 186.45 | |
| 145 | 131.29 | | 230 | 189.65 | |
| 150 | 134.89 | | 235 | 192.75 | |
| 155 | 138.48 | | 240 | 195.95 | |
| 160 | 142.05 | | 245 | 199.05 | |
| 165 | 145.58 | | 250 | 202.05 | |
| 170 | 149.15 | | 255 | 205 | |
| 175 | 152.65 | | 260 | 208 | |
| 180 | 156.10 | | 265 | 211 | |
| 185 | 159.60 | | 270 | 214.10 | |

Fig. 6.1 illustrated the Simulink model used to generate the feed-forward control where SC is fixed at certain value while tuning the CV until it $\lambda_{O2}$=2. The

response of $\lambda_{O2}$ can be referred in Fig. 6.2 when SC=105 A and CV=102.33 V, the value of $\lambda_{O2}$ =2. The lookup-table is designed starting from the minimum to the maximum value of SC. The complete look-up table used as the feedforward control is shown in Table 6.1.

### 6.1.2 Feedback control

The proportional-integral-derivative (PID) controller has been used as a closed-loop controller to overcome the effect of disturbances and also to improve the response with respect to reference signal. The PID controller equation is in the form of (Ogata, 1997):

$$PID_{controller} = K_p + \frac{K_i}{s} + K_d s \tag{43}$$

where Kp, Ki and Kd is proportional, integral and differential gain. The PID controller used in this work has been fine tuned until $\lambda_{O2}$ = 2. Therefore, the final value of the PID controller equation is given by:

$$PID_{controller} = 200\left(1 + \frac{1}{0.6153s} + 0.05s\right) \tag{44}$$

Fig. 6.3 shows the overall control system of feedforward and a closed-loop control implemented in this work. In the Simulink model, the SC acts as a disturbance to the PEMFC systems with a reference input of $\lambda_{O2}$ = 2. The output of $\lambda_{O2}$ needs to be maintained in order to avoid oxygen starvation from happening. The SC being satisfied by the FC enters the control system as a measureable disturbance and therefore justifies by the use of feedforward strategy to compensate for this effect and for this a feedback strategy has to be devised in order to keep $\lambda_{O2}$ as close as possible to the set point.

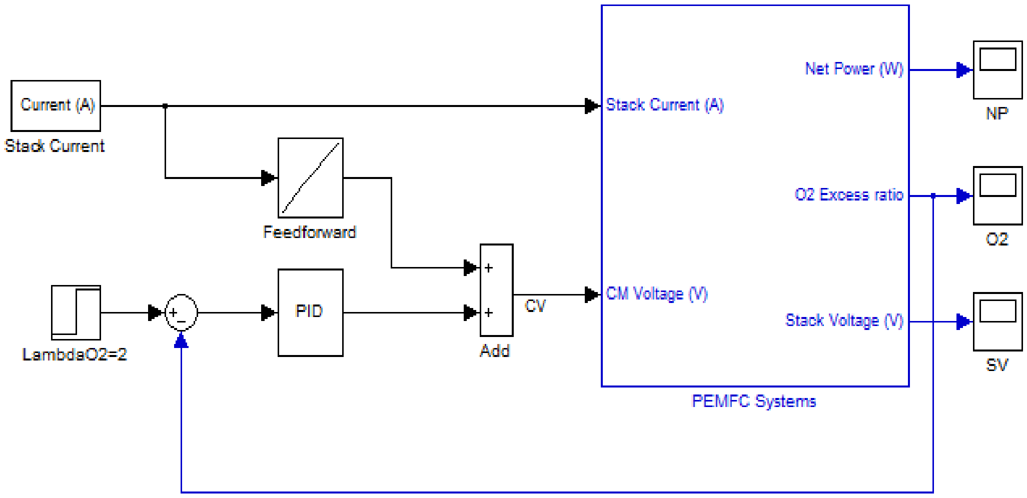


**Fig. 6.3** The Simulink model feedforward-feedback controller of $\lambda_{O2}$

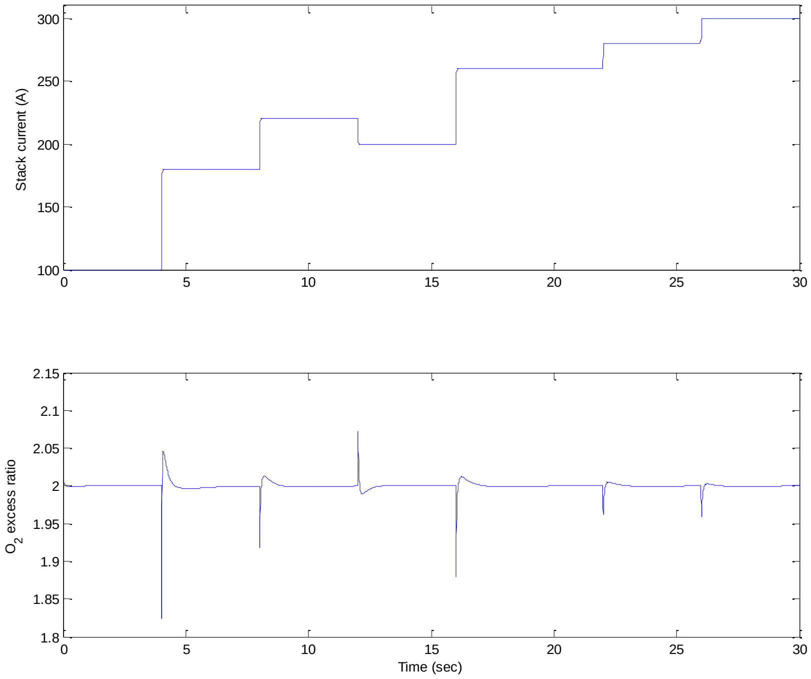


**Fig 6.4** The response of $\lambda_{O2}$ using feedforward and feedback control

Fig. 6.4 shows the response of $\lambda_{O2}$ when both controller are combined. As can be seen, when there is a signal change, $\lambda_{O2}$ also changes and then becomes steady at 2 when the SC is constant. This feed-forward controller produces compensation for the external disturbance; at the same time the PID controller is used to form the feedback control and to control uncompensated effects as well as the steady state (Zhai and Yu, 2008). Fig. 6.5 shows the simplified block diagram of Simulink model build with the combination of feed-forward and feedback control used during the FDI simulation.

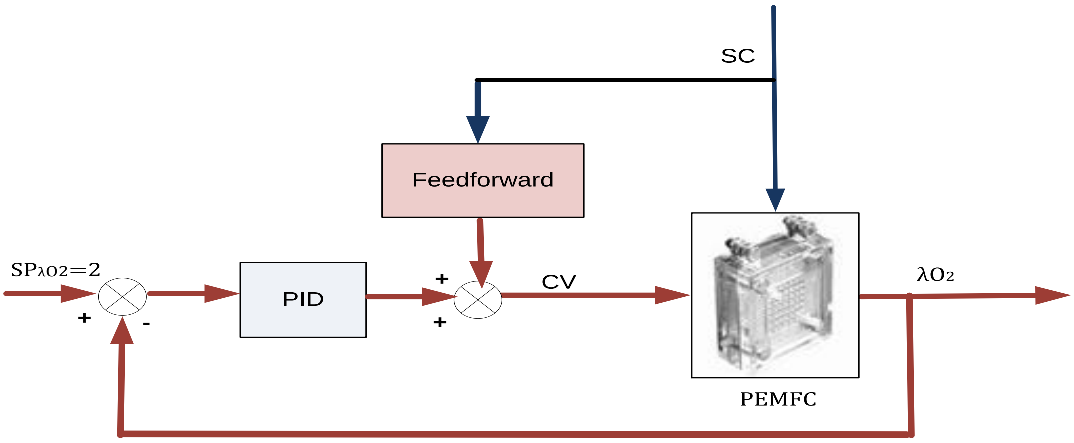


**Fig 6.5** Feedforward plus feedback control scheme

## 6.2 Simulating faults

There are several algorithms for fault simulation, the simplest being the serial fault simulation. Here the fault simulation is performed for one fault at a time. As soon as one fault is detected, the simulation is stopped and a new simulation is started for another fault. This fault simulation method, though simple, is very time consuming. Another method of fault simulation, which simulates more than one fault are called parallel fault simulation. In this work, both algorithms were performed but here only the parallel fault simulations are presented.

The faults acting upon a system can be divided into three types of faults:

i) Actuator faults.

ii) Sensor / Instrument faults.

iii) Component faults.

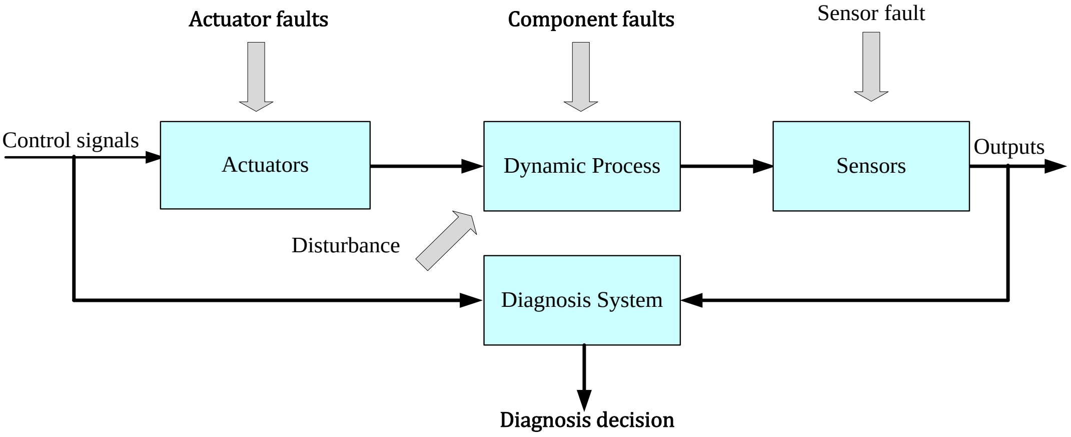


**Fig. 6.6** The block diagram of closed-loop systems

Fig. 6.6 shows the general block diagram of a closed-loop systems with the presence of faults considered during the plant operating. In this research we investigate FDI approaches to diagnose all three types of faults; actuator, component and sensor (instrument) faults. The fault sources chosen to analyse are:

i) Actuator fault –Compressor voltage.

ii) Component fault – Manifold leakage.

iii) Sensor faults – NP sensor, $\lambda_{O2}$ sensor and SV sensor.

The FC system power response depends on air and hydrogen feed, flow and pressure regulation, heat and water management (Pukrushpan *et. al*., 2002). During transient, the FC stack breathing control system is required to maintain optimal temperature, membrane hydration, and partial pressure of the reactants across the membrane in order to avoid degradation of the stack voltage, thus maintain high efficiency and extend the life of the stack (Yang, 1998). The air flow and pressure are the key controlled components of a FC stack for an efficient and dynamic performance of the FC. FC is in risk of oxygen starvation during high current demand and fast load changes. Oxygen starvation is defined as the ratio of the partial pressure of reserving oxygen with that of used oxygen. It comes along with a drop in partial pressure of oxygen (Pukrushpan *et al.*, 2004a). The oxygen ratio, $\lambda_{O2}$, must be kept above a minimum limitation for normal operation.

In this study, five faults are introduced to a known test-bench PEMFC based on the model developed by Michigan University (Kamal and Yu, 2011). The PEMFC simulator was modified to include five possible fault scenarios which may occur during the normal operation of PEMFC systems as shown in Fig. 6.7 in terms of block diagram. The five faults introduced to the PEMFC schematic is shown as in Fig. 6.8.

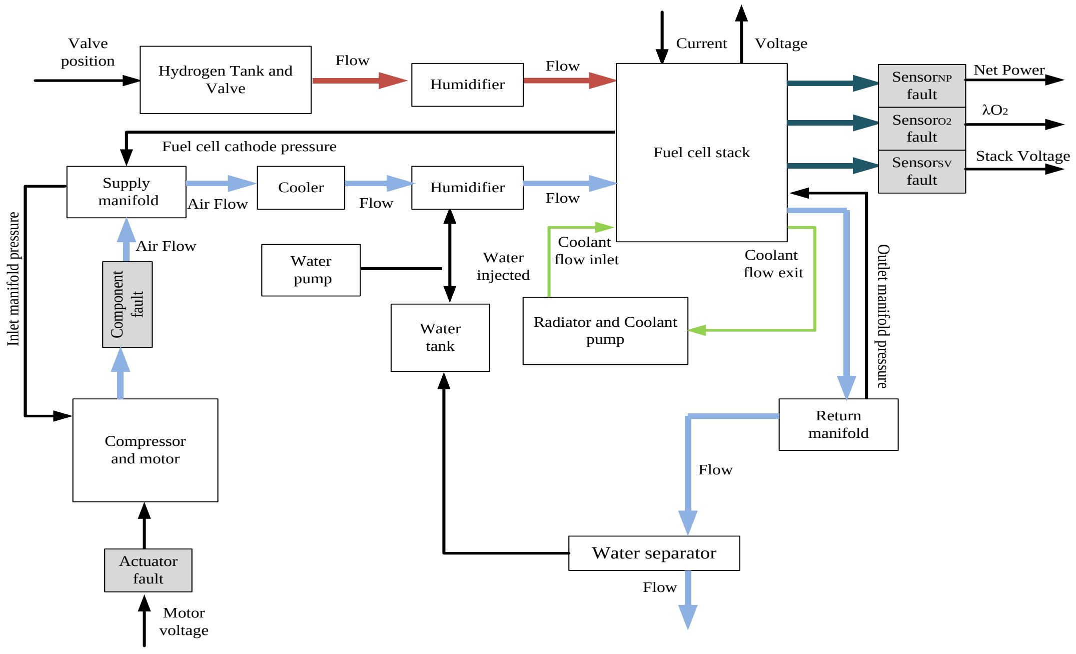


**Fig. 6.7** The overall PEMFC modified with five faults

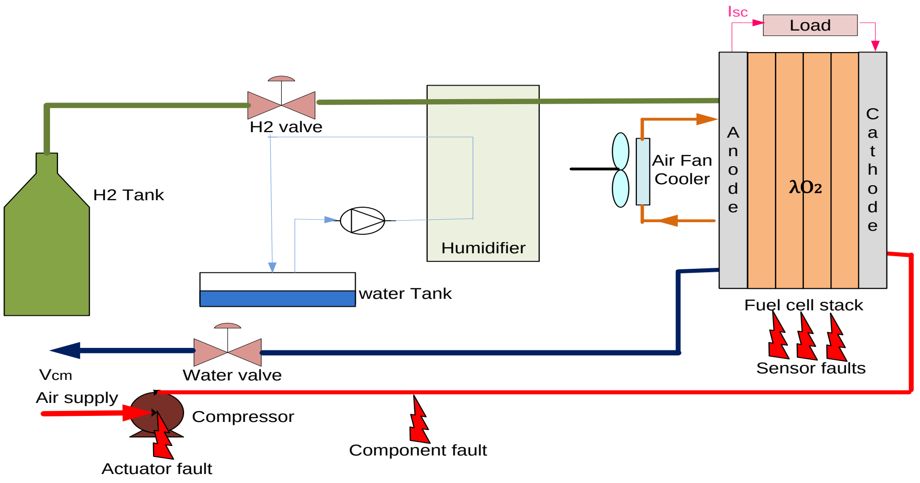


**Fig. 6.8** The schematic of PEMFC systems with five faults

The fault simulation is conducted under two control loop scheme: Firstly is under the open-loop scheme and secondly under closed-loop scheme. For both control systems, five types of faults are being introduced. In this work, all five faults are introduced to the PEMFC dynamic systems. The fault is introduced using a block diagram of a step input.

### 6.2.1 Simulating actuator and component faults

Mostly centrifugal compressor used in FCs is susceptible to surge and choke that limit the efficiency and performance of the compressor. In a high pressure PEMFC, a compressor supplies air to the cathode. The compressor itself consumes up to 30% of FC generated power and therefore, has a direct influence on overall system efficiency (Vahidi *et al*., 2007). The challenge is that oxygen reacts instantaneously as current (load) is drawn from the stack, while the air supply rate is limited by the manifold dynamics, compressor surge and choke constraints (Pukrushpan *et al*., 2004; Vahidi *et al*., 2004). Surge causes large variations in flow and sometimes flow reversal through the compressor and can even damage the compressor (Vahidi *et al*., 2007).Air leakage in the supply manifold makes the pressure in the cathode decrease. Therefore, to collect the FC stack data subjected to the air leak fault, equation (5) in section 4.2.2 is modified to:

$$\frac{dp_{sm}(t)}{dt} = \frac{\gamma R}{M_a^{atm} V_{sm}} \left[ W_{cp}(t) T_{cp}(t) - W_{sm,out}(t) T_{sm}(t) - \Delta l \right] \qquad (45)$$

where $\Delta l$ is used to simulate the leakage from the air manifold, which is subtracted to increase the air outflow from the supply manifold. $\Delta l = 0$ represents that there is no air leakage in the supply manifold.

Most of process applications are connected by a pipe, wire or conduit which experienced leakage during the operation. In PEMFC dynamic systems, the operation of the liquid and gases are connected by the manifold. Leakage is a common fault which normally happens in every process or mechanical operation. If a leakage happens in the system, the pressure during the operation will decrease and this will affect the overall performance of the PEMFC dynamic system. Therefore, in this work a component fault which is leakage is considered happens in the FC systems. The fault appears at the return manifold outlet which implies a reduction of outlet flow. Fig. 6.9 shows the Simulink model where the component and actuator fault being injected to the FC stack process to reflect that faults have occurred in the system.

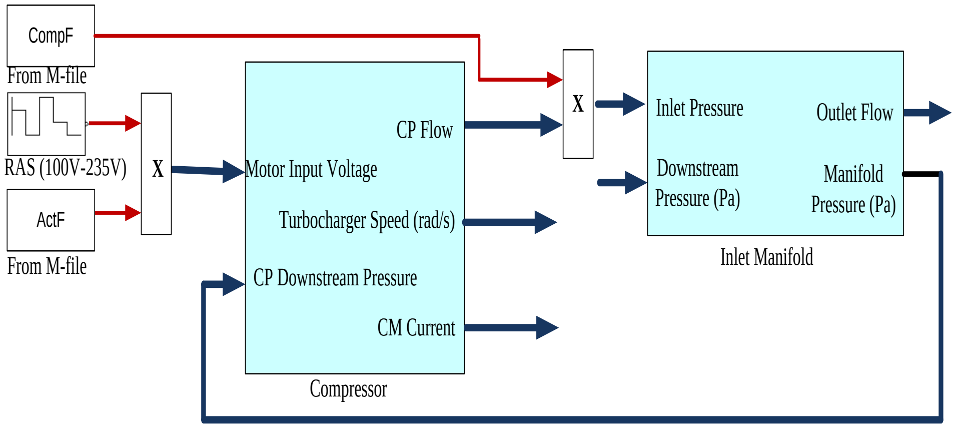


**Fig. 6.9** The Simulink block of component and actuator faults for open-loop systems

The faults simulation of actuator and component for closed-loop system in the Simulink model is presented in Fig. 6.10 which consist of the feedforward and feedback control inside the block diagram.

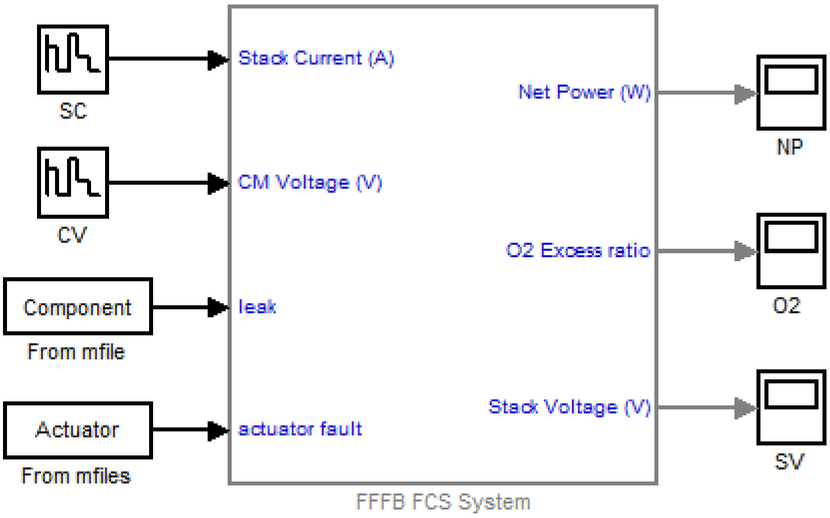


**Fig. 6.10** The Simulink block of component and actuator faults for closed-loop systems

### 6.2.2 Simulating sensor faults

Typically, the FC characteristics are given in the form of a polarization curve, which is the plot of FC voltage versus current density. Since FCs are connected in series to form the stack, the total stack voltage is $V_{st}=\eta.V_{cell}$ and the power is $P_{st}=V_{st}.I_{st}$. Part of the stack power is used to drive the compressor motor. Therefore, $P_{net}=P_{st}-P_{cm}$ (Pukrushpan *et al.*, 2002). While, the oxygen supply to the cathode is one of the key factors in operation of a FC stack. When the current is drawn from a FC, the air supply system should replace the reacted oxygen. Otherwise the cathode will suffer from oxygen starvation which damages the stack and limits the power response of FC (Vahidi *et al.*, 2004). Based on these reasons, fault is introduced at the output of these three sensors used inside the FC stack. Fig. 6.11 shows the construction of faults implemented in the Matlab/Simulink model.

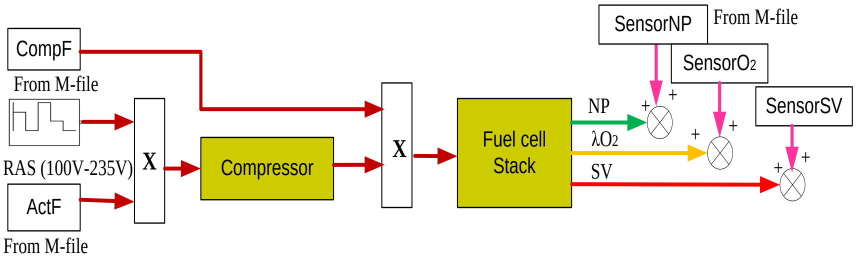


**Fig. 6.11** The simulink block of sensors faults

### 6.2.3 Data collection used in FDI

The implementation of FDI is done in the MATLAB R2009a/Simulink environment. The parameters for simulations are based on an FC prototype vehicle (Pukrushpan *et al.*, 2004a). Before FDI methodology can be applied, healthy data sets from the PEMFC systems is collected using two types of input signals; RAS and step signals. The RAS input signals of SC and CV are generated randomly within their minimum and maximum value stated in Table 6.2 while for the step signals, SC and CV are varied as discussed in Chapter 7.

**Table 6.2** The values of SC and CV

| Parameters | Minimum | Maximum |
|---|---|---|
| SC | 100 A | 300 A |
| CV | 100 V | 235 V |

**Table 6.3** Faulty data sets used during the simulation of RBF and MLP networks model

| Sample | Input type | Control scheme | Types of fault | No. of data | Fault size | Data range |
|---|---|---|---|---|---|---|
| 6data20b | RAS | Open-loop system | $\text{Sensor}_{NP}$ | 200 | +10% | 1000-1200 |
| | | | $\text{Sensor}_{O2}$ | 200 | +10% | 2000-2200 |
| | | | $\text{Sensor}_{SV}$ | 200 | +10% | 3000-3200 |
| | | | Component | 200 | +10% | 4000-4200 |
| | | | Actuator | 200 | +10% | 5000-5200 |
| noF5000 | Step input | Open-loop system | Actuator | 100 | -7% | 500-600 |
| | | | Component | 100 | -7% | 1500-1600 |
| | | | $\text{Sensor}_{NP}$ | 100 | +10% | 2500-2600 |
| | | | $\text{Sensor}_{O2}$ | 100 | +10% | 3500-3600 |
| | | | $\text{Sensor}_{SV}$ | 100 | +5% | 4500-4600 |
| FFFB1500 | Step input | Closed-loop system | $\text{Sensor}_{NP}$ | 50 | +10% | 450-500 |
| | | | $\text{Sensor}_{O2}$ | 50 | +10% | 650-700 |
| | | | $\text{Sensor}_{SV}$ | 50 | +10% | 850-900 |
| | | | Component | 50 | -10% | 1050-1100 |
| | | | Actuator | 50 | -10% | 1250-1300 |

Once the healthy data sets are obtained from these three samples, later five types of faults are injected to the PEMFC systems model as explained above. These faulty data sets are collected because later these faulty data sets are used to test both of networks algorithm. In this work, a lot of samples have been collected which can be found in the conference papers attached as appendixes but only three samples of data sets are shown here to summarize the work that has been conducted as well as to cover different types of controller. Table 6.3 summarizes the data sets which are used in the thesis writing. There are three sample measurement consisting of 6000 samples, 5000 samples and 1500 samples of faulty data obtained during the fault simulation of the model.

## 6.3 Summary

A dynamic FC model suitable for control studies was developed by researchers from University of Michigan. The transient phenomena capture in the model include the flow and inertia dynamics of compressor, the manifold filling dynamics (both anode and cathode), and membrane humidity. These variables affect the FC stack voltage, and thus FC efficiency and power (Pukrushpan *et al*., 2004) as well as the overall of PEMFC performances. Based on these factors, this simulator model has been chosen for FDI analysis where the PEMFC dynamic systems has been modified to introduce five types of faults happens during the normal operation. Table 6.3 summarizes the three data samples used in this work. The purpose of doing the FDI under open-loop systems is to test the effectiveness of the RBF algorithm and MLP algorithm developed in this work before it can be implemented to the closed-loop systems. Besides that, different types of inputs signals are also being injected for this FDI analysis.

# CHAPTER 7

# FAULT DETECTION

## 7.1 Introduction

Process faults have a serious impact on product quality, safety, productivity and pollution level. In order to detect, diagnose and correct these abnormal process behaviors, efficient and advanced automated diagnostic systems are of great importance to modern industries (Rosich *et al*., 2007). Once a fault has been detected and its evolution is monitored, the severity of that fault can be evaluated and a decision can be made on the course of action to take. Monitoring creates the opportunity to strategically plan and schedule outrages and to manage equipment utilization and availability (Gibeault and Kirkup, 1995).

The International Federation of Automatic Control (IFAC) SAFEPROCESS Technical Committee defines a fault as an unpermitted deviation of at least one characteristic property or parameter of the system from the acceptable/usual/standard (Isermann, 1997; Schrick, 1997). Therefore, the fault is a state that may lead to a malfunction or failure of the system (Isermann, 1997). Fault detection is a procedure that reports the occurrence of a fault at a particular time. The task of fault detection is to determine the source of fault and its location.

The implementation of FDI in this work is done in the MATLAB R2009a/Simulink environment. Two different types of signals are fed to the FC stack under open-loop and closed-loop control. The signals are the RAS and step inputs. It is a MIMO system where the inputs are SC and CV while the outputs are NP, $\lambda_{O2}$ and SV. Five main types of faults commonly happen in the process control environment have been considered happened in the FC nonlinear simulation model. Firstly the effect of actuator fault with valve experienced surge or choke during the operation and a leakage in the manifold supply which makes the pressure decrease in the supply. Then, sensors at three outputs experience a malfunction of over reading situation.

## 7.2 Fault detection for open-loop system

Before the algorithm for the RBF network model and MLP network model can be implemented to closed-loop control, it is tested for open-loop control due to its simplicity. From the stability point of view, the open-loop system is easier to build because system stability is not the major problem. On the other hand, stability is a major problem in the closed-loop control system, which may tend to overcorrect errors that can cause oscillations of constant or changing amplitude. Because of these reasons, both; the RBF network model and MLP network model have been tested under open-loop condition.

In this work, the data were divided into two groups; healthy and faulty data set. Firstly, a healthy data set is used to train and test with these algorithms mentioned in Chapter 4 for the both neural network models; the RBF network and the MLP network models. In the training section, once the minimum prediction error models are obtained, both networks are tested to check whether the model structure is satisfied or not using the algorithm mention in Chapter 4. Once the result is satisfactory, then the model structure is tested with faulty data sets.

### 7.2.1 FDI using RBF modelling

As mentioned above, the data sets mentioned in Table 6.3 under Chapter 6 are used to test the RBF algorithm. There are two types of inputs used; RAS and square inputs which are applied to open-loop control as discussed below.

#### 7.2.1.1 RAS inputs

Here, the input signals are randomly generated between the minimum and maximum values in the range of SC and CV. The sampling time was set to 0.1s and consists of 6000 samples. Fig. 7.1 shows the example of RAS excitation inputs injected to both inputs; SC and CV.

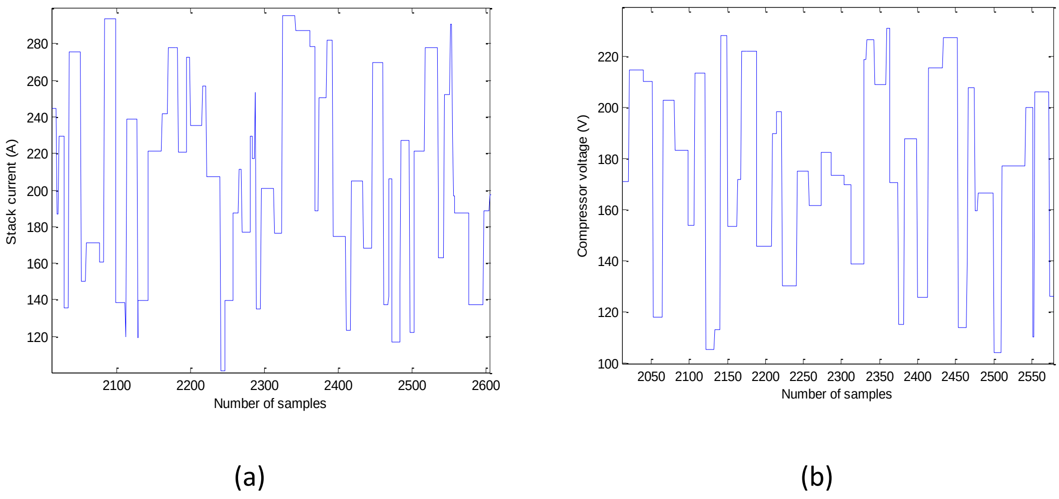


**Fig. 7.1** The RAS signals used as the excitation inputs

Fig. 7.2 shows the healthy and faulty outputs of NP, $\lambda_{O2}$ and SV with respective faults during the injection of RAS as inputs. The five faults simulated in the MATLAB R2009a/Simulink model was simulated by superimposing a positive ten percent. As can be seen here, the faults are very small compared with the plant outputs and with raw eyes these faults cannot be detectable. However, there is a big gap in SV output when sensor$_{SV}$ fault occurred at k=3000 to k=3200.

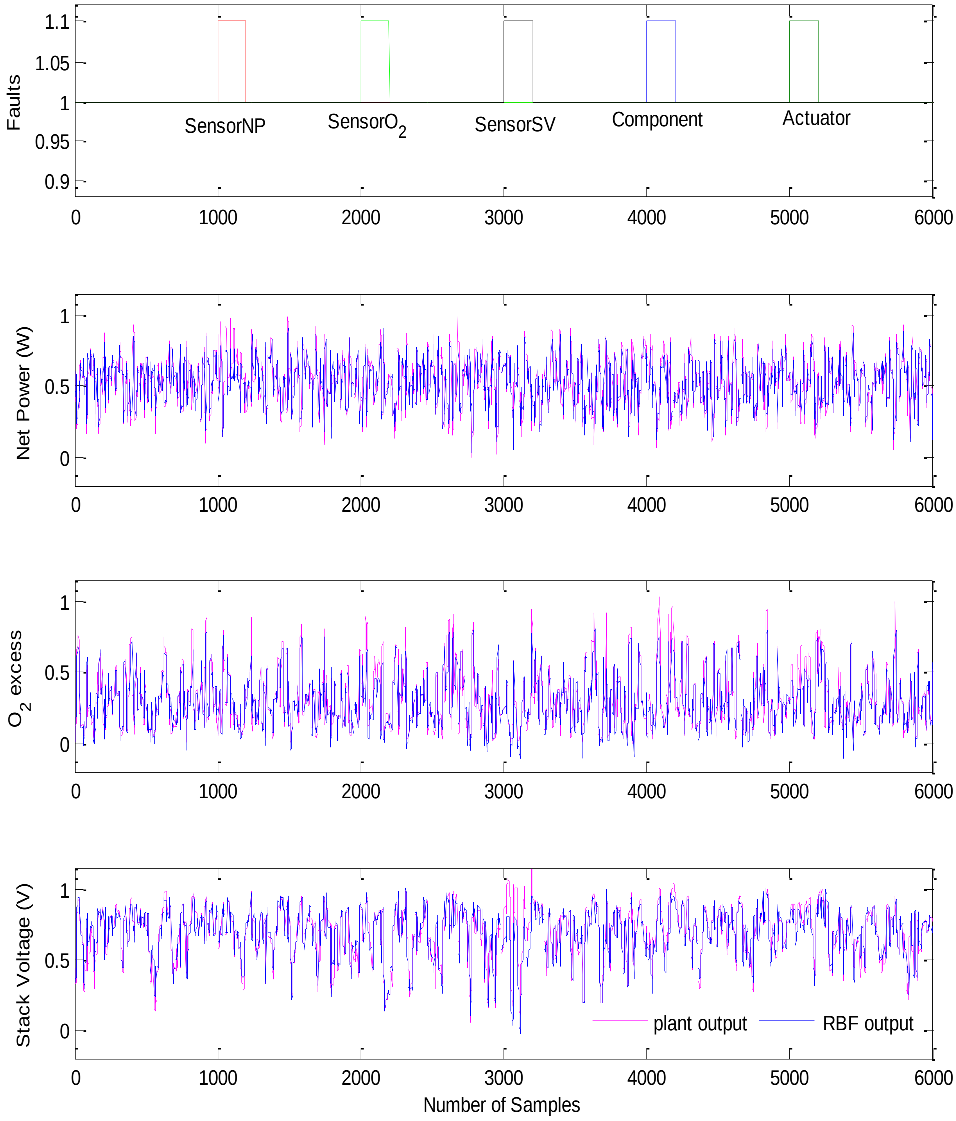


**Fig. 7.2** The testing process of healthy and faulty data sets

Based on Fig 7.2, next step is to calculate the residuals generated from these three outputs measurement based on the difference between the plant outputs and the RBF networks model outputs. Even though the residual signals are obtained, the five faults exiting in these output signals cannot be detected due to noise existence. The mixture of faulty signals and a noisy signal can be referred in Fig 7.3. Anyway there is a change of amplitude at k=3000 to k=3200 due to sensor$_{SV}$ fault.

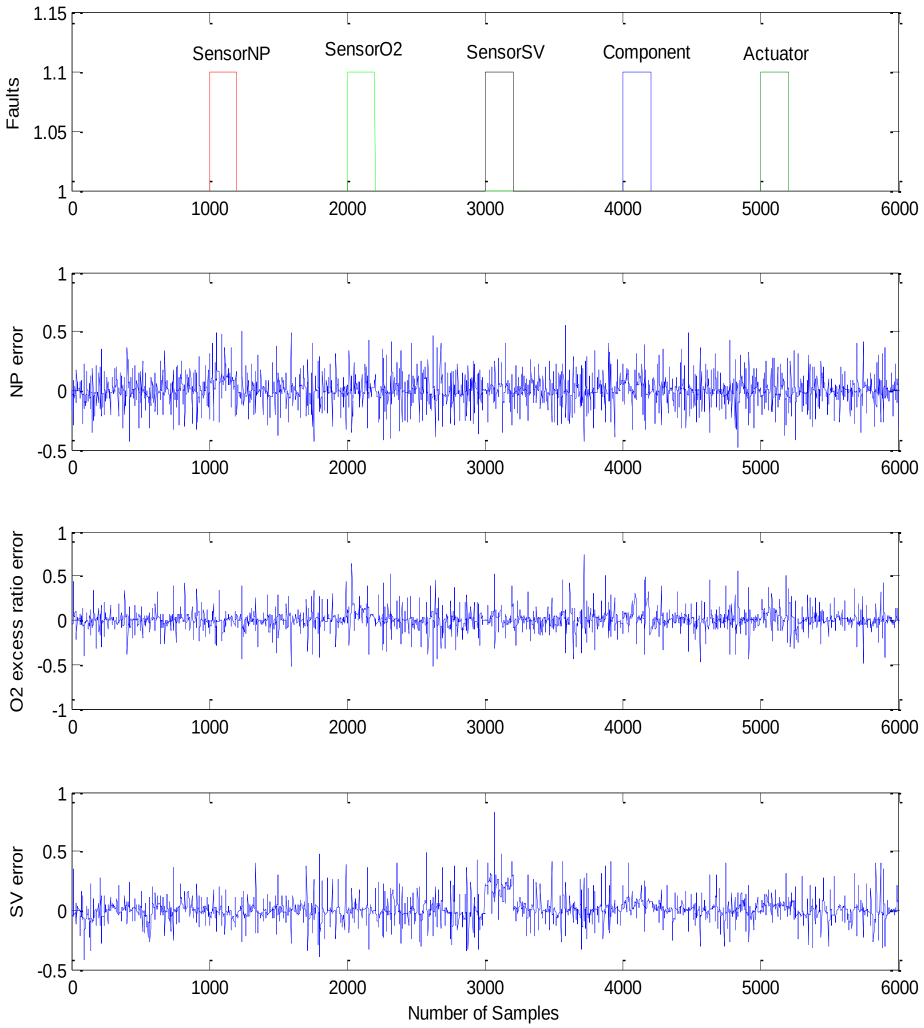


**Fig. 7.3** The modelling prediction errors

The modelling errors obtained from the difference of plant and the RBF networks contains a noisy signals are filtered. The filter equation used in this work is given by:

$$filter\,(i) = 1 * error(i-1) + error(i) \qquad (46)$$

where i is the number of samples used during the simulation.

Therefore, to enhance signal-to-noise ratio the filtered modelling errors are squared. The three squared and filtered model prediction errors are displayed in Fig. 7.4, with the simulated faults on the top of the figure for easy observation. In Fig. 7.4 it can be observed that these signals can be used for fault detection, but cannot be used for fault isolation. From observation, the simulation result shows that there are more than one faults exist in the output signals.

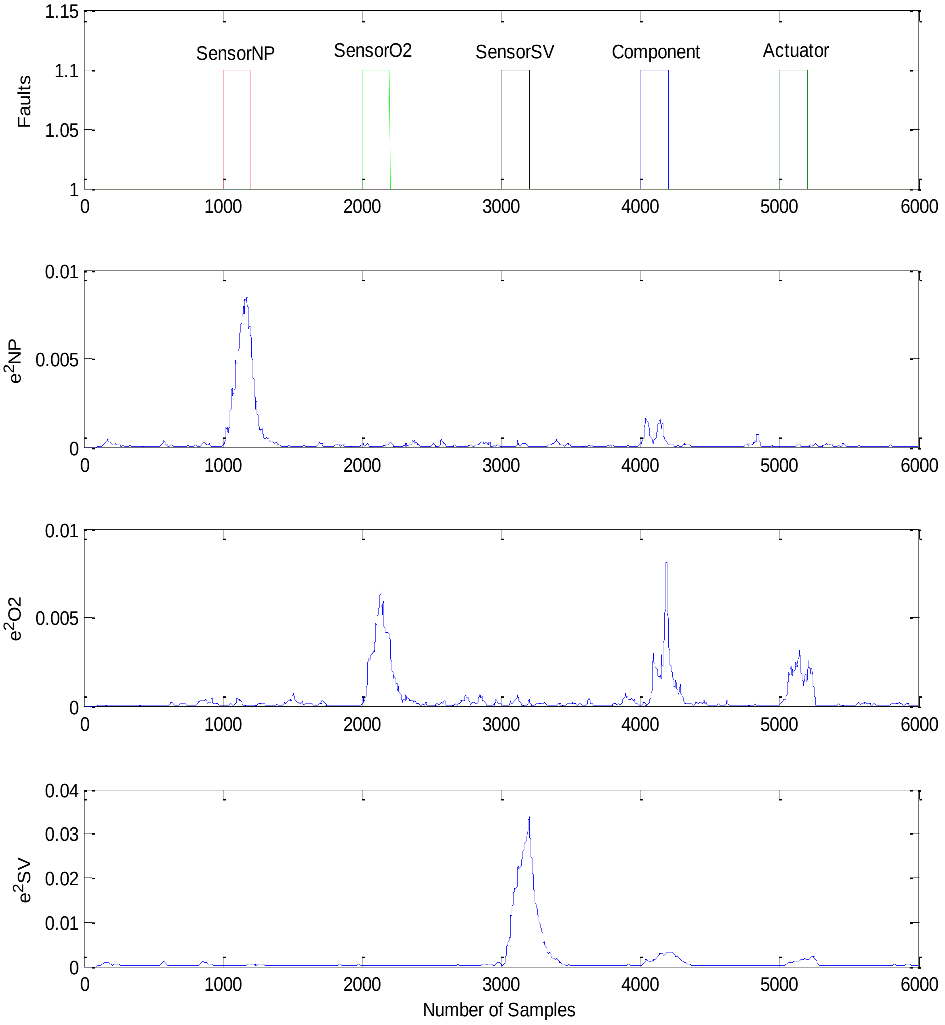


**Fig. 7.4** Filtered and squared model prediction errors

It can be clearly seen, for example, between samples k=1000 and k=3000 in Fig. 7.4 that there is a fault occurred in the filtered squared modelling error signal of output NP and SV based on sensor fault itself. The fault caused by these sensors can clearly be identified in respective outputs. Again, by referring to Fig. 7.4, the filtered squared modelling error of $\lambda_{O2}$ contains three types of faults due to sensor$_{O2}$, component and actuator faults. Therefore it is impossible to identify individual fault happened during the operation of the system according to the fault detection signal only. By applying the residual generation equation, the residual generated is shown in Fig. 7.5.

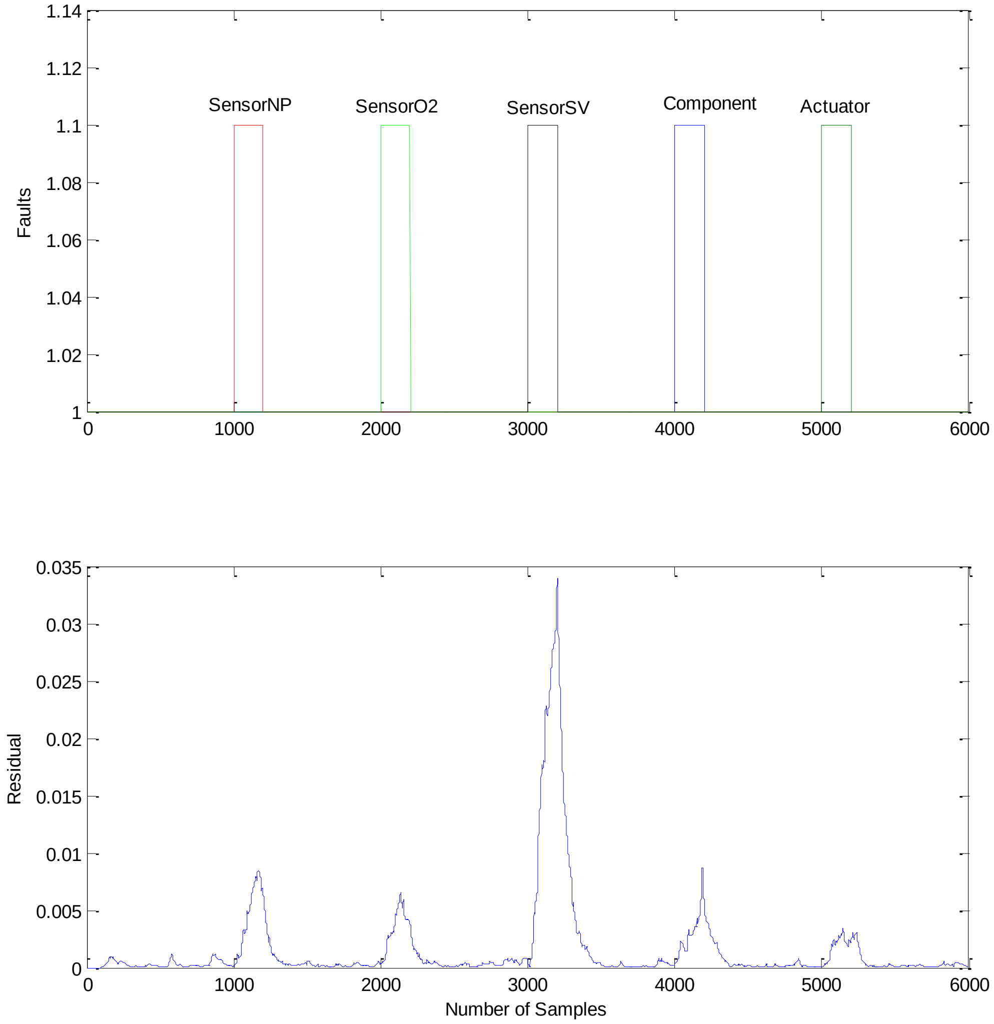
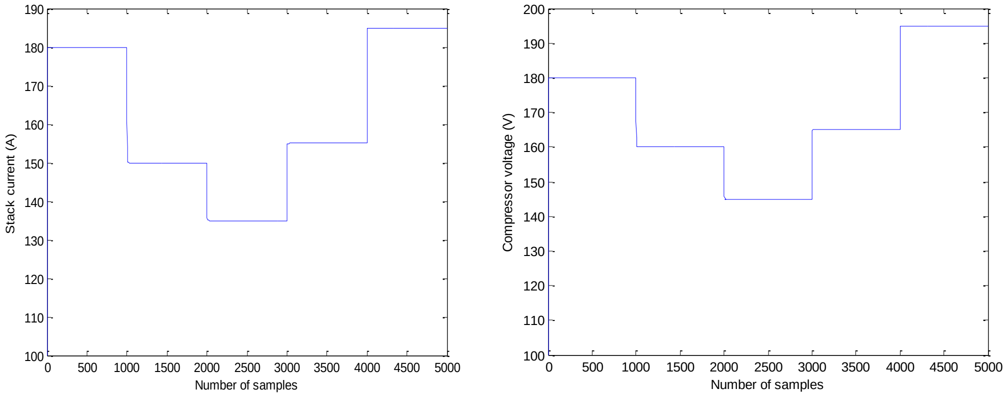


**Fig. 7.5** Residual signal for the five simulated faults

### 7.2.1.2 Step inputs

A series of step changes in SC and CV are applied as excitation inputs as shown in Fig. 7.6 where both signals changing with 100 interval in the signals ranging from 180A to 185A for SC and 180V to 195V for CV. Here, the input signals are randomly generated between the minimum and maximum range of SC values and CV values.



(a) (b)

**Fig. 7.6** The step inputs signals of SC and CV

Fig. 7.7 shows the output simulation of NP, $\lambda_{O2}$ and SV when the faults occur in the system for testing data set. The output signals shows that when there is a fault the signal shape is changed where either it goes down or up at that particular time. However, when there is no fault the signal of plant output is equal to RBF network output. To see more details, Fig. 7.7 had been enlarged. Fig. 7.8 shown a closed-up response of signal at k=1400-2700. Referring to this figure, there are two faults occurred at k=1500-1600 due to component fault and at k=2500-2600 due to sensor$_{NP}$ fault. Here, component fault experienced a -7% fault size which clearly can be seen in Fig. 7.8. The effect of component fault influences the outputs of NP, $\lambda_{O2}$ and SV while sensor$_{NP}$ just influence its own output not the other two outputs with a +10% fault size.

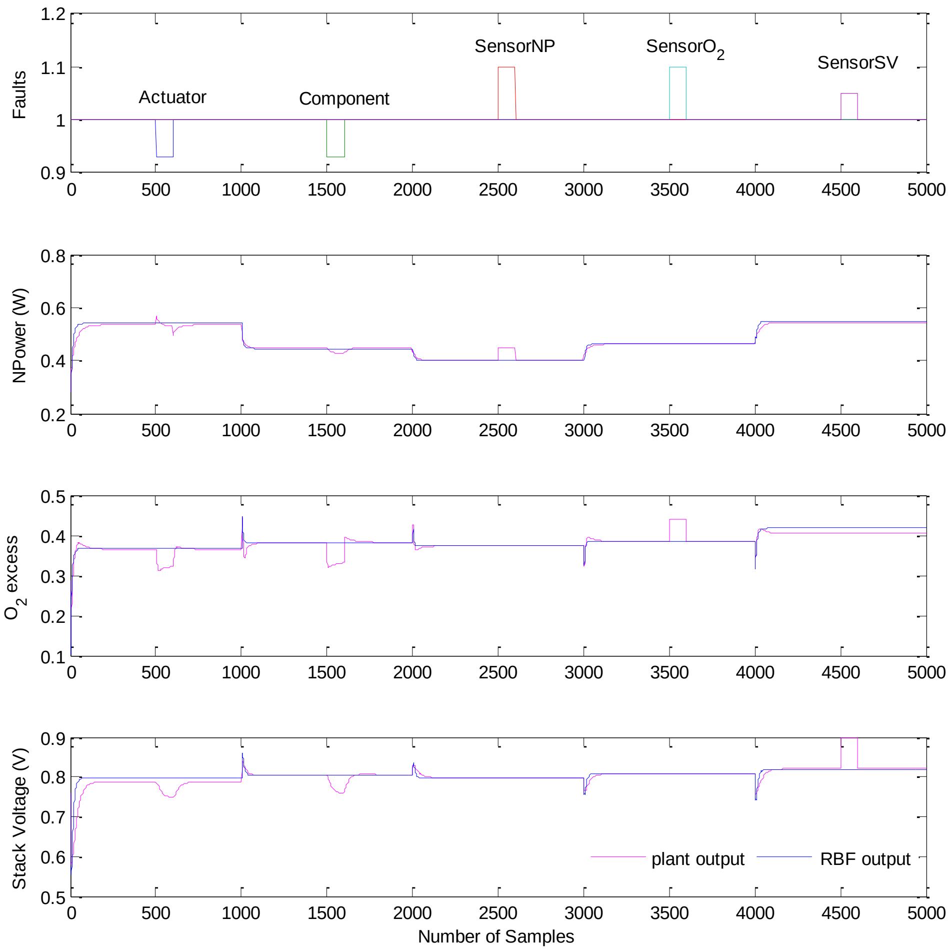


**Fig. 7.7** The testing process of healthy and faulty data sets

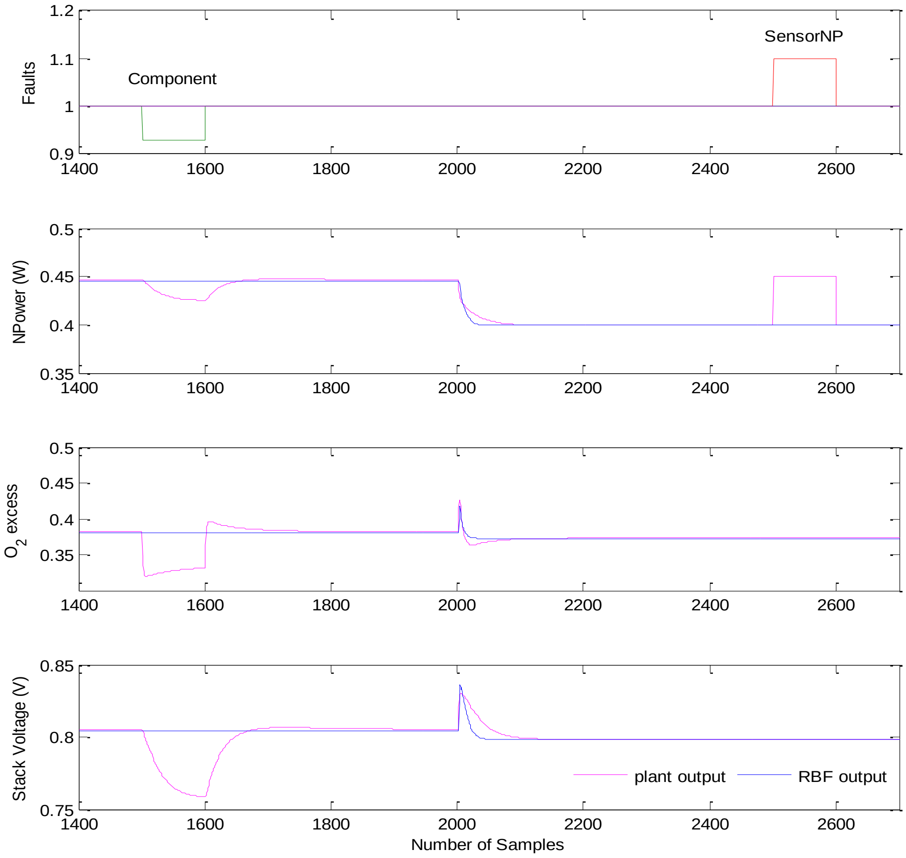


**Fig. 7.8** Magnified outputs from Fig. 7.7

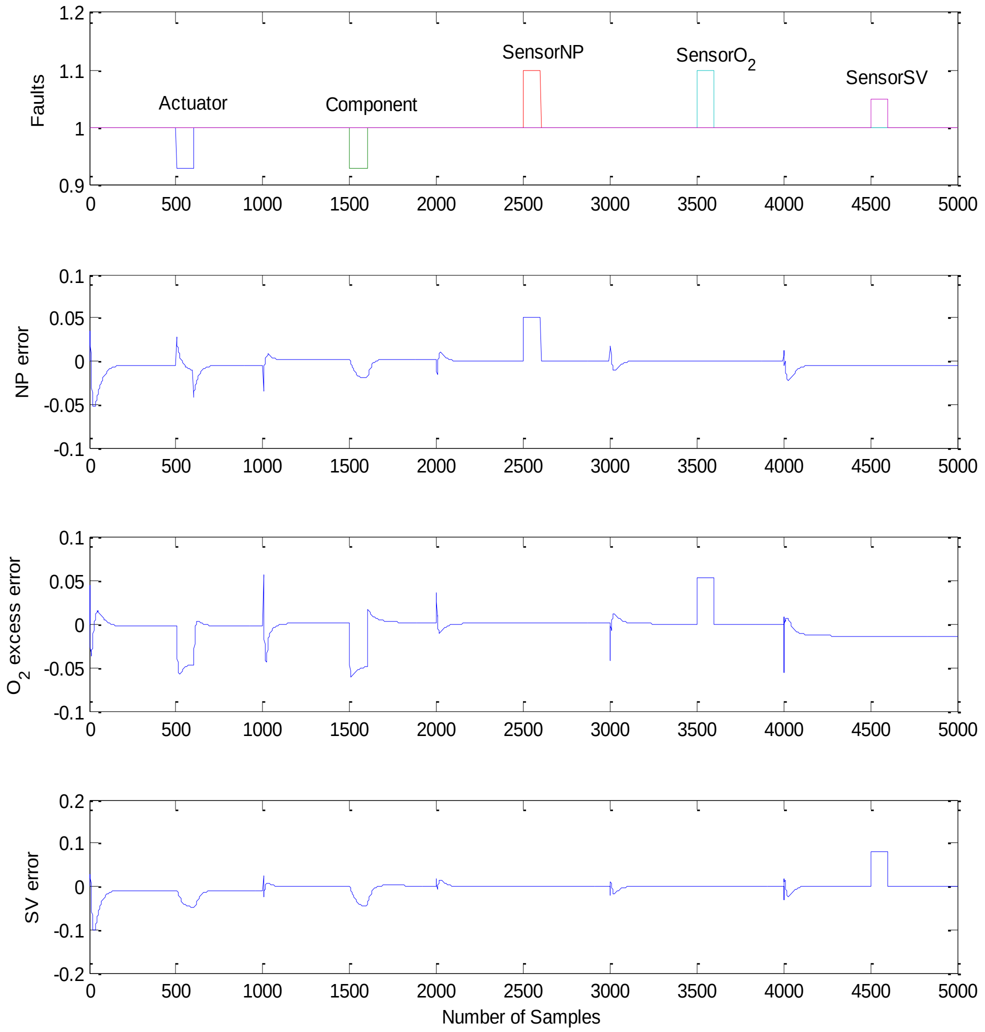


**Fig. 7.9** The modelling prediction errors

The output signals in Fig. 7.9 have negative and positive polarity residual signals which makes the analysis of faults difficult. Thus to make these residual signals easier to analyse, the modelling prediction errors of each outputs are squared and later it will be filtered to makes the signals smoother. Once this has been done, these faults can easily be analysed and determined whether faults have occurred or not in the system. Fig. 7.10 shows the simulation results based on step input signals. From the observation, it is easily to detect that there is a fault due to $sensor_{NP}$ at k=2500 and $sensor_{SV}$ fault at k=4500 due to the fault size is higher than the others.

However it is hardly to define which faults occurred in the output signals of $\lambda_{O2}$ because there are more than one fault signals inside it.

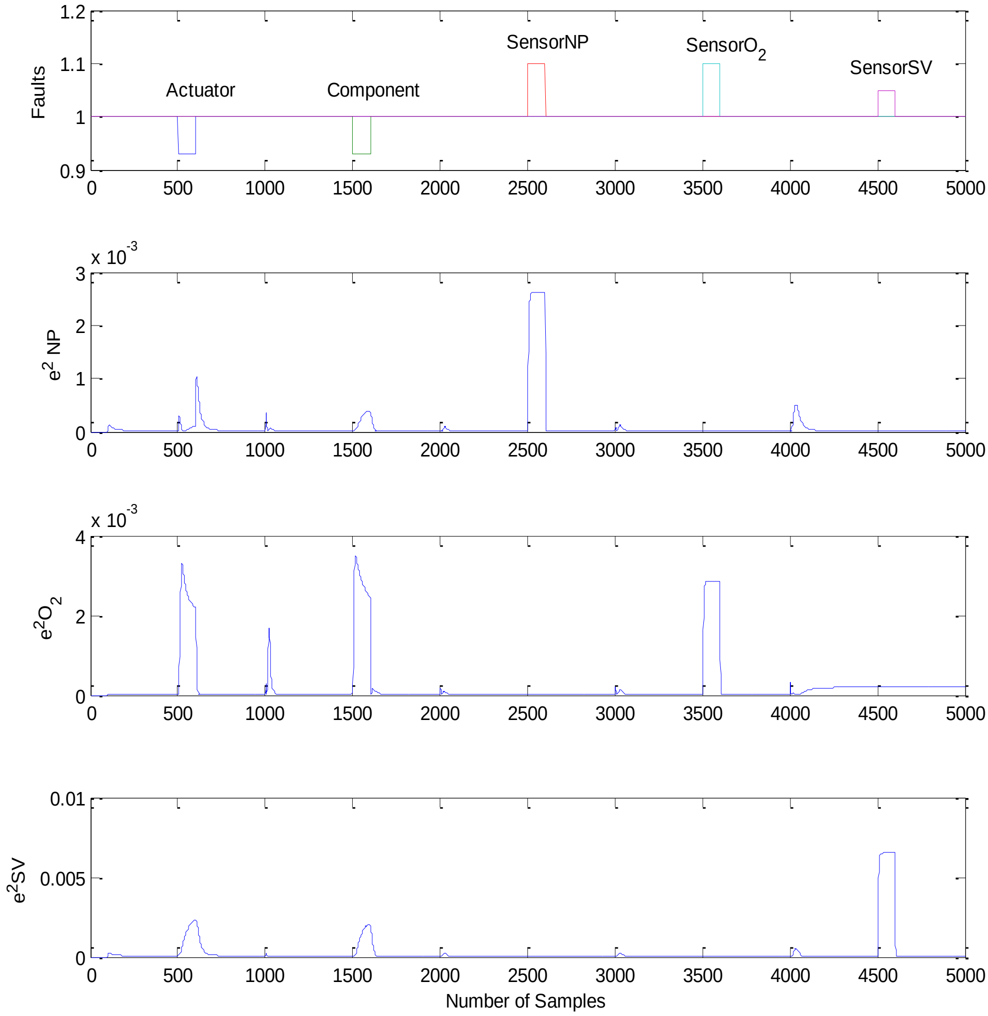


**Fig. 7.10** Filtered and squared model prediction errors

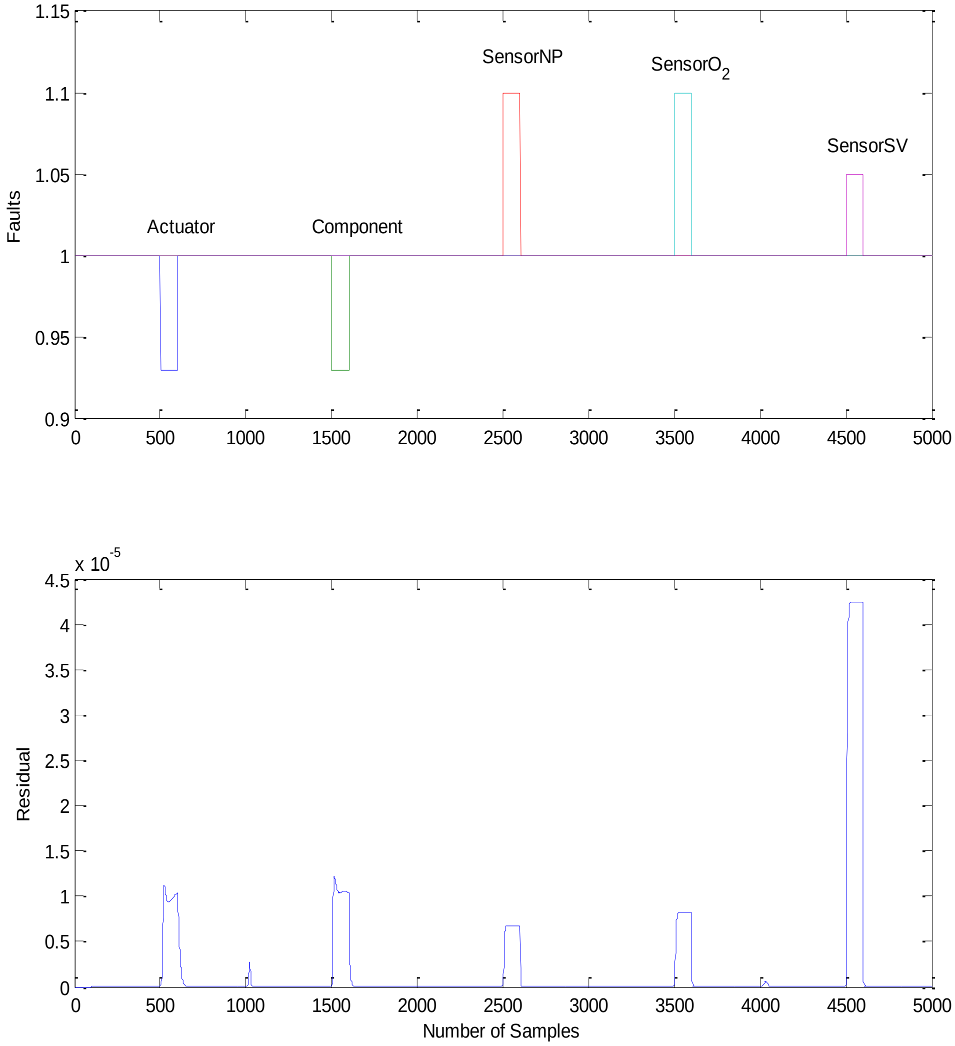


**Fig. 7.11** Residual signal for the five simulated faults

Once the signals being filtered and squared, based on Fig. 7.10, it is hard to identify in $\lambda_{O2}$ outputs which faults belong to whose. Therefore, by applying the residual generator equation, these faults can be detected. This is demonstrated in Fig. 7.11 where the fault size of sensor$_{SV}$ is much higher than the other four faults even though the fault size of sensor$_{SV}$ is the smallest which is around +5%.

### 7.2.2 FDI using MLP modelling

The MLP network model has been applied with the same data samples as in the RBF network model. Here, 6000 samples used for RAS inputs, 5000 samples and 1500 samples of step input signals. The same two signals mentioned in Table 6.3 are applied to open-loop control for further analysis.

#### 7.2.2.1 RAS inputs

6000 data samples are used in the MLP network model where the input signals are randomly generated between the minimum and maximum range of SC values and CV values mentioned in Table 6.2. As can be seen in Fig. 7.12 with the RAS signal as input, it is quite hard to differentiate the signal between the healthy and faulty output signals. However by referring to the output of SV there is a big difference between the plant and the MLP network model happened at k=3000. To make it clearer, Fig. 7.13 shows close-up results of Fig. 7.12 starting from k=1800-3800.

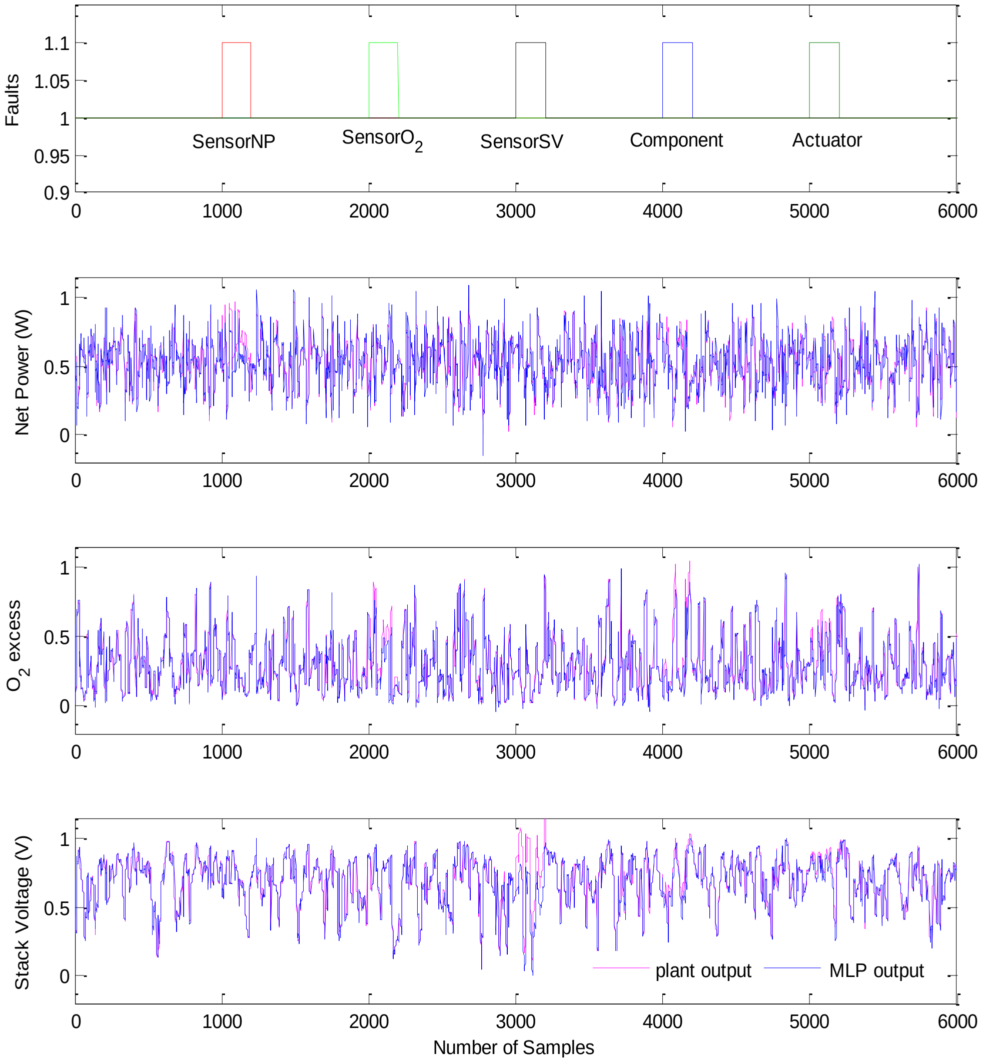


**Fig. 7.12** The testing process of healthy and faulty data sets

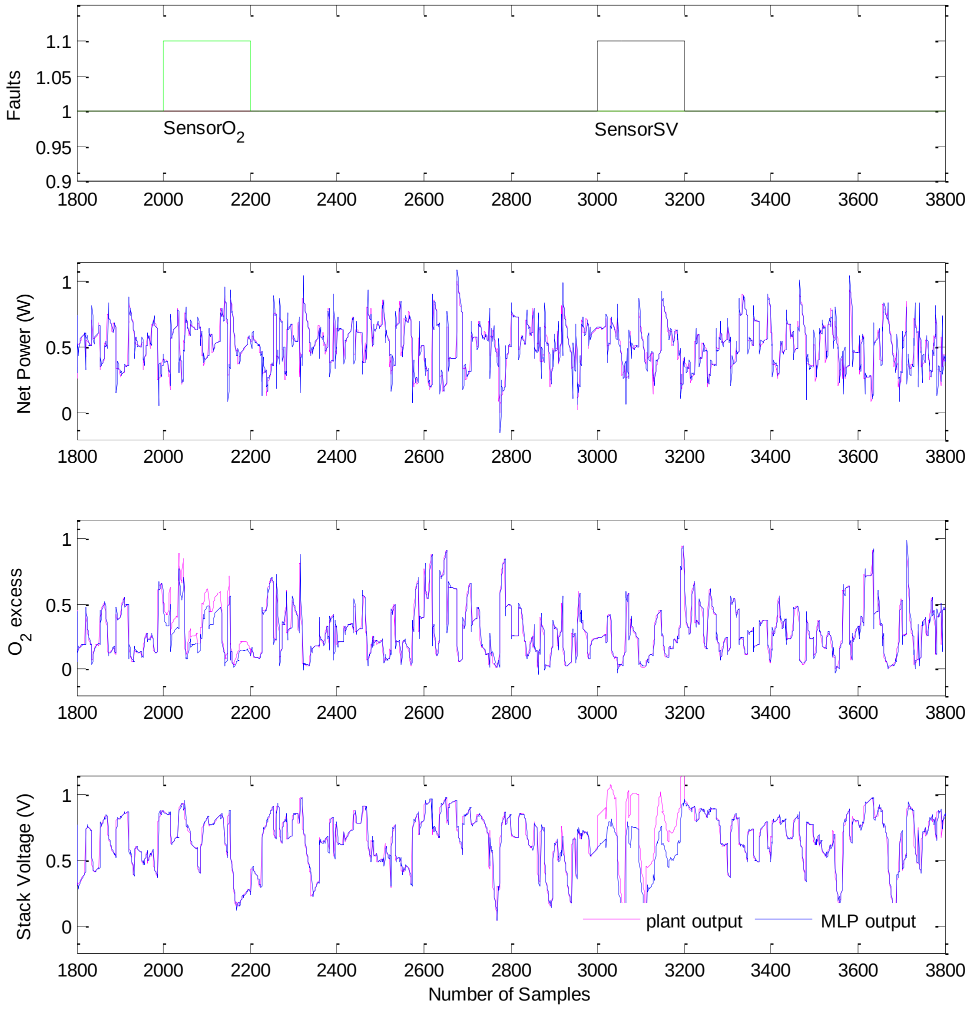


**Fig. 7.13** Close-up view of Fig. 7.12

In Fig. 7.13, two faults occurred due to sensor$_{O2}$ and sensor$_{SV}$. By referring to this, it shows the effect of sensor$_{O2}$ fault is small even though the fault size was +10% at k=200-2200. However, at k=3000-3200, the fault of sensor$_{SV}$ can clearly be seen. It shows that a fault in sensor$_{SV}$ only affected its own output signal.

By applying the same procedure as conducted and mentioned in RBF network model, the modelling prediction errors are found and then being filtered and squared to enhance the signal-noise ratio. Fig. 7.14 and Fig. 7.15 demonstrated the simulation results after implementing those methods towards the residual signals. While Fig. 7.16 show the fault detection based on individual fault after the application of residual generator equation.

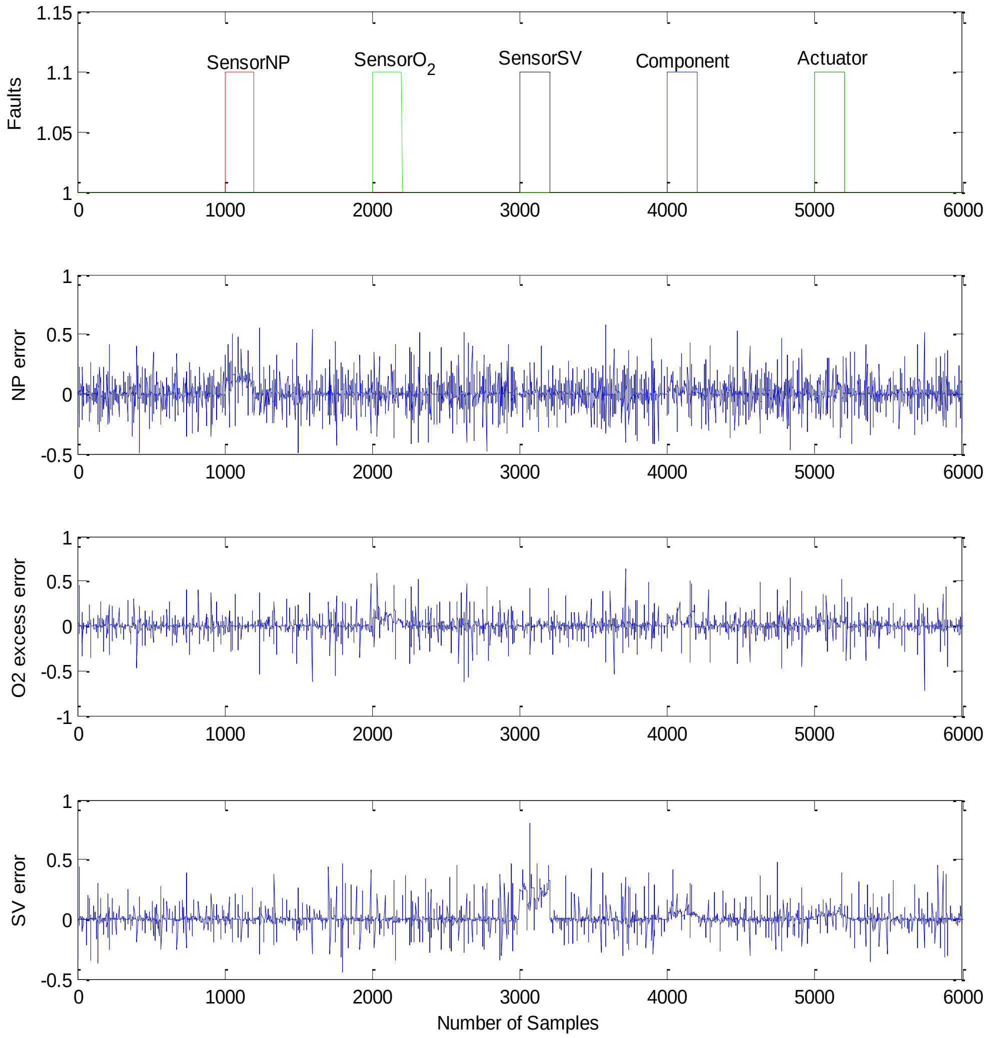


**Fig. 7.14** The modelling prediction errors

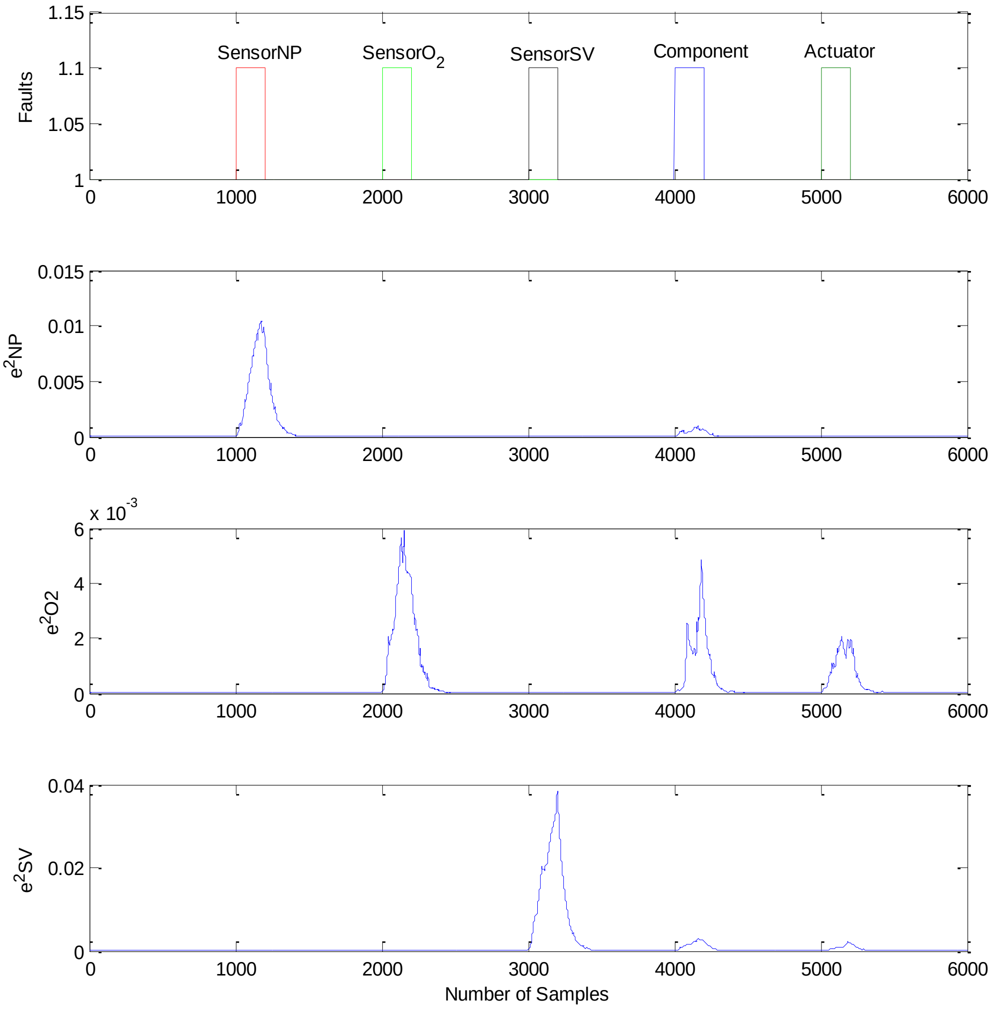


**Fig. 7.15** Filtered and squared model prediction errors

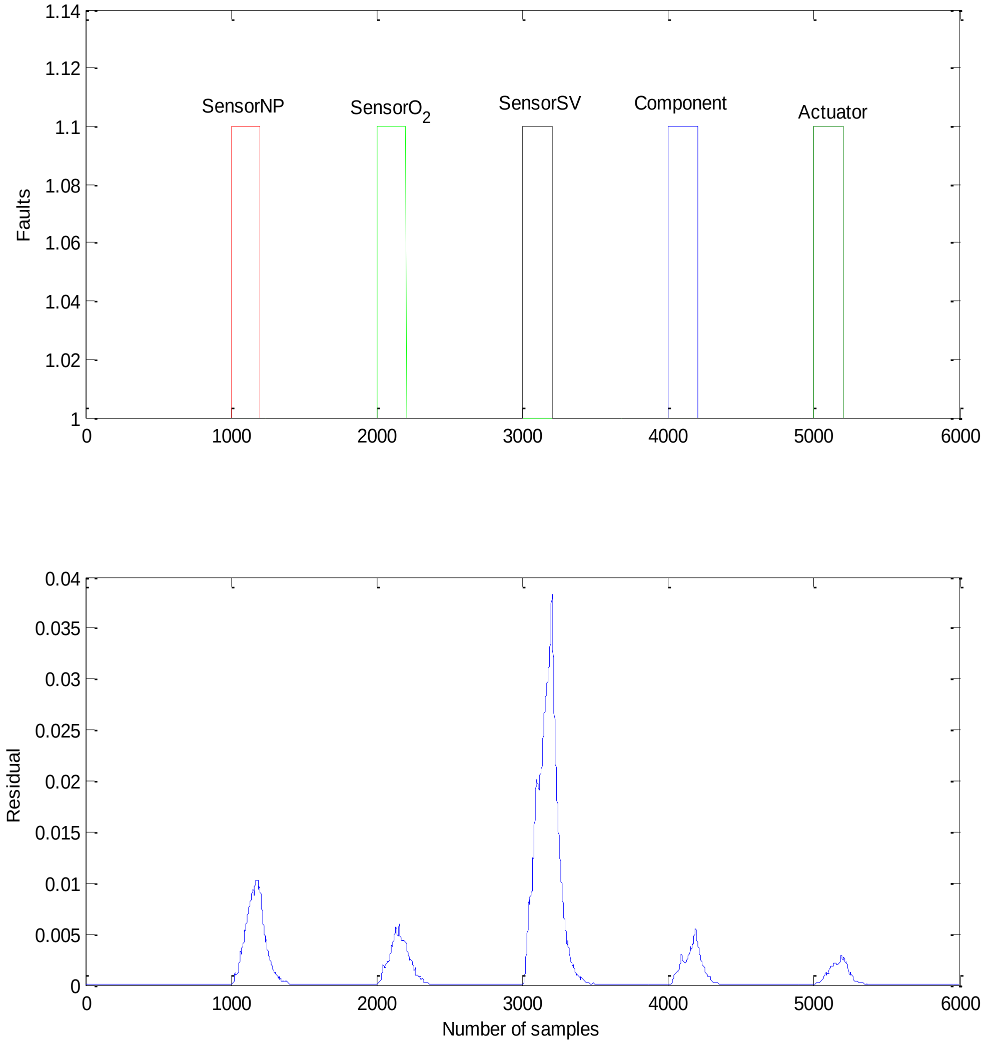


**Fig. 7.16** Residual signal for the five simulated faults

### 7.2.2.2 Step inputs

Next, a step signal injected as inputs, where the change of signal from one state to another is not as in RAS inputs. Due to this situation, the fault occurrence in NP, $\lambda_{O2}$ and SV outputs are clearly been seen as in Fig. 7.17. The changing of state during no fault to fault condition when fault occurred in the plant is obvious. This is shown in Fig. 7.17.

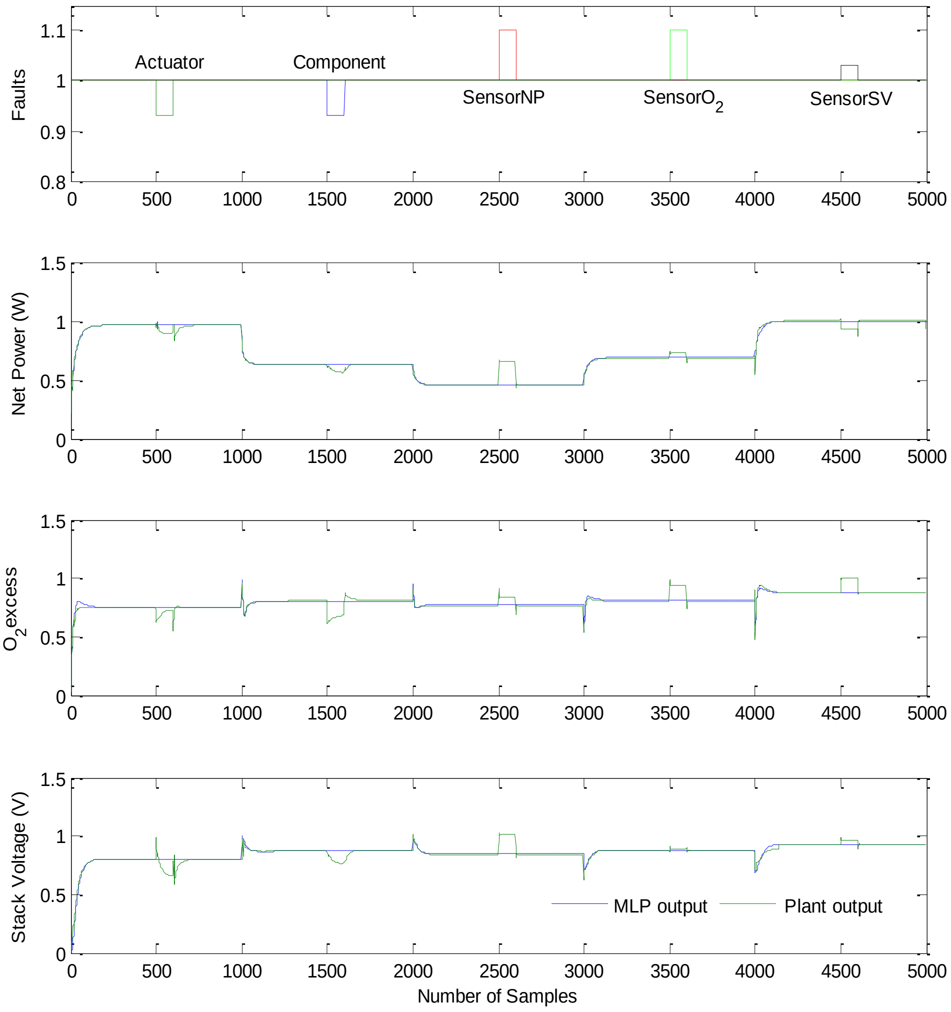


**Fig. 7.17** The testing process of healthy and faulty data sets

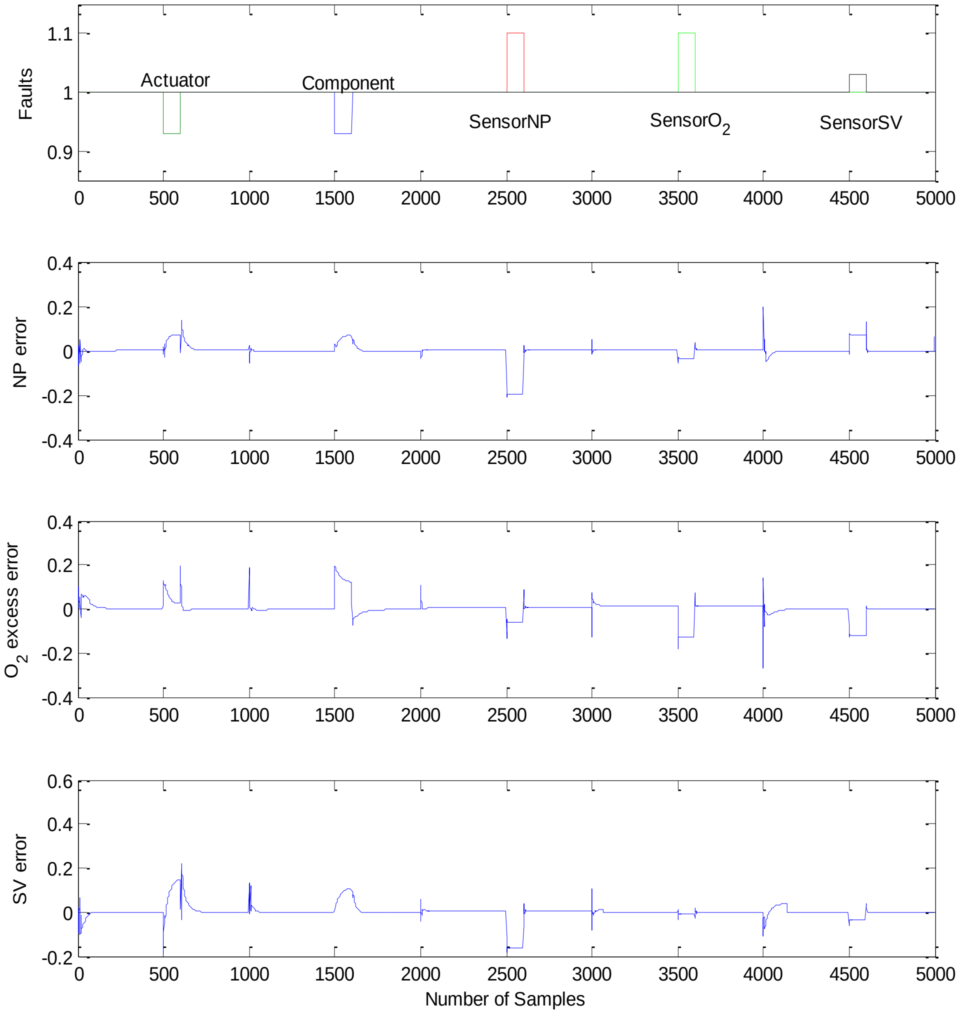


**Fig. 7.18** The modelling prediction errors

The difference between the plant output and the MLP model prediction is used to generate residual signal which is later used to perform fault detection. These signals are filtered and squared to change the negative amplitude of residuals as shown in Fig. 7.18 while Fig 7.19 shows that more than one signals occurred inside the output signals. Based on this result, it is difficult to detect faults in the plant outputs. Therefore, by using the equation of residual generator, these five faults can easily be detectable as shown in Fig. 7.20.

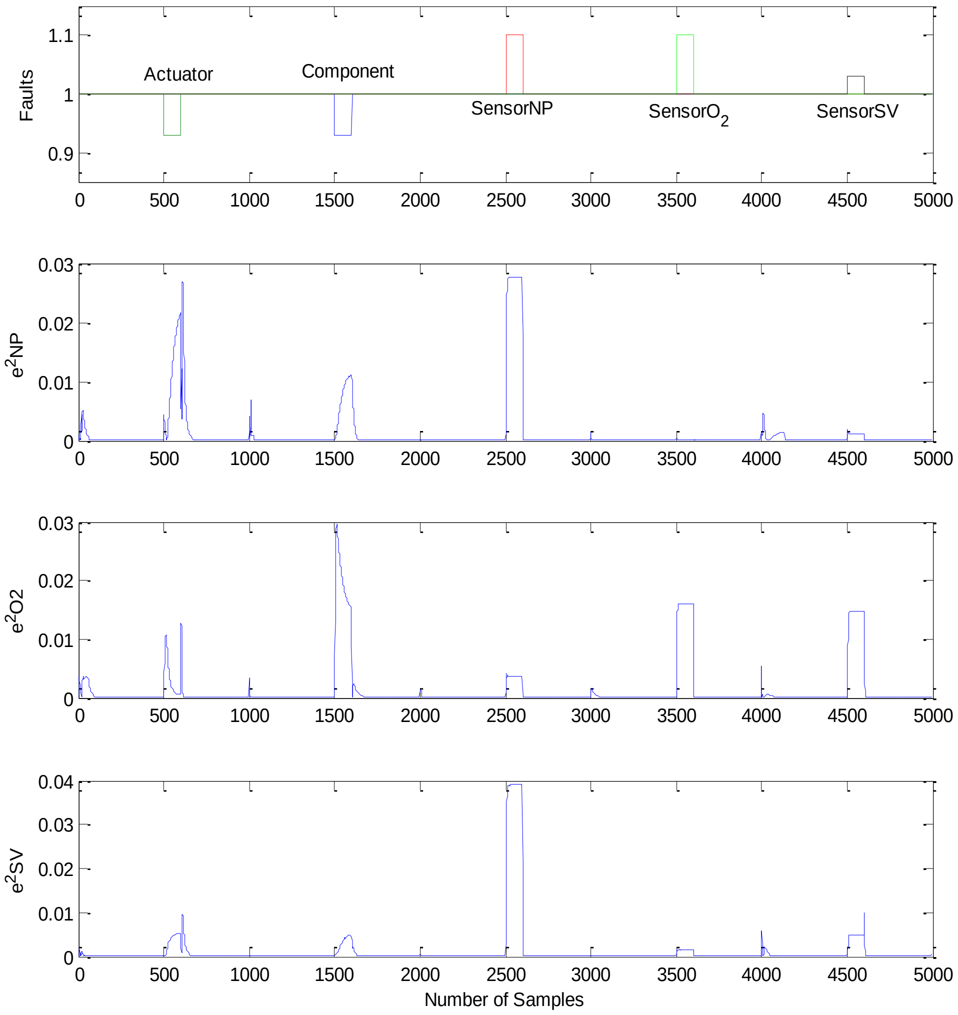


**Fig. 7.19** Filtered and squared model prediction errors

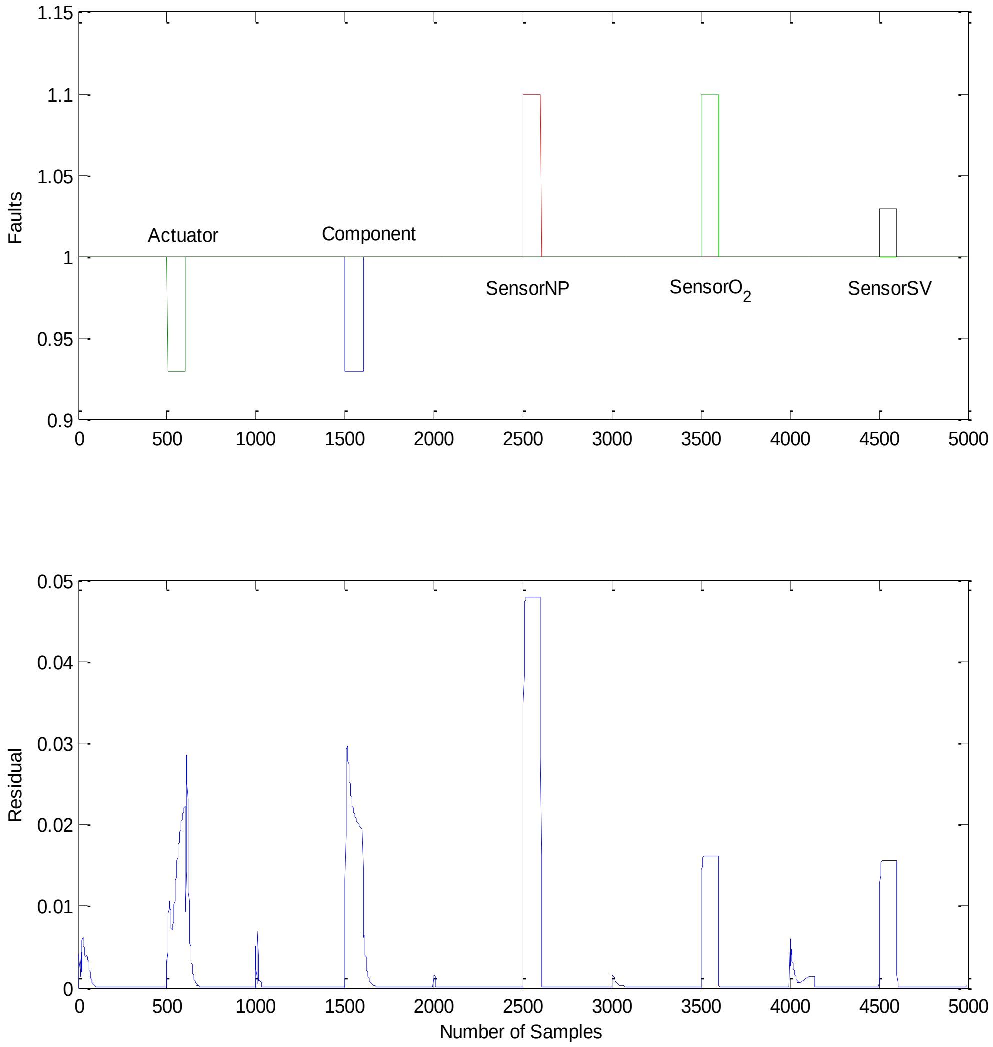
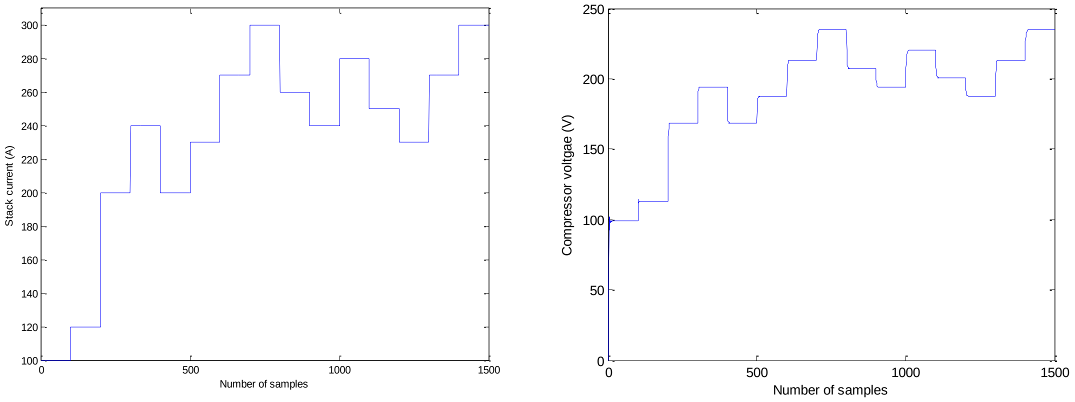


**Fig. 7.20** Residual signal for the five simulated faults

## 7.3 Fault detection for closed-loop system

After these two networks being tested on open-loop control, next they are implemented to closed-loop control. The simulated faults can be further referred in Table 6.3. Fig. 7.21 shows the type of inputs used to train and test the networks as part of the simulation process. Here, $\lambda_{O2}$ is set at 2 and being used as a reference signal to monitor the performance of $\lambda_{O2}$.



(a) (b)

**Fig. 7.21** The step inputs used in the feedforward-feedback control

### 7.3.1 RBF modelling

Firstly, the RBF network is implemented to perform the closed-loop control using the faulty data sets. The simulation is conducted using 1500 data samples obtained with five faults simulated is superimposed with ±10% fault size. The same procedures and methods applied in the previous section are applied to this controller while doing the fault detection. Fig. 7.22 shows the testing process between healthy and faulty data set while Fig, 7.23 demonstrates the model prediction errors of the signal obtained.

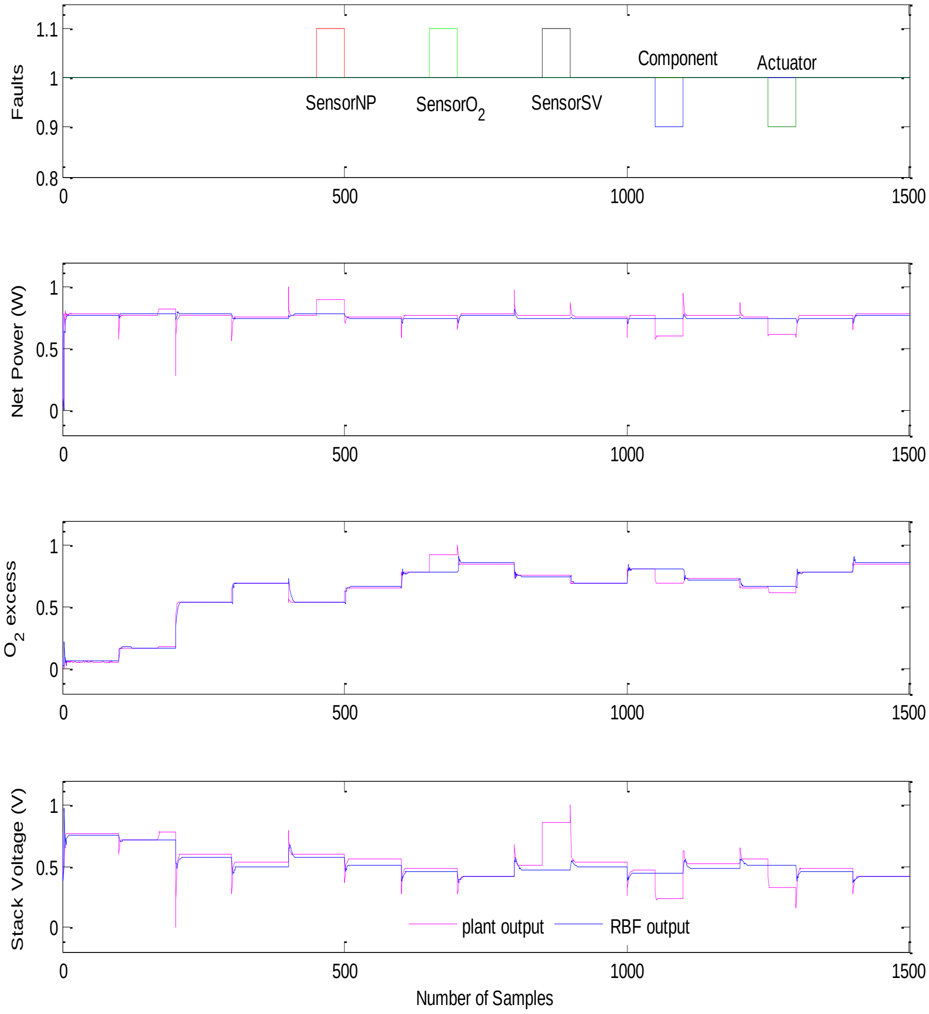


**Fig. 7.22** The testing process of healthy and faulty data sets

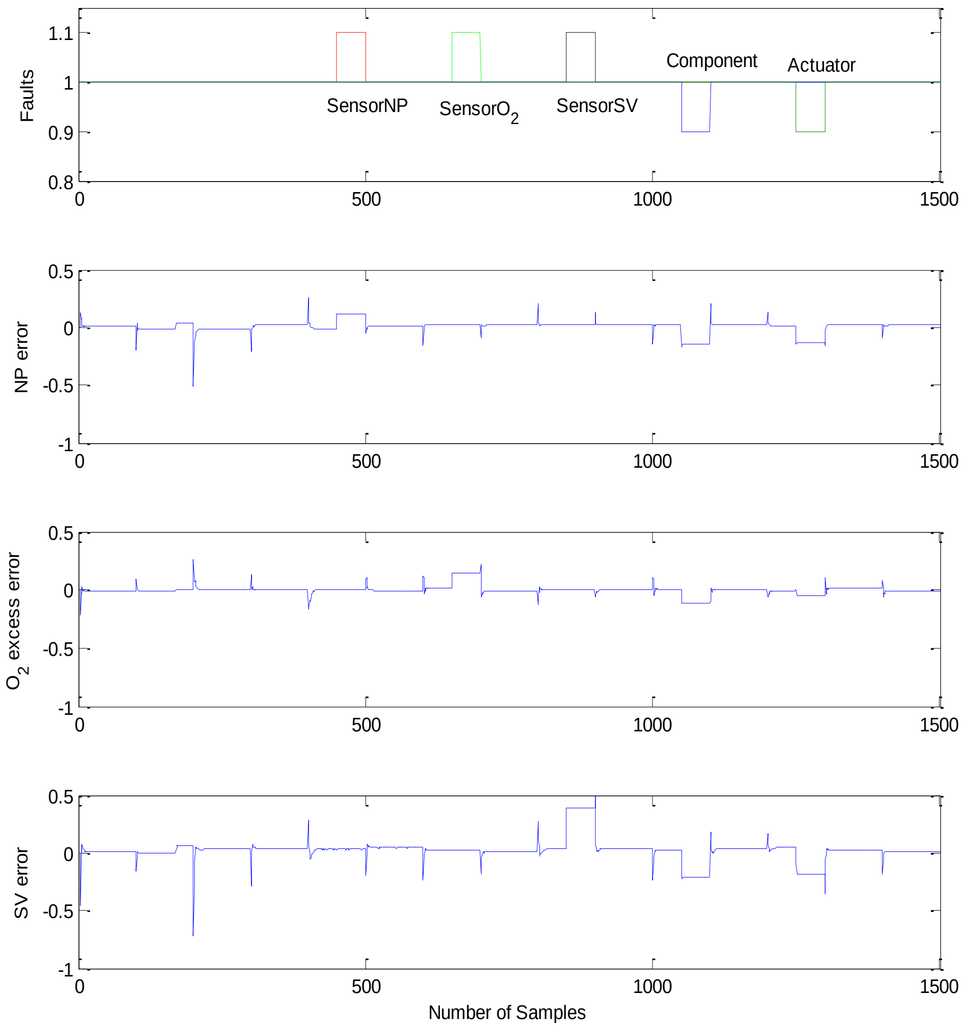


**Fig. 7.23** The modelling prediction errors

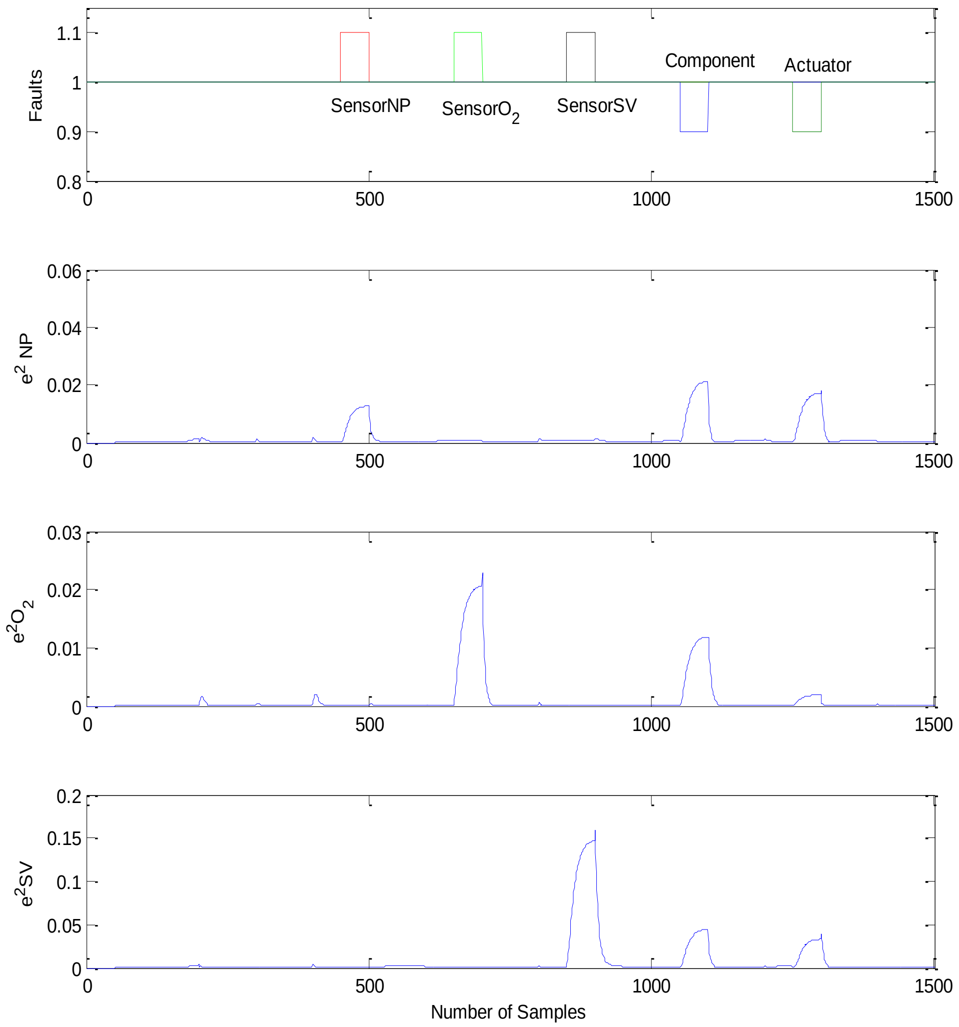


**Fig. 7.24** Filtered and squared model prediction errors

Later the prediction error signals are filtered and squared and again in order to detect faults using the residual generator. Fig. 7.24 shows the filtered modelling prediction errors while Fig. 7.25 shows the fault detection of RBF network model.

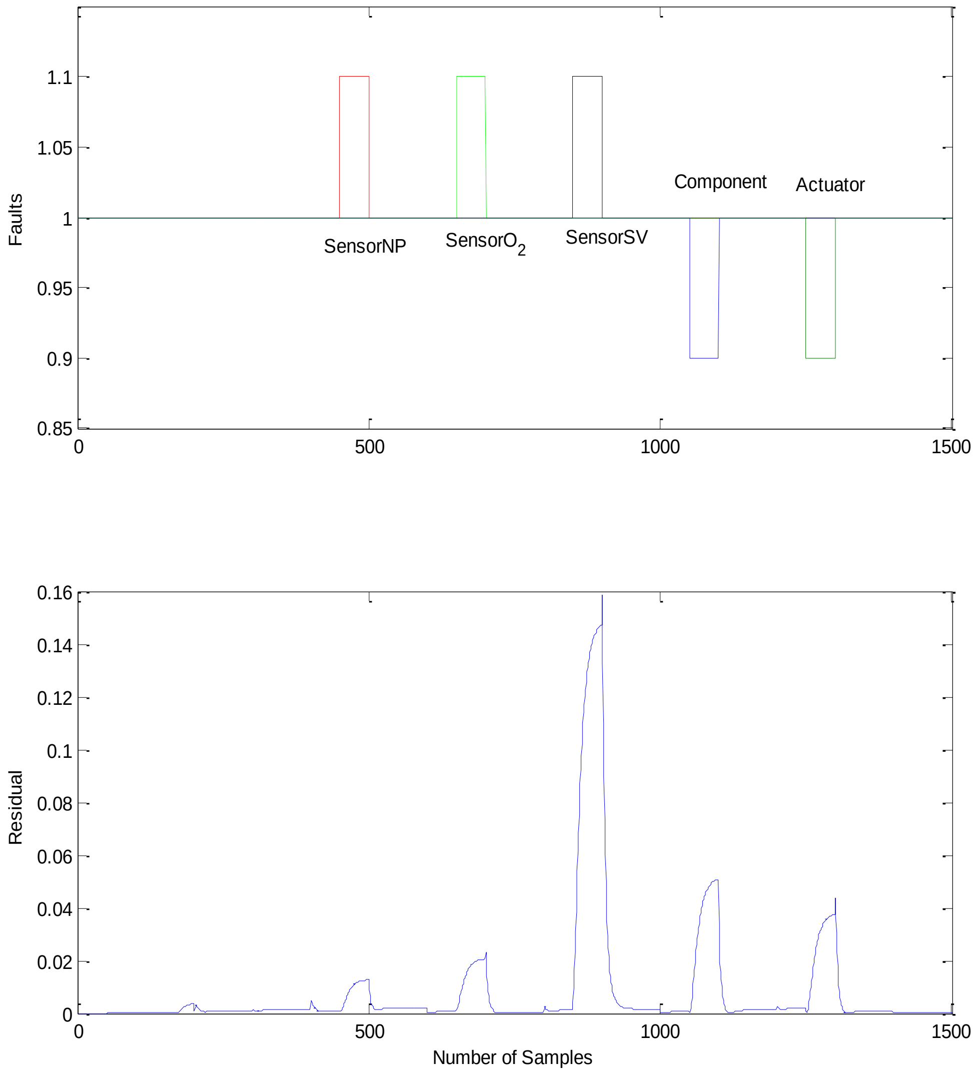


**Fig. 7.25** Residual signal for the five simulated faults

### 7.3.2 MLP modelling

The MLP network model for closed-loop control used step inputs of 1500 data samples. Fig. 7.26 illustrates the testing process of MLP network between healthy and faulty data sets. As can be seen here, the outputs plant change where there is a fault occurred and become constant when there is no fault in the plant systems.

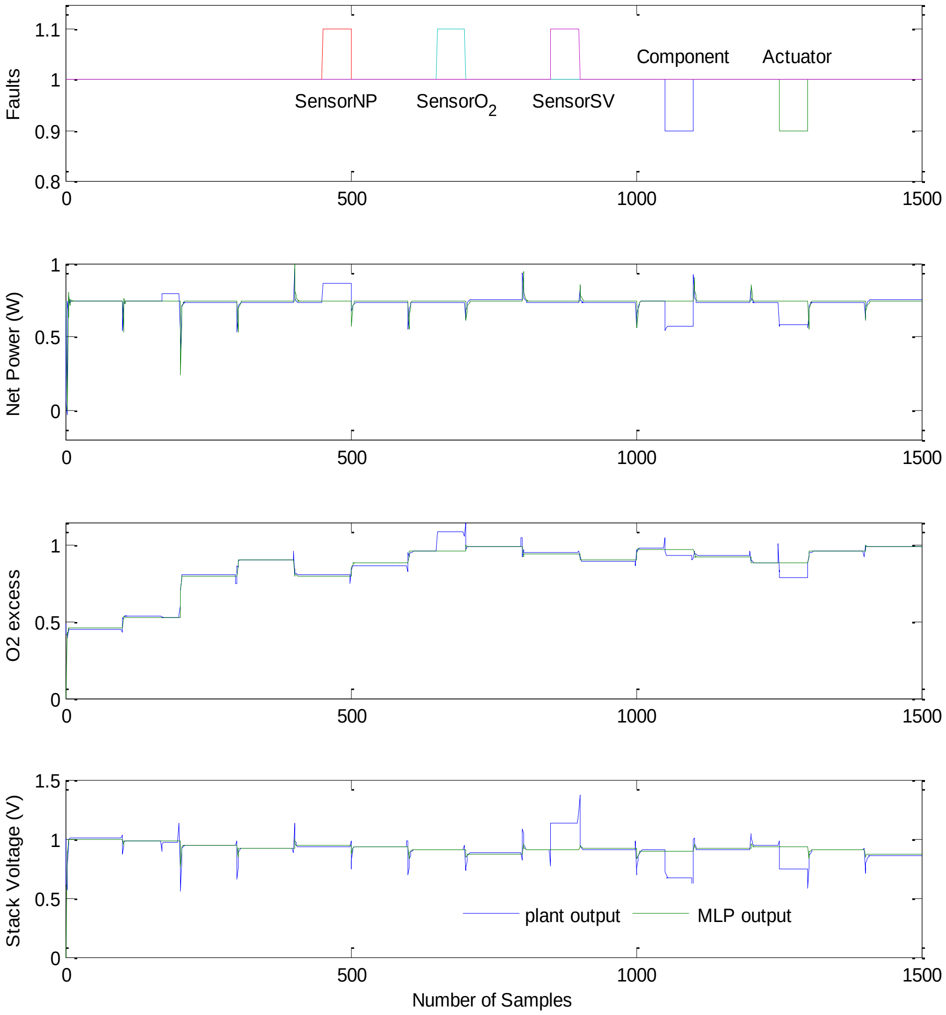


**Fig. 7.26** The testing process of healthy and faulty data sets

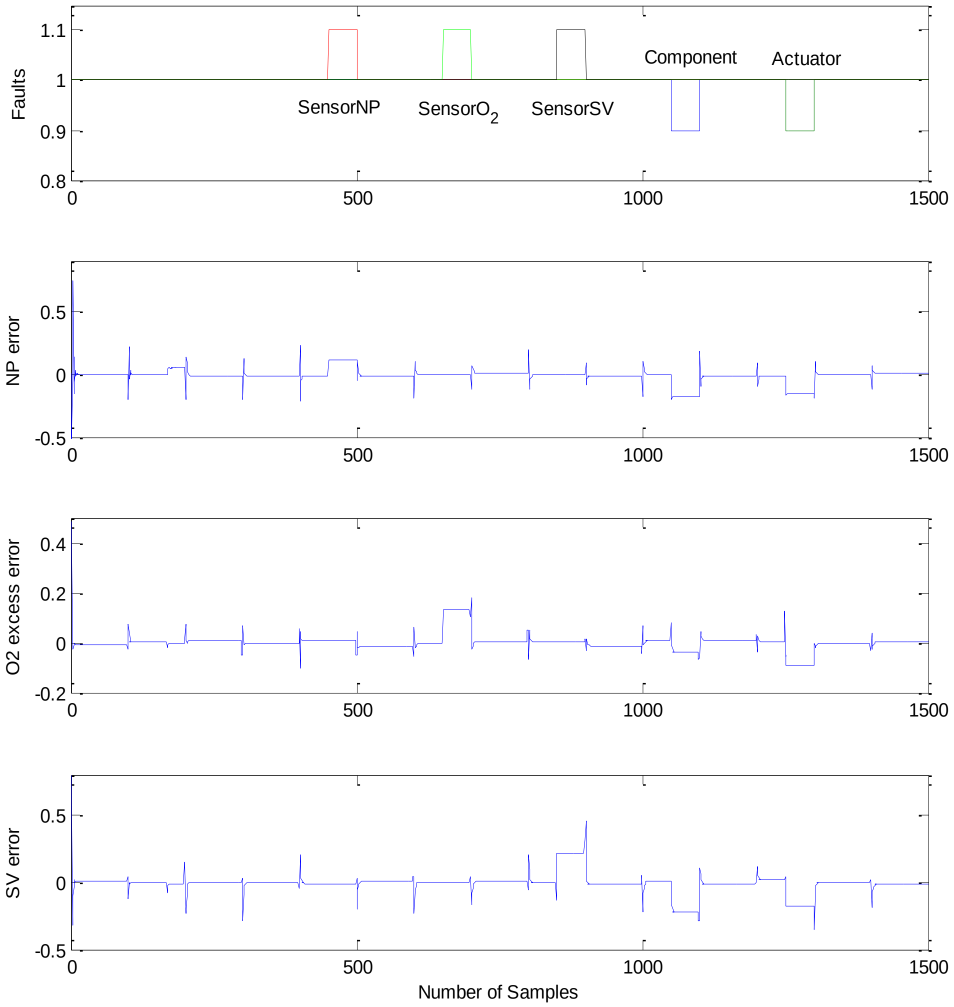


**Fig. 7.27** The modelling prediction errors

In order to perform fault detection, the residual signals are obtained then filtered and squared for easy analysis. Fig. 7.27 and Fig. 2.28 shows the corresponding results.

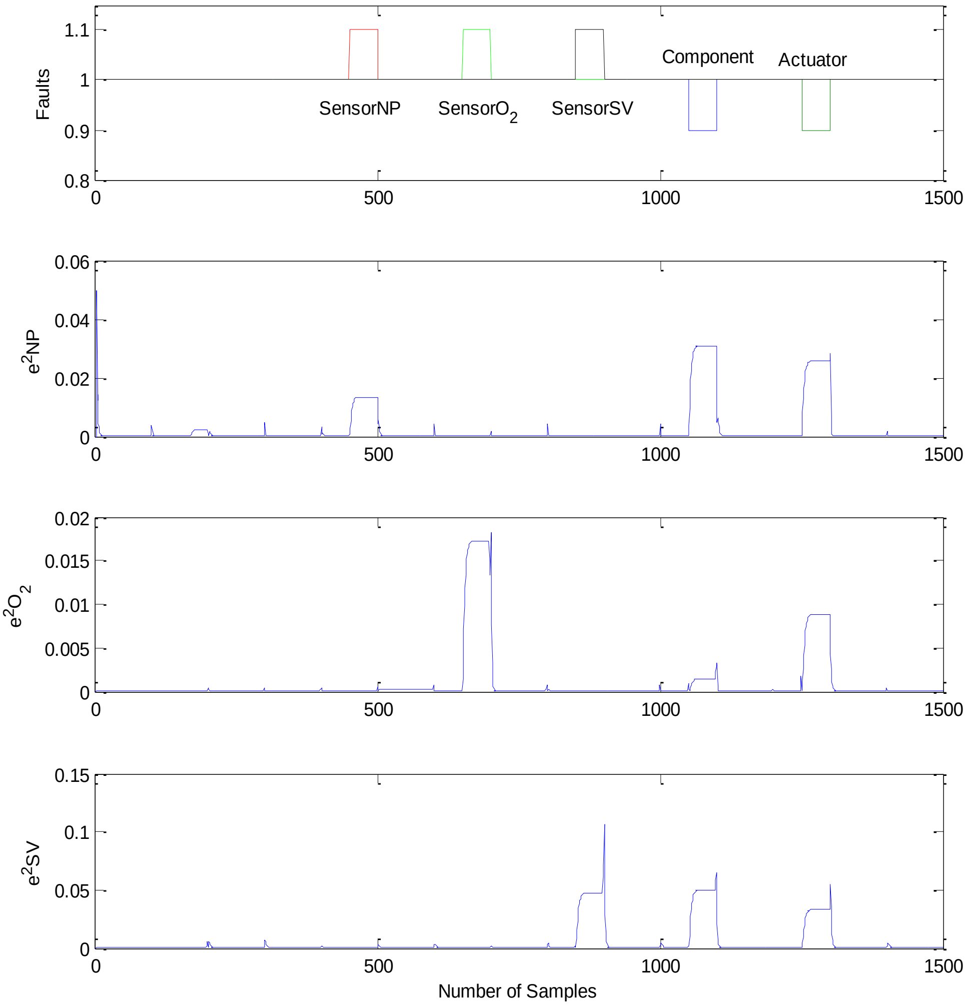


**Fig. 7.28** Filtered and squared model prediction errors

Once the filtered modelling prediction errors are obtained, then the fault detect can be perform. In Fig. 7.28 the output signals show that there are more than one fault exists in the signal. The fault cannot be determined therefore once the residual generator being applied, these five faults can be detectable. After the signals have been filtered due to a lot of noises and disturbances, a threshold is set to indicate a false signal and a faulty signal. Based on Fig. 7.29 the threshold is at 0.008 where it shows that five faults simulated have been detected.

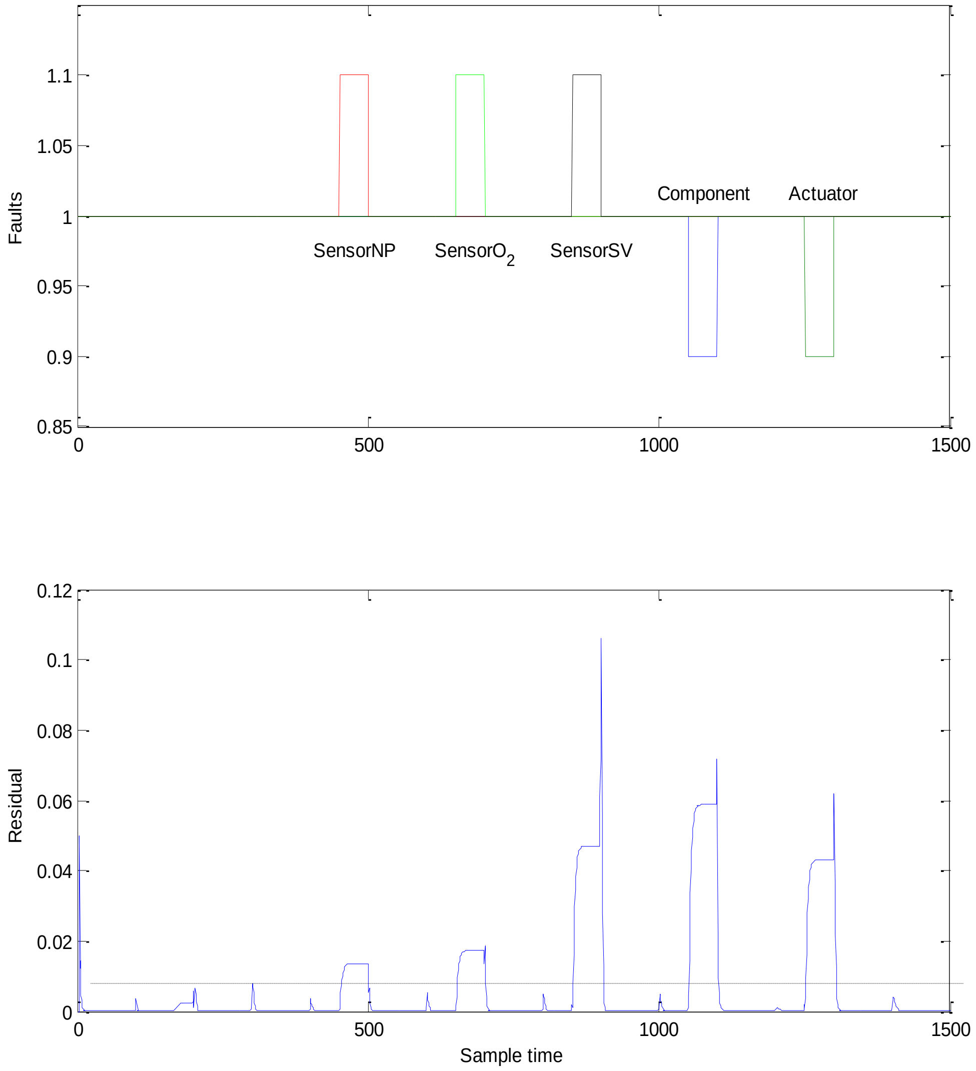


**Fig. 7.29** Residual signal for the five simulated faults

## 7.4 Summary

In this chapter, a new independent neural network model-based fault detection method has been presented and tested. An advantage of this new method is that it does not require the knowledge of the fault magnitude to provide a diagnostic. To prove this, a PEMFC simulator based on a model presented in the literature (Pukrushpan *et al*., 2004a; 2004b) has been used and tested. The simulation model was modified to include five set of possible faults scenarios proposed in this thesis work. This modified simulator allows imposing a determined fault scenario and analysing its behaviour. In this work the RBF network model and MLP network model are used to perform fault detection. From the observation it shows that, both methods are able to detect the five simulated faults. It shows that this proposed fault detection method is able to detect all five faults either in open-loop control or closed-loop control with different kind of input signals. The fault detection plays an important part in fault isolation procedure because if the fault cannot be detected, then the fault isolation cannot be performed.

# CHAPTER 8

# FAULT ISOLATION

## 8.1 Introduction

For fault isolation, the residuals generated should not only be sensitive to faults, they also need to be able to distinguish between different types of faults. There are two approaches to generate such residuals which facilitate fault isolation. One method, known as the directional residual approach, is to generate residual vectors that lie in a specified direction in the residual subspace corresponding to each type of fault. The fault isolation problem is then transformed into one of determining the direction of the residual vector. The other method is the structured residual approach, in which each residual vector is designed to be sensitive to a single or selective set of faults, and insensitive to the rest (Chen, 1995). Structured residuals are usually characterized by an incidence matrix in which the rows correspond to residuals and columns correspond to faults. A “1” in the incidence matrix represents coupling between a residual and a fault, and a “0” represents no coupling. For isolation, all columns must be different. A special case in which each residual is designed to respond to a single fault is known as a diagonal structure (Patton & Chen, 1997). The design consists of two stages, the first one is to specify isolable structured, and second is to make each residual robust. In the following, fault isolation schemes based on structured residual sets are presented. Note that, each residual is designed using eigenstructure assignment approach.

## 8.2 RBF classifier

The residual signals have different structures for different faults. This feature can be used to isolate these faults. From references (Himmelblau *et al*., 1991; Watanabe *et al*., 1989; Sorsa *et al*., 1991; Naidu *et al*., 1990; Willis *et al*., 1991), it can be seen that the neural network is used as a fault classifier. A neural network is used to examine the possible faults or abnormal features in the process plant outputs and fives a fault classification signal to declare whether the system is in faulty condition or not. Here, the outputs system is used as input to fault classifier. The successful detection of a fault is followed by the fault isolation, which will distinguish or isolate a particular fault from others (Patton & Chen, 1997).

The RBF network is well known for its powerful ability to classify components with different features from a mixed signal. Fault isolation in this study is implemented by adding another RBF network as a classifier (Kamal and Yu, 2012). The model prediction error vector obtained from the fault detection part is caused by the faults, in addition to modelling error and noise and is a nonlinear function of the faults. As this vector is multi-dimension, three dimensions in this study, it will have different structures for different faults. The RBF classifier uses this feature to classify these faults. Based on this idea, the RBF classifier is designed with three inputs to receive the three elements of model prediction error vector, and five outputs with each being dedicated to one fault.

This RBF classifier is a nonlinear static network. The network is trained with a set of data including five subsets. Each subset of data is collected when the system is subjected to one of the five simulated faults. Then the classifier is trained with its target vector arranged in the following way: For the data subset with the first fault occurrence, the target for the first output is “1”, while the target for the other outputs are “0”. For the data subset with the second fault occurrence, the target for the second output is “1”, while the target for the other outputs are “0”, and so on so forth, until the final faults.

Here two types of signals have been used as input signals. Therefore, to demonstrate how fault isolation used as the structured residual, the target matrix of the fault is demonstrated using one of the sample instead of the other two data sets. In the RAS execution, 6000 samples of data were collected with the first fault occurring during *k* = 1001~1200, the second fault occurring during *k* = 2001-2200, and so on. Then, the generated filtered and squared model prediction error vector from the fault detection part was used as the input data of the RBF classifier. Correspondingly, the target matrix $X_0$ has 6000 rows and 5 columns. The entries from the 1000th row to the 1200th row in the first column are “1”, while the other entries are “0”. The arrangement for columns 2 to 5 is done in the same way. This is shown as in Table 8.1 for example arrangement of 6000 samples to perform fault isolation.

**Table 8.1** The target matrix in training the RBF classifier

| Rows | Xo | | | | |
|---|---|---|---|---|---|
| 1001~1200 | [1 | 0 | 0 | 0 | 0]$^T$ |
| 2001~2200 | [0 | 1 | 0 | 0 | 0]$^T$ |
| 3001~3200 | [0 | 0 | 1 | 0 | 0]$^T$ |
| 4001~4200 | [0 | 0 | 0 | 1 | 0]$^T$ |
| 5001~5200 | [0 | 0 | 0 | 0 | 1]$^T$ |

The target matrix in Table 8.1 was used in training of the RBF classifier. The centres and widths of the network were chosen using the K-means clustering algorithm and the p-nearest centre algorithm. The weights were trained with using the RLS algorithm with the following data, $\mu= 0.99999$, $w(0) =1.0\times10^{-6}\times U_{(nh\times3)}$, $P(0)=1.0\times10^{8}\times I_{(nh)}$; where I is an identity matrix and *U* is a ones matrix. The RBF networks model only used the three rows of the PEMFC outputs matrix which contain the values of NP, $\lambda_{O2}$ and SV. After training the data set is tested using the RBF classifier algorithm where the location of faults are located and classify according to time of fault occurred. Fig. 8.1 illustrates the process of RBF classifier implemented in this study where five faults can be located and classify.

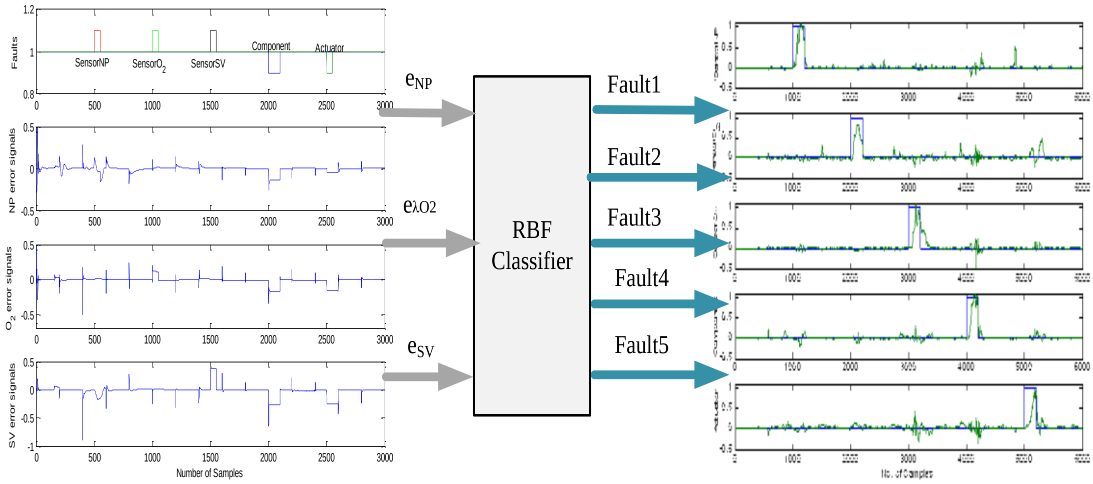


**Fig. 8.1** The concept of RBF classifier

### 8.2.1 Fault isolation for open-loop systems

In this work the fault isolation was implemented in open-loop system to verify whether both networks can function successfully in classifying faults in the process plant. As mentioned earlier, for open-loop control, two types of excitation signals are used as inputs for both network models. The first signal is the RAS while the second signal is a step signal. The structured residuals for RAS signals are set as in Table 1 while the structured residual setting for step input signals are as in Table 8.2. These two tables are referred for both networks; RBF network and MLP network.

**Table 8.2** The target matrix in training the RBF classifier

| Rows | Xo | | | | |
|---|---|---|---|---|---|
| 501~600 | [1 | 0 | 0 | 0 | $0]^T$ |
| 1501~1600 | [0 | 1 | 0 | 0 | $0]^T$ |
| 2501~2600 | [0 | 0 | 1 | 0 | $0]^T$ |
| 3501~3600 | [0 | 0 | 0 | 1 | $0]^T$ |
| 4501~4600 | [0 | 0 | 0 | 0 | $1]^T$ |

### 8.2.1.1 Fault isolation based on RBF residual signals

When RAS input signal applied, signals are fed into the RBF classifier in the sample interval, k = 1001-1200, k=2001-3000, k=3001-400, k=4001-5000 and k=5001-6000. While in step signal inputs, the first fault was introduced at k=501-600 and the second fault was introduced at k=1501-1600. To make the research easy to monitored and analysed, the distant between one faults to other have been fix at certain value of time. Refering to Table 8.2, firstly sensor$_{NP}$ fault is trained by the RBF classifier where when the fault is '1' at k=501-600, it is corresponding to this but at k=601-500, it is '0'. Then, at k=601-700, the RBF classier will classify sensor$_{O2}$ fault by trained it to be '1' at that time else becomes '0' at k=0-600 and k=701-500. This step is done until all faults being trained in order to classify faults in the signals. By doing this, later the RBF classifier is able to classify faults according to their occurrence. Based on these figures, they show that when there is a fault, the amplitude is '1' while when there is no fault the signal is '0'. Fig. 8.2 and Fig. 8.3 show how this RBF classifier works.

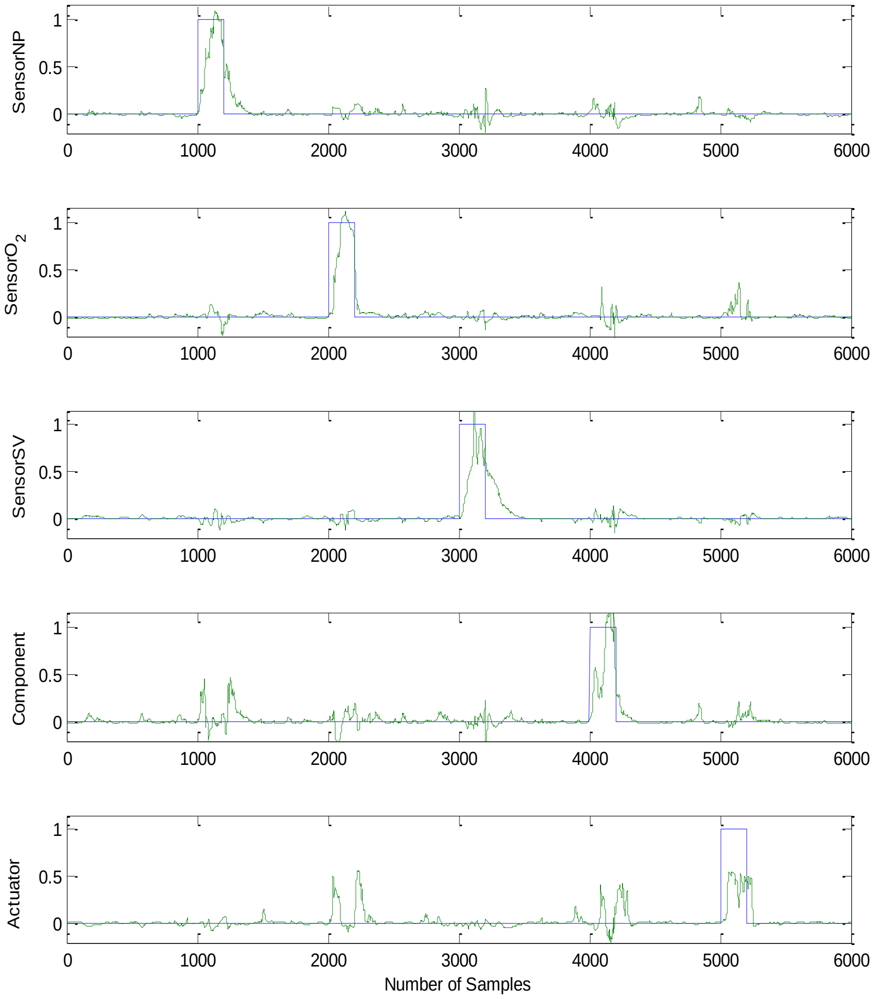


**Fig. 8.2** RBF network classifier outputs using RAS inputs

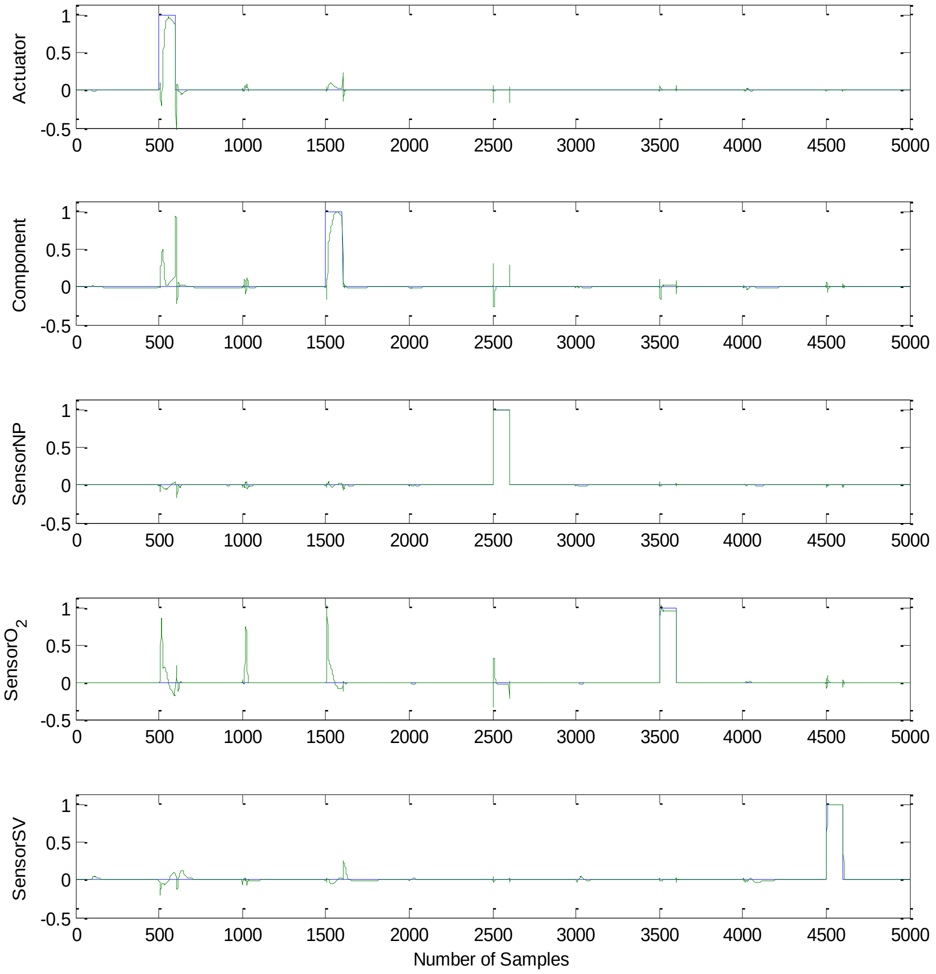


**Fig. 8.3** RBF network classifier outputs using step inputs

After trained the network with these structured residual signals, the network is tested. However the residual signal itself consist a lot of distortion signals in it. Therefore, the RBF classifier outputs are filtered for these two data samples and the filtered signals are displayed in Fig. 8.4 and Fig. 8.5. It is obvious that the filtered fault isolation signals are much smoother and the robustness of the signal is greatly enhanced. The faults mentioned in Table 8.1 and Table 8.2 was successful classify according to the individual fault trained by the RBF classifier. By setting certain range of threshold, the dotted line as shown in Fig. 8.4 and Fig. 8.5 false alarms can be eliminated and faults can easily be classified.

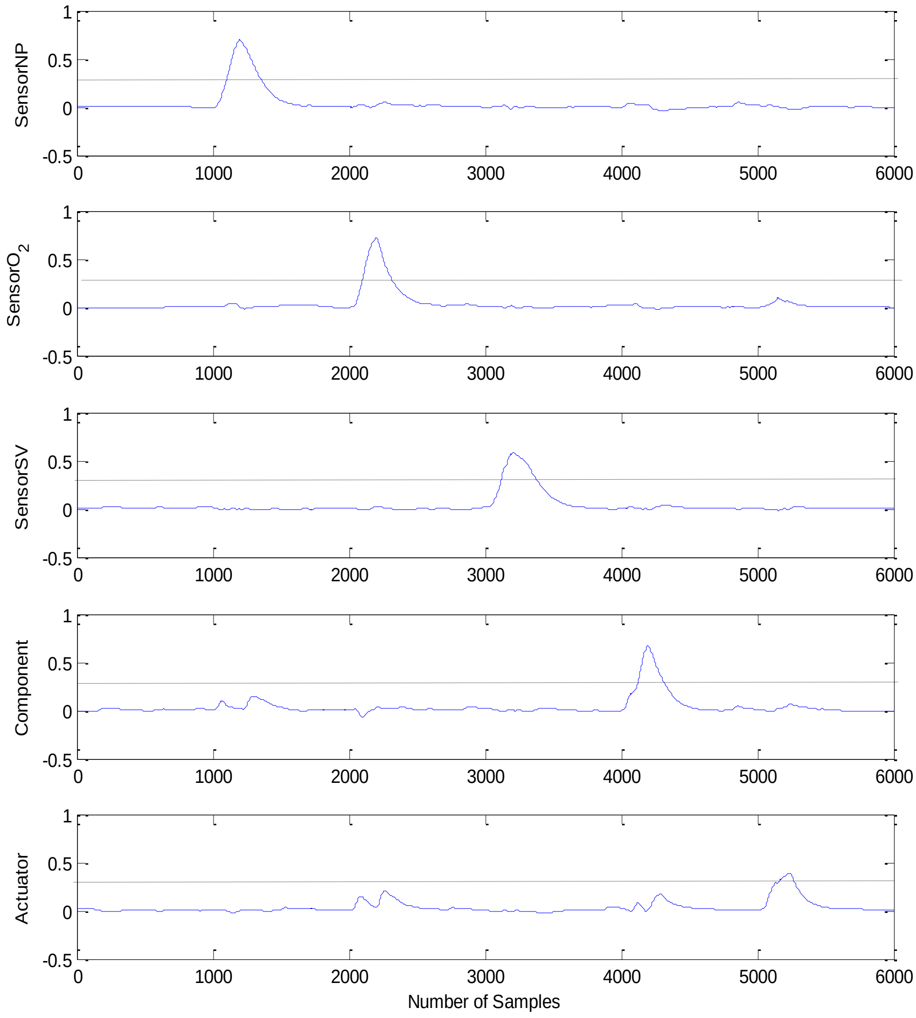


**Fig. 8.4** Filtered RBF classifier output with RBF residual

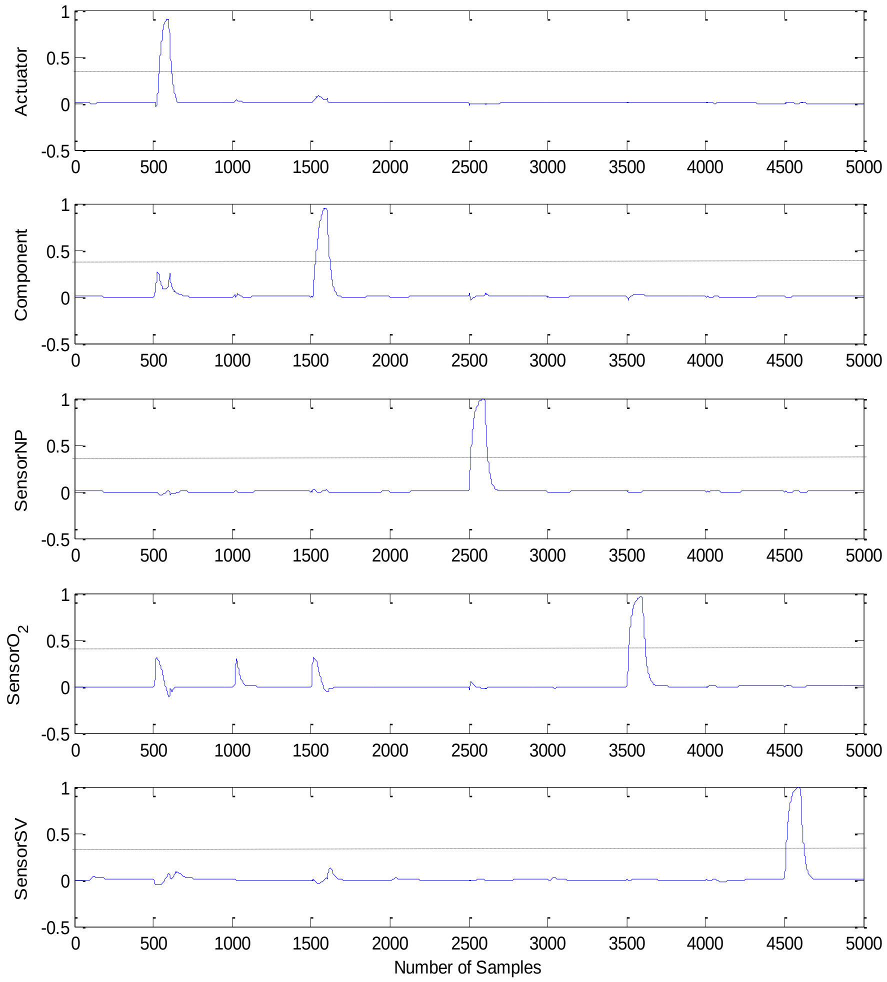


**Fig. 8.5** Filtered RBF classifier output with RBF residual

### 8.2.1.2 Fault isolation based on MLP residual signals

The same concept and approach was applied to the MLP network residual signals. The residual signals obtained at three outputs of the MLP network are used to perform fault isolation based on structured residual signals. The RBF classifier will train faults according to Table 8.1 and Table 8.2. Classification done by RBF classifier is explained in section 8.2.1.1. The simulation result of MLP network classier can be referred in Fig. 8.6 and Fig. 8.7.

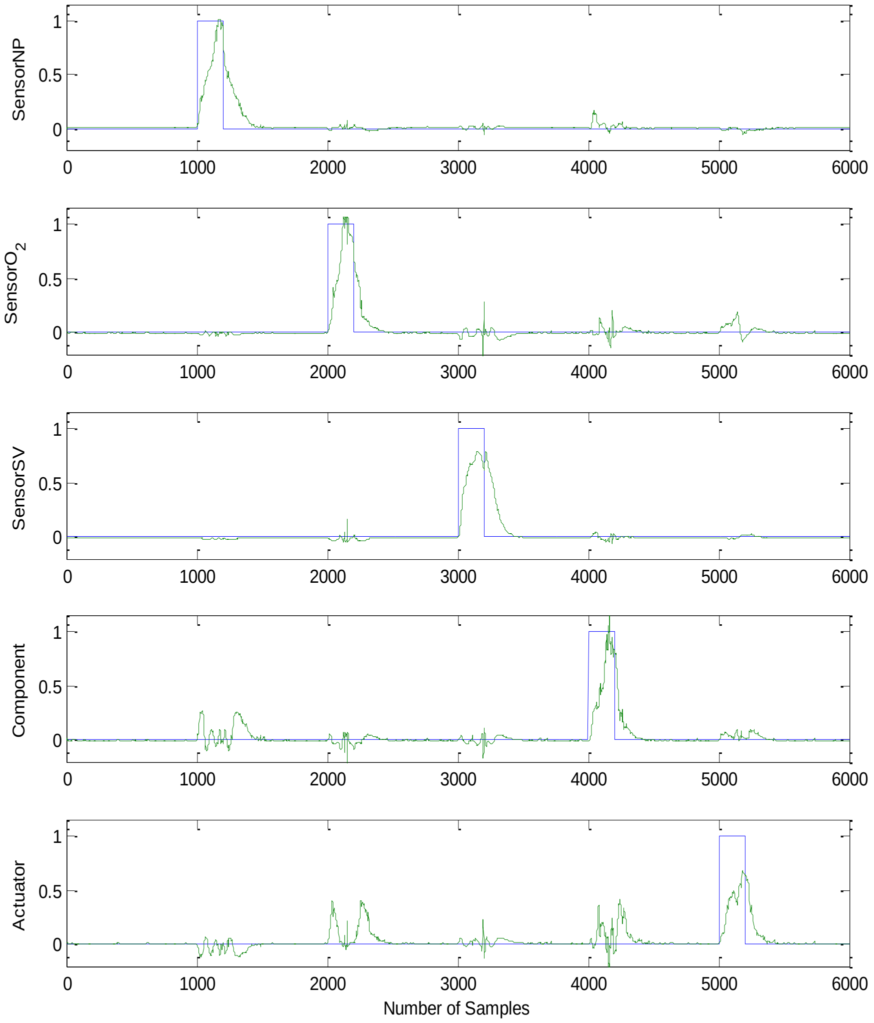


**Fig. 8.6** RBF classifier outputs for MLP residual using RAS inputs

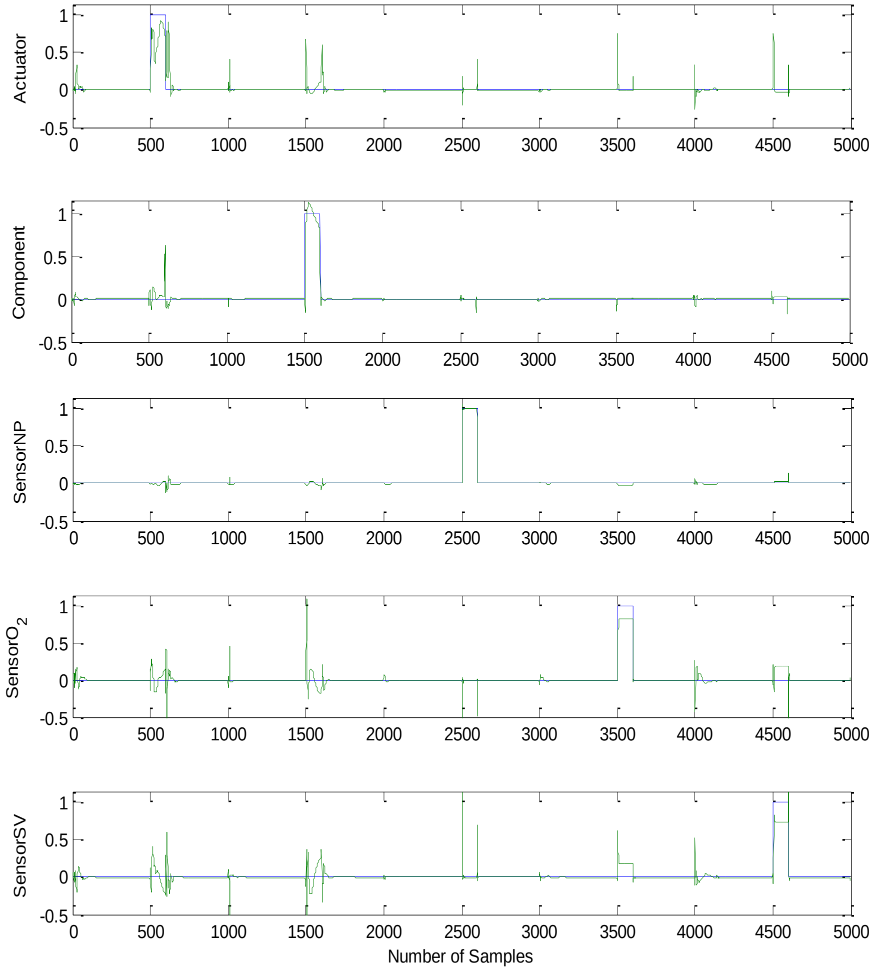


**Fig. 8.7** RBF classifier outputs for MLP residual using step inputs

After the training of the network has done, it is tested to check whether the RBF classifier can classify faults according to the trained one. These trained signals have to be filtered to get smoother signals. Five faults introduced to the FC stack are able to be isolated and classified according to the trained structured as mentioned in Table 8.1 and Table 8.2. The detail of the training involved is explained in previous results. Based on this explanation, Fig. 8.8 and Fig. 8.9 shows the results of MLP outputs obtained from the RBF classifier after it is filtered.

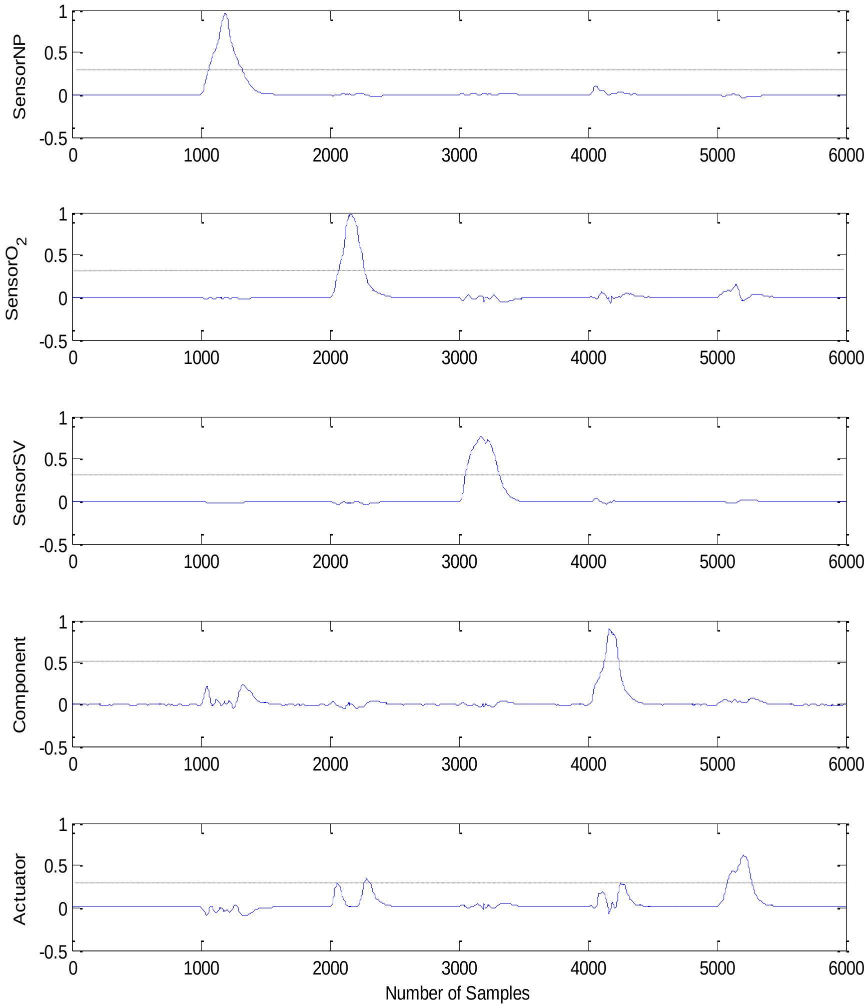


**Fig. 8.8** Filtered RBF classifier output with MLP residual

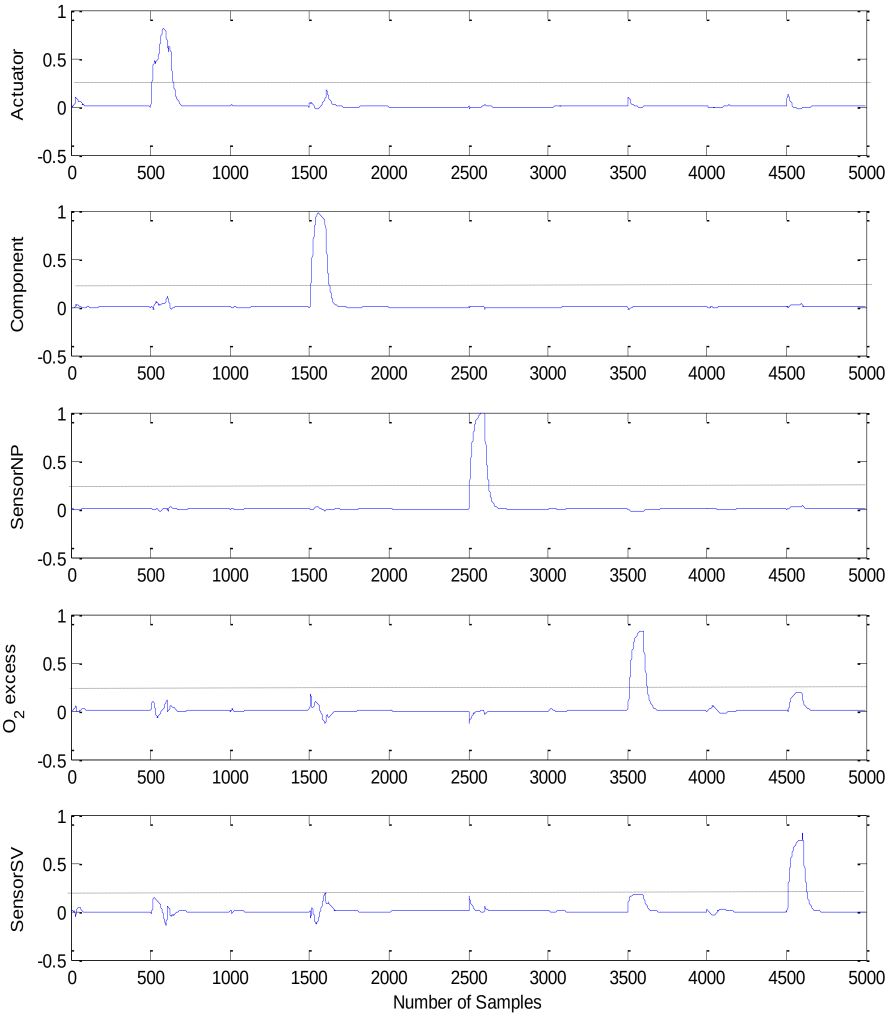


**Fig. 8.9** Filtered RBF classifier output with MLP residual

### 8.2.2 Fault isolation for closed-loop systems

All the industrial processes are controlled using a closed-loop control scheme. Therefore, the closed-loop control are also implemented in this work to see if the proposed algorithms are able to classify all faults correctly due to the complexity of the process plant. The data set used here are the same for both networks; RBF network model and MLP network model. Here, 1500 samples are used to perform fault isolation where the input signal of SC is a step input while set point is being set at 2 ($\lambda_{O2}$=2). Table 8.3 shows the structured faults implemented in this simulation.

**Table 8.3** The target matrix in training the RBF classifier

| Rows | Xo | | | | |
|---|---|---|---|---|---|
| 451~500 | [1 | 0 | 0 | 0 | 0]$^{T}$ |
| 651~700 | [0 | 1 | 0 | 0 | 0]$^{T}$ |
| 851~900 | [0 | 0 | 1 | 0 | 0]$^{T}$ |
| 1051~1100 | [0 | 0 | 0 | 1 | 0]$^{T}$ |
| 1251~1300 | [0 | 0 | 0 | 0 | 1]$^{T}$ |

The fault isolation was tested with RBF and MLP network classifier based on their respective residual signals. Referring to the block diagram in Fig. 5.6, both networks go through the same process where all the residual signals are injected to the RBF classifier in order to classify their faults according to the source of faults. The five outputs of the classifier are displayed in Fig. 8.10 and Fig. 8.11 for RBF outputs and MLP outputs respectively. As can be seen in Fig. 8.10 and Fig. 8.11, the RBF classifier will be trained with '1' if there is a fault exist and '0' if there is no fault. From the residual signals, the RBF classier will classify a fault signal with value of '1' where it is sensitive to the fault represented by it, while is insensitive to all the other four faults. This is represented in Fig 8.10 and Fig. 8.11. In this way, all the five considered faults can be isolated easily. The RBF classifier successfully suppressed the corresponding output value for the no-fault-occurring period, while promoted the corresponding outputs value for the fault-occurring period.

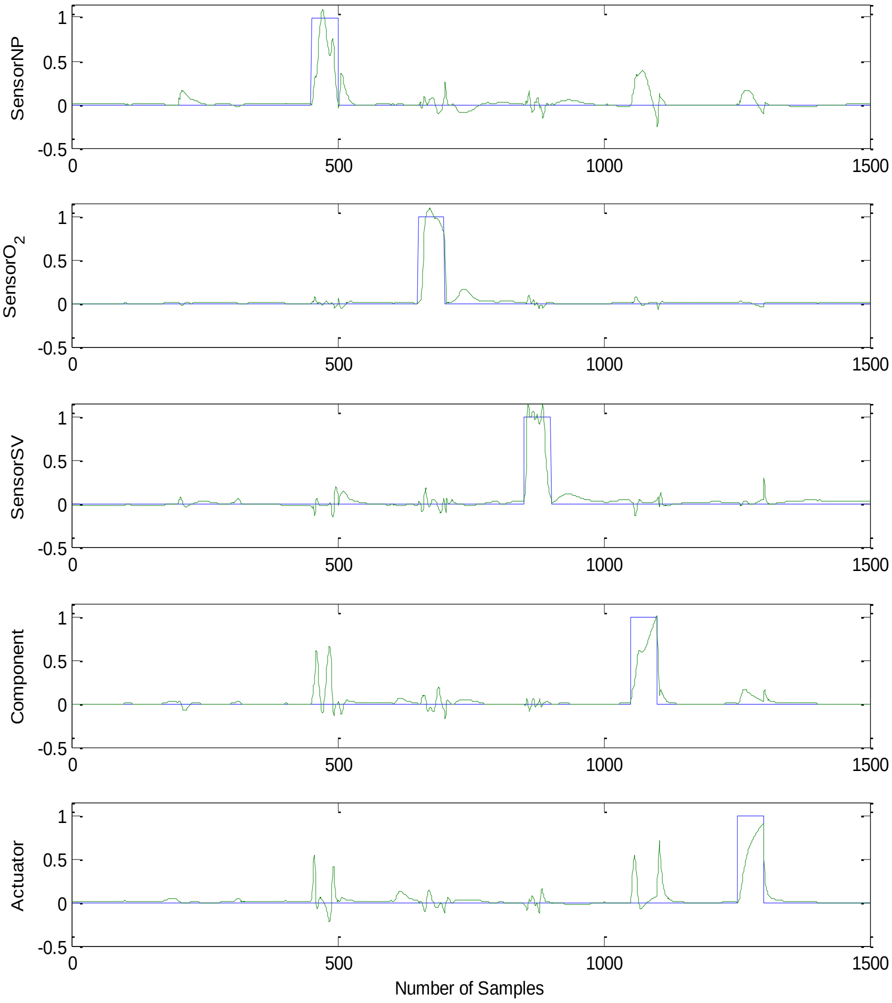


**Fig. 8.10** RBF classifier outputs with RBF residual

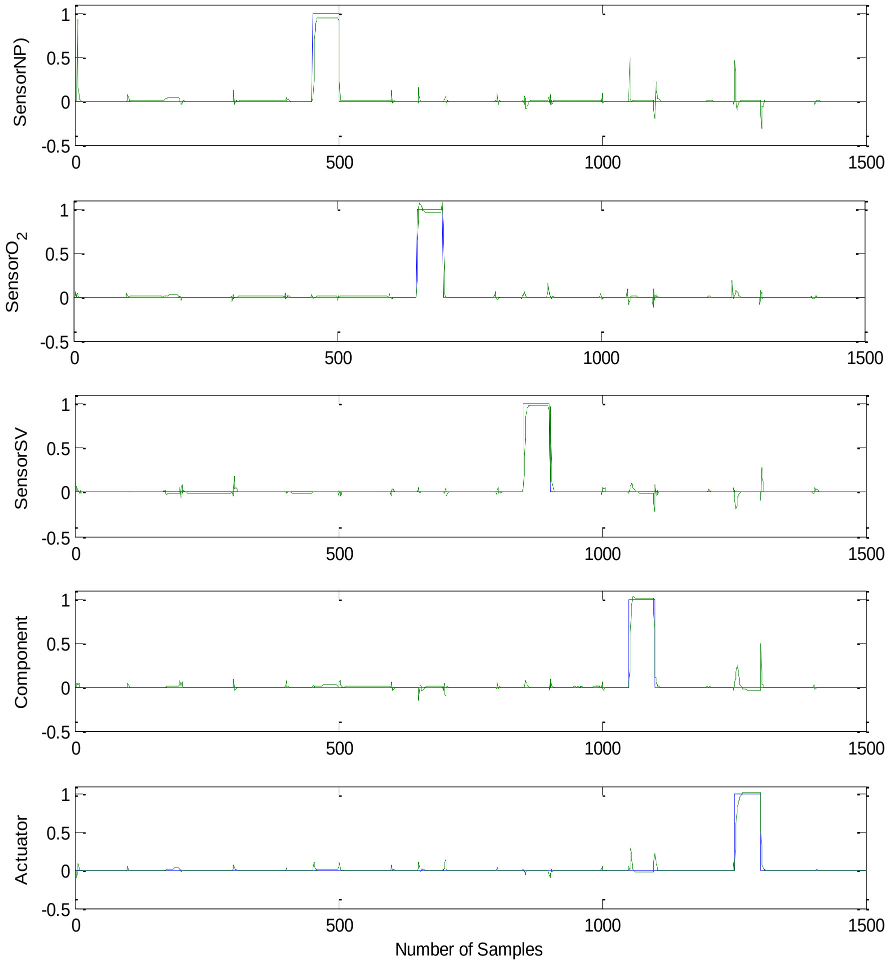


**Fig. 8.11** RBF classifier outputs with MLP residual

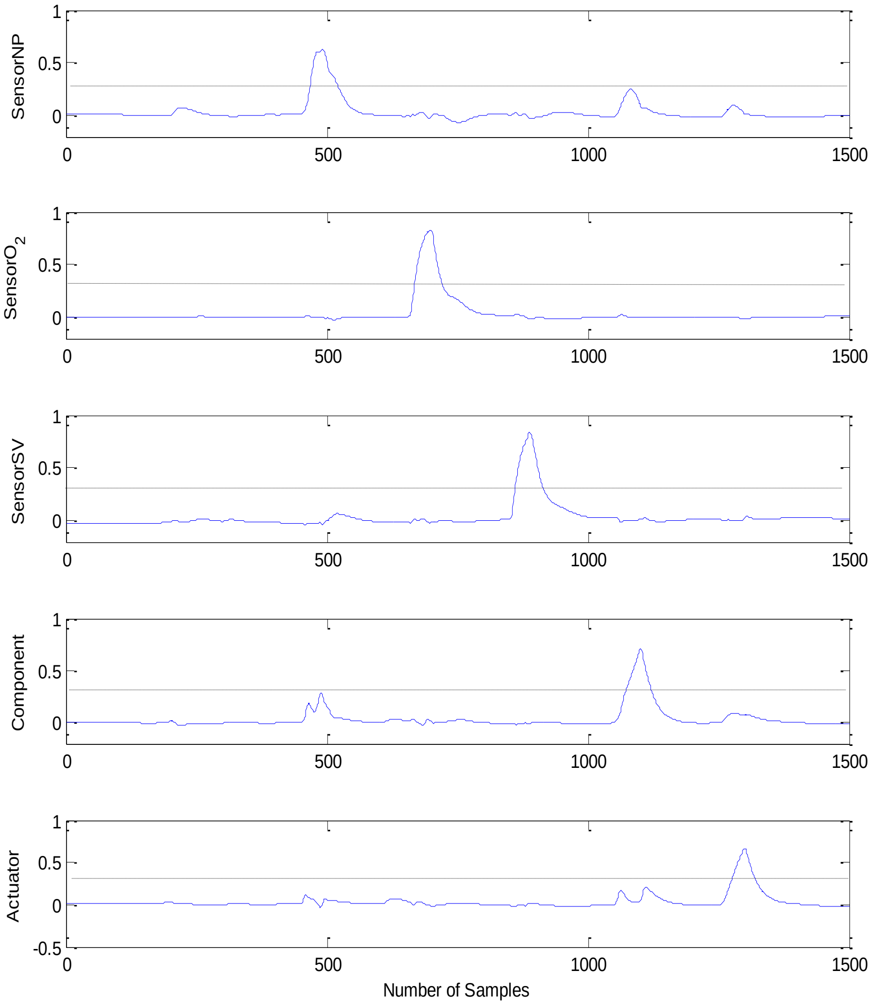


**Fig. 8.12** Filtered RBF classifier output with RBF residual

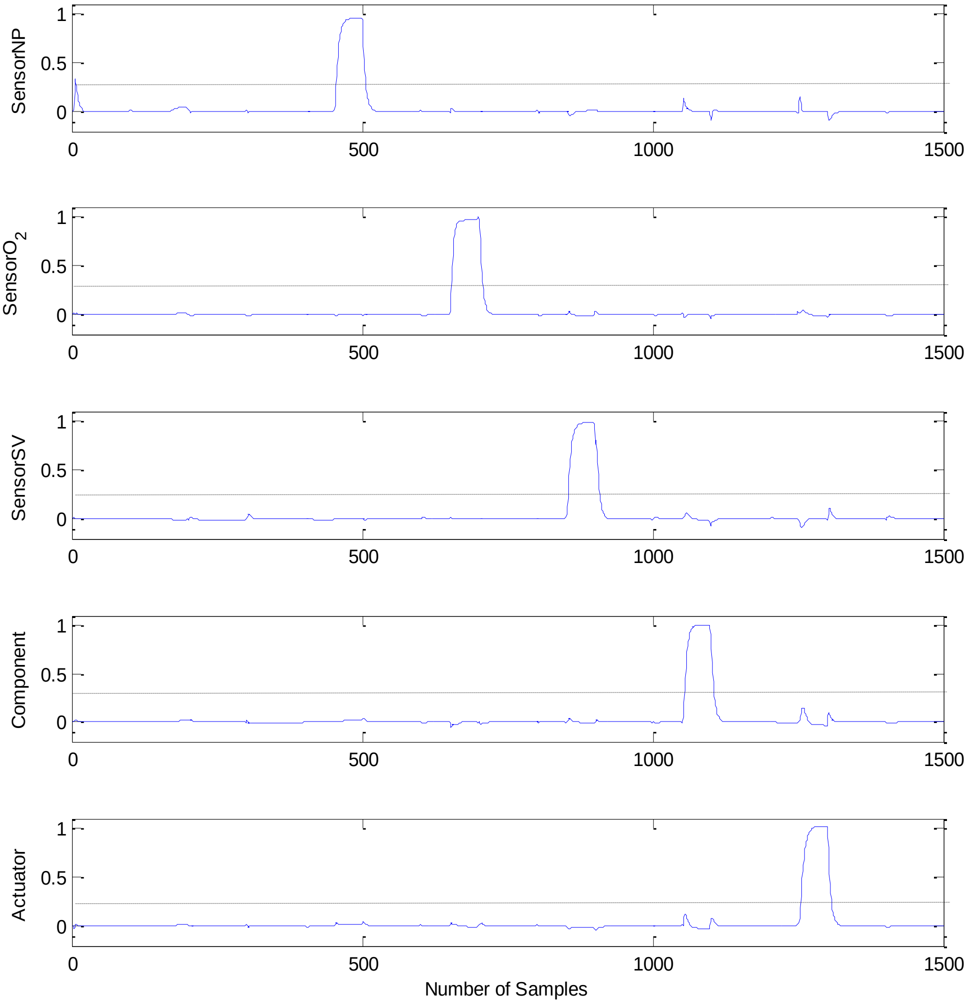


**Fig. 8.13** Filtered RBF classifier output with MLP residual

Once the trained has been completed, RBF classifier is tested and later these outputs for both networks are filtered and displayed as in Fig. 8.12 and Fig. 8.13. It is obvious that the filtered fault isolation signals are much smoother and the robustness of the signal to modelling errors. It is important to isolate the malfunction devices in the systems for easy troubleshooting and maintenance purposes. By doing this step, the device can easily be replaced and any appropriate action can be taken quickly and therefore it can save time and increase productivity. Fig. 8.12 and Fig. 8.13 show the final results of RBF classifier while doing fault isolation for closed-loop system using the residual outputs of RBF and MLP network. The dotted line in these two figures will eliminated false alarms in the signals.

## 8.3 Summary

The implementation of RBF classifier is done in the MATLAB R2009a/Simulink environment. In this work, three results of two control schemes are presented. The fault isolation was tested for open-loop and closed-loop control schemes. These residual signals are transformed into a structured residual set based on individual fault at their respective time. In this work it is found in the simulations that the RBF classifier with 22 hidden layer nodes is most suitable for the classification task in this research for both network residual outputs. As far as accuracy of prediction is concerned, the performance of the RBF model using the K-means clustering technique is satisfactory. The developed model is sensitive to these five faults. The simulation result shows that all five faults are successfully isolated. This model will be useful for the optimal design and real-time FDI of the PEMFC dynamic systems.

# CHAPTER 9

# CONCLUSION AND FUTURE WORK

## 9.1 Conclusion and discussion

The goal of this research is to develop a simple, new and effective method of getting relevant information of faults or malfunctions occurring in the FC stack. This research presents a basic concept and approach in independent model-based FDI diagnostic techniques. Because of the nonlinear behaviour and the complexity of the mathematical model, we focused on the basic concept and those which we think close a gap in existing theory and may gain some relevance for future research and practical applications. For this purpose, we presented intelligent FDI using independent model-based neural networks for PEMFC dynamic systems. Two types of neural networks model have been implemented in this work; RBF networks model and MLP networks model for open-loop and closed-loop control.

Most other techniques used the model-based observer to perform fault diagnosis which used either parameter estimation or parity space approach. On the other hand most of the model-based used the output of the plant as inputs to the neural networks or to system parameters. In this case if the plant experienced faults during the operation and execution the output used is not sensitive to faults then the signal itself is contaminated with faulty signals. If the fault size or amplitude is small then it is not obvious because it may distinguish when threshold applied else the signal cannot be distinguished due to this faulty signal.

Based on this we develop an independent model-based which is not affected when faults occurred or happened in the plant systems. Our study demonstrates that the independent model-based technique does not affected the input signals to the RBF network and MLP network because their inputs are design so that it is not dependent to the output signals of the process plant. The proposed models have shown an excellent achievement with healthy and faulty data sets. Faulty data set consist of five types of faults and was simulated on a PEMFC simulation model developed by University of Michigan. The residual signals produce in the modelling prediction errors was used as inputs to another neural network act as a classifier.

Fig. 9.1 shows the filtered modelling prediction error for RBF and MLP network due to step input signals for open-loop control. It shows that there is a $\text{sensor}_{\text{NP}}$ fault in the output of NP in the residual signal of MLP network. However, using the RBF residual signal, the fault of $\text{sensor}_{\text{NP}}$ cannot be detected. Based on the observation, it is obvious that there is more than one faults occurring in the filtered squared errors display at the outputs. Therefore to detect these faults, the equation of residual generator is applied in order to make these five simulated faults detectable.

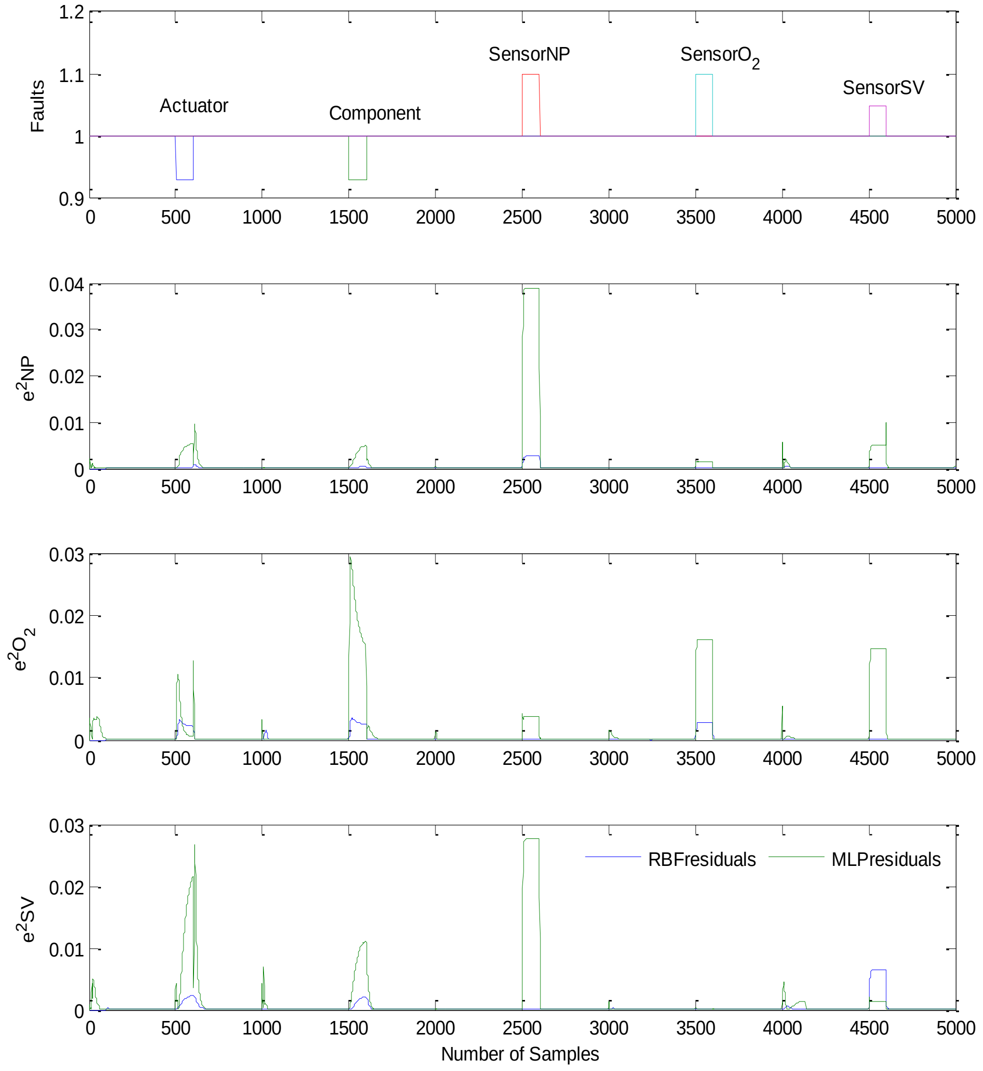


**Fig. 9.1** Filtered and squared model prediction errors

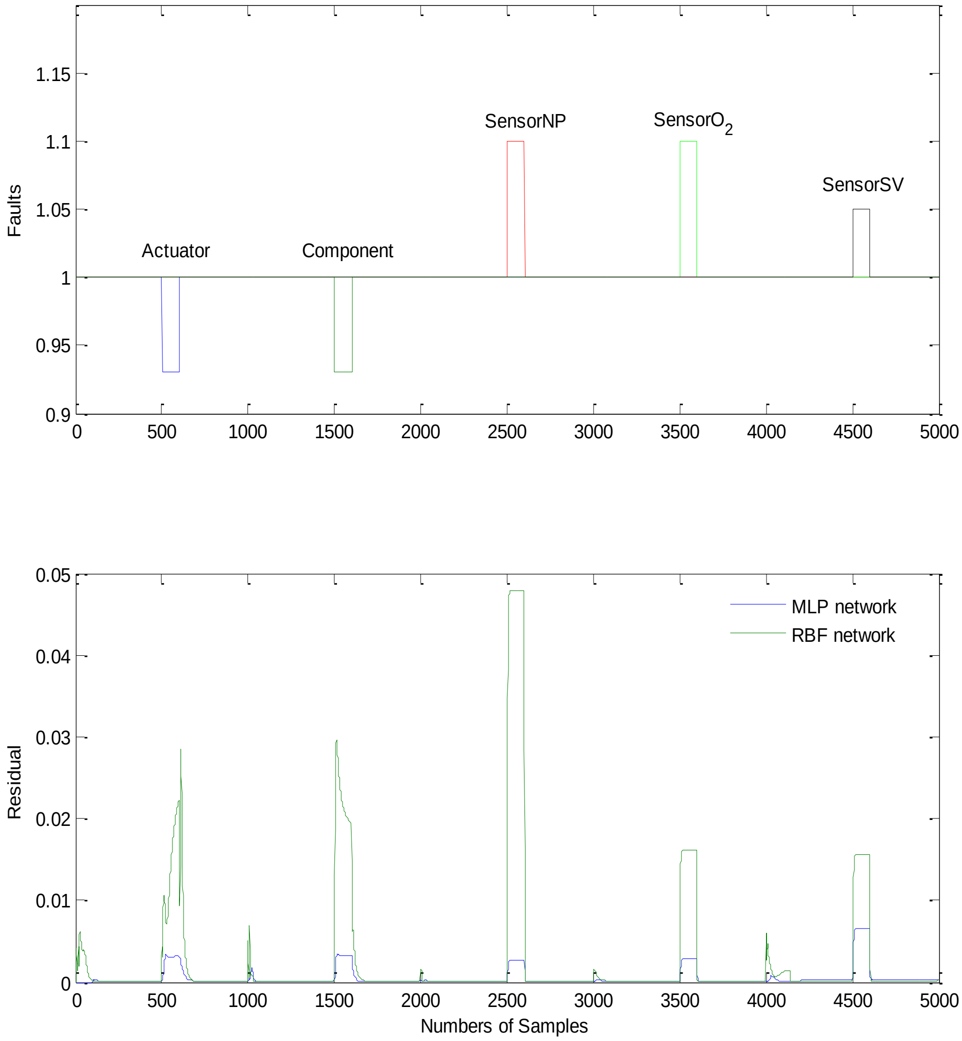


**Fig. 9.2** Residual signal for the five simulated faults

Based on Fig. 9.2, it shows that all five faults can be detected. However the amplitude of fault size in RBF network model is higher than MLP network model. The features of fault at k=501-600 and k=1501-1600 for both networks are not the same. The fault signals of MLP network model is not smooth at this time compared with the other three signals. The reason maybe because of the change in the state of input signals. That is why these two faults are not smoother.

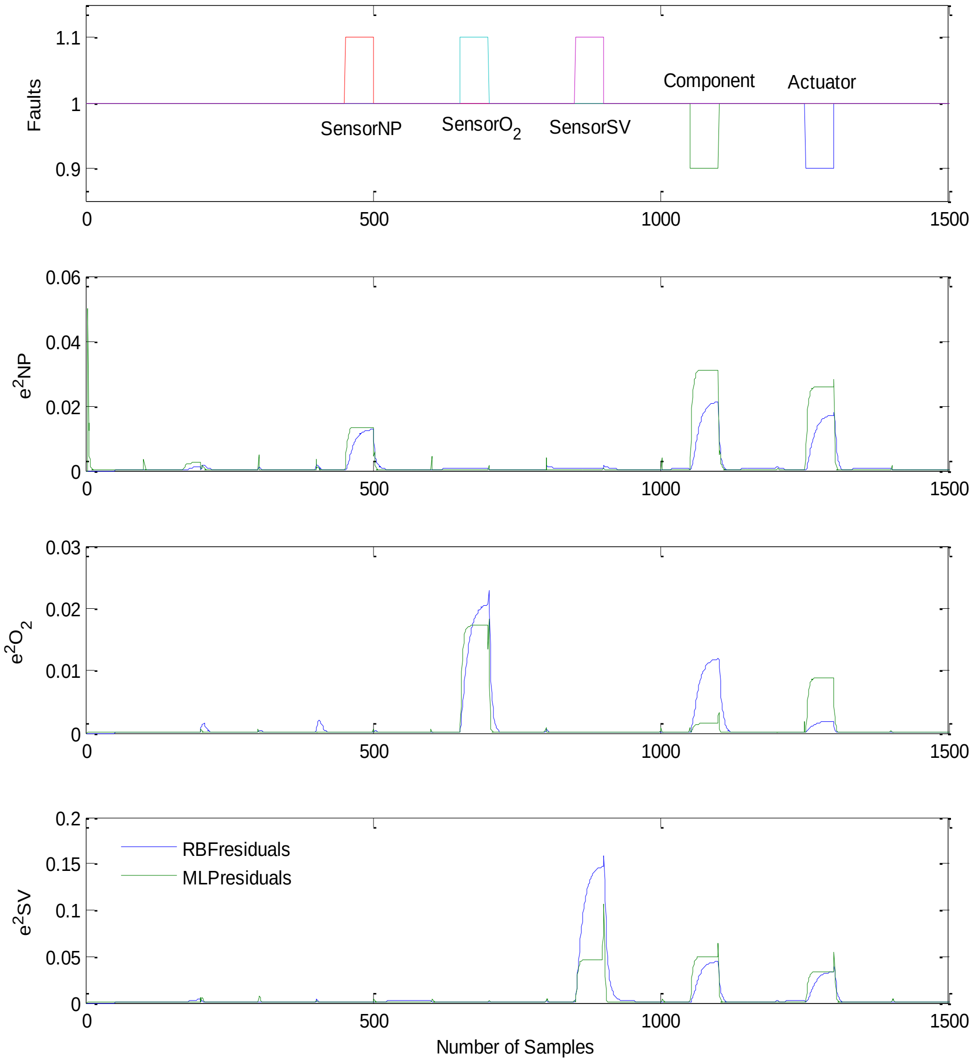


**Fig. 9.3** Filtered and squared model prediction errors

The above result is based on closed-loop control. Here, the filtered squared error signals pattern for both networks is quite similar as shown in Fig. 9.3. However, fault cannot be detected straight away due to more than one faults occurred their output signals. Therefore, Fig. 9.4 presented the fault detection of five faults after performing the residual generator. From the observation all five faults are detectable and it is important because later these signals are fed to the RBF classier to isolate these five faults. The faults are similar expect for the amplitude of sensor$_{SV}$ fault in RBF network model is higher than MLP network model.

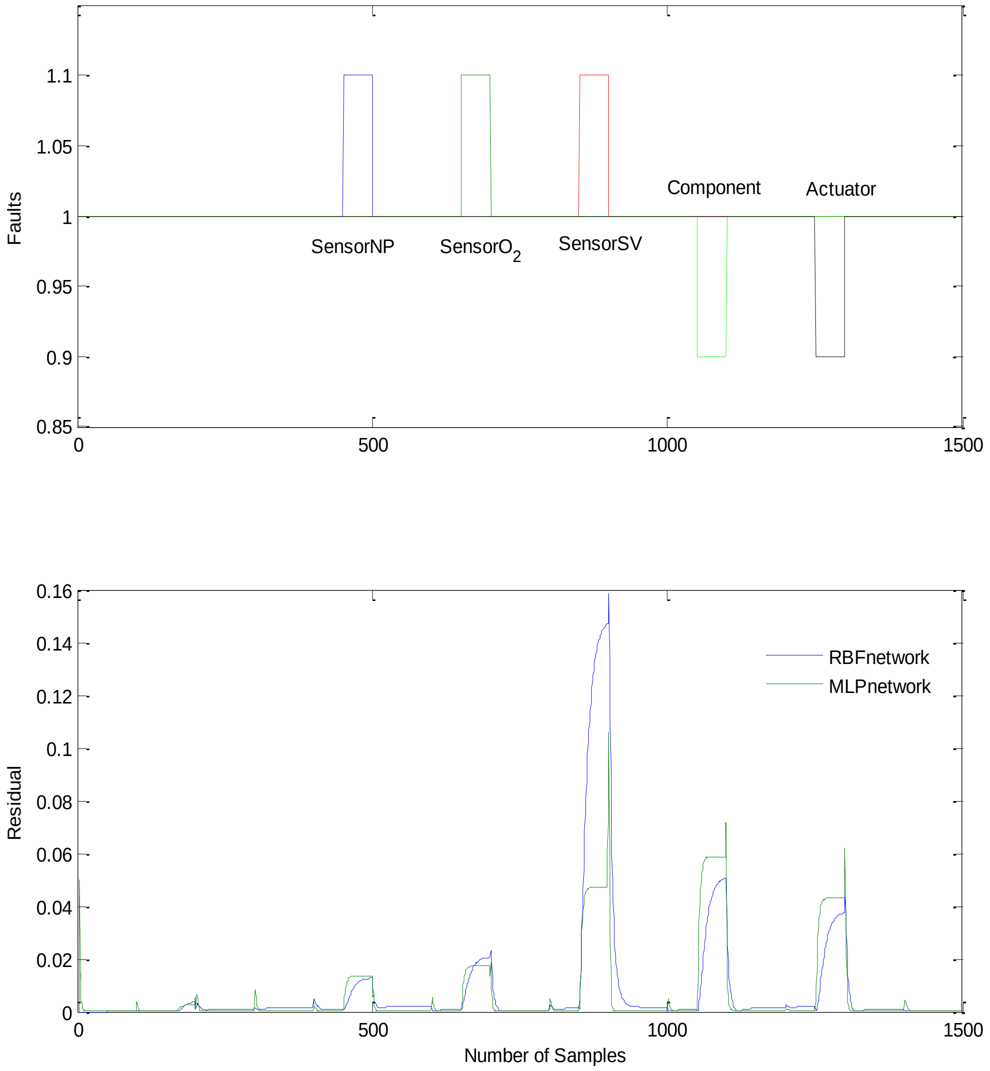


**Fig. 9.4** Residual signal for the five simulated faults

As a conclusion, we can conclude that the independent RBF network model and the MLP network model are able to perform fault detection where all the simulated five faults are detectable either in open-loop or closed-loop control. To perform fault isolation, RBF classifier is applied to both networks which prove that these five faults can be isolated. The developed method has a big potential to be applied to real world of FC stack. Also, the method is not limited to FC systems, and can be applied to other multivariable nonlinear dynamic systems with some modifications.

## 9.2 Future work

This work forms the basis for developing an independent neural network models for FDI strategies for PEMFC systems. It is considered as a starting point before the reconfiguration part takes place. The basic idea behind model-based fault diagnosis is the generation of residuals, consisting of the difference between the process plant and the estimated model. It is well known that the core element of model-based fault detection in control systems is the generation of residual signals which act as indicator to the controller of the process plant.

The reliability and the effectiveness of this FDI approach need to be tested in real-time application. It is extremely useful if this method can be linked in designing of practical applications. The proposed models need further tests in larger environment before deployed for practical applications in the real world. For future work the following suggestions need to be considered:

1) Conduct a simulation for fault detection based on multiple faults occurring at the same location and perform fault isolation.
2) Varies the input signals and the fault types to validate the effectiveness of the RBF network and MLP network algorithm to detect faults.
3) Implement a fuzzy logic to perform fault classification.
4) Later, the proposed FDI method needs to be tested with a real test rig.

5) By doing the above, these models can be further extended to control and reconfigure the controller while adjusting the systems performance and reliability due to faults existence.

# REFERENCES

Athans, M., Fekri, S., & Pascoal, A. (2005, July). Issues on robust adaptive feedback control. In *Proceedings of 16th IFAC world congress, Prague, Czech Republic*.

Angeli, C. (2010). Diagnostic Expert Systems: From Expert’s Knowledge to Real—Time Systems. *Advanced Knowledg Based Systems (Model, Applications & Search)*, *1*, 50-73.

Ayoubi, M. (1995, June). Neuro-fuzzy structure for rule generation and application in the fault diagnosis of technical processes. In *American Control Conference, Proceedings of the 1995* (Vol. 4, pp. 2757-2761). IEEE.

Basseville, M. (1988). Detecting changes in signals and systems—a survey. *Automatica*, *24*(3), 309-326.

Basseville, M. and A. Benveniste (Eds) (1985). Detection of Abrupt Changes in Signals and Dynamical Systems. *Lecture Notes in Control and Information Sciences*, Vol.77, Springer, Berlin.

Bavarian, M., Soroush, M., Kevrekidis, I. G., & Benziger, J. B. (2010). Mathematical Modeling, Steady-State and Dynamic Behavior, and Control of Fuel Cells: A Review†. *Industrial & engineering chemistry research*, *49*(17), 7922-7950.

Beard, R. V. (1971). Failure accommodation in linear systems through self-reorganization', 1971. *PhD, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, MA.*

Bethoux, O., Hilairet, M., & Azib, T. (2009, November). A new on-line state-of-health monitoring technique dedicated to PEM fuel cell. In *Industrial Electronics, 2009. IECON'09. 35th Annual Conference of IEEE* (pp. 2745-2750). IEEE.

Bhagwat, A., Srinivasan, R., & R Krishnaswamy, P. (2003). Multi-linear model-based fault detection during process transitions. *Chemical Engineering Science*, *58*(9), 1649-1670.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford university press.

Buchholz, M., Eswein, M., & Krebs, V. (2008, September). Modelling PEM fuel cell stacks for FDI using linear subspace identification. In *Control Applications, 2008. CCA 2008. IEEE International Conference on* (pp. 341-346). IEEE.

Chang, D. H., & Islam, S. (2000). Estimation of soil physical properties using remote sensing and artificial neural network. *Remote Sensing of Environment*, *74*(3), 534-544.

Chen, J., Patton, R. J., & Zhang, H. Y. (1996). Design of unknown input observers and robust fault detection filters. *International Journal of Control*, *63*(1), 85-105.

Chiang, L. H., Braatz, R. D., & Russell, E. L. (2001). *Fault detection and diagnosis in industrial systems*. Springer.

Chong, E. K., & Zak, S. H. (2013). *An introduction to optimization* (Vol. 76). John Wiley & Sons.

Chow, E., & Willsky, A. S. (1984). Analytical redundancy and the design of robust failure detection systems. *Automatic Control, IEEE Transactions on*, *29*(7), 603-614.

Clancey, W. J., & Shortliffe, E. H. (1984). Introduction: Medical artificial intelligence programs. *Readings in Medical Artificial Intelligence. The First Decade. Massachusetts, Addison-Wesley Publishing Company*, 1-17.

De Lira, S., Puig, V., & Quevedo, J. (2009). PEM Fuel Cells System Robust LPV model-based Fault Diagnosis. In *20th International Workshop on Principles of Diagnosis* (pp. 91-98).

Delashmit, W. H., & Manry, M. T. (2005). Recent developments in multilayer perceptron neural networks. In *Proceedings of the seventh Annual Memphis Area Engineering and Science Conference, MAESC*.

Delmaire, G., Cassar, J. P., & Staroswiecki, M. (1994, December). Identification and parity space techniques for failure detection in SISO systems including modelling errors. In *Decision and Control, 1994., Proceedings of the 33rd IEEE Conference on* (Vol. 3, pp. 2279-2285). IEEE.

Diao, Y., & Passino, K. M. (2002). Intelligent fault-tolerant control using adaptive and learning methods. *Control Engineering Practice*, *10*(8), 801-817.

Ding, X., & Frank, P. M. (1990). Fault detection via factorization approach. *Systems & control letters*, *14*(5), 431-436.

Ding, X., Guo, L., & Frank, P. M. (1994). Parameterization of linear observers and its application to observer design. *IEEE transactions on automatic control, 39*(8), 1648-1652.

Duan, G. R., Patton, R. J., Chen, J., & Chen, Z. (1997, August). A parametric approach for robust fault detection in linear systems with unknown disturbances. In *Proc. IFAC Symp. SAFEPROCESS '97, 318* (Vol. 322).

Escobet, T., Feroldi, D., De Lira, S., Puig, V., Quevedo, J., Riera, J., & Serra, M. (2009). Model-based fault diagnosis in PEM fuel cell systems. *Journal of Power Sources*, *192*(1), 216-223.

Frank, P. M. (1987). Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey. *Automatica*, *26*(3), 459-474.

Frank, P. M. (1994). Enhancement of robustness in observer-based fault detection†. *International Journal of control*, *59*(4), 955-981.

Frank, P. M. (1996). Analytical and qualitative model-based fault diagnosis–a survey and some new results. *European Journal of control*, *2*(1), 6-28.

Frank, P. M., & Keller, L. (1980). Sensitivity discriminating observer design for instrument failure detection. *Aerospace and Electronic Systems, IEEE Transactions on*, (4), 460-467.

Frank, P. M., & Köppen-Seliger, B. (1997). Fuzzy logic and neural network applications to fault diagnosis. *International Journal of Approximate Reasoning*, *16*(1), 67-88.

Gatzke, E. P., & Doyle Iii, F. J. (2002). Use of multiple models and qualitative knowledge for on-line moving horizon disturbance estimation and fault diagnosis. *Journal of Process Control*, *12*(2), 339-352.

Ge, W., & FANG, C. Z. (1988). Detection of faulty components via robust observation. *International Journal of Control*, *47*(2), 581-599.

Gertler, J. J. (1988). Survey of model-based failure detection and isolation in complex plants. *IEEE Control Syst. Mag.*, *8*(6), 3-11.

Gertler, J. (1997). Fault detection and isolation using parity relations. *Control engineering practice*, *5*(5), 653-661.

Gertler, J., & Singer, D. (1990). A new structural framework for parity equation-based failure detection and isolation. *Automatica*, *26*(2), 381-388.

Gertler, J. J., & Monajemy, R. (1995). Generating directional residuals with dynamic parity relations. *Automatica*, *31*(4), 627-635.

Gertler, J. (1991, September). Analytical redundancy methods in fault detection and isolation. In *Proceedings of IFAC/IAMCS symposium on safe process* (Vol. 1, pp. 9-21).

Gibeault, J. P., & Kirkup, J. K. (1995, September). Early detection and continuous monitoring of dissolved key fault gases in transformers and shunt reactors. In *Electrical Electronics Insulation Conference, 1995, and Electrical Manufacturing & Coil Winding Conference. Proceedings* (pp. 285-293). IEEE.

Gomm, J. B., Williams, D., Evans, J. T., Doherty, S. K., & Lisboa, P. J. G. (1996). Enhancing the non-linear modelling capabilities of MLP neural networks using spread encoding. *Fuzzy sets and systems, 79*(1), 113-126.

Gomm, J. B., & Yu, D. L. (2000). Selecting radial basis function network centers with recursive orthogonal least squares training. *Neural Networks, IEEE Transactions on*, *11*(2), 306-314.

Grujicic, M., Chittajallu, K. M., Law, E. H., & Pukrushpan, J. T. (2004). Model-based control strategies in the dynamic interaction of air supply and fuel cell. *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, *218*(7), 487-499.

Himmelblau, D. M., Barker, R. W., & Suewatanakul, W. (1991). Fault classification with the aid of artificial neural networks.

Hou, M., & Muller, P. C. (1992). Design of observers for linear systems with unknown inputs. *Automatic Control, IEEE Transactions on, 37*(6), 871-875.

Hou, M., & Müller, P. C. (1994). Fault detection and isolation observers. *International Journal of Control*, *60*(5), 827-846.

Hwang, I., Kim, S., Kim, Y., & Seah, C. E. (2010). A survey of fault detection, isolation, and reconfiguration methods. *Control Systems Technology, IEEE Transactions on*, *18*(3), 636-653.

Ibrir, S., & Cheddie, D. (2009, July). Model-based estimation of PEM fuel-cell systems. In *Control Applications,(CCA) & Intelligent Control,(ISIC), 2009 IEEE* (pp. 1397-1402). IEEE.

Ingimundarson, A., Bravo, J. M., Puig, V., & Alamo, T. (2005a, December). Robust fault diagnosis using parallelotope-based set-membership consistency tests. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on* (pp. 993-998). IEEE.

Ingimundarson, A., & Hägglund, T. (2005b). Closed-loop performance monitoring using loop tuning. *Journal of Process Control*, *15*(2), 127-133.

Ingimundarson, A., Stefanopoulou, A. G., & McKay, D. A. (2008). Model-based detection of hydrogen leaks in a fuel cell stack. *Control Systems Technology, IEEE Transactions on*, *16*(5), 1004-1012.

Isermann, R. (1984). Process fault detection based on modeling and estimation methods—a survey. *Automatica*, *20*(4), 387-404.

Isermann, R. (1987). Experiences with process fault detection methods via parameter estimation. In *System Fault Diagnostics, Reliability and Related Knowledge-Based Approaches* (pp. 3-33). Springer Netherlands.

Isermann, R. (1993). Fault diagnosis of machines via parameter estimation and knowledge processing—tutorial paper. *Automatica*, *29*(4), 815-835.

Isermann, R. (1997). Supervision, fault detection and fault diagnosis methods-An introduction. *Control Eng Practice 5*(5), 639-652.

Isermann, R., & Balle, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical processes. *Control engineering practice*, *5*(5), 709-719.

Jones, H. L. (1973). Failure detection in linear system. Ph.D. Thesis, MIT, Cambridge, MA.

Kamal, M., & Yu, D. (2011). Model-based fault detection for proton exchange membrane fuel cell systems. *International Journal of Engineering, Science and Technology*, *3*(9), 1-15.

Kamal, M. M., & Yu, D. (2012, September). Fault detection and isolation for PEMFC systems under closed-loop control. In *Control (CONTROL), 2012 UKACC International Conference on* (pp. 976-981). IEEE.

Kamal, M. M., Yu, D. W., & Yu, D. L. (2014). Fault detection and isolation for PEM fuel cell stack with independent RBF model. *Engineering Applications of Artificial Intelligence*, *28*, 52-63.

Kang, H. (1993). Stability and control of fuzzy dynamic systems via cell-state transitions in fuzzy hypercubes. *Fuzzy Systems, IEEE Transactions on*, *1*(4), 267-279.

Keller, J. Y. (1999). Fault isolation filter design for linear stochastic systems. *Automatica*, *35*(10), 1701-1706.

Kashaninejad, M., Dehghani, A. A., & Kashiri, M. (2009). Modeling of wheat soaking using two artificial neural networks (MLP and RBF). *Journal of food engineering*, *91*(4), 602-607.

Lebbal, M. E., & Lecœuche, S. (2009). Identification and monitoring of a PEM electrolyser based on dynamical modelling. *International Journal of Hydrogen Energy*, *34*(14), 5992-5999.

Lecture notes of ME 475: Note 9: Closed-loop control. Introduction to Mechatronics Department of mechanical engineering, University of Saskatchewan, Saskatoon, SK

Leontaritis, I. J., & Billings, S. A. (1985). Input-output parametric models for non-linear systems part I: deterministic non-linear systems. *International journal of control*, *41*(2), 303-328.

Li, M., He, S., & Li, X. (2009). Complex radial basis function networks trained by QR-decomposition recursive least square algorithms applied in behavioral modeling of nonlinear power amplifiers. *International Journal of RF and Microwave Computer-Aided Engineering*, *19*(6), 634-646.

Li, X. R., & Bar-Shalom, Y. (1993). Performance prediction of the interacting multiple model algorithm. *Aerospace and Electronic Systems, IEEE Transactions on*, *29*(3), 755-771.

Lippmann, R. P. (1987). An introduction to computing with neural nets. *ASSP Magazine, IEEE*, *4*(2), 4-22.

Maybeck, P. S. (1999). Multiple model adaptive algorithms for detecting and compensating sensor and actuator/surface failures in aircraft flight control systems. *International Journal of Robust and Nonlinear Control*, *9*(14), 1051-1070.

Mehra, R. K. and I. Peshon (1971). An innovation approach to fault detection and diagnosis in dynamic systems. *Automatica*, 7, 637-640.

Mironovski, L. A. (1979). Functional diagnosis of linear dynamic systems. *Automn Remote Control*, *40,* 1198-1205.

Mironovski, L. A. (1980). Functional diagnosis of dynamic system--A survey. *Automn Remote Control*, *41*, 1122-1143.

Morales, R. A. G., Riascos, L. A., & Miyagi, P. E. (2008). FAULT DIAGNOSIS IN FUEL CELLS BASED ON BAYESIAN NETWORKS.

Moustafa, A.A. (2011). Performance evaluation of artificial neural networks for spatial data. *Analysis, Comtemporary Engineering Sciences 4(4), 149-163.*

Naidu, S. R., Zafiriou, E., & McAvoy, T. J. (1990). Use of neural networks for sensor failure detection in a control system. *Control Systems Magazine, IEEE*, *10*(3), 49-55.

Narendra, K. S., & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *Neural Networks, IEEE Transactions on*, *1*(1), 4-27.

Nazari, J., & Ersoy, O. K. (1992). Implementation of back-propagation neural networks with MatLab.

Nomura, H., Hayashi, I., & Wakami, N. (1992, March). A learning method of fuzzy inference rules by descent method. In *Fuzzy Systems, 1992., IEEE International Conference on* (pp. 203-210). IEEE.

Norvilas, A., Negiz, A., DeCicco, J., & Çinar, A. (2000). Intelligent process monitoring by interfacing knowledge-based systems and multivariate statistical monitoring. *Journal of Process Control*, *10*(4), 341-350.

Ogata, K. Modern control engineering, 1997. *ISBN: 0-13-227307-1*, 299-231.

Ou, S., & Achenie, L. E. (2005). Artificial neural network modeling of PEM fuel cells. *Journal of Fuel Cell Science and Technology*, *2*(4), 226-233.

Patton, R. J. (1988). Robust fault detection using eigenstructure assignment.

Patton, R. J., & Chen, J. (1991, December). Robust fault detection using eigenstructure assignment: a tutorial consideration and some new results. In

*Decision and Control, 1991., Proceedings of the 30th IEEE Conference on* (pp. 2242-2247). IEEE.

Patton, R. J., & Chen, J. (1994). Review of parity space approaches to fault diagnosis for aerospace systems. *Journal of Guidance, Control, and Dynamics*, *17*(2), 278-285.

Patton, R. J., & Chen, J. (1997). Observer-based fault detection and isolation: robustness and applications. *Control Engineering Practice*, *5*(5), 671-682.

Patton, R. J., & Chen, J. (1999). Robust model-based fault diagnosis for dynamic systems.

Patton, R. J., Frank, P. M., & Clarke, R. N. (1989). *Fault diagnosis in dynamic systems: theory and application*. Prentice-Hall, Inc..

Patton, R. J., Chen, J., & Siew, T. M. (1994b, March). Fault diagnosis in nonlinear dynamic systems via neural networks. In *Control, 1994. Control'94. International Conference on* (Vol. 2, pp. 1346-1351). IET.

Patton, R.J., 1994a. Robust model-based fault diagnosis: the state of art. In: Proceeding of IFAC Sym. SAFEPROCESS’94, 1-24.

Patton, R. J., Lopez-Toribio, C. J., & Uppal, F. J. (1999). Artificial intelligence approaches to fault diagnosis. In *Condition Monitoring: Machinery, External Structures and Health (Ref. No. 1999/034), IEE Colloquium on* (pp. 5-1). IET.

Ploix, S., & Adrot, O. (2006). Parity relations for linear uncertain dynamic systems. *Automatica*, *42*(9), 1553-1562.

Porfírio, C. R., Almeida Neto, E., & Odloak, D. (2003). Multi-model predictive control of an industrial c3/c4 splitter. *Control engineering practice*, *11*(7), 765-779.

Prasad, P. R., Davis, J. F., Jirapinyo, Y., Josephson, J. R., & Bhalodia, M. (1998). Structuring diagnostic knowledge for large-scale process systems. *Computers & chemical engineering*, *22*(12), 1897-1905.

Puig, V., Quevedo, J., Escobet, T., & Stancu, A. (2003). Passive robust fault detection using linear interval observers. *IFAC Safe Process*.

Pukrushpan, J. T., Peng, H., & Stefanopoulou, A. G. (2002, January). Simulation and analysis of transient fuel cell system performance based on a dynamic reactant flow model. In *ASME 2002 International Mechanical Engineering Congress and Exposition* (pp. 637-648). American Society of Mechanical Engineers.

Pukrushpan, J.T. 2003. Modeling and Control of Fuel cell systems and fuel processors, PhD thesis, Department of Mechanical Engineering, The University of Michigan.

Pukrushpan, J. T., Peng, H., & Stefanopoulou, A. G. (2004a). Control-oriented modeling and analysis for automotive fuel cell systems. *Journal of dynamic systems, measurement, and control*, *126*(1), 14-25.

Pukrushpan, J. T., Stefanopoulou, A. G., & Peng, H. (2004b). *Control of fuel cell power systems: principles, modeling, analysis and feedback design*. Springer.

Quan, R., Huang, L., Chen, Q., & Quan, S. (2009, August). Study on online monitoring system of PEMFC resistance based on electrochemical impedance spectroscopy. In *Electronic Measurement & Instruments, 2009. ICEMI'09. 9th International Conference on* (pp. 3-932). IEEE.

Rao, M., Sun, X., & Feng, J. (2000). Intelligent system architecture for process operation support. *Expert Systems with Applications*, *19*(4), 279-288.

Riascos, L. A., Simões, M. G., & Miyagi, P. E. (2006). Fault identification in fuel cells based on Bayesian network diagnosis. In *ABCM Symposium Series in Mechatronics* (Vol. 2, pp. 757-764).

Riascos, L. A. M., Simoes, M. G., & Miyagi, P. E. (2008). On-line fault diagnostic system for proton exchange membrane fuel cells. *Journal of Power Sources, 175*(1), 419-429.

Rhee, F. C. H., & Krishnapuram, R. (1993). Fuzzy rule generation methods for high-level computer vision. *Fuzzy Sets and Systems*, *60*(3), 245-258.

Rodrigues, M., Theilliol, D., Adam-Medina, M., & Sauter, D. (2008). A fault detection and isolation scheme for industrial systems based on multiple operating models. *Control Engineering Practice*, *16*(2), 225-239.

Rosich, A., Sarrate, R., Puig, V., & Escobet, T. (2007, December). Efficient optimal sensor placement for model-based FDI using an incremental algorithm. In *Decision and Control, 2007 46th IEEE Conference on* (pp. 2590-2595). IEEE.

Rosich, A., Sarrate, R., & Nejjari, F. (2013). On-line model-based fault detection and isolation for PEM fuel cell stack systems. *Applied Mathematical Modelling*.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by error propagation, Parallel Distributed Processing, vol. 1, 318-362.

Schneider, R., & Frank, P. M. (1994, August). Fuzzy logic based threshold adaption for fault detection in robots. In *Control Applications, 1994., Proceedings of the Third IEEE Conference on* (pp. 1127-1132). IEEE.

Schrick, D. (1997). Remarks on terminology in the field of supervision, fault detection and diagnosis. In *Preprints of the 3rd IFAC Symposium on Fault*

*Detection, Supervision and Safety for Technical Processes SAFEPROCESS'97* (pp. 959-964).

Sneider, H., & Frank, P. M. (1996). Observer-based supervision and fault detection in robots using nonlinear and fuzzy logic residual evaluation. *Control Systems Technology, IEEE Transactions on*, *4*(3), 274-282.

Sorsa, T., & Koivo, H. N. (1993). Application of artificial neural networks in process fault diagnosis. *Automatica*, *29*(4), 843-849.

Staroswiecki, M., Cassar, J. P., & Cocquempot, V. (1993). Generation of optimal structured residuals in the parity space.

Steiner, N. Y., Candusso, D., Hissel, D., & Moçoteguy, P. (2010). Model-based diagnosis for proton exchange membrane fuel cells. *Mathematics and Computers in Simulation*, *81*(2), 158-170.

Stengel, R. F. (1991). Intelligent failure-tolerant control. *Control Systems, IEEE*, *11*(4), 14-23.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *Systems, Man and Cybernetics, IEEE Transactions on*, (1), 116-132.

Tian, G., Wasterlain, S., Endichi, I., Candusso, D., Harel, F., François, X., ... & Kauffmann, J. M. (2008). Diagnosis methods dedicated to the localisation of failed cells within PEMFC stacks. *Journal of Power Sources*, *182*(2), 449-461.

Uraikul, V., Chan, C. W., & Tontiwachwuthikul, P. (2007). Artificial intelligence for monitoring and supervisory control of process systems. *Engineering Applications of Artificial Intelligence*, *20*(2), 115-131.

Vaidyanathan, R., & Venkatasubramanian, V. (1992). Representing and diagnosing dynamic process data using neural networks. *Engineering Applications of Artificial Intelligence*, *5*(1), 11-21.

Vahidi, A., Stefanopoulou, A., & Peng, H. (2004, June). Model predictive control for starvation prevention in a hybrid fuel cell system. In *American Control Conference, 2004. Proceedings of the 2004* (Vol. 1, pp. 834-839). IEEE.

Vahidi, A., Kolmanovsky, I., & Stefanopoulou, A. (2007). Constraint handling in a fuel cell system: A fast reference governor approach. *Control Systems Technology, IEEE Transactions on*, *15*(1), 86-98.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., & Yin, K. (2003a). A review of process fault detection and diagnosis: Part III: Process history based methods. *Computers & chemical engineering*, *27*(3), 327-346.

Venkatasubramanian, V., Rengaswamy, R., Yin, K., & Kavuri, S. N. (2003b). A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. *Computers & chemical engineering*, *27*(3), 293-311.

Viswanadham, N. and R. Srichander (1987). Fault detection using unknown-input observers. *Control Theory and Advanced Technology,* Vol.3, pp. 91-101. MITA.

Wang, S. W., Yu, D. L., Gomm, J. B., Page, G. F., & Douglas, S. S. (2006). Adaptive neural network model based predictive control for air–fuel ratio of SI engines. *Engineering Applications of Artificial Intelligence*, *19*(2), 189-200.

Watanabe, K., Matsuura, I., Abe, M., Kubota, M., & Himmelblau, D. M. (1989). Incipient fault diagnosis of chemical processes via artificial neural networks. *AIChE Journal*, *35*(11), 1803-1812.

Welch, G., & Bishop, G. (1995). An introduction to the Kalman filter.

Willis, M. J., Di Massimo, C., Montague, G. A., Tham, M. T., & Morris, A. J. (1991, May). Artificial neural networks in process engineering. In *Control Theory and Applications, IEE Proceedings D* (Vol. 138, No. 3, pp. 256-266). IET.

Willsky, A. S. (1976). A survey of design methods for failure detection in dynamic systems. *Automatica*, *12*(6), 601-611.

Willsky, A. S., & Jones, H. L. (1976). A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *Automatic Control, IEEE Transactions on*, *21*(1), 108-112.

Woodland, P. C. (1989, October). Weight limiting, weight quantisation and generalisation in multi-layer perceptrons. In *Artificial Neural Networks, 1989., First IEE International Conference on (Conf. Publ. No. 313)* (pp. 297-300). IET.

Xia, Q., & Rao, M. (1999). Dynamic case-based reasoning for process operation support systems. *Engineering Applications of Artificial Intelligence*, *12*(3), 343-361.

Xue, X., Tang, J., Sammes, N., & Ding, Y. (2006). Model-based condition monitoring of PEM fuel cell using Hotelling $T^2$ control limit. *Journal of power sources*, *162*(1), 388-399.

Yager, R. R. (1987). Using approximate reasoning to represent default knowledge. *Artificial Intelligence*, *31*(1), 99-112.

Yousfi-Steiner, N., Moçotéguy, P., Candusso, D., & Hissel, D. (2009). A review on polymer electrolyte membrane fuel cell catalyst degradation and starvation issues: causes, consequences and diagnostic for mitigation. *Journal of Power Sources*, *194*(1), 130-145.

Yousfi Steiner, N., Hissel, D., Moçotéguy, P., & Candusso, D. (2011). Diagnosis of polymer electrolyte fuel cells failure modes (flooding & drying out) by neural networks modeling. *International journal of hydrogen energy*, *36*(4), 3067-3075.

Yu, D., & Shields, D. N. (1995). Fault diagnosis in bi-linear systems-A survey. In *Proceedings of the third European control conference* (Vol. 1, pp. 360-365).

Yu, D., Shields, D. N., & Daley, S. (1996). A hybrid fault diagnosis approach using neural networks. *Neural Computing & Applications*, *4*(1), 21-26.

Yu, D., & Shields, D. N. (1997). A bilinear fault detection filter. *International Journal of Control*, *68*(3), 417-430.

Yu, D. L., Gomm, J. B., & Williams, D. (1999). Sensor fault diagnosis in a chemical process via RBF neural networks. *Control Engineering Practice*, *7*(1), 49-55.

Yu, D. L., Chang, T. K., & Yu, D. W. (2005). Adaptive neural model-based fault tolerant control for multi-variable processes. *Engineering Applications of Artificial Intelligence*, *18*(4), 393-411.

Zadeh, L. A. (1971). Similarity relations and fuzzy orderings. *Information sciences*, *3*(2), 177-200.

Zhai, Y. J., & Yu, D. L. (2008). Radial-basis-function-based feedforward—feedback control for air—fuel ratio of spark ignition engines. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, *222*(3), 415-428.

Zhou, Y., Wang, D., Li, J., Yi, L., & Huang, H. 2011. Fuzzy Logic Based Interactive Multiple Model Fault Diagnosis for PEM Fuel Cell Systems.

Zimmermann, H. J. (1992). *Fuzzy {Set Theory {and Its Applications Second, Revised Edition*. Kluwer academic publishers.