Report No.: ORNL/TM-2002/196

NERI PROJECT 99-119.
TASK 2. DATA-DRIVEN PREDICTION OF PROCESS VARIABLES. FINAL REPORT

Belle R. Upadhyaya
(Principal Investigator)

Baofu Lu
Ke Zhao

Nuclear Engineering Department
The University of Tennessee
Knoxville, TN 37996-2300

J. A. Mullens
J. March-Leuba
Oak Ridge National Laboratory

September 2002

Prepared by the
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831
Managed by
UT-Battelle, LLC
For
U.S. Department of Energy
Under contract DE-AC05-00OR22725

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

TABLE OF CONTENTS

INTRODUCTION

This report describes the detailed results for task 2 of DOE-NERI project number 99-119 entitled ***Automatic Development of Highly Reliable Control Architecture for Future Nuclear Power Plants.*** This project is a collaboration effort between the Oak Ridge National Laboratory (ORNL,) The University of Tennessee, Knoxville (UTK) and the North Carolina State University (NCSU). UTK is the lead organization for Task 2 under contract number DE-FG03-99SF21906.

Under task 2 we completed the development of data-driven models for the characterization of sub-system dynamics for predicting state variables, control functions, and expected control actions. We have also developed the *Principal Component Analysis (PCA)* approach for mapping system measurements, and a nonlinear system modeling approach called the *Group Method of Data Handling (GMDH)* with rational functions, and includes temporal data information for transient characterization.

The majority of the results are presented in detailed reports for Phases 1 through 3 of our research, which are attached to this report.

TASK 2.2 DETECTION OF SIMULTANEOUS FAULTS

Under this task, we completed the development of a fault detection and isolation module that combines system operational knowledge (including system simulation) and a rule-based logic for *FDI* of both single and dual faults in dissimilar sensor and field devices. In addition, we have developed a complimentary approach that quantifies the prediction errors using a fault pattern classification technique.

The above techniques have been applied to a laboratory process control loop using both simulation and actual loop measurements. The techniques have been demonstrated for detecting and isolating faults in sensors and devices in a U-tube steam generator (UTSG) in a pressurized water reactor (PWR) using a full-scope PWR simulator developed by North Carolina State University. The application to the laboratory system and preliminary application to a PWR steam generator were described in the Phase 1 Report.

TASK 2.3 IMPLEMENTATION OF ON-LINE DIAGNOSTICS SYSTEM

The key contributions of Task 2 during Phase-3 of the project include the following: 1. Development of data-driven system models using *Group Method of Data Handling (GMDH)*, *Principal Component Analysis (PCA)* and *Adaptive Network Fuzzy Inference System (ANFIS)*, 2. Fault detection by tracking model residuals of selected process variables and control functions, and 3. Fault isolation using a rule-based technique, a residual pattern classification technique, and a multi-observer digraph approach. Fault diagnosis, during both steady state and transient operations, is demonstrated with applications to a nuclear plant steam generator. A full-scope physics model of the steam generator in a pressurized water reactor (PWR) has been used to generate an extensive database of normal plant operation and faulty operation data. Some of the faults being monitored include: degradation of turbine control valve, steam generator water level sensor drift, feed water flow meter sensor offset, dead band error in feed control valve, steam pressure sensor drift and steam flow meter offset. The type of degradations used in the study include several dual faults that are selected from the above single device faults.

Group method of data handling (GMDH)

The GMDH constructs a model of a desired output as a function of a set of related inputs from a subsystem, by a successive polynomial approximation (Farlow, 1984). The general relationship has the form shown in Equation (1) where $\{x_1, x_2, \ldots, x_m\}$ is a vector of input variables and y is the variable to be predicted. This formulation can be extended to the prediction of multiple outputs $\{y_1, y_2, \ldots, y_n\}$. An efficient numerical algorithm has been developed for applications to process control loops (Upadhyaya et al.).

$$y = a + \sum_{i=1}^{m} b_i x_i + \sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij} x_i x_j + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m} d_{ijk} x_i x_j x_k + \cdots \qquad (1)$$

Principal component analysis (PCA)

PCA makes use of the property of the data that for normal operation the measurements can be characterized by a low dimension hyper-surface. Faulty conditions in one or more of the field devices lead to deviations from the surface. These deviations from the surface, in terms of prediction residuals, can be used for fault detection. The pattern of the residuals of the various measurements may be established for each type of fault under consideration.

Consider an (*m x n*) data matrix **X,** with *n* samples along the rows and each sample consisting of m measurements. PCA decomposes **X** into a product of scores (**T**) and orthogonal loadings (**P**) as (Kaistha and Upadhyaya, 2001)

$$\mathbf{X} = \mathbf{T}\mathbf{P}^{\mathbf{T}} + \mathbf{E} \qquad (2)$$

where **E** contains the residuals. The principal components (PCs) in the successive columns of **P** are obtained such that maximum variance in **X** is explained. Thus, if the data are highly collinear, the first few PCs explain most of the variability in the data and are retained. The residuals in **E** constitute the unexplained variation in the data and contain the higher PCs that are rejected. The PCs are obtained as the right singular vectors of the data matrix **X**, using its singular value decomposition. The PCA method can be generalized to include nonlinear forms of the measurement vector (Kaistha & Upadhyaya, 2001).

Adaptive network fuzzy inference system (ANFIS)

ANFIS is a data-driven modeling approach that combines the system knowledge with the learning capability of an artificial neural network (Jang, 1993). The system knowledge is represented by rules. The membership functions of each of the input signals are estimated using the training data and a neural network model. This step introduces nonlinearity in the estimated weights for all the postulated rules. For each fuzzy rule, the output is computed using a linear model of the input signals. The strength of this approach lies in the ability to use prior knowledge, and to update membership functions that provide a better model for the desired output.

Fault detection

The first step in the FDI implementation is the detection of possible faults in sensors and other devices. The GMDH, PCA or the ANFIS model is used to compute the residuals between the measured variable and its prediction from other measurements. This calculation is performed for all the variables considered in the analysis. If the residual RMS value exceeds a preset alarm level, then we declare that a possible error exists in one or more of the devices. In this study, we have considered anomalies in one or two devices at a time. Once a fault is detected, the next step is to isolate a single or a dual fault.

Fault isolation using parallel approaches

The first step in the fault isolation procedure is to compute the residual sequence between the measurements and the model-estimated values of the set of variables used in the analysis. For a steam generator system the number of state variables and control functions considered is less than m = 15. For the *GMDH* and *ANFIS* models, the residuals are calculated as the difference between the measurement and the model prediction. The residuals are calculated similarly for the PCA model, where all the state variables considered in the multivariate model are used for residual computation. Thus, if **x** is a sample measurement vector, the residual vector **e** is given by (P is the matrix of principal components)

$$\mathbf{e} = \mathbf{x}(\mathbf{I} - \mathbf{P}\mathbf{P}^{\mathbf{T}}) \qquad (3)$$

The first approach used for fault isolation is the development of a **rule base** for each of the fault types. The rule base describes the directional and magnitude variations in the residuals of all the variables considered in the analysis. The fault isolation is then performed by comparing the residual pattern with each of the pre-established residual patterns (similar to a template matching) for all the faults. The pattern with the best *match* is then used for deciding about the fault type.

The second approach for fault isolation uses the **PCA model of the residuals** for each known fault. The first principal component of this model is used as a fault signature. For a given test case, the residual vector is computed using the data-driven model. This

vector is then projected on to the selected PC direction and the corresponding cosine of the angle is determined. If this measure is close to unity
(> 0.9), then the fault is isolated. This procedure is repeated for all the fault directions.

The third approach for fault isolation is the **multi-model digraph** technique. For a set of m models of the measurements, identify the measurements that have propagated their faults by tracking backwards until a model gives insignificant residual, or its output has not been corrupted. Next, reconstruct all the corrupted outputs by tracking forward from the identified fault origin to the input nodes of the detected model. Compute the residual of the measurement in question using all the reconstructed inputs. If the reconstructed residuals and the original residuals are consistent then a local fault is isolated. For the case of a dual fault, the reconstructed residuals of the local device would deviate from the original residual, indicating an additional fault in the input signal.

The simultaneous implementation of the above techniques increases the confidence of fault isolation.

Applications of the FDI *method to a steam generator system*

Both normal operation and faulty operation data were generated using a full-scope PWR simulation code. The following measurements are considered in the following applications: narrow range (NR) SG water level sensor, feed water flow transmitter, steam flow transmitter, steam pressure transmitter, turbine control valve (TCV) position, feed control valve (FCV) position.

Figure 1shows the plot of the residual directions of the measurements for the case when there is a bias fault in the narrow range SG level sensor. Note that the NR direction signature has a maximum value (≈0.9). The direction signatures for the steam flow and feed flow are not insignificant, primarily because their settings change because of error in the NR sensor and the resulting feedback. Note that each of the fault direction plots illustrates nine steady-state operating conditions.

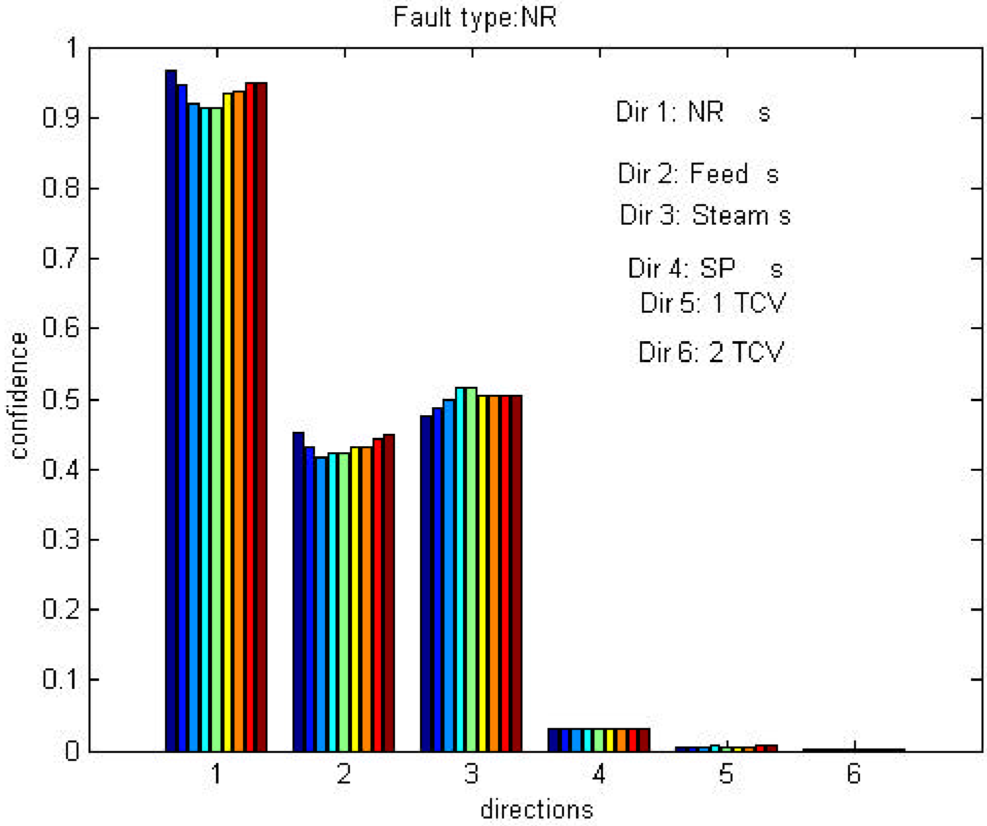
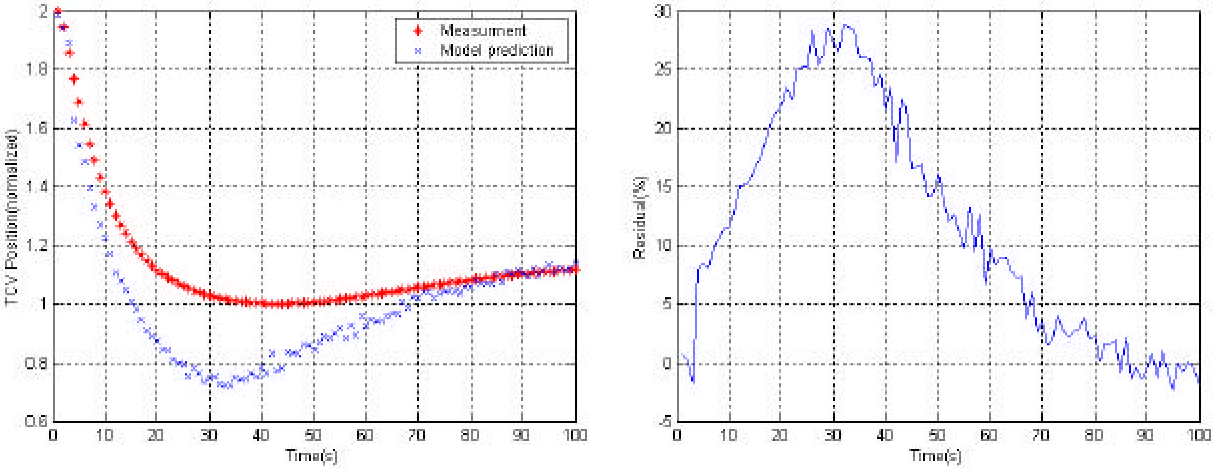


Figure 1. Narrow range SG water level sensor bias fault for the case of steady-state plant operation. The residual directional features are plotted for six different measurements at nine operating levels. The confidence level for the NR fault has the largest value.

An example of tracking turbine control valve fault during transient power operation is shown in Figure 2. A change in the actuator time constant has been simulated. Figure 2a is a plot of the measured and model-predicted values of the TCV position. The residual between the two variables is plotted in Figure 2b. The application illustrates that the time-dependent *GMDH* model is able to track the valve error during the transient operation.

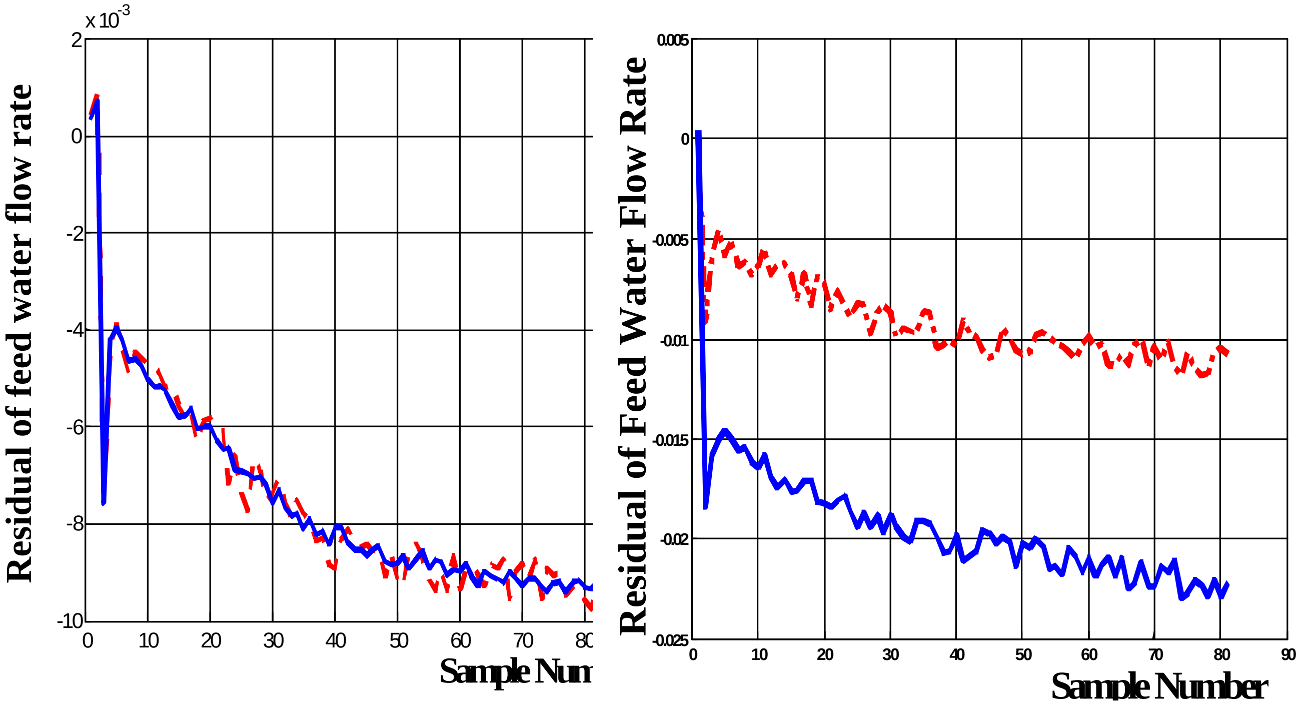


(a) **(b)**

Figure 2. A comparison of the measured and predicted values of the TCV position is made in (a).This is the case of fault detection during a plant transient. The residual plot is shown in (b).

The last example illustrates the application of *ANFIS* modeling and the multi-model digraph for isolating both single and dual faults. The single fault considered is the feed water flow transmitter error. The dual fault considers simultaneous errors in the SG pressure transmitter and the feed water flow transmitter.

Figure 3 shows the plots of feed water flow measurement residual. Among other signals the model uses the SG pressure. The residual magnitude exceeds the acceptable limit, indicating a possible fault in the feed water flow transmitter. The model predictions using the SG pressure and using the reconstructed SG pressure are the same, thus indicating that the SG pressure is not in error. In the case when the SG pressure transmitter has an error, the two prediction residuals do not match, as shown in Figure 3b. This indicates that the SG pressure transmitter is also faulty, in addition to the faulty feed water flow sensor. The model-based directional graph is able to detect both single and dual faults.



(a) (b)

-----after SG pressure is reconstructed

___before SG pressure is reconstructed

Figure 3. Figure 3. Prediction residual of feed water flow transmitter using measured and reconstructed SG pressure for the case of single fault (a) and simultaneous dual faults (b).

ATTACHMENTS

The following attachments are included

Attachment I Task 2 Phase 1 Report

Attachment II Task 2 Phase 2 Report

Attachment III Task 2 Phase 3 Report

Attachment I
Task 2 Phase 1 Report

Task 2. Phase 1
Advanced Monitoring and Diagnostics

SUMMARY

This report describes the tasks performed and the progress made by The University of Tennessee (UTK) during 1999-2000 on the DOE-NERI project entitled ***Automatic Development of Highly Reliable Control Architecture for Future Nuclear Power Plants.*** UTK is collaborating with the Instrumentation & Controls Division of ORNL (lead organization) and the North Carolina State University (NCSU). The objective of the UTK research task is to develop an on-line monitoring system for *fault detection and isolation (FDI)* of sensors and field devices in a nuclear power plant. In this research emphasis is given to process instrumentation in a nuclear power plant such as temperature, pressure, flow, level transmitters, and measurements of control functions. Field devices include valve actuators, control modules, spray and heater systems, pumps, and other similar equipment. The goal of this task is to provide diagnostics information to a system executive for enhanced decision-making by the plant control system.

The following R&D tasks have been accomplished during this reporting period:

- Development of data-driven models for the characterization of sub-system dynamics for predicting state variables, control functions, and expected control actions.
- Development of a nonlinear system modeling approach called the *Group Method of Data Handling (GMDH)* with rational functions.
- Development of the *Principal Component Analysis (PCA)* approach for mapping system measurements.
- Development of a fault detection and isolation module that combines system operational knowledge (including system simulation) and a rule-based logic for both single and dual faults in dissimilar sensor and field devices.
- Development of a complimentary approach that quantifies the prediction errors using a fault pattern classification technique.

The above techniques have been applied to a laboratory process control loop using both simulation and actual loop measurements. The techniques have been demonstrated for detecting and isolating faults in sensors and devices in a U-tube steam generator (UTSG) in a pressurized water reactor (PWR). Only simulation data were used in the latter case.

During the second phase of this project (FY 2000), the FDI system will be implemented for a UTSG system as part of a full-scope PWR plant being developed by NCSU. The current methods will be further developed and extended to fault detection during plant transients. This phase will also include the development of minimum requirements for application to an existing PWR, and the limitations imposed by the measurements. The information generated by the FDI module will be interfaced with the system executive and the control design system. A paper was presented at the American

Nuclear Society Annual Meeting, June 2000, and another paper will be presented at the ANS Topical Meeting on NPIC&HMIT, November 2000.

1. **INTRODUCTION**

1.1 Background and Motivation

Existing and new generation of nuclear power plants have economic and reliability concerns as addressed by overall plant performance, unscheduled downtime and the long-term management of critical assets. The key to achieving these needs is to develop an integrated approach for monitoring, control, fault detection and diagnosis of plant components such as sensors, actuators, control devices and other equipment. Several methods developed by industry and academia, for monitoring isolated sensors and system components were reported [1-8]. Model-based local sensor validation and fault diagnosis methods were developed for specific applications [3,8]. These approaches assume that a system fault being monitored occurs in a specific plant component and in an isolated fashion. Fault detection and isolation (FDI) of sensors and field devices is an important step towards the implementation of an automated and intelligent process control strategy [12].

A large-scale system, such as a nuclear power plant, has several feedback control loops. This makes the identification and isolation of faults in these interconnected systems highly complex. Even when a sensor used for set point control is faulty, the control system through feedback, tries to vary the actuating signals until the error in the set point is eliminated. The *sensor-alone* type validation will fail in this situation. It is therefore necessary to consider fault detection and isolation at the system level rather than at the device level. The objective of this R&D task is to develop an on-line sensor and field device monitoring and fault detection system, when simultaneous faults may occur in two or more of these devices. This goal will be achieved by a two-step approach: (1) Development of data-driven models for predicting multiple variables, using rational function approximation and group method of data handling; (2) A decision-making module that uses system functional knowledge base and pattern classification algorithms, that will be deployed in a distributed configuration. High priority will be given to the computational efficiency of these techniques, with the capability to change the module structure with changing plant conditions. The intrinsic merit of the project lies in the development of an autonomous global monitoring and fault detection approach that would be executed with minimal human interaction.

1.2 Objectives of R&D and Definition of Tasks

The objective of this research task is to develop an on-line monitoring system for fault detection and isolation of sensors and field devices in a nuclear power plant. The sensor suite consists of major process variables in a plant, such as temperature, pressure, flow, level, and control functions. Field devices in a power plant include, but are not limited to, valve actuators, control modules, spray and heater systems, pumps, and similar equipment. The objectives of this R&D are being accomplished through the completion of the following technical tasks:

- Review of literature and previous work.

- Characterization of sub-system dynamics using data-driven models for predicting state variables, control functions, and expected control actions.
- Development of a *Group Method of Data Handling (GMDH)* modeling algorithm with rational function approximation.
- Development of a *Principal Component Analysis (PCA)* algorithm with linear and nonlinear mapping.
- Development of an FDI module that combines system operational knowledge and a rule-based logic for both single and dual faults in dissimilar sensors and field devices.
- Development of a complimentary module that quantifies the prediction error using a fault pattern classification technique.
- Demonstration of the FDI system with application to an experimental process control loop.
- Demonstration of the FDI system with application to a U-tube steam generator (UTSG) in a full-scope simulation model of a 1,300 MWe PWR.
- Development of minimum requirements for FDI system implementation.
- Extension of the techniques for the case of fault detection during plant transients.
- Identification of realistic faults in a PWR and establish the characteristics of transient faults as compared with steady-state faults.
- Interfacing the FDI module with control system module via the system executive and development of a graphical user interface (GUI) for the *FDI* system demonstration.
- Identification of issues in technology transfer to nuclear power industry.
- **Deliverables:** Annual Reports and a Final Report.
  FDI software system and *User's Manual.*
  Conference and journal manuscripts.

1.3 Summary of Significant Accomplishments During 1999-2000

The following major milestones were accomplished during this reporting period:

- Development and testing of the ***GMDH*** modeling module for state and control function prediction.
- Development and testing of the ***PCA*** mapping method for system modeling.
- Development and testing of the ***FDI*** module for both single and dual/simultaneous faults.
- Rule-based decision making.
- Fault pattern clustering approach.
- Demonstration of the *GMDH* method using single and dual faults in a laboratory process control loop.
- Demonstration of the *PCA* approach with application to a PWR steam generator (UTSG) system.
- Preparation of the following manuscripts for publication.

- *Detection and Isolation of Multiple Faults in Nuclear Plant Systems*, ANS Annual Meeting, San Diego, June 2000.

- *Fault Detection and Isolation of Nuclear Power Plant Sensors and Field Devices,* ANS Topical Meeting on NPIC & HMIT, November 2000.

1.4 FDI Architecture and Issues in Developing a Robust FDI Algorithm

Figure 1.1 shows the functional modules of the FDI system being developed in this project. Both *GMDH* and *PCA* modeling of process measurements are considered. This provides a crosschecking of prediction techniques applied to the measurements. Fault isolation is based on either a rule-based algorithm or a pattern classification algorithm. The following issues must be considered in developing a robust FDI algorithm.

- Sensor faults may not be detected in a closed-loop control system.
- Redundancies in sensors and controllers are used in nuclear power plants (NPPs).
- Separation of process variations from sensor/field-device faults must be considered.
- Noise levels in measurements can increase false alarms. It may be necessary to pre-process signals to eliminate this effect at different sub-bands.
- The use of physics models and data-driven models to understand and characterize the process dynamics.

1.4 Organization of the Report

The group method of data handling *(GMDH)* algorithm is described in Section 2 and the principal component analysis *(PCA)* is discussed in Section 3. The application of GMDH to the fault detection and isolation of multiple faults in an experimental process control loop is presented in Sections 4. Section 5 describes the application of PCA and pattern classification approach to a U-tube steam generator in a PWR. Concluding remarks and plans for Phase 2 are given in Section 6.

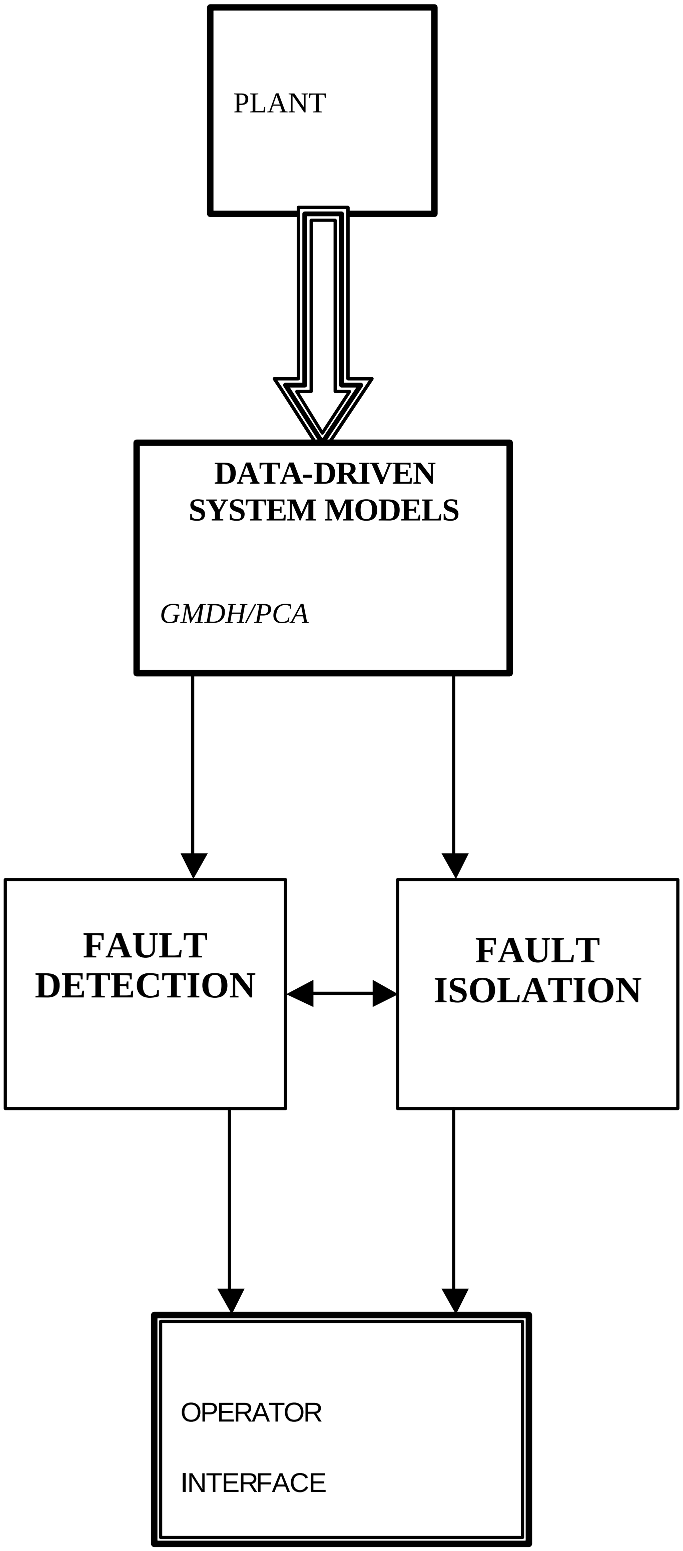


Figure 1.1. Schematic showing FDI system functional modules.

2. GROUP METHOD OF DATA HANDLING (GMDH) APPROACH FOR MEASUREMENT CHARACTERIZATION

2.1 The GMDH Method with Rational Function Approximation

The objective of this sub-task is to characterize the mapping among process variables and control functions using self-organizing and data-driven modeling. The so-called *Group Method of Data Handling (GMDH)* is an algebraic method for predicting system states, controller and actuator functions. A new algorithm, that will create appropriate prediction models for different nuclear plant sub-systems, will be developed by a rational function approximation of the original GMDH algorithm [11,12]. The GMDH approach has the advantage over artificial neural networks in not requiring tedious network training procedures. It is also easy to update the prediction models during plant operation.

The GMDH constructs a model, of a desired output as a function of a set of related inputs from a subsystem, by a successive polynomial approximation. The general relationship has the form shown in Equation (2.1) where $\{x_1, x_2, \ldots , x_m\}$ is a vector of input variables and y is the variable to be predicted. This formulation can be extended to the prediction of multiple outputs $\{y_1, y_2, \ldots , y_n\}$ as well.

$$y = a + \sum_{i=1}^{m} b_i x_i + \sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij} x_i x_j + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m} d_{ijk} x_i x_j x_k + \cdots \qquad (2.1)$$

A typical node of a GMDH modeling layer is a basic quadratic predictor using variables $[x_i, x_j]$. The model parameters such as {A, B, C, D, E, F}, are estimated from a least-squares fit using **N** observations of the input and output variables.

$$y = A + Bx_i + Cx_j + Dx_i^2 + Ex_j^2 + Fx_i x_j \qquad (2.2)$$

Figure 2.1 illustrates that the predicted values of *y* are propagated to successively higher layers of the algorithm, with the approximation of y_{pred} improving at successive stages. At each stage of the approximation, y_{pred} is formed from pairs of input signals (to that layer), and new values of the predicted variable are propagated pair-wise to the next layer. The iteration is continued until the mean-squared error between the predicted and the measured values of the output variable attains a desired value.

Parsimony in model fitting is achieved by comparing the fractional prediction errors from one generation to the next, and by terminating the algorithm when the error is a minimum or when the difference between errors from successive approximation stages is less than a preset limit [12].

The GMDH approach described above uses polynomial approximation. This polynomial set may be satisfactory in establishing some of the relationships of interest. In characterizing the subsystems in a nuclear power plant it may be necessary to use terms containing rational functions (for example, ratios of polynomials in x_1 and x_2). The expression (2.3) represents a set of such terms that forms a *complete* set of terms in a given domain.

$$\left\{\begin{array}{l} 1,(x_1,x_2),(x_1^2,x_2^2),(x_1x_2),(\frac{1}{x_1},\frac{1}{x_2}),(\frac{1}{x_1^2},\frac{1}{x_2^2}),(\frac{1}{x_1+x_2},\frac{1}{x_1x_2}),(\frac{x_1}{x_2},\frac{x_2}{x_1}), \\ (\frac{x_1}{x_1+x_2},\frac{x_2}{x_1+x_2}),(\frac{x_1+x_2}{x_1},\frac{x_1+x_2}{x_2}),\ldots \end{array}\right\} \tag{2.3}$$

The new set should facilitate the development of prediction models with a minimum number of terms. The computational efficiency of establishing these models will be enhanced by a systematic choice of the terms in the set shown in Expression (2.3).

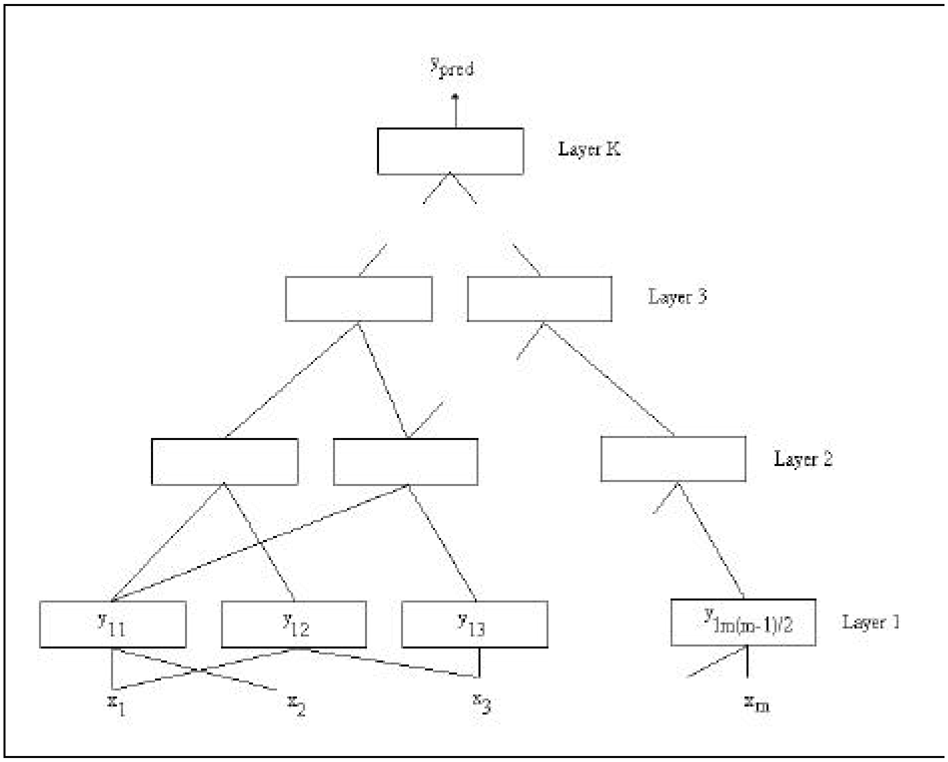


Figure 2.1. GMDH network showing m inputs and K layers.

2.2 The GMDH Algorithm

The following steps explain the procedure used in developing data-driven models using the group method of data handling.

- Consider *N* observations of *m* variables X ≡ {x_1, x_2, … , x_m} and the measurements of the variable to be estimated, Y ≡ {y_1, y_2, … , y_N}.

- Divide the data into a training set (*nt*) and a test set (*N-nt*).

- For each pair {x_i, x_j} and Y, compute the regression polynomial

  $$y = A + Bx_i + Cx_j + Dx_i^2 + Ex_j^2 + Fx_i x_j$$

  A total of m(m-1)/2 polynomials are computed.

- Create new observations, Z, for each of the new m(m-1)/2 variables.
- Screening out the least effective variables: Compute the SSE

$$r_j^2 = \frac{\sum_{i=1}^{nt} (y_i - z_{ij})^2}{\sum_{i=1}^{nt} y_i^2} \qquad (2.4)$$

- Pick those new *inputs* for which $r_j < R$ (choice of the user).

- Repeat the stage-wise computation until the method starts over-fitting the data. Plot the smallest of {r_j} at each stage and look for a *minimum*. This is called the minimum *Ivakhnenko* polynomial.

- Using the best-fit model, compute the prediction errors using the test data of length (*N-nt*). Check if the error r_{best} is satisfactory.

2.3 Enhancement of the GMDH Algorithm

To improve model building with a minimum number of layers, the set of terms in the regression model is generalized to include rational functions of {x_1, x_2, … , x_m}.

- **The choice of terms in the regression is made according to a binary selector: For example, for k=8, the binary number is between 0 and 255 (a total 256 input vectors).**

- *Example:* **model number 179 has the terms [1 0 1 1 0 0 1 1]**
- **Choose ~ ten best-fit models. From this set, choose the model with the least number of terms!**

- **To avoid unlimited increase in the number of nodes in a higher GMDH layer, use the best m nodes for the succeeding layer.** ***All layers have the same number of nodes, m.***

- **Make sure that the number of input variables in the first layer is m > 2, in order to avoid the termination of GMDH after the first (input) layer.**

- **To avoid long training times, limit the maximum number of layers for a single model (30 was suggested in this study, since no improvement was observed beyond this level).**

2.4 Application of the GMDH Algorithm in an FDI System

The choice of the measurement set {x_1, x_2 , … , x_m}, for each predictor *y*, is determined from the knowledge of the system, simulation studies, and parametric analysis such as pair-wise correlations.

- **Generate the prediction models using the fault-free data.**

- **Computes the residual errors for all the state and control functions of interest.**

- **When the error exceeds a pre-set threshold, a fault is detected.**

- **Isolate single/multiple faults.**

3. PRINCIPAL COMPONENT ANALYSIS (PCA) FOR MEASUREMENT CHARACTERIZATION

3.1 Introduction to Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a data characterization method that extracts the directions of maximum variability in a data matrix X (also the matrix of measurements of process signals). PCA is similar to fitting a hyper-plane in the measurement space for the normal operation data, and uses a matrix decomposition method. In case of redundancy in X, the first few principal components (PCs) may be sufficient to explain most of the variability in the data. The data may then be represented as the projection on to the sub-space of the retained PCs with minimal loss of information. The squared sum of errors (SSE) the perpendicular distance of the test data from the PC hyper-plane, and should be small for normal operation (see Figure 3.1). Fault detection is performed by evaluating the SSE (or residuals) after projecting the test data on to the PC hyper-plane. A large value of SSE indicates a possible fault in the system.

PCA uses a fundamental result of linear algebra, called the *Singular Value Decomposition (SVD)*. The following references are suggested [13, 16-23, 25-33, 39].

Singular Value Decomposition (SVD)

SVD: Every *(N* x *m)* matrix *A* can be decomposed into $A = U\,S\,V^T$, where *U* and *V* are orthogonal matrices, and *S* is diagonal.

$$\boldsymbol{A = U}\ S\ \boldsymbol{V^T} = [\mathbf{u_1} .. \mathbf{u_r} .. \mathbf{u_N}].\ \mathbf{Diag}[s_{\mathbf{1}} ..\ s_{\mathbf{r}}].\ [\mathbf{v_1} .. \mathbf{v_r} .. \mathbf{v_m}]^{\mathbf{T}} \tag{3.1}$$

- *U(NxN)* and *V(mxm)* are orthogonal matrices: $U^TU = V^TV = I$.
- The matrix *S* has the *singular values* $\sigma_1, \ldots, \sigma_r$ on its diagonal and zero elsewhere. The dimension r < N and r < m.

Remark:

The singular values $\{\sigma_i\}$ are not eigenvalues of *A*. But $\{\sigma_i^2\}$ are eigenvalues of A^TA.

Definitions:

A (Nxm) is a rectangular matrix. Its row space (each row has *m* elements) is r-dimensional (inside R^m) and its column space (each column has *N* elements) is r-dimensional (inside R^N).

- Required to choose orthonormal bases: Row space basis: $[v_1, \dots , v_r]$ and column space basis: $[u_1, \dots , u_r]$.
  We want orthonormal bases that also diagonalize *A*.

- For a *(2x2)* matrix *A*

$$A[v_1\ v_2] = [\sigma_1 u_1\ \sigma_2 u_2] = [u_1\ u_2] \begin{bmatrix} \sigma_1 & \\ & \sigma_2 \end{bmatrix}$$

$AV = U\Sigma$ and $U^TU = V^TV = I$

- The singular value decomposition (SVD) of *A* is given by

$A = USV^{-1} = USV^T$, S is diagonal (*)

- From (*)

$$A^TA = (U\Sigma V^T)^T(U\Sigma V^T) = V\Sigma^TU^TU\Sigma V^T$$

$$\Longrightarrow \qquad A^TA = VS^TSV^T$$

- For the symmetric matrix A^TA, the columns of *V* are its eigenvectors corresponding to its eigenvalues $\{\sigma_1^2, \dots , \sigma_r^2\}$. This indicates how to calculate the matrix *V*.

- Once $\{v_i\}$ are known, the $\{u_i\}$ are calculated from the equations

$$Av_i = \sigma_i u_i,\ i = 1, 2, \dots , r$$

- ***Remark:***
- The vectors $\{u_i\}$ can be calculated directly from AA^T.

- $AA^T = (USV^T)(VS^TU^T) = USS^TU^T$.

- The columns of *U* are the eigenvectors of AA^T (and correspond to the same eigenvalues as those of A^TA).

- ***Example:***

$$A = \begin{pmatrix} 2 & 2 \\ -1 & 1 \end{pmatrix} \qquad A^TA = \begin{pmatrix} 5 & 3 \\ 3 & 5 \end{pmatrix}$$

- Eigenvalues of A^TA are $\sigma_1^2 = 2$, $\sigma_2^2 = 8$

- $v_1 = [-1/2 \quad 1/\sqrt{2}\,]^t$ and $v_2 = [1/\sqrt{2} \quad 1/\sqrt{2}\,]^t$

- By computing the normalized values of Av_1 and Av_2, we get

- $u_1 = [0 \;\; 1]$ and $u_2 = [\,1 \;\; 0]$

- Now verify the results

$$A = USV^T = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \sqrt{2} & 0 \\ 0 & 2\sqrt{2} \end{pmatrix} \begin{pmatrix} -1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix}$$

Remarks:
The matrices *U* and *V* contain orthonormal basis for all four fundamental subspaces:

- First *r* columns of *V*: row space of *A*
- Last *m-r* columns of *V*: null space of *A*
- First *r* columns of *U*: column space of *A*
- Last *N-r* columns of *U*: null space of A^T

3.3 Principal Component Analysis of Process Data

Consider a data matrix *X (N x m)* with *m* variables and *N* independent measurements.

$$\boldsymbol{X} = \begin{array}{cc} \begin{pmatrix} x_{11} & \dots & x_{1m} \\ x_{21} & \dots & x_{2m} \\ & & \\ x_{N1} & \dots & x_{Nm} \end{pmatrix} & N \updownarrow \\ \xleftrightarrow{\quad m \quad} & \end{array} \tag{3.2}$$

Decompose *X* into a product of scores (T) and loadings (P) as

$$X = TP^T + E \tag{3.3}$$

Where E represents the residuals (error) after projection on to the principal axes or the hyper-plane. The PCs are ordered such that the successive PCs explain the contribution to *X* in descending order of the lengths of principal axes of the hyperellipsoid of the data space.

Now consider the SVD of the data matrix *X*:

$$\boldsymbol{X} = \mathbf{U}\mathrm{S}\,\mathbf{V}^{\mathbf{T}} \tag{3.4}$$

- **Where**
  U is an orthogonal matrix *(NxN)* spanning the column space of *X*.
  V is an orthogonal matrix spanning the row space of *X*.
  S **is a diagonal matrix of singular values of *X* in decreasing magnitude.**

Interpretation of the PCA of Process Data

Consider a column vector b *(mx1)* in the row space of *X* so that

$$\boldsymbol{X}\mathbf{b} = \mathbf{U}\mathrm{S}\,(\mathbf{V}^{\mathbf{T}}\mathbf{b})$$

The term in the parentheses represents a rotation of the reference from unit circle to V.

- **Multiplication by** S **corresponds to a scaling of vector b in the V frame by the corresponding singular values and transforms b to the column space of *X*.**

- **The final vector is in the U-frame and multiplication by U transforms to the unit hyper sphere in $\mathbf{R}^{\mathbf{N}}$.**

- **The columns of V are the principal components or directions and the singular values are the lengths of the principal axes of the hyper-ellipsoid.**

- **The scores T are the projections on to the PCs and are obtained as**

$$\mathbf{T} = \boldsymbol{X}\mathbf{V} = \mathbf{U}\mathrm{S}\,\mathbf{V}^{\mathbf{T}}\mathbf{V} = \mathbf{U}\mathrm{S}$$

- **The scores are decorrelated**

$$\mathbf{T}^{\mathbf{T}}\mathbf{T} = (\mathbf{U}\mathrm{S}\,)^{\mathbf{T}}\,\mathbf{U}\mathrm{S}\ = \mathrm{S}\,^{\mathbf{T}}\mathbf{U}^{\mathbf{T}}\mathbf{U}\mathrm{S}\ = \mathrm{S}\,^{\mathbf{T}}\mathrm{S}$$

- **PCA thus represents a rotation of the $\mathbf{I}_{\mathbf{mxm}}$ reference frame to the PC reference frame, so that the data is uncorrelated in the PC frame.**

- **Retaining *p* of a maximum rank (*X*) PCs, the data matrix may be written as**

$$\boldsymbol{X} = \mathbf{U}_{\mathbf{p}}\mathrm{S}_{\mathbf{p}}\mathbf{V}_{\mathbf{p}}^{\mathbf{T}} + \mathbf{E} \tag{3.5}$$

- **For a test sample vector x *(mx1)*, the scores (t) and the errors (e) are given by**

$$\mathbf{t} = \mathbf{x}\mathbf{V}_{\mathbf{p}} \tag{3.6}$$

$$\mathbf{e} = \mathbf{x} - \mathbf{t}\mathbf{V}_p^T = \mathbf{x} - \mathbf{x}\mathbf{V}_p\mathbf{V}_p^T = \mathbf{x}(\mathbf{I}-\mathbf{V}_p\mathbf{V}_p^T) \quad (3.7)$$

- **The fault vector for each case may then be generated from the error vector e.**

The principal component analysis described above performs a linear transformation of the signals. The PCA may be generalized so that the data matrix *X* would consist of nonlinear terms in the measurements. This generalization is somewhat similar to the use of rational functions in GMDH and is described in Section 2.

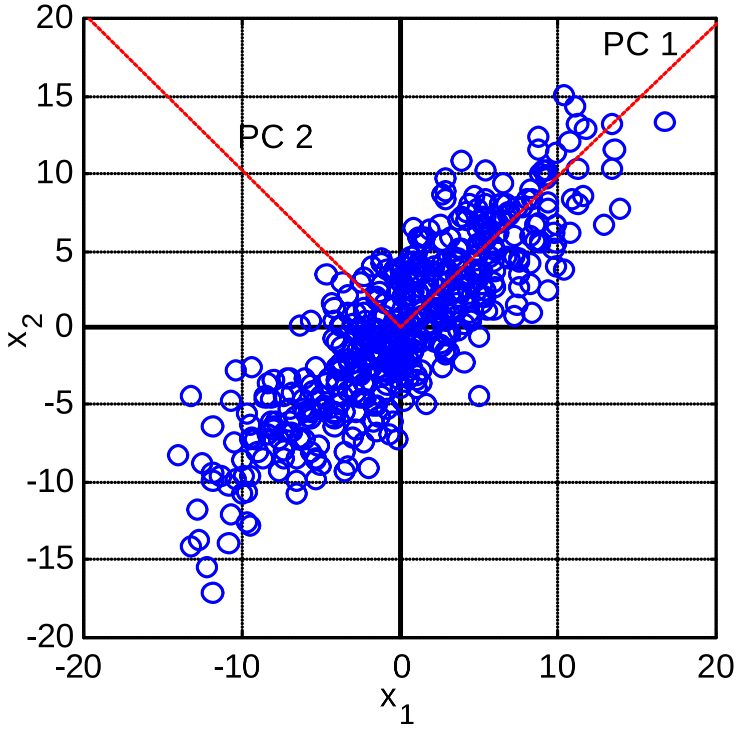


Figure 3.1. Illustration of the principal component analysis (PCA) for a two-dimensional measurement system.

4. APPLICATION OF GMDH AND RULE-BASED APPROACH FOR FAULT DETECTION AND ISOLATION IN A PROCESS CONTROL LOOP

4.1 Introduction

This section describes the results of the application of the GMDH model prediction method and a rule-based approach for fault detection and isolation in a laboratory process control loop. Both single and dual faults were imposed on various devices in this loop. A description of the experimental facility (along with the sensors and devices used) and results of loop response simulation are also presented.

The following are the steps in implementing the FDI algorithm:

1. Generation of a fault-free database. Various system operational conditions must be considered here.
2. Determination of a qualitative relationship among different loop components through linear correlation analysis.
3. Determination of quantitative relationships among different loop components through the GMDH technique.
4. Development of a rule-based decision module for fault detection and isolation. This is accomplished by simulating and characterizing a defined fault in each loop component.

4.2 GMDH Models and Rational Function Approximation for State and Control Function Prediction

The *Group Method of Data Handling (GMDH)* is an algebraic method for predicting system states, controller and actuator functions. This is described in Section 2. The GMDH constructs a model, of a desired output as a function of a set of related inputs from a subsystem, by a successive polynomial approximation. The general relationship has the form shown in Equation (4.1) where $\{x_1, x_2,\ldots,x_m\}$ is a vector of input variables and y is the variable to be predicted. This formulation can be extended to the prediction of multiple outputs $\{y_1, y_2, \ldots , y_n\}$.

$$y = a + \sum_{i=1}^{m} b_i x_i + \sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij} x_i x_j + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m} d_{ijk} x_i x_j x_k + \cdots \qquad (4.1)$$

Figure 4.1 shows a typical node of a GMDH modeling layer with the basic quadratic predictor. The model parameters such as {A, B, C, D, E, F}, are estimated from a least-squares fit using **N** observations of the input and output variables.

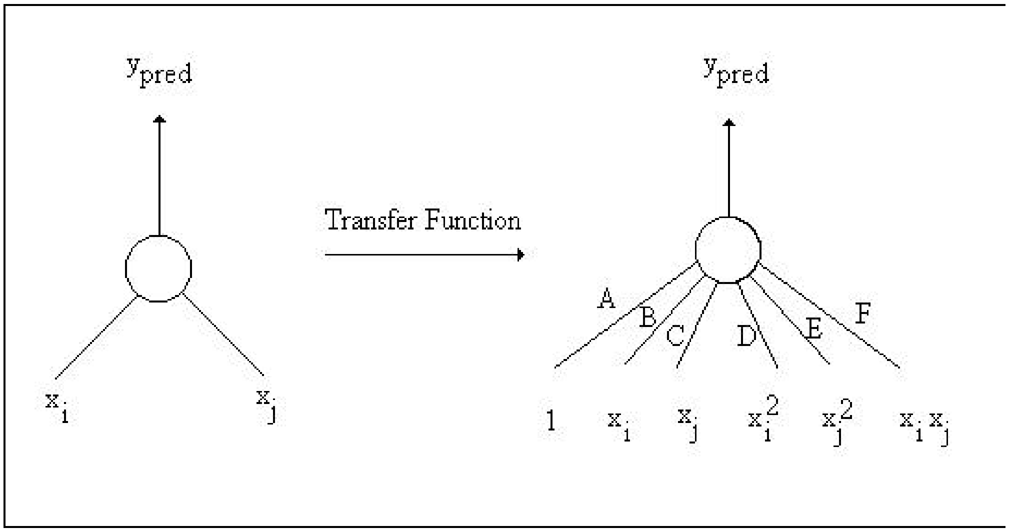


Figure 4.1. A node of the GMDH model predictor. This node uses a second order polynomial transfer function.

In application to nuclear plant subsystems, a systematic study has to be performed in establishing models that are valid for a range of operating conditions. The level of complexity of the fault detection and identification algorithm depends on the importance of the equipment or the asset being considered, the ease of real-time monitoring and communication, and the multiplicity of devices.

4.3 Development of a Mathematical Model of the Laboratory Process Control System

Theoretical and experimental studies were performed for feasibility studies of the Fault Detection and Isolation (FDI) method proposed in this work. For the theoretical study, a simulation model of a process control loop, including sensors, controllers, and actuators, was developed. This model was implemented in the Matlab-Simulink^{TM} programming environment. For experimental studies a low-pressure water loop (LPWL) system was designed and built in the Nuclear Engineering Department [12]. A LabView program was developed to acquire loop measurements and to control the experiment. Known faults were imposed on different devices, such as pressure transmitters, motor-operated valves, and control elements. The purpose of the model and the test system was to provide useful data and an environment for developing and testing the proposed FDI algorithm. Data from all available sensors for normal loop operation were used to build a database.

The fault detection and isolation system is first tested using a Simulink™ model of a process control loop. The control loop is shown in Figure 4.2 and consists of the following major components: orifice flow meter (RMT flow meter), water level sensor (pressure transmitter), turbine flow meters (two), three motor-operated valves (MOVs) with valve position signals, main circulating pump, and a software-driven proportional-integral controller for the tank water level. Figure 4.3 shows the main screen of the Simulink™ model. A list of all system variables available is given in Table 4.1.

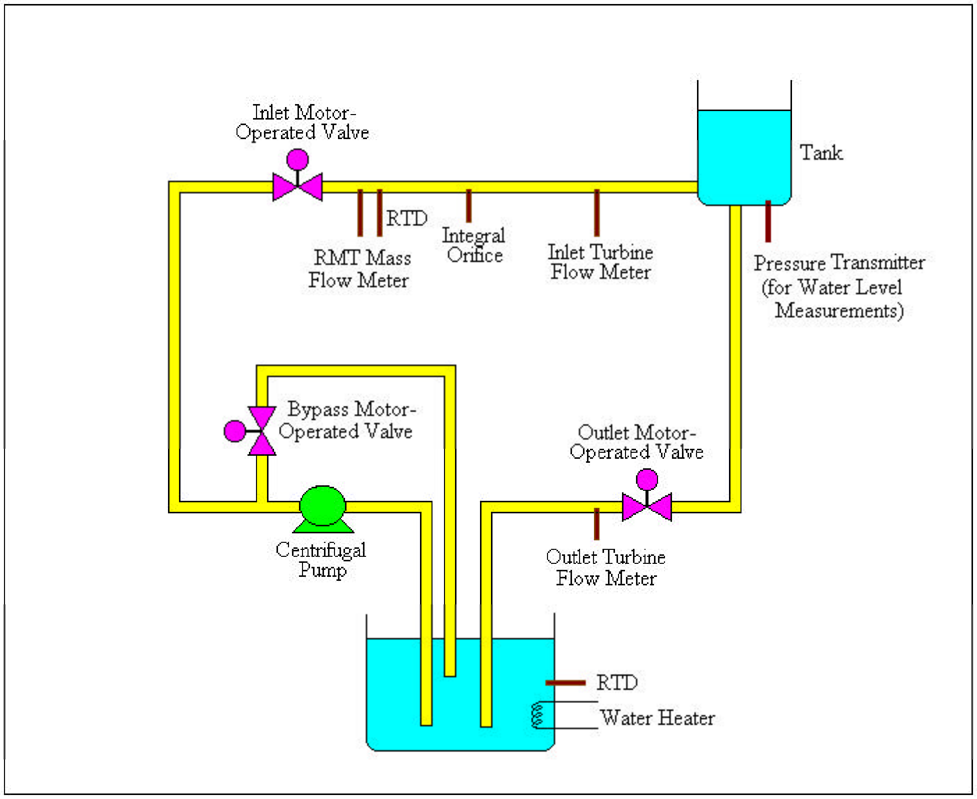


Figure 4.2. A schematic of the low-pressure water loop system, showing the various sensors and field devices.

Simulation models used for the control loop include (1) mass balance of water in the tank, P-I controller model, and first order sensor models. An example of steady state process representation is shown for the pump model with the following parameters.

- IFR = Inlet Flow Rate through tank inlet piping.
- BFR = Bypass Flow Rate through bypass valve
- IMOVP = Inlet MOV Position.
- BMOVP = Bypass MOV Position
- PD = Pump Discharge.

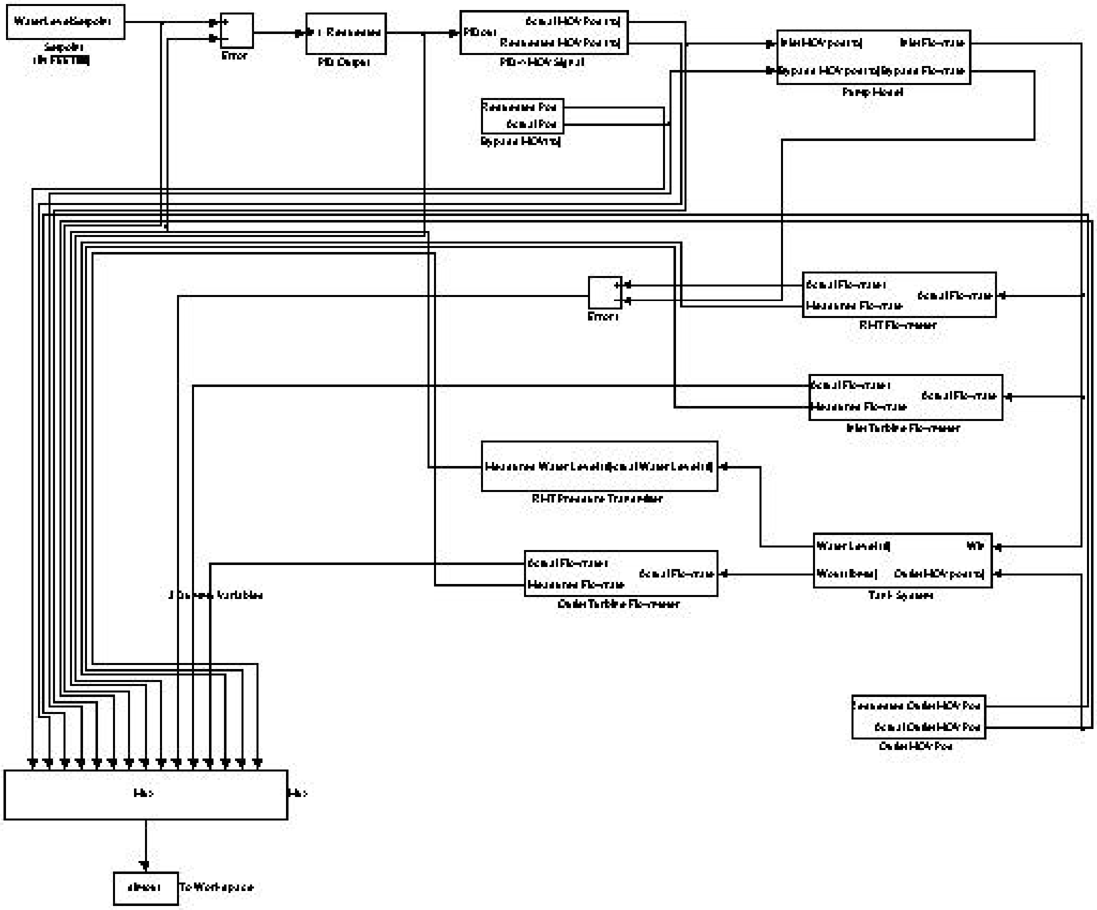


Figure 4.3. Physical model of the low-pressure process control loop represented by Matlab-Simulink [12].

Steady-State Pump Model:

$$
\begin{aligned}
IFR &= \frac{IMOVP}{IMOVP + BMOVP} * PD \\
BFR &= \frac{BMOVP}{IMOVP + BMOVP} * PD
\end{aligned}
\tag{4.2}
$$

Table 4.1. System variables for the process control loop

	Variable (measurement)
1	Bypass MOV position Setpoint
2	Inlet MOV position Setpoint
3	Outlet MOV position Setpoint
4	Measured Bypass MOV position
5	Measured Inlet MOV position
6	Measured Outlet MOV position
7	Water Level Setpoint
8	Measured Water Level
9	PID output
10	Tank Water Temperature set point
11	Tank Water Temperature
12	Heater Element PID Controller Action
13	Measured RMT Inlet Flow rate
14	Measured Turbine Inlet Flow rate
15	Measured Turbine Outlet Flow rate

GMDH prediction models were developed directly from the measurements for tank inlet flow rate, tank outlet flow rate, tank water level, and level controller signal using the following functional relationships.

- *Inlet Flow Rate* = f (Bypass MOV position, Inlet MOV position)
- *Outlet Flow Rate* = f (Tank water level, Outlet MOV position)
- *Tank Water Level* = f (Inlet flow rate, Outlet MOV position)
- *Controller Output* = f (Bypass MOV position, Tank water level, Outlet flow rate)

(4.3)

4.4 Types of (Device) Faults Studied in this Research

Many types of faults can occur in a process control loop such as sensor faults, actuator faults, controller faults, pump failure, leaks in piping, etc. This study limits itself to those faults that can lead to significant error in the GMDH prediction models. Faults are introduced in one or more devices during the experiments through the computer interface. The following is a list of single faults (7) and dual faults (21) that are simulated using the model.

Single Faults:

Water level sensor drift (sensor fault).
Outlet turbine flow meter drift (sensor fault).
Outlet MOV positioning device drift (actuator fault).
Bypass MOV position (actuator fault).
Inlet MOV positioning device drift (actuator fault).
RMT flow meter drift (sensor fault).
Water Level Controller (controller fault).

Multiple Faults (Dual Faults):

Inlet MOV Fault and Water Level Sensor Fault
Inlet MOV Fault and RMT Flow meter Fault
Inlet MOV Fault and Outlet MOV position
Inlet MOV Fault and Bypass MOV Position
Inlet MOV Fault and PID controller Fault
Water Level Sensor Fault and RMT Flow meter Fault
Water Level Sensor Fault and Outlet MOV position
Water Level Sensor Fault and Bypass MOV Position
Water Level Sensor Fault and Outlet Turbine Flow meter
RMT Flow meter Fault and Outlet MOV position
RMT Flow meter Fault and Bypass MOV Position
RMT Flow meter Fault and Outlet Turbine Flow meter
Outlet MOV position Fault and Bypass MOV Position
Outlet MOV position Fault and Outlet Turbine Flow meter
Bypass MOV Position Fault and Outlet Turbine Flow meter
PI controller Fault and Inlet MOV Fault
PI controller Fault and Water Level Sensor Fault
PI controller Fault and RMT Flow meter Fault
PI controller Fault and Outlet MOV position
PI controller Fault and Bypass MOV Position
PI controller Fault and Outlet Turbine Flow meter.

4.5 Implementation of the FDI Algorithm

The basic steps for developing an FDI algorithm are:

- Generation of the Fault-Free Database
- Generating Qualitative Relationships among Loop Components
- Generating Quantitative Relationships among Loop Components
- Development of a Rule-based Decision Module for Fault Detection and Isolation.

4.5.1. Generation of the Fault-Free Database

A fault-free database was generated using the theoretical model for different device configurations. Bypass and Outlet MOV positions were systematically changed one at a time. Water level set point was also changed. The Inlet MOV being a part of the water level control system, its position cannot be set manually; its position is set directly by the water level controller. No faulty devices were allowed in this phase. About 1,235 cases were simulated and the data generated were stored in a database. This database is used to obtain both qualitative and quantitative relationships among the loop devices.

4.5.2. Generating Qualitative Relationships Among Loop Components

To obtain a qualitative relationship among the loop components, the correlation coefficient method was applied. From this analysis, sets of related variables were defined, although the characterization of these relationships through mathematical expressions is not obtained in this step.

Even though sometimes a large set of variables with high correlation existed, groups of small number of variables were selected for modeling using GMDH. The variables selected in each group were those from the components that are physically close to each other. For example the bypass and inlet MOV positions determine the flow that goes through the inlet piping. One could also use the flow rate that goes through the outlet piping in the above correlation set, however this variable would not bring new information to this relationship. Creating a small number of models with local variables makes the fault detection very efficient, that is, it is easy to isolate a faulty component. This efficiency is reflected in the rule-based expert system. The simpler the expert system, the easier it is to develop and maintain it.

Four relationships were defined for this particular loop system. With this relationships the FDI algorithm is be able to isolate basically all possible faults that may happen in that loop. The relationships are

- Inlet flow rate as a function of bypass and inlet MOV positions.
- Outlet flow rate as a function of tank water level and outlet MOV position.
- Tank water level as a function of inlet flow rate and outlet MOV position.
- Level controller output value as a function of bypass MOV position, tank water level and outlet flow rate.

4.5.3. Generating Quantitative Relationships Among Loop Components

For characterization of these four relationships through mathematical equations, the GMDH method was applied to the same fault-free database. A coefficient matrix representing each model is generated by the GMDH. This matrix is used to obtain the predicted (or analytical) component value. This predicted value is then later used for fault detection. The main task of the GMDH algorithm is to find the model that best maps the input/output set for given basic functions. After finding the best model, a residual value is computed between the predicted value and the target value. This residual is stored in memory (RESIDUAL variable). This variable is used later for comparison purposes among all residuals from other runs. Each run with a different set of basic functions is tagged with a number stored in a variable called JobCounter. This variable is used later to recover the best of the GMDH models. The loop runs 2^n times, where n is the number of basic functions to be tested.

At the end of this process, the variable called RESIDUAL contains the residuals of all possible combinations of basic functions (2^n). Using the Matlab function called SORT ([y, index] = sort(residual)) the residuals are sorted from smallest to the highest values in the variable Y. The INDEX variable contains the JobCounter case numbers associated with the residuals in Y. Normally the first JobCounter (index(1)) should be used because it represents the best set of basic functions for the GMDH model structure.

After establishing the mathematical relationship among a set of loop variables, the model can be used for generating an analytical redundant measurement. This redundant measurement is the prediction of the GMDH model.

The Fault Detection and Isolation algorithm uses the comparison between the predicted and measured values of the process variable under study. If the error (residual) between the predicted and measured values is higher than a pre-defined threshold, the algorithm assumes that a fault exists. In this case, a second module is executed; namely, the fault isolation module. This module is based on a rule-based expert system. This expert system is described in the next section.

As mentioned earlier in this section, the GMDH algorithm uses the fault-free database to obtain the best model that maps the input variables (independent variable) to the output variable (dependent variable). The algorithm splits the input data into two sets of data. The first set is used to find the best model, while the second monitors for model over fitting. Figure 4.4 is a plot of the training output data along with the value predicted by the model for the best set of basic functions. Figure 4.5 is a plot of the test data set with the predicted values. Finally, Figure 4.6 is a plot of the error (in percent) between the predicted and the expected output results of the best GMDH model for predicting the flow rate in the tank inlet piping as a function of Bypass and Inlet MOV positions. The plots indicate that the GMDH maps the input-output variables satisfactorily for both the training data and the test data. Figures 4.7 through 4.15 show similar analysis for the Outlet Flow Rate, PID Controller Output and Tank Water Level.

As an example, the RMS error between the measurement and model prediction for the tank outlet water flow rate is 0.11%.

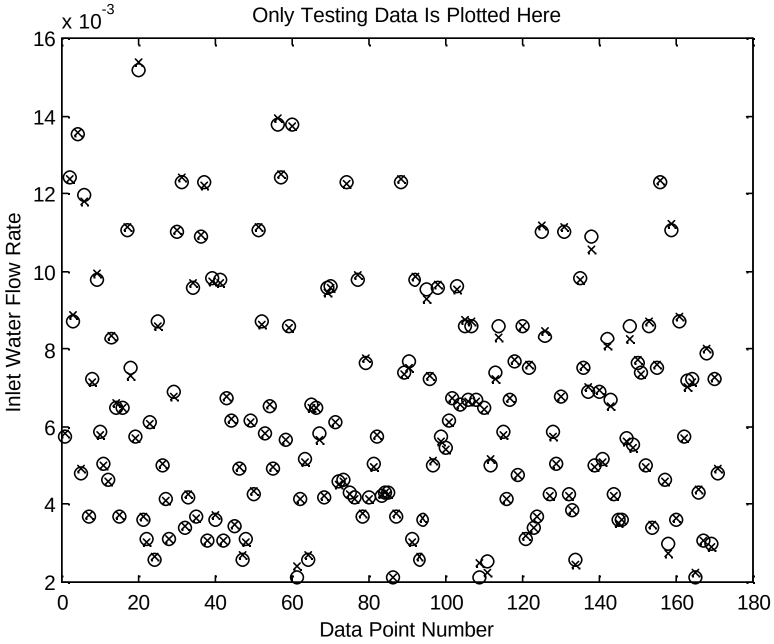


Figure 4.4. The GMDH predicted values ("o") against the test output values ("x") for the Inlet Flow rate as a function of Bypass MOV and Inlet MOV positions

.

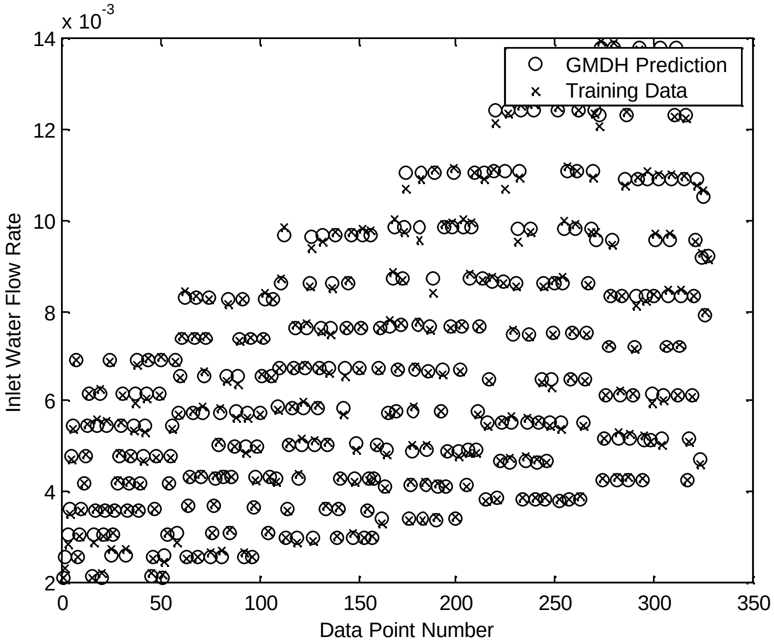


Figure 4.5. The GMDH predicted values ("o") against the training output values ("x") for the Inlet Flow rate as a function of Bypass MOV and Inlet MOV positions.

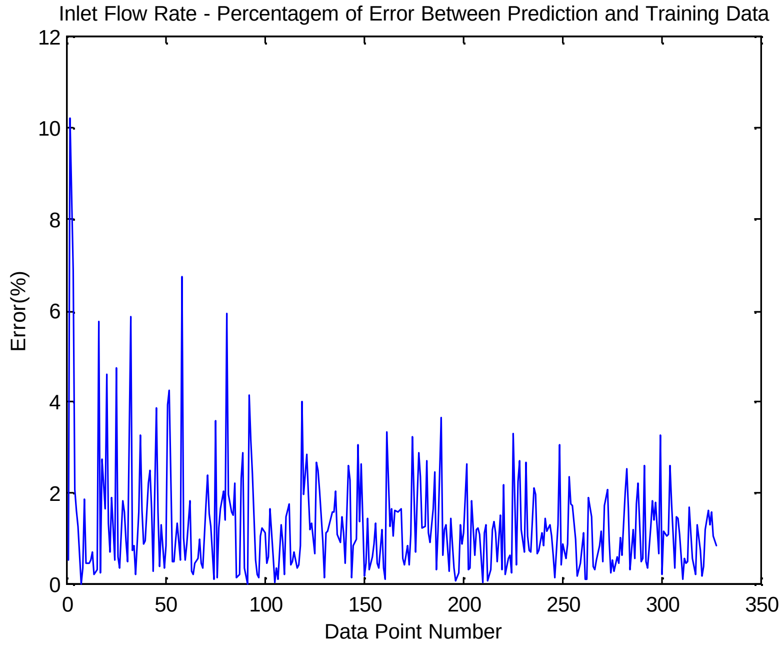


Figure 4.6. The error (%) between the predicted values and the measurements for the Inlet Flow rate as a function of Bypass MOV and Inlet MOV positions.

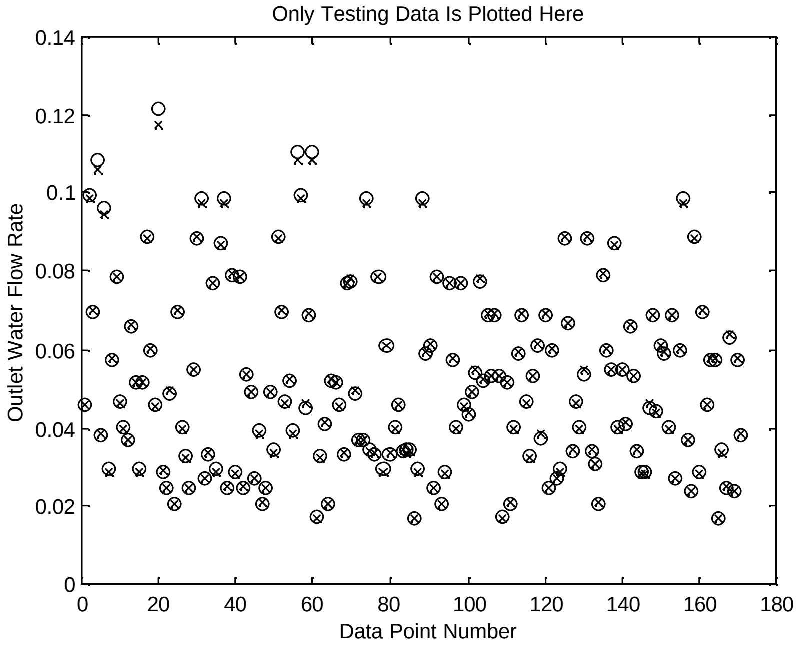


Figure 4.7. The GMDH predicted values ("o") against the test output values ("x") for the Outlet Flow rate as a function of Water Level and Outlet MOV positions.

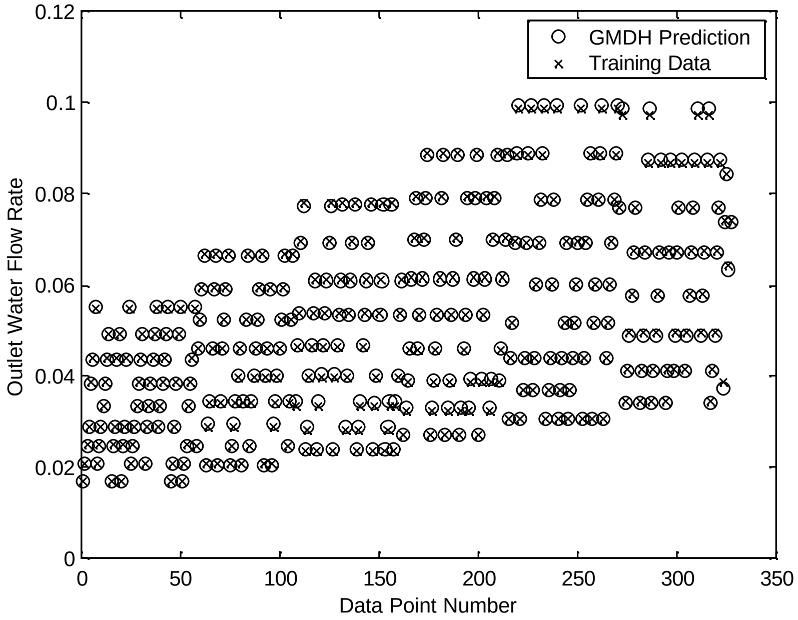


Figure 4.8. The GMDH predicted values ("o") against the training output values ("x") for the Outlet Flow rate as a function of Water Level and Outlet MOV positions.

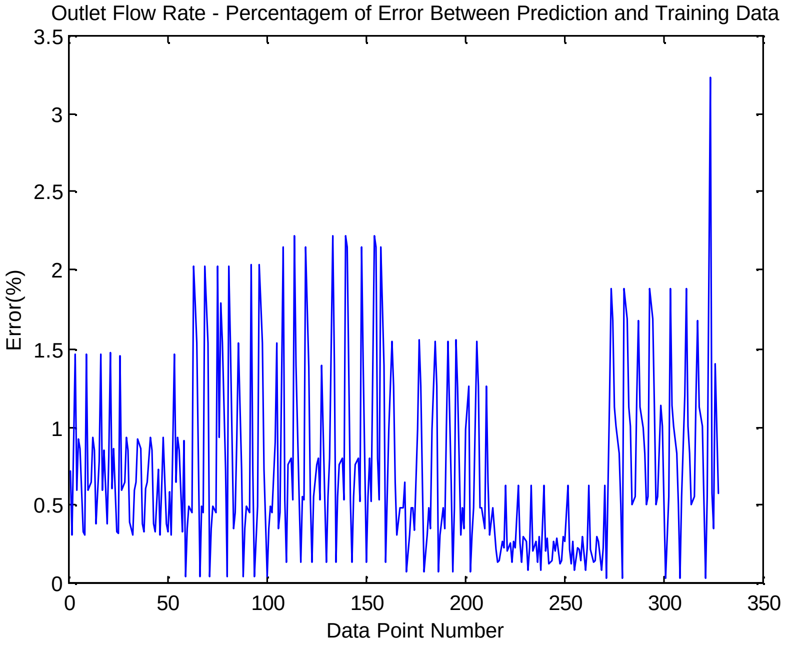


Figure 4.9. The error (%) between the predicted values and the measurements for the Outlet Flow rate as a function of Water Level and Outlet MOV positions.

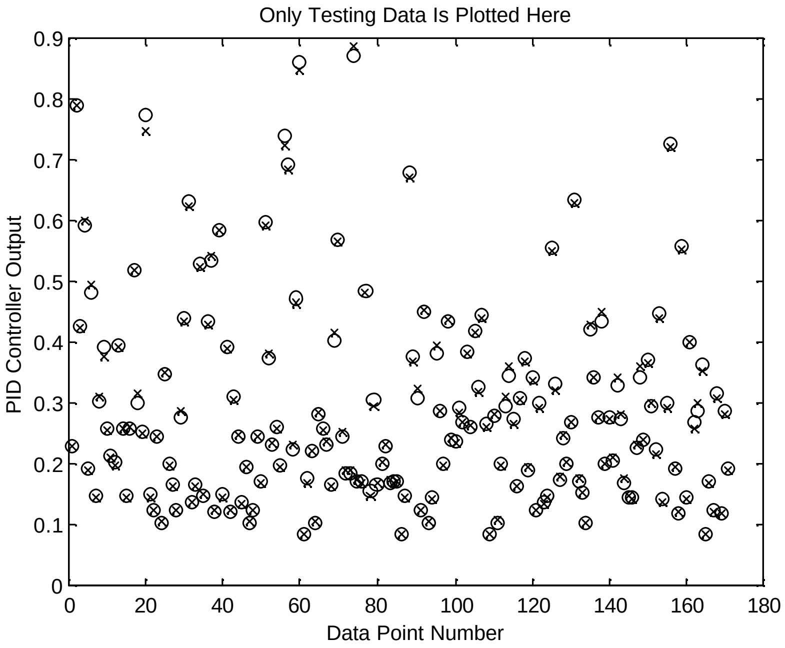


Figure 4.10. The GMDH predicted values ("o") against the test output values (x) for the PID Controller Output as a function of Bypass MOV, Water Level and Outlet flow rate.

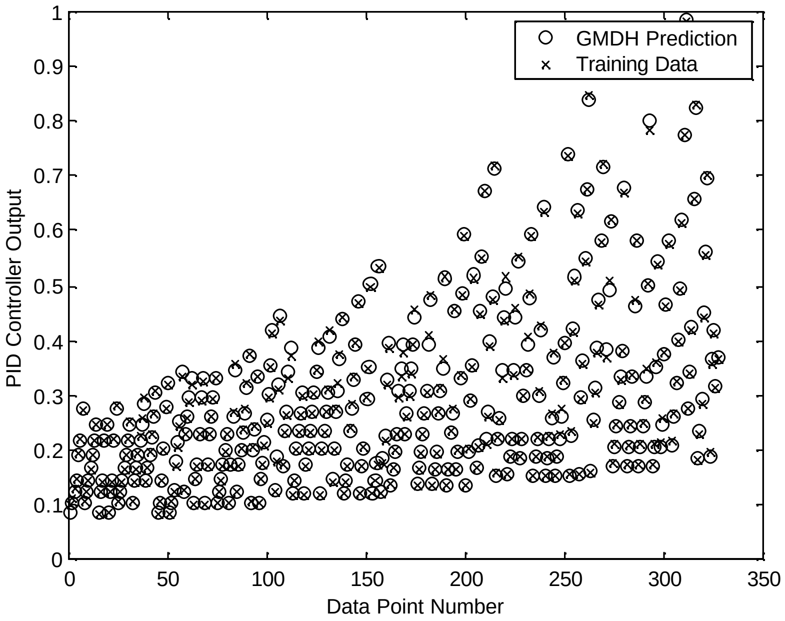


Figure 4.11. The GMDH predicted values ("o") against the training output values ("x") for PID Controller Output as a function of Bypass MOV, Water Level and Outlet flow rate.

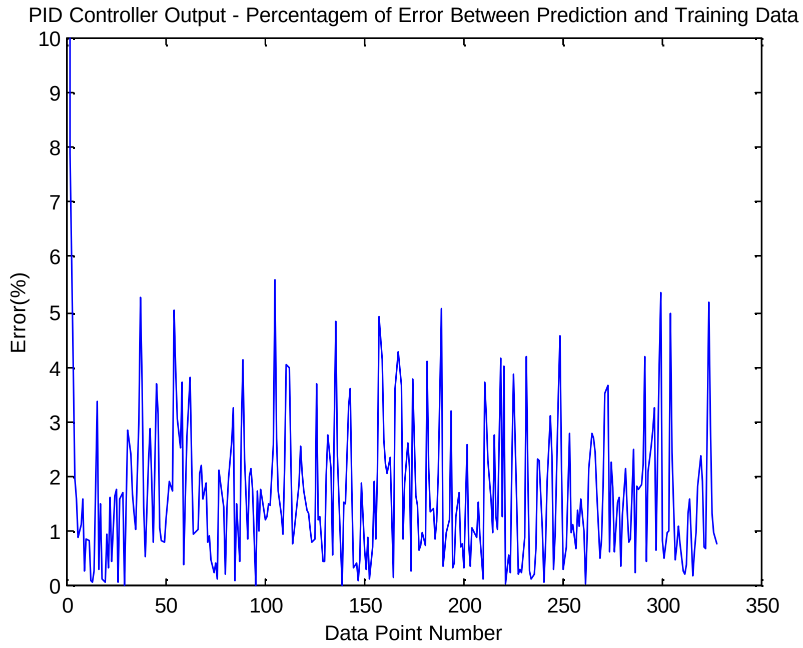


Figure 4.12. The error (%) between the predicted values and the measurements for the level Controller Output as a function of Bypass MOV, Water Level and Outlet flow rate.

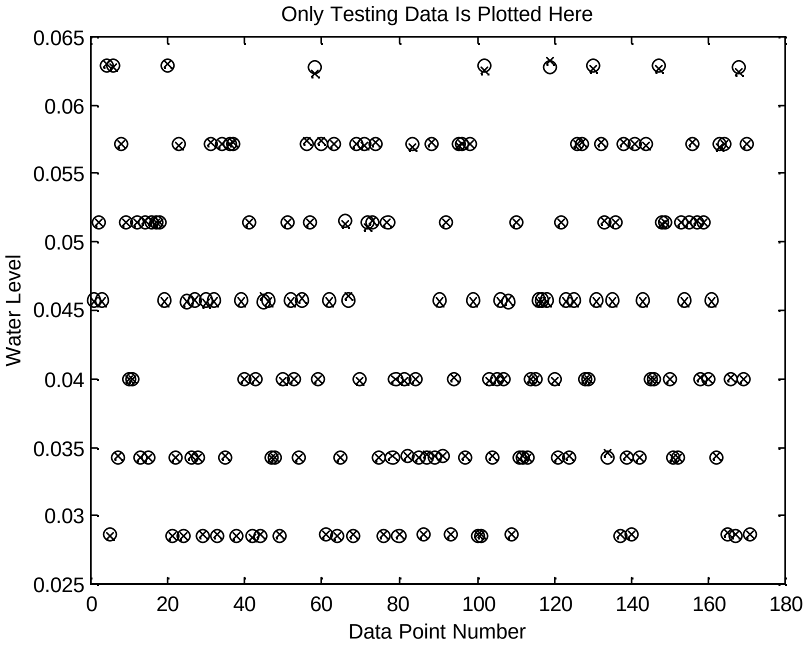


Figure 4.13. The GMDH predicted values ("o") against the test output values (x) for the Tank Water Level as a function of Inlet Flow rate and Outlet MOV positions.

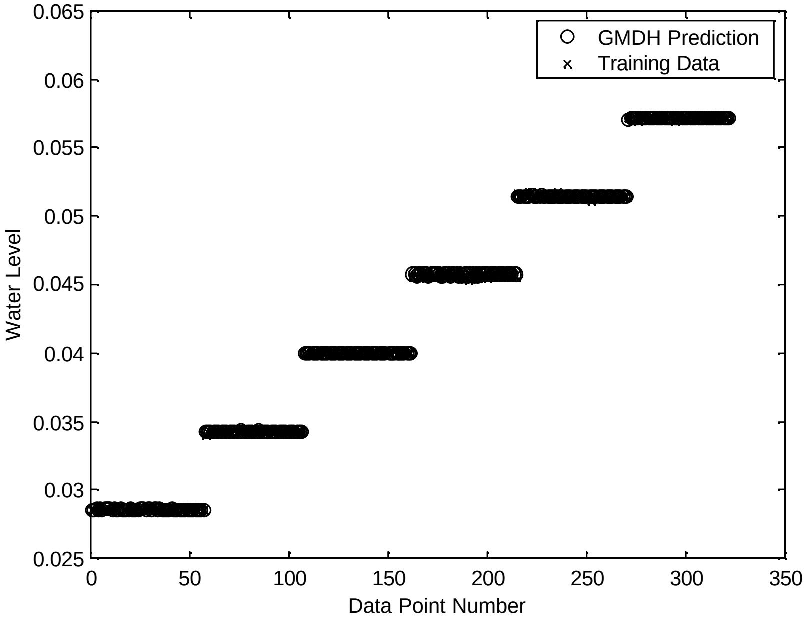


Figure 4.14. The GMDH predicted values ("o") against the training output values ("x") for the Tank Water Level as a function of Inlet Flow rate and Outlet MOV positions.

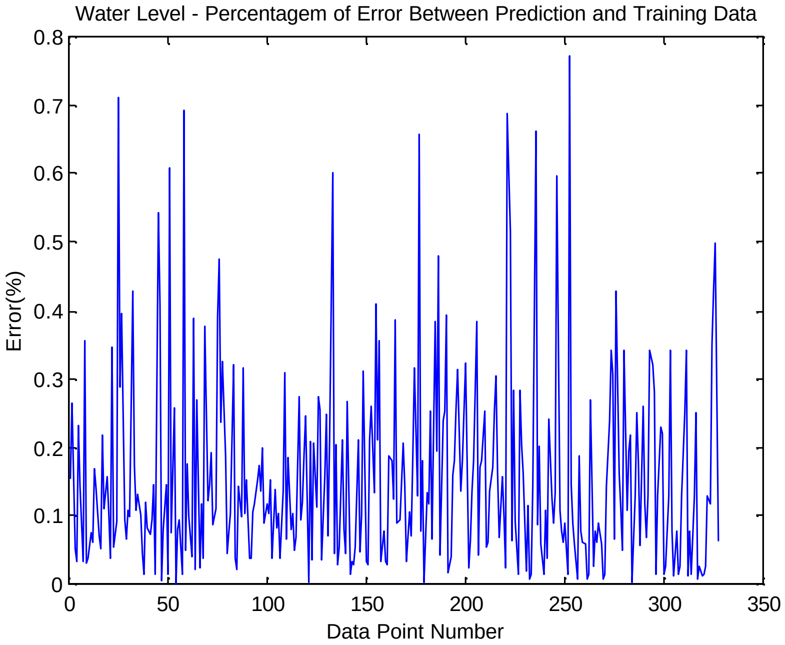


Figure 4.15. The error (%) between the predicted values and the measurements for the Tank Water Level as a function of Inlet Flow rate and Outlet MOV positions.

4.5.4. Development of the Rule-based Decision Module for Fault Isolation

The simulation data are generated for various steady-state conditions of the tank water level. GHDH models are then generated for four variables – inlet flow rate, outlet flow rate, controller output, and water level. The prediction errors (or the residuals) are used to develop a comprehensive rule base. Table 4.2 summarizes the behavior of the prediction errors for each of the faults (both single and dual simultaneous faults). A cell with a '+' sign indicates that the measured value is greater than the predicted value, and conversely for a cell with a '-' sign. A cell with no entry indicates that the predicted value is within 3% of the measured value. This threshold level is set by the user.

During normal system operation, the predicted values obtained by the GMDH methods are very close to those obtained by sensor measurements. However, when a fault occurs in one of the system components, one or more of these analytical redundant values will not match with their measurements.

During the experiments, the faults mentioned in the previous section were simulated. The basic type of fault imposed on the loop devices was a drift type. In all the studied fault cases, the imposed drift was about 10% of the device nominal value. Figure 4.16 shows a typical fault profile imposed on the tank water level sensor output. The profile is of the form *tanh(x)*.

The rule-based expert system uses the system behavior characteristics for each simulated fault. For most of the cases, the system dynamic behavior responds differently for each loop component fault. For these cases, each multidimensional residual vector (vector whose components are made of each individually generated residual) is unique. In cases where there are two or more different faults that produce the same system behavior, other system variables was supplied to the expert system:

- Water level set point error.
- Outlet MOV position error.
- Bypass MOV position error.
- Inlet MOV position error.

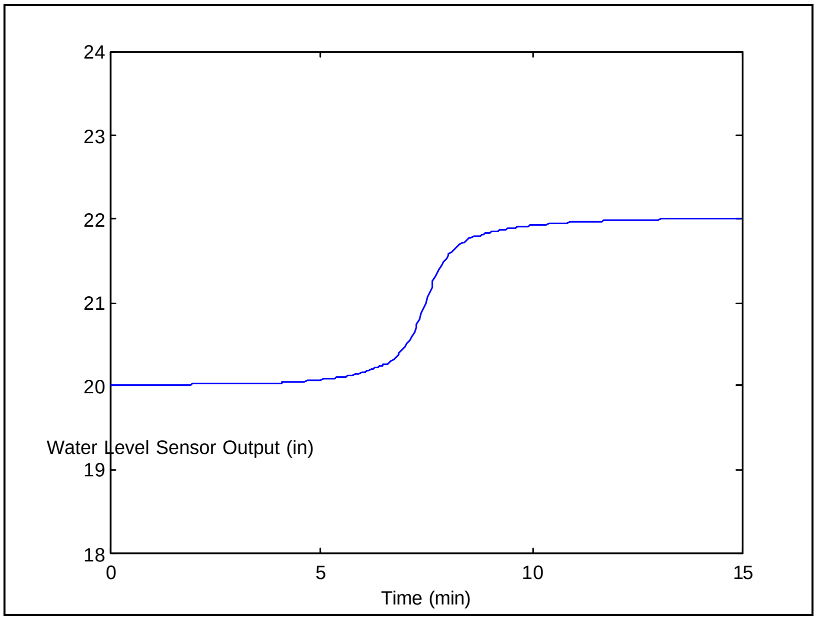


Figure 4.16. Imposed fault profile on the tank water level sensor output.

Table 4.2. Summary of behavior of residuals for various fault cases.

Qualitative error between GMDH prediction values and the measured values for Single and Multiple Fault cases				
FAULTS	GMDH Predictions			
	Inlet Flow rate	Outlet Flow rate	Controller Output	Water Level
Inlet MOV Fault	+		-	
Water Level Sensor Fault		-		+
RMT Flow Meter Fault	+			-
Outlet MOV position		-		+
Bypass MOV position	+		-	
Outlet Turbine Flow meter		-	+	
PID Controller Fault				
Inlet MOV Fault and Water Level Sensor	+	-	-	+
Inlet MOV Fault and RMT Flow meter	+		-	-
Inlet MOV Fault and Outlet MOV position	+	-	-	+
Inlet MOV Fault and Bypass MOV Position				
Inlet MOV Fault and Outlet Turbine Flow meter	+	-	+	
Water Level Sensor and RMT Flow meter	+	-		
Water Level Sensor and Outlet MOV position		-		+
Water Level Sensor and Bypass MOV Position	-	-	+	+
Water Level Sensor and Outlet Turbine Flow meter		-	+	+
RMT Flow meter and Outlet MOV position	+	-		+
RMT Flow meter and Bypass MOV Position			+	-
RMT Flow meter and Outlet Turbine Flow meter	+	-	+	-
Outlet MOV position and Bypass MOV Position	-	-	+	+
Outlet MOV position and Outlet Turbine Flow meter		-	+	+
Bypass MOV Position and Outlet Turbine Flow meter	-	-	+	
PID controller Fault and Inlet MOV Fault	+		-	
PID controller Fault and Water Level Sensor		-		+
PID controller Fault and RMT Flow meter	+			-
PID controller and Outlet MOV position	+	-		+
PID controller and Bypass MOV Position	-		+	
PID controller and Outlet Turbine Flow meter		-	+	

Using the results shown in Table 4.2 and additional measurements, a rule-based expert system for fault isolation was developed. The rules consist of ***if-then*** statements based on the characteristic behavior of each pre-defined fault. A complete list of all the rules, used for both single and dual-fault situations, is given below.

```
1)  IF (Inlet Flow rate Error > Threshold)
                AND (Controller Output Error > Threshold)
                AND (Bypass MOV Position Error < Threshold)

                THEN Loop Faulty Component = Inlet MOV

2)  IF (Outlet Flow rate Error > Threshold)
                AND (Water Level Error > Threshold)
                AND (Outlet MOV Position Error < Threshold)

                THEN Loop Faulty Component = Water Level Sensor

3)  IF (Inlet Flow rate Error > Threshold)
                AND (Water Level Error > Threshold)

                        THEN Loop Faulty Component = Inlet Flow rate

4)  IF (Outlet Flow rate Error > Threshold)
                AND (Water Level Error > Threshold)
                AND (Outlet MOV Position Error > Threshold)

                        THEN Loop Faulty Component = Outlet MOV

5)  IF (Inlet Flow rate Error > Threshold)
                AND (Controller Output Error > Threshold)
                AND (Bypass MOV Position Error > Threshold)

                        THEN Loop Faulty Component = Bypass MOV

6)  IF (Outlet Flow rate Error > Threshold)
                AND (Controller Output Error > Threshold)

                        THEN Loop Faulty Component = Outlet Flow meter

7)  IF (Water Level Set point Error > Threshold)

                        THEN Loop Faulty Component = Water Level Controller


8)  IF (Inlet Flow rate Error > Threshold)
                AND (Outlet Flow rate Error > Threshold)
                AND (Controller Output Error > Threshold)
                AND (Water Level Error > Threshold)
                AND (Inlet MOV Position Error > Threshold)


                THEN Loop Faulty Component = Inlet MOV and Water Level sensor

9)  IF (Inlet Flow rate Error > Threshold)
                AND (Controller Output Error > Threshold)
```

```
             AND (Water Level Error > Threshold)
             AND (Inlet MOV Position Error > Threshold)

             THEN Loop Faulty Component = Inlet MOV and RMT Flow meter Fault


10) IF (Inlet Flow rate Error > Threshold)
             AND (Outlet Flow rate Error > Threshold)
             AND (Controller Output Error > Threshold)
             AND (Water Level Error > Threshold)
             AND (Outlet MOV Position Error > Threshold)
             AND (Inlet MOV Position Error > Threshold)

             THEN Loop Faulty Component = Inlet MOV and Outlet MOV Position


11) IF (Bypass MOV Position Error > Threshold)
             AND (Inlet MOV Position Error > Threshold)

             THEN Loop Faulty Component = Inlet MOV and Bypass MOV Position


12) IF (Inlet Flow rate Error > Threshold)
             AND (Outlet Flow rate Error > Threshold)
             AND (Controller Output Error > Threshold)
             AND (Inlet MOV Position Error > Threshold)

             THEN Loop Faulty Component = Inlet MOV and Outlet Turbine Flow meter


13) IF (Inlet Flow rate Error > Threshold)
             AND (Outlet Flow rate Error > Threshold)

             THEN Loop Faulty Component = Water Level sensor and RMT Flow meter


14) IF (Outlet Flow rate Error > Threshold)
             AND (Water Level Error > Threshold)
             AND (Outlet MOV Position Error > Threshold)

                  THEN Loop Faulty Component = Water Level sensor and Outlet MOV


15) IF (Inlet Flow rate Error > Threshold)
             AND (Outlet Flow rate Error > Threshold)
             AND (Controller Output Error > Threshold)
             AND (Water Level Error > Threshold)
             AND (Bypass MOV Position Error > Threshold)

                  THEN Loop Faulty Component = Water Level sensor and Bypass MOV


16) IF (Outlet Flow rate Error > Threshold)
```

AND (Controller Output Error > Threshold)
AND (Water Level Error > Threshold)

THEN Loop Faulty Component = Water Level sensor and Outlet Turbine

17) IF (Inlet Flow rate Error > Threshold)
AND (Outlet Flow rate Error > Threshold)
AND (Water Level Error > Threshold)
AND (Outlet MOV Position Error > Threshold)

THEN Loop Faulty Component = RMT Flow meter and Outlet MOV

18) IF (Controller Output Error > Threshold)
AND (Water Level Error > Threshold)
AND (Bypass MOV Position Error > Threshold)

THEN Loop Faulty Component = RMT Flow meter and Bypass MOV

19) IF (Inlet Flow rate Error > Threshold)
AND (Outlet Flow rate Error > Threshold)
AND (Controller Output Error > Threshold)
AND (Water Level Error > Threshold)

THEN Loop Faulty Component = RMT Flow meter and Outlet Turbine

20) IF (Inlet Flow rate Error > Threshold)
AND (Outlet Flow rate Error > Threshold)
AND (Controller Output Error > Threshold)
AND (Water Level Error > Threshold)
AND (Outlet MOV Position Error > Threshold)
AND (Bypass MOV Position Error > Threshold)

THEN Loop Faulty Component = Outlet MOV Position and Bypass MOV

21) IF (Outlet Flow rate Error > Threshold)
AND (Controller Output Error > Threshold)
AND (Water Level Error > Threshold)
AND (Outlet MOV Position Error > Threshold)

THEN Loop Faulty Component = Outlet MOV Position and Outlet Turbine Flow meter

22) IF (Inlet Flow rate Error > Threshold)
AND (Outlet Flow rate Error > Threshold)
AND (Controller Output Error > Threshold)
AND (Bypass MOV Position Error > Threshold)

THEN Loop Faulty Component = Bypass MOV Position and Outlet Turbine Flow meter

23) IF (Inlet Flow rate Error > Threshold)
AND (Controller Output Error > Threshold)
AND (Water Level Set point Error > Threshold)
AND (Inlet MOV Position Error > Threshold)

THEN Loop Faulty Component = PID Controller and Inlet MOV faults

24) IF (Outlet Flow rate Error > Threshold)
AND (Water Level Error > Threshold)
AND (Water Level Set point Error > Threshold)

THEN Loop Faulty Component = PID Controller and Water Level sensor

25) IF (Inlet Flow rate Error > Threshold)
AND (Water Level Error > Threshold)
AND (Water Level Set point Error > Threshold)

THEN Loop Faulty Component = PID Controller and RMT Flow meter

26) IF (Inlet Flow rate Error > Threshold)
AND (Outlet Flow rate Error > Threshold)
AND (Water Level Error > Threshold)
AND (Water Level Set point Error > Threshold)
AND (Outlet MOV Position Error > Threshold)

THEN Loop Faulty Component = PID Controller and Outlet MOV Position

27) IF (Inlet Flow rate Error > Threshold)
AND (Controller Output Error > Threshold)
AND (Water Level Set point Error > Threshold)
AND (Bypass MOV Position Error > Threshold)

THEN Loop Faulty Component = PID Controller and Bypass MOV Position

28) IF (Outlet Flow rate Error > Threshold)
AND (Controller Output Error > Threshold)
AND (Water Level Set point Error > Threshold)

THEN Loop Faulty Component = PID Controller and Outlet Turbine Flow meter

Remarks:

Using the above rules we could detect and isolate single and dual faults occurring in the experimental water loop, except in one case. This case occurs when we have Outlet MOV position fault (single fault) or Water Level sensor fault and Outlet MOV position fault (dual fault). In both these cases, the values of the GMDH predictions have similar behavior. In such cases, additional information or measurements are necessary to resolve the faulty situation.

4.6 Results of Simulation of Fault Detection and Isolation Studies

The following is a summary of the results of a simulation study of the fault detection and isolation simulation performed using the Matlab-Simulink model. All the single and double fault cases were simulated. The results of this study demonstrate that the performance of the FDI algorithm in detecting faulty sensors/devices is excellent.

```
Predicted Inlet Flow rate: 0.010424
Model Inlet Flow rate: 0.010346
Percentage of Variation: -0.754702 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.082766
Percentage of Variation: -0.645251 (%)

Predicted PID Output: 0.493633
Model PID Output: 0.494193
Percentage of Variation: 0.113352 (%)

Predicted Water Level: 0.042740
Model Water Level: 0.042857
Percentage of Variation: 0.273263 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.494193
Measured Position: 0.494193
```

```
Percentage of Variation: 0.000000 (%)

Fault Type =

No fault detected
```

```
Predicted Inlet Flow rate: 0.009717
Model Inlet Flow rate: 0.010346
Percentage of Variation: 6.081036 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.082766
Percentage of Variation: -0.645251 (%)

Predicted PID Output: 0.493633
Model PID Output: 0.444940
Percentage of Variation: -10.943678 (%)

Predicted Water Level: 0.042740
Model Water Level: 0.042857
Percentage of Variation: 0.273263 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.444940
Measured Position: 0.494193
Percentage of Variation: -11.069578 (%)


Fault Type =

Inlet MOV Fault
```

```
Process Time: 0.520000
Predicted Inlet Flow rate: 0.009277
Model Inlet Flow rate: 0.009317
Percentage of Variation: 0.434580 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.074539
Percentage of Variation: -11.753406 (%)

Predicted PID Output: 0.419792
Model PID Output: 0.415886
Percentage of Variation: -0.939251 (%)

Predicted Water Level: 0.038533
Model Water Level: 0.042857
Percentage of Variation: 10.089986 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.415886
Measured Position: 0.415886
Percentage of Variation: 0.000000 (%)


Fault Type =

Water Level Sensor Fault
```

```
Predicted Inlet Flow rate: 0.010424
Model Inlet Flow rate: 0.011396
Percentage of Variation: 8.531987 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.082766
Percentage of Variation: -0.645251 (%)

Predicted PID Output: 0.493633
Model PID Output: 0.494193
Percentage of Variation: 0.113352 (%)

Predicted Water Level: 0.047109
Model Water Level: 0.042857
Percentage of Variation: -9.921735 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.494193
Measured Position: 0.494193
Percentage of Variation: 0.000000 (%)


Fault Type =

RMT Flow meter Fault
```

```
Predicted Inlet Flow rate: 0.008303
Model Inlet Flow rate: 0.008434
Percentage of Variation: 1.554577 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.067471
Percentage of Variation: -23.460349 (%)

Predicted PID Output: 0.365378
Model PID Output: 0.356373
Percentage of Variation: -2.526843 (%)

Predicted Water Level: 0.034919
Model Water Level: 0.042857
Percentage of Variation: 18.521490 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.812597
Percentage of Variation: 9.711405 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.356373
Measured Position: 0.356373
Percentage of Variation: 0.000000 (%)


Fault Type =

Outlet MOV Position
```

```
Predicted Inlet Flow rate: 0.011022
Model Inlet Flow rate: 0.010346
Percentage of Variation: -6.540570 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.082766
Percentage of Variation: -0.645251 (%)

Predicted PID Output: 0.493633
Model PID Output: 0.544582
Percentage of Variation: 9.355584 (%)

Predicted Water Level: 0.042740
Model Water Level: 0.042857
Percentage of Variation: 0.273263 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.771373
Percentage of Variation: -10.196140 (%)

Inlet MOV Setpoint Position: 0.544582
Measured Position: 0.544582
Percentage of Variation: 0.000000 (%)


Fault Type =

Bypass MOV Position
```

```
Predicted Inlet Flow rate: 0.010424
Model Inlet Flow rate: 0.010346
Percentage of Variation: -0.754702 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.074537
Percentage of Variation: -11.756441 (%)

Predicted PID Output: 0.419776
Model PID Output: 0.494193
Percentage of Variation: 15.058347 (%)

Predicted Water Level: 0.042740
Model Water Level: 0.042857
Percentage of Variation: 0.273263 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.494193
Measured Position: 0.494193
Percentage of Variation: 0.000000 (%)


Fault Type =

Outlet Turbine Flow meter
```

```
Predicted Inlet Flow rate: 0.006892
Model Inlet Flow rate: 0.007089
Percentage of Variation: 2.779387 (%)

Predicted Outlet Flow rate: 0.056598
Model Outlet Flow rate: 0.056690
Percentage of Variation: 0.162781 (%)

Predicted PID Output: 0.293820
Model PID Output: 0.283548
Percentage of Variation: -3.622405 (%)

Predicted Water Level: 0.029344
Model Water Level: 0.029355
Percentage of Variation: 0.037995 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.029355
Percentage of Variation: 31.505365 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.283548
Measured Position: 0.283548
Percentage of Variation: 0.000000 (%)


Fault Type =

PID controller Fault
```

```
Predicted Inlet Flow rate: 0.008474
Model Inlet Flow rate: 0.009308
Percentage of Variation: 8.955284 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.074463
Percentage of Variation: -11.867831 (%)

Predicted PID Output: 0.419165
Model PID Output: 0.365631
Percentage of Variation: -14.641400 (%)

Predicted Water Level: 0.038494
Model Water Level: 0.042857
Percentage of Variation: 10.180673 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.365631
Measured Position: 0.415208
Percentage of Variation: -13.559418 (%)


Fault Type =

Inlet MOV Fault and Water Level Sensor Fault
```

```
Predicted Inlet Flow rate: 0.009710
Model Inlet Flow rate: 0.011386
Percentage of Variation: 14.724183 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.082766
Percentage of Variation: -0.645251 (%)

Predicted PID Output: 0.493633
Model PID Output: 0.444491
Percentage of Variation: -11.055774 (%)

Predicted Water Level: 0.047068
Model Water Level: 0.042857
Percentage of Variation: -9.824562 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.444491
Measured Position: 0.494193
Percentage of Variation: -11.181801 (%)


Fault Type =

Inlet MOV Fault and RMT Flow meter Fault
```

```
Predicted Inlet Flow rate: 0.007382
Model Inlet Flow rate: 0.008439
Percentage of Variation: 12.520947 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.067509
Percentage of Variation: -23.391954 (%)

Predicted PID Output: 0.365648
Model PID Output: 0.308802
Percentage of Variation: -18.408437 (%)

Predicted Water Level: 0.034939
Model Water Level: 0.042857
Percentage of Variation: 18.476632 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.812823
Percentage of Variation: 9.686386 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.308802
Measured Position: 0.356672
Percentage of Variation: -15.501652 (%)


Fault Type =

Inlet MOV Fault and Outlet MOV Position
```

```
Predicted Inlet Flow rate: 0.010424
Model Inlet Flow rate: 0.010346
Percentage of Variation: -0.754702 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.082766
Percentage of Variation: -0.645251 (%)

Predicted PID Output: 0.493633
Model PID Output: 0.494193
Percentage of Variation: 0.113352 (%)

Predicted Water Level: 0.042740
Model Water Level: 0.042857
Percentage of Variation: 0.273263 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.769764
Percentage of Variation: -9.966345 (%)

Inlet MOV Setpoint Position: 0.494193
Measured Position: 0.543446
Percentage of Variation: -9.966345 (%)


Fault Type =

Inlet MOV Fault and Bypass MOV Position
```

```
Predicted Inlet Flow rate: 0.009719
Model Inlet Flow rate: 0.010346
Percentage of Variation: 6.061601 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.074539
Percentage of Variation: -11.753406 (%)

Predicted PID Output: 0.419792
Model PID Output: 0.445071
Percentage of Variation: 5.679639 (%)

Predicted Water Level: 0.042740
Model Water Level: 0.042857
Percentage of Variation: 0.273263 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.445071
Measured Position: 0.494193
Percentage of Variation: -11.036939 (%)


Fault Type =

Inlet MOV Fault and Outlet Turbine Flow meter
```

```
Predicted Inlet Flow rate: 0.009253
Model Inlet Flow rate: 0.010346
Percentage of Variation: 10.557606 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.074363
Percentage of Variation: -12.018408 (%)

Predicted PID Output: 0.418342
Model PID Output: 0.414320
Percentage of Variation: -0.970680 (%)

Predicted Water Level: 0.042740
Model Water Level: 0.042857
Percentage of Variation: 0.273263 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.414320
Measured Position: 0.414320
Percentage of Variation: 0.000000 (%)


Fault Type =

Water Level Sensor Fault and RMT Flow meter Fault
```

```
Predicted Inlet Flow rate: 0.005767
Model Inlet Flow rate: 0.005879
Percentage of Variation: 1.901189 (%)

Predicted Outlet Flow rate: 0.065678
Model Outlet Flow rate: 0.047030
Percentage of Variation: -39.651459 (%)

Predicted PID Output: 0.248721
Model PID Output: 0.245957
Percentage of Variation: -1.123806 (%)

Predicted Water Level: 0.030823
Model Water Level: 0.042857
Percentage of Variation: 28.079230 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000024 (%)

Outlet MOV Setpoint Position: 0.800000
Measured Position: 0.716747
Percentage of Variation: 10.406626 (%)

Bypass MOV Setpoint Position: 0.800000
Measured Position: 0.800000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.245957
Measured Position: 0.245957
Percentage of Variation: 0.000000 (%)


Fault Type =

Water Level Sensor Fault and Outlet MOV Position
```

```
Predicted Inlet Flow rate: 0.009888
Model Inlet Flow rate: 0.009290
Percentage of Variation: -6.436993 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.074316
Percentage of Variation: -12.088435 (%)

Predicted PID Output: 0.417960
Model PID Output: 0.456164
Percentage of Variation: 8.375028 (%)

Predicted Water Level: 0.038419
Model Water Level: 0.042857
Percentage of Variation: 10.354989 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: -0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.771463
Percentage of Variation: -10.209068 (%)

Inlet MOV Setpoint Position: 0.456164
Measured Position: 0.456164
Percentage of Variation: 0.000000 (%)


Fault Type =

Water Level Sensor Fault and Bypass MOV Position
```

```
Predicted Inlet Flow rate: 0.009270
Model Inlet Flow rate: 0.009310
Percentage of Variation: 0.439513 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.066202
Percentage of Variation: -25.827818 (%)

Predicted PID Output: 0.356337
Model PID Output: 0.415394
Percentage of Variation: 14.217085 (%)

Predicted Water Level: 0.038505
Model Water Level: 0.042857
Percentage of Variation: 10.155823 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.415394
Measured Position: 0.415394
Percentage of Variation: 0.000000 (%)


Fault Type =

Water Level Sensor Fault and Outlet Turbine Flow meter
```

```
Predicted Inlet Flow rate: 0.008226
Model Inlet Flow rate: 0.009410
Percentage of Variation: 12.578092 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.066955
Percentage of Variation: -24.412211 (%)

Predicted PID Output: 0.361678
Model PID Output: 0.352275
Percentage of Variation: -2.669276 (%)

Predicted Water Level: 0.038910
Model Water Level: 0.042857
Percentage of Variation: 9.209935 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: -0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.809483
Percentage of Variation: 10.057462 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.352275
Measured Position: 0.352275
Percentage of Variation: 0.000000 (%)


Fault Type =

RMT Flow meter Fault and Outlet MOV Position
```

```
Predicted Inlet Flow rate: 0.011006
Model Inlet Flow rate: 0.011371
Percentage of Variation: 3.216201 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.082766
Percentage of Variation: -0.645251 (%)

Predicted PID Output: 0.493633
Model PID Output: 0.543179
Percentage of Variation: 9.121466 (%)

Predicted Water Level: 0.047005
Model Water Level: 0.042857
Percentage of Variation: -9.677425 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.769386
Percentage of Variation: -9.912258 (%)

Inlet MOV Setpoint Position: 0.543179
Measured Position: 0.543179
Percentage of Variation: 0.000000 (%)


Fault Type =

RMT Flow meter Fault and Bypass MOV Position
```

```
Predicted Inlet Flow rate: 0.010424
Model Inlet Flow rate: 0.011405
Percentage of Variation: 8.604518 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.074290
Percentage of Variation: -12.127503 (%)

Predicted PID Output: 0.417748
Model PID Output: 0.494193
Percentage of Variation: 15.468661 (%)

Predicted Water Level: 0.047147
Model Water Level: 0.042857
Percentage of Variation: -10.010503 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.494193
Measured Position: 0.494193
Percentage of Variation: 0.000000 (%)


Fault Type =

RMT Flow meter Fault and Outlet Turbine Flow meter
```

```
Predicted Inlet Flow rate: 0.008802
Model Inlet Flow rate: 0.008313
Percentage of Variation: -5.874404 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.066505
Percentage of Variation: -25.253264 (%)

Predicted PID Output: 0.358482
Model PID Output: 0.384859
Percentage of Variation: 6.853725 (%)

Predicted Water Level: 0.034422
Model Water Level: 0.042857
Percentage of Variation: 19.681264 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: -0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.806761
Percentage of Variation: 10.359944 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.772520
Percentage of Variation: -10.359944 (%)

Inlet MOV Setpoint Position: 0.384859
Measured Position: 0.384859
Percentage of Variation: 0.000000 (%)


Fault Type =

Outlet MOV position Fault and Bypass MOV Position
```

```
Predicted Inlet Flow rate: 0.008150
Model Inlet Flow rate: 0.008305
Percentage of Variation: 1.869022 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.057826
Percentage of Variation: -44.052817 (%)

Predicted PID Output: 0.300900
Model PID Output: 0.348203
Percentage of Variation: 13.584896 (%)

Predicted Water Level: 0.034388
Model Water Level: 0.042857
Percentage of Variation: 19.762091 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: -0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.806353
Percentage of Variation: 10.405262 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.348203
Measured Position: 0.348203
Percentage of Variation: 0.000000 (%)


Fault Type =

Outlet MOV Position Fault and Outlet Turbine Flow meter
```

```
Predicted Inlet Flow rate: 0.011028
Model Inlet Flow rate: 0.010346
Percentage of Variation: -6.591704 (%)

Predicted Outlet Flow rate: 0.083300
Model Outlet Flow rate: 0.074254
Percentage of Variation: -12.183014 (%)

Predicted PID Output: 0.417447
Model PID Output: 0.545020
Percentage of Variation: 23.407045 (%)

Predicted Water Level: 0.042740
Model Water Level: 0.042857
Percentage of Variation: 0.273263 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.771993
Percentage of Variation: -10.284769 (%)

Inlet MOV Setpoint Position: 0.545020
Measured Position: 0.545020
Percentage of Variation: 0.000000 (%)


Fault Type =

Bypass MOV Position Fault and Outlet Turbine Flow meter
```

```
Predicted Inlet Flow rate: 0.006483
Model Inlet Flow rate: 0.007318
Percentage of Variation: 11.415346 (%)

Predicted Outlet Flow rate: 0.058432
Model Outlet Flow rate: 0.058516
Percentage of Variation: 0.144084 (%)

Predicted PID Output: 0.305244
Model PID Output: 0.263691
Percentage of Variation: -15.758305 (%)

Predicted Water Level: 0.030289
Model Water Level: 0.030300
Percentage of Variation: 0.038683 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.030300
Percentage of Variation: 29.298986 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.263691
Measured Position: 0.292718
Percentage of Variation: -11.008097 (%)


Fault Type =

PID controller Fault and Inlet MOV Fault
```

```
Predicted Inlet Flow rate: 0.006499
Model Inlet Flow rate: 0.006612
Percentage of Variation: 1.709294 (%)

Predicted Outlet Flow rate: 0.058357
Model Outlet Flow rate: 0.052788
Percentage of Variation: -10.550544 (%)

Predicted PID Output: 0.270009
Model PID Output: 0.264491
Percentage of Variation: -2.086258 (%)

Predicted Water Level: 0.027503
Model Water Level: 0.030262
Percentage of Variation: 9.117532 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.030262
Percentage of Variation: 29.387886 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.264491
Measured Position: 0.264491
Percentage of Variation: 0.000000 (%)


Fault Type =

PID controller Fault and Water Level Sensor Fault
```

```
Predicted Inlet Flow rate: 0.006892
Model Inlet Flow rate: 0.007797
Percentage of Variation: 11.614730 (%)

Predicted Outlet Flow rate: 0.056598
Model Outlet Flow rate: 0.056690
Percentage of Variation: 0.162781 (%)

Predicted PID Output: 0.293820
Model PID Output: 0.283548
Percentage of Variation: -3.622405 (%)

Predicted Water Level: 0.032285
Model Water Level: 0.029355
Percentage of Variation: -9.983092 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.029355
Percentage of Variation: 31.505365 (%)

Outlet MOV Setpoint Position: 0.900000
Measured Position: 0.900000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.283548
Measured Position: 0.283548
Percentage of Variation: 0.000000 (%)


Fault Type =

PID controller Fault and RMT Flow meter Fault
```

```
Predicted Inlet Flow rate: 0.004881
Model Inlet Flow rate: 0.005080
Percentage of Variation: 3.916024 (%)

Predicted Outlet Flow rate: 0.050652
Model Outlet Flow rate: 0.040638
Percentage of Variation: -24.642490 (%)

Predicted PID Output: 0.206564
Model PID Output: 0.204004
Percentage of Variation: -1.254906 (%)

Predicted Water Level: 0.026851
Model Water Level: 0.033143
Percentage of Variation: 18.984938 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.033143
Percentage of Variation: 22.667085 (%)

Outlet MOV Setpoint Position: 0.800000
Measured Position: 0.717135
Percentage of Variation: 10.358170 (%)

Bypass MOV Setpoint Position: 0.800000
Measured Position: 0.800000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.204004
Measured Position: 0.204004
Percentage of Variation: 0.000000 (%)


Fault Type =

PID controller Fault and Outlet MOV Position
```

```
Predicted Inlet Flow rate: 0.006142
Model Inlet Flow rate: 0.005770
Percentage of Variation: -6.447694 (%)

Predicted Outlet Flow rate: 0.046378
Model Outlet Flow rate: 0.046162
Percentage of Variation: -0.468441 (%)

Predicted PID Output: 0.242884
Model PID Output: 0.264704
Percentage of Variation: 8.243300 (%)

Predicted Water Level: 0.030251
Model Water Level: 0.030252
Percentage of Variation: 0.003177 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.030252
Percentage of Variation: 29.411539 (%)

Outlet MOV Setpoint Position: 0.800000
Measured Position: 0.800000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.800000
Measured Position: 0.882154
Percentage of Variation: -10.269225 (%)

Inlet MOV Setpoint Position: 0.264704
Measured Position: 0.264704
Percentage of Variation: 0.000000 (%)


Fault Type =

PID controller Fault and Bypass MOV Position
```

```
Predicted Inlet Flow rate: 0.005814
Model Inlet Flow rate: 0.005920
Percentage of Variation: 1.786318 (%)

Predicted Outlet Flow rate: 0.047512
Model Outlet Flow rate: 0.042621
Percentage of Variation: -11.476917 (%)

Predicted PID Output: 0.219426
Model PID Output: 0.248216
Percentage of Variation: 11.598555 (%)

Predicted Water Level: 0.031041
Model Water Level: 0.031037
Percentage of Variation: -0.011790 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.031037
Percentage of Variation: 27.579513 (%)

Outlet MOV Setpoint Position: 0.800000
Measured Position: 0.800000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.800000
Measured Position: 0.800000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.248216
Measured Position: 0.248216
Percentage of Variation: 0.000000 (%)


Fault Type =

PID controller Fault and Outlet Turbine Flow meter
```

```
Predicted Inlet Flow rate: 0.008299
Model Inlet Flow rate: 0.008174
Percentage of Variation: -1.518967 (%)

Predicted Outlet Flow rate: 0.065678
Model Outlet Flow rate: 0.065395
Percentage of Variation: -0.432238 (%)

Predicted PID Output: 0.384712
Model PID Output: 0.388666
Percentage of Variation: 1.017533 (%)

Predicted Water Level: 0.042803
Model Water Level: 0.042857
Percentage of Variation: 0.127335 (%)

Water Level Setpoint: 0.042857
Measured Water Level: 0.042857
Percentage of Variation: 0.000000 (%)

Outlet MOV Setpoint Position: 0.800000
Measured Position: 0.800000
Percentage of Variation: 0.000000 (%)

Bypass MOV Setpoint Position: 0.800000
Measured Position: 0.800000
Percentage of Variation: 0.000000 (%)

Inlet MOV Setpoint Position: 0.388666
Measured Position: 0.388666
Percentage of Variation: 0.000000 (%)


Fault Type =

No fault detected
```

4.7 Implementation of the FDI Algorithm for the Experimental Control Loop

The performance of the FDI algorithm was checked in the low-pressure water loop experimental system described earlier. All the steps for the development and implementation of the FDI algorithm performed in the simulations were repeated for this experimental setup. The steps are:

- Generation of fault free database
- Definition of qualitative relationships among loop devices
- Quantitative definition of those relationships by applying the GMDH method
- Development of a rule-based expert system

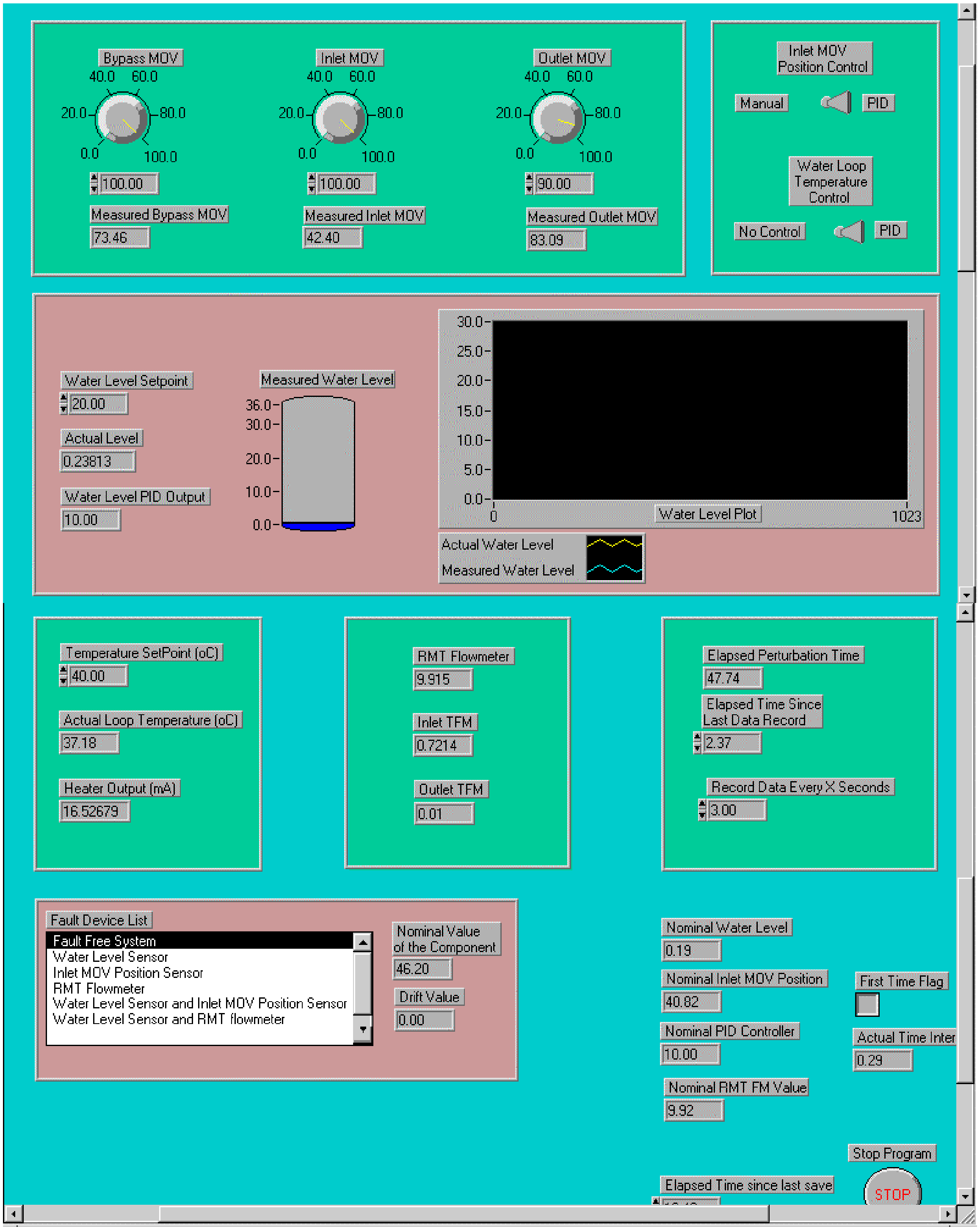
4.7.1. LabView Program

A LabView program was developed for controlling the experiment, data acquisition, and for system perturbations (imposed single faults).

LabView programs are called virtual instruments (VIs). The VIs have three main parts: the front panel, the block diagram, and the icon/connector. The front panel provides the means for setting input values and viewing outputs from the VI block diagram. Because the front panel is analogous to a front panel of a real instrument, the inputs are called controls and the outputs are called indicators. Each front panel has an accompanying block diagram, which is the program. The block diagram is built using the graphical programming language G. The icon/connector turns a VI into an object (sub-VI) that is used as a subroutine in the block diagrams or other VIs.

The LabView program performs not only data acquisition and experiment control, but also artificially creates the faults in different loop components. A ***TANH(X)***-type function was used as drift. For single fault, the drift was introduced into each loop component signal one at a time. For multiple faults, the drift was introduced into two loop component signals at a time. A maximum drift of 10 % of the sensor's nominal value prior to the fault was added in a fifteen-minute time interval. For example, if the water level sensor output is sending a signal representing 20 inches of water before the fault is imposed, a drift of 2 inches was added during a fifteen minutes of perturbation.

Figure 4.17 shows the main panel of the LabView program. The selection of the loop component (or components) to introduce the drift is made by the user in the front panel of the LabView program. A list box at the lower left corner of the main screen makes the selection. According to the choice of the fault in the list box of the front panel, the drifts are added to the loop component.



Bypass MOV
100.00
Measured Bypass MOV
73.46

Inlet MOV
100.00
Measured Inlet MOV
42.40

Outlet MOV
90.00
Measured Outlet MOV
83.09

Inlet MOV Position Control

Water Loop Temperature Control

Water Level Setpoint
20.00

Actual Level
0.23813

Water Level PID Output
10.00

Measured Water Level

Water Level Plot

Actual Water Level

Measured Water Level

Temperature SetPoint (oC)
40.00

Actual Loop Temperature (oC)
37.18

Heater Output (mA)
16.52679

RMT Flowmeter
9.915

Inlet TFM
0.7214

Outlet TFM
0.01

Elapsed Perturbation Time
47.74

Elapsed Time Since Last Data Record
2.37

Record Data Every X Seconds
3.00

Fault Device List
- Fault Free System
- Water Level Sensor
- Inlet MOV Position Sensor
- RMT Flowmeter
- Water Level Sensor and Inlet MOV Position Sensor
- Water Level Sensor and RMT flowmeter

Nominal Value of the Component
46.20

Drift Value
0.00

Nominal Water Level
0.19

Nominal Inlet MOV Position
40.82

Nominal PID Controller
10.00

Nominal RMT FM Value
9.92

First Time Flag
□

Actual Time Inter
0.29

Stop Program

Elapsed Time since last save

Figure 4.17. Computer monitor display of the LabView program for data acquisition from the experimental control loop.

4.7.2. Data Filtering

An interesting fact observed during the experiments in the loop is that even when the experimental water loop system was operated at steady-state conditions, a low frequency oscillation was observed all the time. This oscillation is due to two factors, both coming from the Inlet motor-operated valve. The first factor is that the valve's actuator has an internal dead-band, that is, even though the water level controller output is increasing, the valve stays steady in its current position until the controller's output reaches a threshold. After reaching this threshold, the motor actuator moves the valve to its new position. The second factor is that the valve has a free motion of approximately five-degree in angle. These two factors cause the system to oscillate at a low frequency around the set point value. To solve this problem a simple filter was designed. Figure 4.18 shows the signal with oscillation and the signal after low-pass filtering.

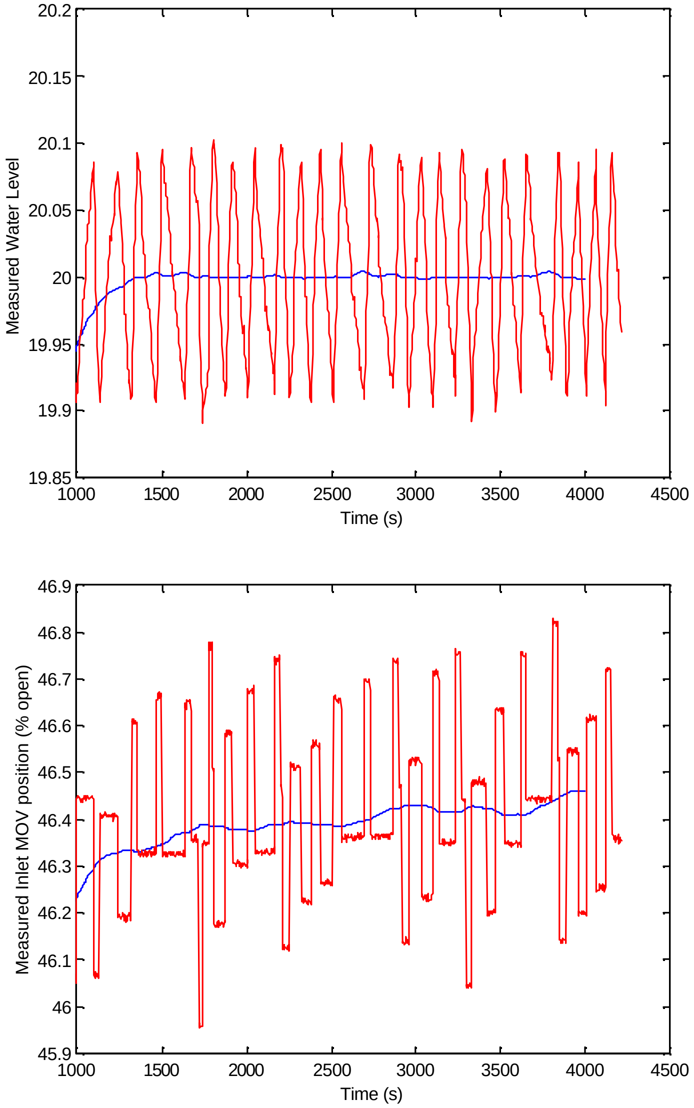


Figure 4.18. Measured and filtered data (water level and MOV position).

4.7.3. Generating the Database

Obtaining a fault-free database is important for the development of a FDI algorithm. This database should contain as many normal system conditions as possible. This database was used for generating prediction models and allowing the FDI algorithm to determine whether changes in the sensors' outputs are due to changes in the system set point or due to a loop component malfunction.

For generating the fault-free database, a LabView program was developed to automatically change the system normal operational conditions every ten minutes. The outlet motor-operated valve position was changing from 80% to 100% open (in steps of 5%). The water level set point was also changed from 10 to 28 inches (in steps of 2 inches). All 15 available variables were collected every three seconds for each perturbation. To check the system reproducibility, the complete set was run two times. The results are shown in Figure 4.19.

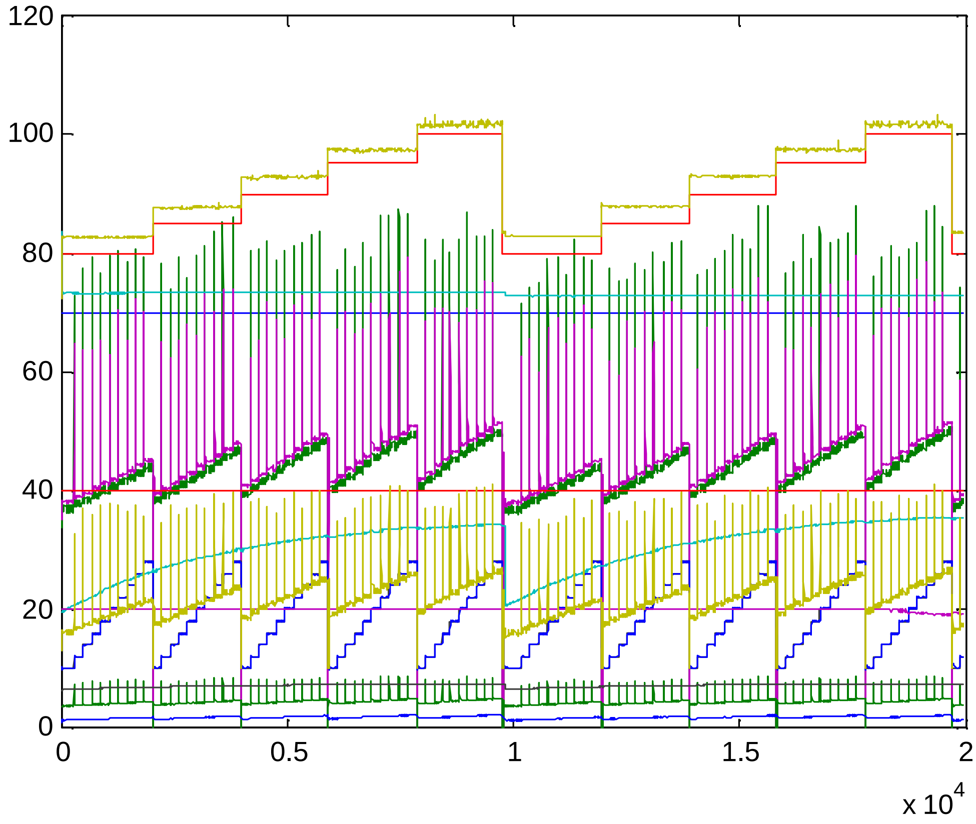


Figure 4.19. Fault-free database for the experimental system.

4.7.4. Generation of Qualitative Relationships for the Control Loop Components

The qualitative relationship among loop components were obtained by the same process described earlier, that is, by applying correlation coefficients method to the data in the fault-free database. Because the theoretical model was based on this experimental loop, the same qualitative relationships among components were obtained. However, we could not use the same theoretical quantitative relationships (models) for the variable prediction, due to the simplicity of the theoretical model. The GMDH method was applied to the experimental fault free data to get the mathematical models for variable prediction. About half of the data were used for training and the other half for performance evaluation.

Figures 4.20 through 4.31 show both the measured and the predicted values for the training and testing data set. The predicted values are very close to the measured ones. The error between these values (in percentage) is also shown.

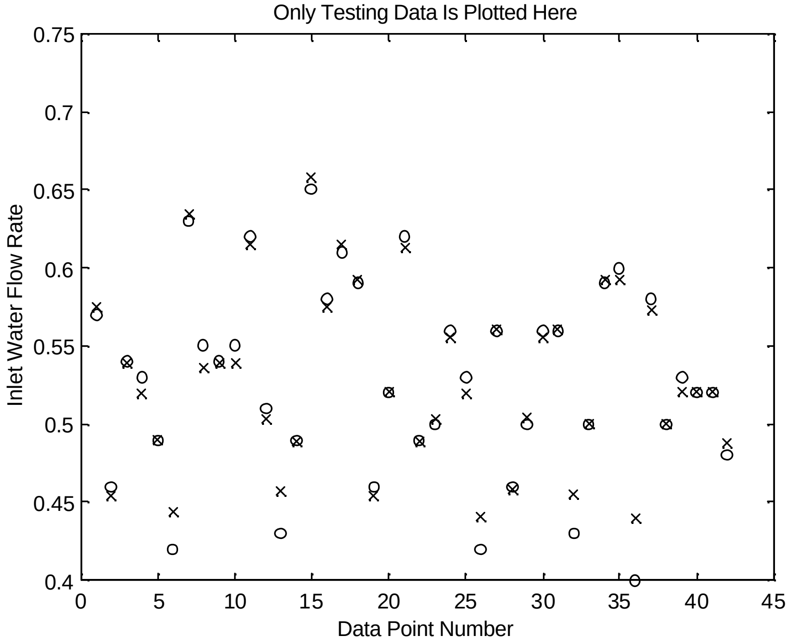


Figure 4.20. The GMDH predicted values ("o") against the test output values (x) for the Inlet Flow rate as a function of Bypass MOV and Inlet MOV positions.

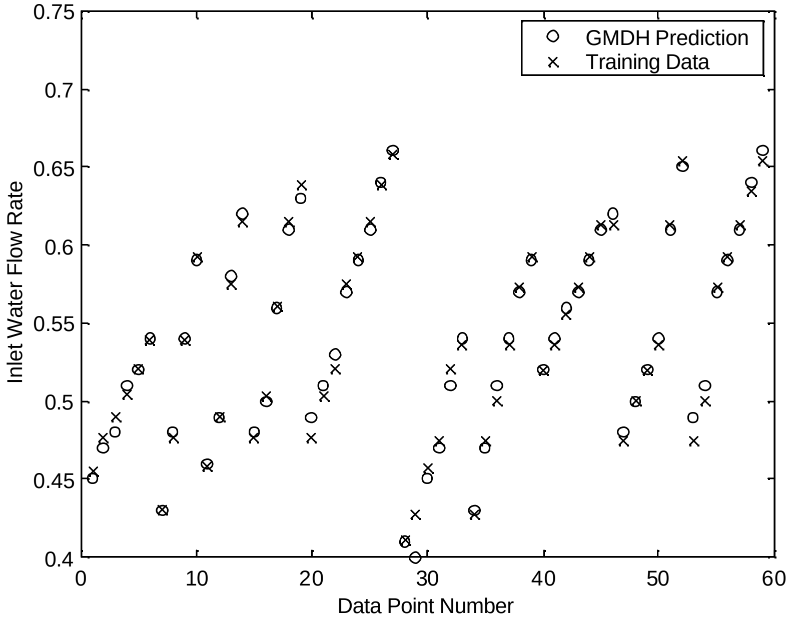


Figure 4.21. The GMDH predicted values ("o") against the training output values (x) for the Inlet Flow rate as a function of Bypass MOV and Inlet MOV positions.

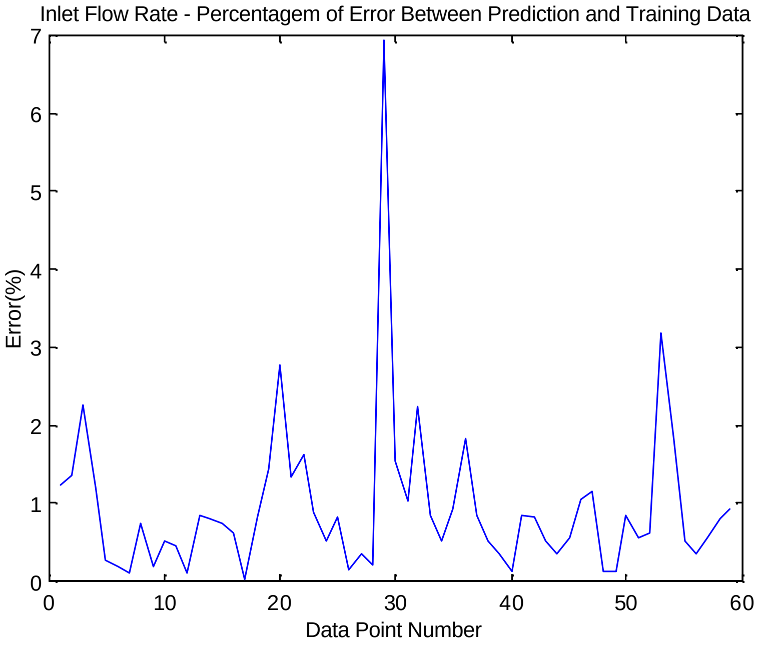


Figure 4.22. The error (%) between the predicted values and the measurements for the Inlet Flow rate as a function of Bypass MOV and Inlet MOV positions.

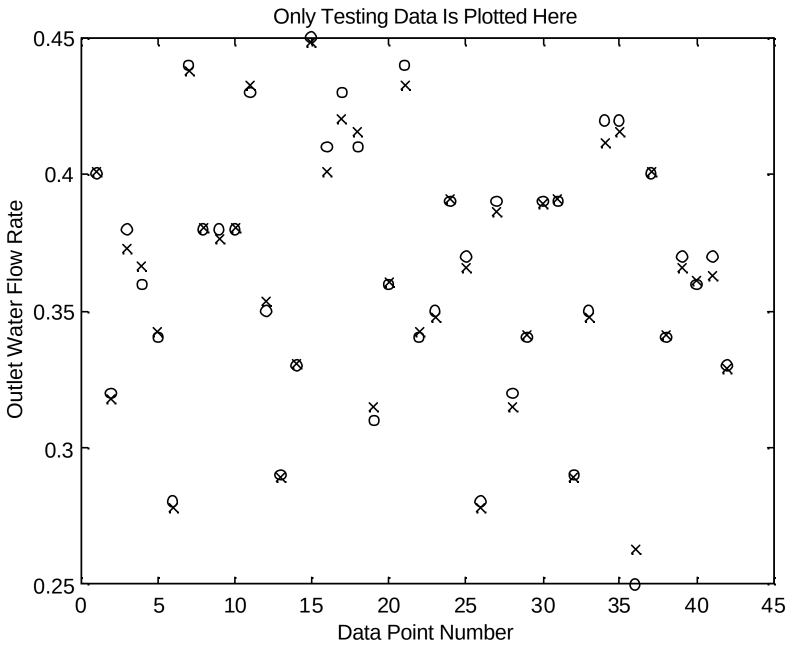


Figure 4.23. The GMDH predicted values ("o") against the test output values (x) for the Outlet Flow rate as a function of Water Level and Outlet MOV positions.

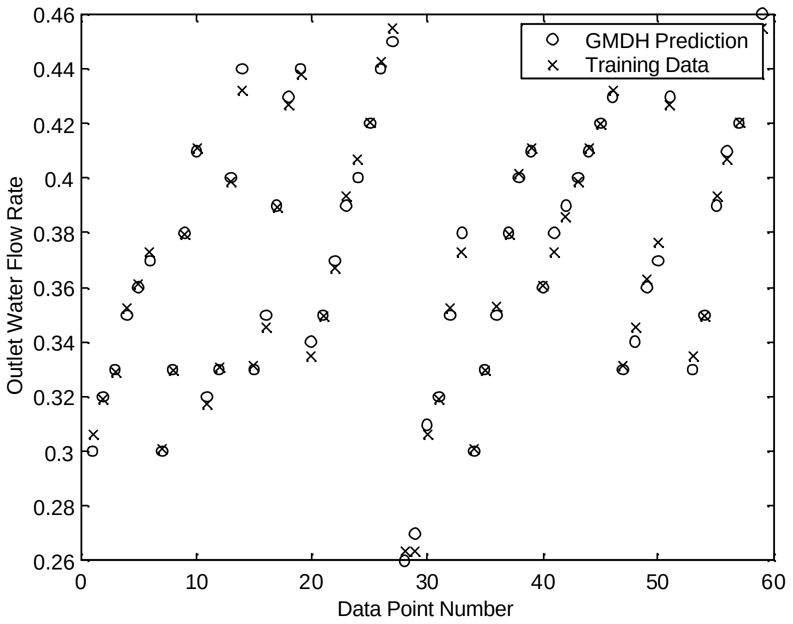


Figure 4.24. The GMDH predicted values ("o") against the training output values (x) for the Outlet Flow rate as a function of Water Level and Outlet MOV positions.

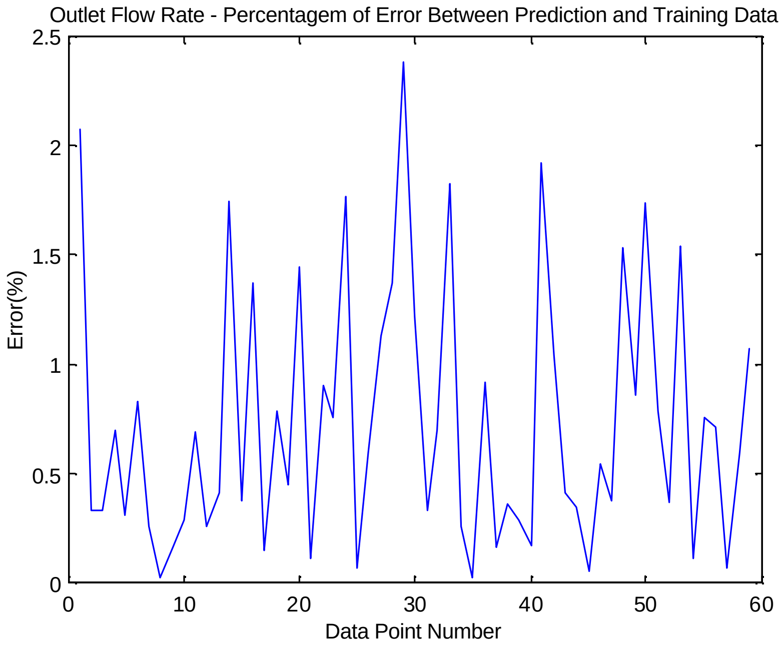


Figure 4.25. The error (%) between the predicted values and the measurements for the Outlet Flow rate as a function of Water Level and Outlet MOV positions.

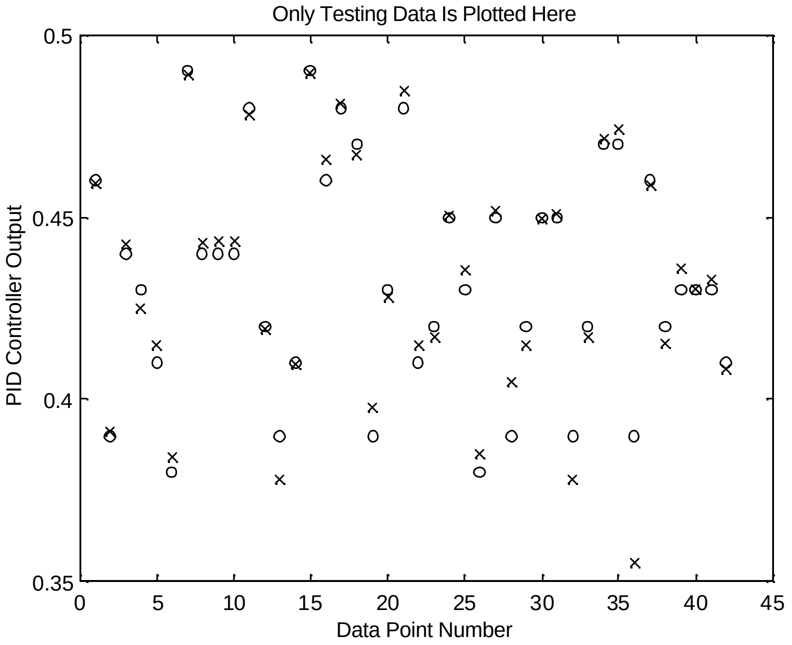


Figure 4.26. The GMDH predicted values ("o") against the test output values (x) for the PID Controller Output as a function of Bypass MOV, Water Level and Outlet flow rate.

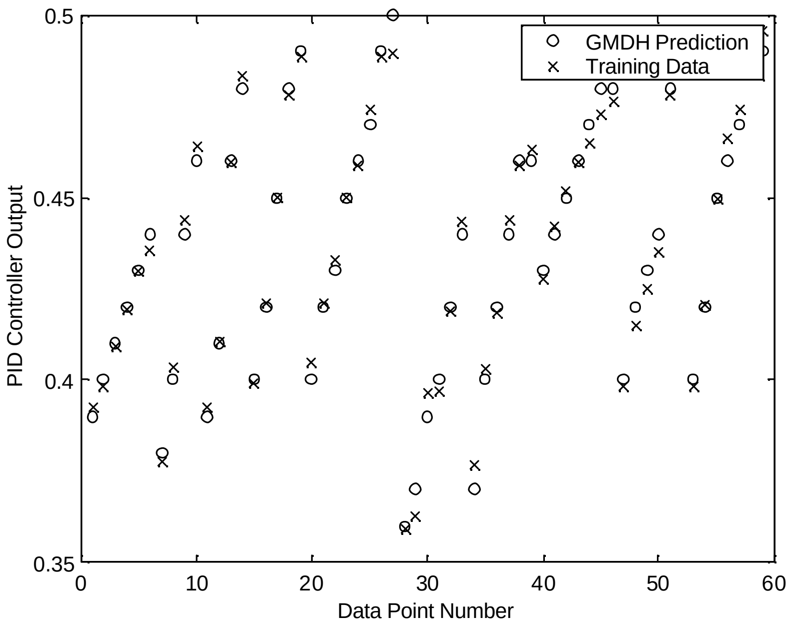


Figure 4.27. The GMDH predicted values ("o") against the training output values (x) for PID Controller Output as a function of Bypass MOV, Water Level and Outlet flow rate.

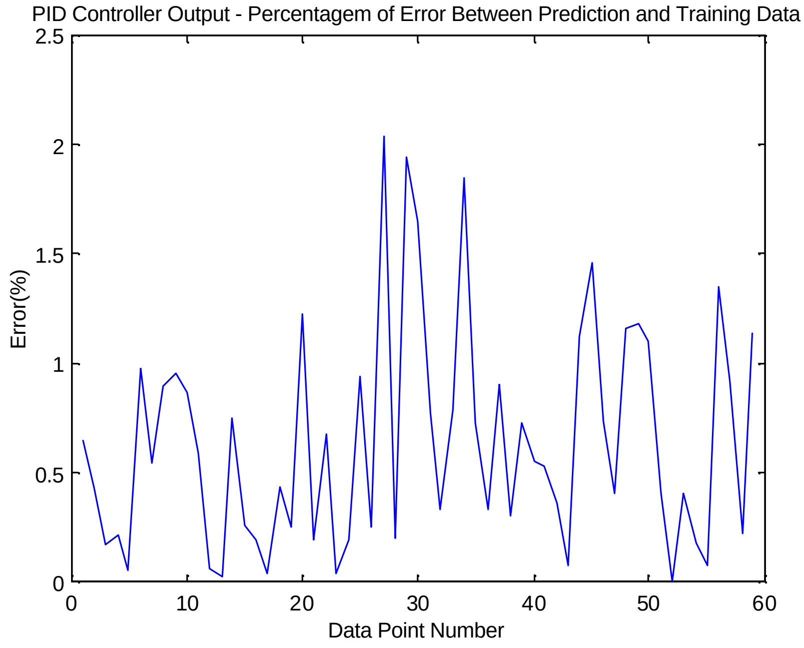


Figure 4.28. The error (%) between the predicted values and the measurements for the PID Controller Output as a function of Bypass MOV, Water Level and Outlet flow rate.

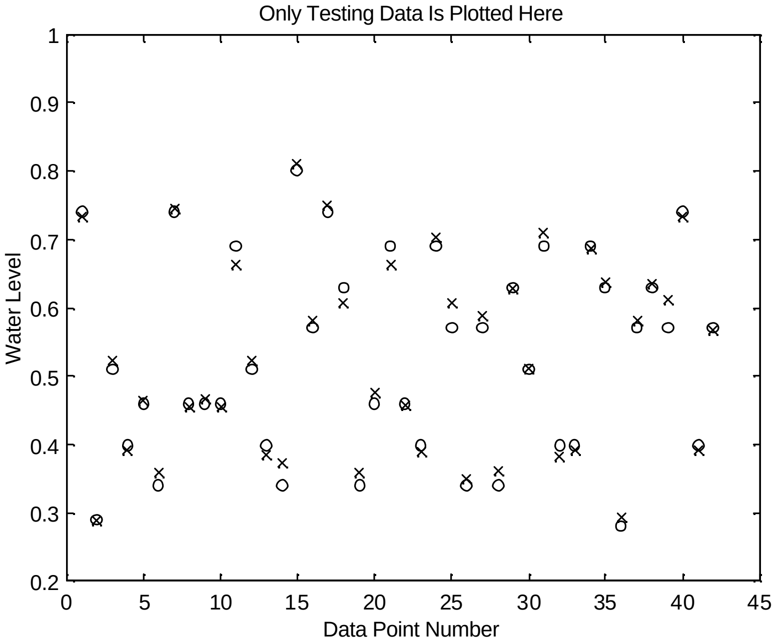


Figure 4.29. The GMDH predicted values ("o") against the test output values (x) for the Tank Water Level as a function of Inlet Flow rate and Outlet MOV positions.

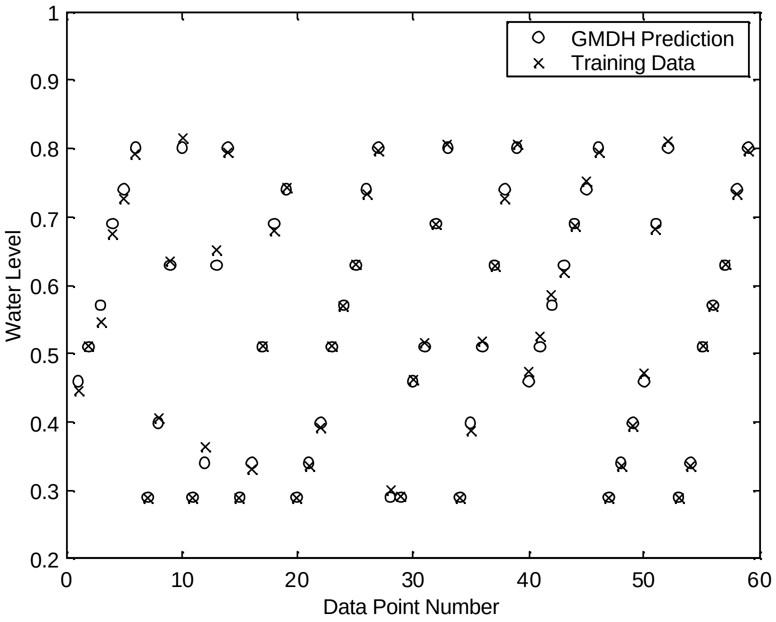


Figure 4.30. The GMDH predicted values ("o") against the training output values (x) for the Tank Water Level as a function of Inlet Flow rate and Outlet MOV positions.

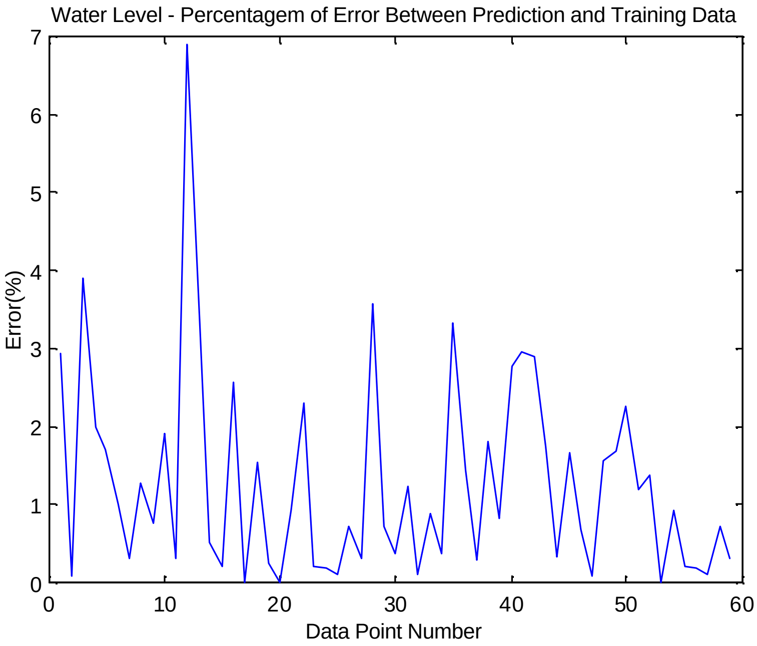


Figure 4.31. The error (%) between the predicted values and the measurements for the Tank Water Level as a function of Inlet Flow rate and Outlet MOV positions.

4.7.5. Experimental Results

Several experiments were performed in the low-pressure water loop system for checking the performance of the fault detection and isolation algorithm developed in this research. The following experiments were performed:

Single faults:

- Inlet MOV position fault
- Water level sensor fault
- RMT flow meter fault

Double faults:

- Inlet MOV position fault and Water level sensor fault
- Inlet MOV position fault and RMT flow meter fault
- Water level sensor fault and RMT flow meter fault

The following is a summary of the results of analysis of the experimental data for the imposed faults:

```
Predicted Inlet Flow rate: 0.379999
Model Inlet Flow rate: 0.411055
Percentage of Variation: 7.555140 (%)

Predicted Outlet Flow rate: 0.263051
Model Outlet Flow rate: 0.238558
Percentage of Variation: -2.093661 (%)

Predicted PID Output: 0.353009
Model PID Output: 0.337290
Percentage of Variation: -4.660607 (%)

Predicted Water Level: 0.321814
Model Water Level: 0.326577
Percentage of Variation: -1.479895 (%)

Water Level Setpoint: 0.285714
Measured Water Level: 0.283587
Percentage of Variation: 0.744660 (%)

Outlet MOV Setpoint Position: 0.800000
Measured Position: 0.799758
Percentage of Variation: 0.030226 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700292
Percentage of Variation: -0.041646 (%)

Inlet MOV Setpoint Position: 0.337290
Measured Position: 0.326077
Percentage of Variation: 3.324477 (%)


Fault Type =

Inlet MOV Fault
```

```
Predicted Inlet Flow rate: 0.437513
Model Inlet Flow rate: 0.441625
Percentage of Variation: -0.939755(%)

Predicted Outlet Flow rate: 0.258975
Model Outlet Flow rate: 0.276997
Percentage of Variation: -6.958854(%)

Predicted PID Output: 0.354104
Model PID Output: 0.344888
Percentage of Variation: 2.602761 (%)

Predicted Water Level: 0.313953
Model Water Level: 0.259312
Percentage of Variation: -21.071753 (%)

Water Level Setpoint: 0.285714
Measured Water Level: 0.277843
Percentage of Variation: 2.754893 (%)

Outlet MOV Setpoint Position: 0.800000
Measured Position: 0.799898
Percentage of Variation: 0.012799 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.699989
Percentage of Variation: 0.001584 (%)

Inlet MOV Setpoint Position: 0.367339
Measured Position: 0.359821
Percentage of Variation: -2.046650 (%)


Fault Type =

Water Level Sensor Fault
```

```
Predicted Inlet Flow rate: 0.586977
Model Inlet Flow rate: 0.443550
Percentage of Variation: -32.336015 (%)

Predicted Outlet Flow rate: 0.263210
Model Outlet Flow rate: 0.256999
Percentage of Variation: 2.359590 (%)

Predicted PID Output: 0.362095
Model PID Output: 0.366655
Percentage of Variation: 1.243559 (%)

Predicted Water Level: 0.421617
Model Water Level: 0.286259
Percentage of Variation: -47.285338 (%)

Water Level Setpoint: 0.285714
Measured Water Level: 0.286259
Percentage of Variation: -0.190490 (%)

Outlet MOV Setpoint Position: 0.800000
Measured Position: 0.801131
Percentage of Variation: -0.141426 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700161
Percentage of Variation: -0.022940 (%)

Inlet MOV Setpoint Position: 0.366655
Measured Position: 0.355885
Percentage of Variation: -2.937322 (%)


Fault Type =

RMT Flow meter Fault
```

```
Predicted Inlet Flow rate: 0.400076
Model Inlet Flow rate: 0.446718
Percentage of Variation: 10.441050 (%)

Predicted Outlet Flow rate: 0.263192
Model Outlet Flow rate: 0.279461
Percentage of Variation: 5.821468 (%)

Predicted PID Output: 0.364706
Model PID Output: 0.349312
Percentage of Variation: -4.406872 (%)

Predicted Water Level: 0.434217
Model Water Level: 0.286012
Percentage of Variation: -51.817437 (%)

Water Level Setpoint: 0.285714
Measured Water Level: 0.286012
Percentage of Variation: -0.104350 (%)

Outlet MOV Setpoint Position: 0.800000
Measured Position: 0.799745
Percentage of Variation: 0.031818 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700043
Percentage of Variation: -0.006199 (%)

Inlet MOV Setpoint Position: 0.349312
Measured Position: 0.364997
Percentage of Variation: -4.490270 (%)


Fault Type =

Inlet MOV Fault and Water Level Sensor Fault
```

```
Predicted Inlet Flow rate: 0.455162
Model Inlet Flow rate: 0.533609
Percentage of Variation: 14.701316 (%)

Predicted Outlet Flow rate: 0.334127
Model Outlet Flow rate: 0.342804
Percentage of Variation: 2.531280 (%)

Predicted PID Output: 0.404606
Model PID Output: 0.380670
Percentage of Variation: -6.288062 (%)

Predicted Water Level: 0.398560
Model Water Level: 0.287169
Percentage of Variation: -38.789047 (%)

Water Level Setpoint: 0.285714
Measured Water Level: 0.287169
Percentage of Variation: -0.509260 (%)

Outlet MOV Setpoint Position: 1.000000
Measured Position: 0.998297
Percentage of Variation: 0.170297 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.700367
Percentage of Variation: -0.052437 (%)

Inlet MOV Setpoint Position: 0.380670
Measured Position: 0.397486
Percentage of Variation: -4.417556 (%)


Fault Type =

Inlet MOV Fault and RMT Flow meter Fault
```

```
Predicted Inlet Flow rate: 0.466461
Model Inlet Flow rate: 0.505581
Percentage of Variation: 7.737662 (%)

Predicted Outlet Flow rate: 0.304979
Model Outlet Flow rate: 0.324003
Percentage of Variation: 5.871538 (%)

Predicted PID Output: 0.392468
Model PID Output: 0.393446
Percentage of Variation: 0.248585 (%)

Predicted Water Level: 0.318716
Model Water Level: 0.309679
Percentage of Variation: -2.835419 (%)

Water Level Setpoint: 0.285714
Measured Water Level: 0.277838
Percentage of Variation: 2.756431 (%)

Outlet MOV Setpoint Position: 1.000000
Measured Position: 1.003776
Percentage of Variation: -0.377636 (%)

Bypass MOV Setpoint Position: 0.700000
Measured Position: 0.699858
Percentage of Variation: 0.020284 (%)

Inlet MOV Setpoint Position: 0.393447
Measured Position: 0.404018
Percentage of Variation: -2.686962 (%)


Fault Type =

Water Level Sensor Fault and RMT Flow meter Fault
```

4.8 Summary of the GMDH Approach

The performance of the FDI algorithm using the GMDH prediction and rule-based decision making was excellent with applications to detect and isolate single and multiple faults. The fault isolation could also be made using a classification of the prediction error patterns for each fault-type. Figure 4.32 shows the prediction error of the four variables (inlet flow rate, outlet flow rate, tank water level, and controller output) for the six single-fault cases. Twenty samples each were simulated for the six fault scenarios. Each fault case has a unique residual pattern. In some cases two faults may have similar error behavior. In these cases it is necessary to have additional information about sensors/devices, such as set point values.

The system diagnostics was performed in two modules. The first module monitors the system variables and checks if the changes in them are due to changes in system operating level or due to device faults. The second module was used when an incipient fault is detected. A rule-based logic algorithm was developed to associate the measurements to one or multiple fault types. This module can be implemented for on-line applications and was evaluated for false and missed alarm rates.

The rule-base is often plant specific. In this project the GMDH module for state, controller, and actuator function prediction and the decision-making module are evaluated using a simulation model of an existing process control loop. The rule-base is established for detecting single faults and two simultaneous faults in sensors and other field devices. It is important to note that no assumption is made as to the nature of the incipient fault and the types of devices where the faults may occur.

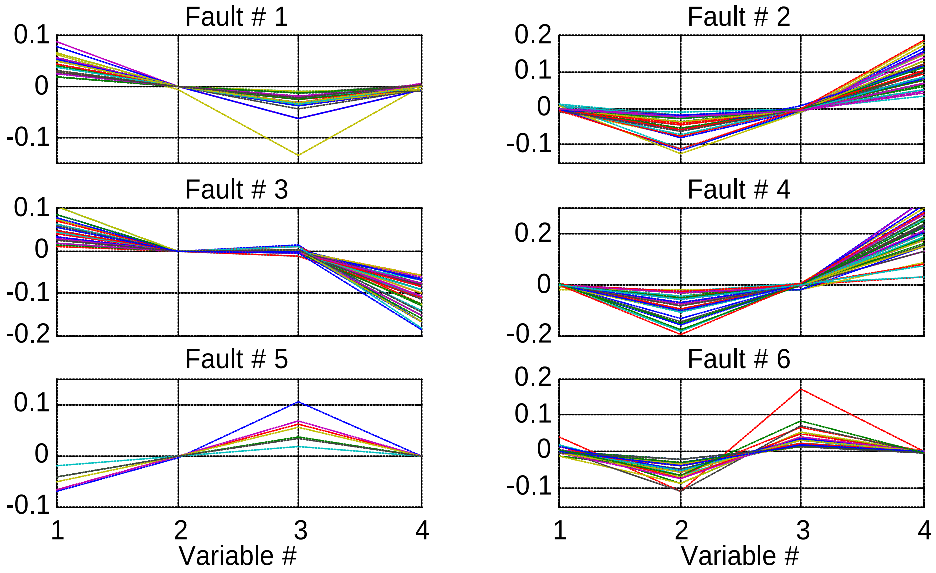


Figure 4.32. Plots of residual patterns for the six single device faults: inlet MOV, Water level sensor, Orifice flow meter, outlet MOV position, Bypass MOV position, and outlet turbine flow meter. The four variables are: 1. Inlet flow rate, 2. Outlet flow rate, 3. Controller output, and 4. Water level.

5. APPLICATION OF *PCA* AND PATTERN CLASSIFICATION APPROACH FOR FAULT DETECTION AND ISOLATION IN A *PWR* U-TUBE STEAM GENERATOR

5.1 Introduction

The objective of this application is to demonstrate the fault isolation in a complex system such as a U-tube steam generator (UTSG) in a pressurized water reactor (PWR) plant. The approach integrates the measurement characterization using the principal component analysis (PCA) and a pattern classification approach. Normal operational data and measurements with device faults were simulated using a Simulink model of a PWR plant with a UTSG. The focus is on the fault isolation for the UTSG system.

Review of Principal Component Analysis (PCA)

This section provides a brief description of PCA and the prediction error parameters used for fault detection. For the details of PCA refer to Section 3 and the list of References.

Principal Component Analysis (PCA) is an exploratory data analysis technique that is very popular for analyzing data sets with significant redundant (collinear) information. Consider an I by J matrix **X** where each row represents an observation. PCA decomposes the matrix as a product of scores (**T**) and loadings (**P**) as

$$\mathbf{X} = \mathbf{T}\mathbf{P}^{\mathbf{t}} + \mathbf{E} \qquad (5.1)$$

where **E** represents the residuals remaining after projection onto the principal component loadings. The PCs are obtained so that the successive PCs explain the maximum amount of variance in **X**. Thus, if the data are highly redundant with high correlation among the measurements, the first few PCs that explain most of the variation in the data can be retained and the higher PCs rejected leading to small residuals in **E**. PCA is thus a very efficient data compression technique.

To understand PCA better, consider **X** such that its columns are auto-scaled to zero mean and unit variance. The successive PCs then represent the directions of the principal axes of the hyper-ellipsoid that contains the data and are in descending order of the length of the principal axes. PCA is closely related to the singular value decomposition (SVD) of **X,** which decomposes **X** as

$$\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}' \qquad (5.2)$$

where

U: Orthogonal matrix (n by n) spanning the column space of **X**
S: Diagonal matrix (n by m) with the singular values in descending order

V: Orthogonal matrix (m by m) spanning the row space of **X**

For understanding, consider a column vector **b** (m by 1) in the row space of **X** so that

$$\mathbf{Xb} = \mathbf{US}\ (\mathbf{V^t b}) \qquad (5.3)$$

The term in the brackets represents a rotation of the reference frame from $\mathbf{I}_{JxJ}$ to **V**. Multiplication by **S** represents the scaling of the representation of **b** in the **V** frame by the corresponding singular values and transforms **b** to the column space of **X**. The representation of the resulting vector is, however, in the **U** frame and the multiplication by **U** transforms to the $\mathbf{I}_{IxI}$ reference frame. In terms of PCA, the columns of **V** are the PCs and the singular values are the lengths of the principal axes of the hyper-ellipsoid. The scores **T** are the projections onto the PCs and are obtained as

$$\begin{aligned}\mathbf{T} &= \mathbf{XV}\\ &= \mathbf{USV^tV}\\ &= \mathbf{US}\end{aligned}$$

It is worth mentioning that the scores are decorrelated or $\mathbf{T^tT}$ is a diagonal matrix since

$$\begin{aligned}T'T &= (XV)^t XV\\ &= (\mathbf{USV^tV})^t(\mathbf{USV^tV})\\ &= \mathbf{S^tU^tUS}\\ &= \mathbf{S^tS}\\ &= \mathbf{S^2}\end{aligned}$$

PCA thus represents a rotation of the $\mathbf{I}_{JxJ}$ reference frame to the PC reference frame so that in the PC frame, the data appears uncorrelated. Note that in cases where high redundancy exists in the data, the higher PCs would correspond to very small singular values and can be rejected with negligible loss of information. This leads to the residuals **E** and we have

$$\mathbf{X} = \mathbf{U_rS_rV_r^t} + \mathbf{E}$$

The subscript r is used to emphasize the fact that only r of a maximum of rank(**X**) PCs are retained.

Any new test sample **x** can be projected on the retained PCs to obtain the scores (**t**) and the residuals (**e**) as

$$\mathbf{t} = \mathbf{xV_r}$$

and

$$\begin{aligned}\mathbf{e} &= \mathbf{x} - \mathbf{tV_r^t}\\ &= \mathbf{x} - \mathbf{xV_rV_r^t}\\ &= \mathbf{x}(\mathbf{I} - \mathbf{V_rV_r^t}) \qquad (5.4)\end{aligned}$$

In applications to monitoring industrial processes, PCA is typically used to build a model of the nominal operation data. Multivariate SPC charts can then be used to detect abnormalities in the PC space and the residuals space. For the PC space, the Hotelling's T^2 statistic can be used while a sum-of-squared error can be used for the residuals. Table

5.1 summarizes the PCA equations, the monitoring statistics, and the expressions for obtaining the confidence limits on the statistics.

Application to a PWR Steam Generator System

A U-tube steam generator (UTSG) in a typical PWR is chosen to illustrate the application of PCA data characterization and pattern classification approach for fault isolation. The steam generator is used because of its importance in transferring energy from the primary system to the balance-of-plant system, and because of its dynamic complexity. Schematic of a typical UTSG is shown in Figure 5.1. The total number of signals used in the PCA analysis is m=18 and are listed in Table 5.2. Variables that directly influence the performance of the UTSG are used, including reactor power set point and the steam generator water level set point. The simulation study consisted of the following specifications:

- Number of measurement samples for PC estimation: 500
- Reactor power range: 75% - 100%
- Six devices were selected for fault analysis. Drift magnitudes of 0.5 – 2% of nominal values were imposed on the following devices:
    - SG water level sensor
    - Steam pressure sensor
    - Steam flow rate sensor
    - Feedwater flow rate sensor
    - Feedwater valve signal
    - Steam valve signal.

Table 5.3 shows the behavior of the prediction errors of the six variables (without the SG water level). The faults correspond to the devices associated with the UTSG only. The simulation takes into account the interaction in all the control loops in a typical PWR. Twenty samples each for the various fault scenarios were simulated. This study demonstrates the application of PCA for data characterization using eighteen variables. By retaining seven PCs to model the hyper-plane that best fits the data, five of the six single device faults are detected. The fault corresponding to a drift in the level sensor is not detected. The reason is that there are no other measurements that are highly correlated with the level. The absence of such alternative measurements makes drifts in the steam generator inventory *unobservable*. This situation may be resolved by using set point information and redundant measurements. For the other cases, alternate highly correlated measurements are available so that the faults are *observable*. The residuals produce unique fault patterns. By applying the rule base, all the observable fault scenarios are correctly isolated.

PCA Model

PC Equation (r retained PCs)

$$\mathbf{X} = \mathbf{T_r V_r} + \mathbf{E} = \mathbf{U_r S_r V_r^T} + \mathbf{E}$$

Sample Scores

$$\mathbf{t_i} = \mathbf{x_i V_r}$$

Sample Residuals

$$\mathbf{e_i} = \mathbf{x_i} - \mathbf{t_i V_r^T} = \mathbf{x_i}\left(\mathbf{I} - \mathbf{V_r V_r^T}\right)$$

Multivariate Sample Statistics

Hotelling's T^2

$$T_i^2 = \mathbf{t_i}\mathrm{cov}(\mathbf{t})^{-1}\mathbf{t_i^T} = \mathbf{t_i ?_r^{-1} t_i^T}$$

$$\text{where } \mathbf{?_r} = \frac{\mathbf{S_r^2}}{I-1}$$

Q statistic

$$Q_i = \mathbf{e_i e_i^T} = \mathbf{x_i}\left(\mathbf{I} - \mathbf{V_r V_r^T}\right)\mathbf{x_i^T}$$

Confidence Limits

Hotelling's T^2

$$T_{r,I,a}^2 = \frac{r(I-1)}{I-r} F_{r,I-r,a}$$

Q Statistic

$$Q_a = T_1\left[\frac{c_a\sqrt{2T_2h_0^2}}{T_1} + 1 + \frac{T_2h_0(h_0-1)}{T_1^2}\right]^{\frac{1}{h_0}}$$

$$\text{where } \Theta_i = \mathrm{trace}\left(\frac{\mathbf{EE^T}}{I-1}\right)^i \text{ for } i = 1, 2, 3$$

Variable Descriptions

$\mathbf{X}$	Data Matrix (I by J)
$\mathbf{V_r}$	Retained PC loadings (J by r)
$\mathbf{T_r}$	Scores on retained PCs (I by r)
$\mathbf{E}$	Residuals remaining (I by J)
$\mathbf{x_i}$	i^{th} sample row vector (1 by J)
$\mathbf{t_i}$	i^{th} sample scores (1 by r)
F	Critical value of F distribution
c_a	standard normal deviate

Table 5.1 PCA model equations and statistics

Table 5.2. Nuclear power plant measurements used in the UTSG system (PCA)

Index	**Description**	*Index*	**Description**
1	SG level	10	Pressurizer level
2	SG pressure	11	Pressurizer pressure
3	SG steam flow	12	**Primary side PC output**
4	SG FW flow	13	Pressurizer LC Output
5	SG LC output	14	**Turbine power**
6	Reactivity	15	Turbine RPM
7	Hot leg temperature	16	RPM controller output
8	Cold leg temperature	17	Power set point
9	T_{avg} controller output	18	Level set point

SG: steam generator; FW: feed water; PC: pressure controller;
LC: level controller; RPM: revolutions per minute.

Table 5.3. Fault isolation table for the nuclear plant data. All the faults are associated with the UTSG system

Variables ? *Fault Scenarios*?	2	3	4	5	10	14	15	17
SG Pressure	+						-	
SG Steam flow		+						
SG FW flow			+					
SG FW valve				-				
SG Steam valve		+	+	+	+	-		-

SG: Steam Generator; FW: Feed Water

Figure 5.2 shows the plots of PCA eigenvalues, Q-statistic, prediction errors and the fault directions. The patterns of prediction errors and fault directions indicate clearly the type of fault. The reason for the failure in not being able to identify the level senor error was due to the fact that the SG level was fixed for all power levels. This is not true in practice. The future work will use a full-scope PWR simulator that will include the water level dynamics. The knowledge of the level set point and fluid inventory balance may also be used to monitor a level sensor measurement in situations where the level is maintained constant for all conditions.

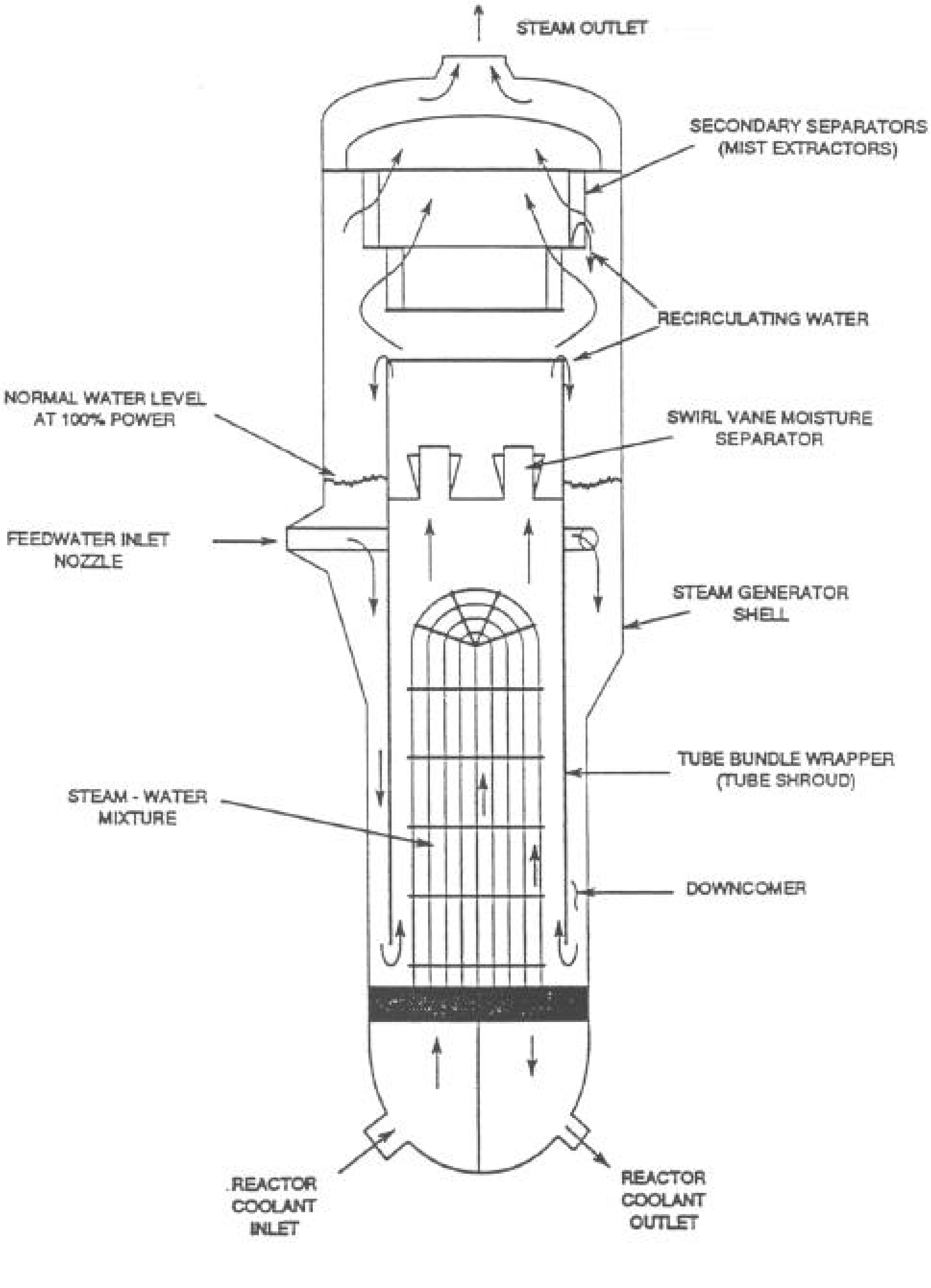


Figure 5.1. Schematic of a U-tube steam generator (UTSG) used in a pressurized water reactor (PWR) plant.

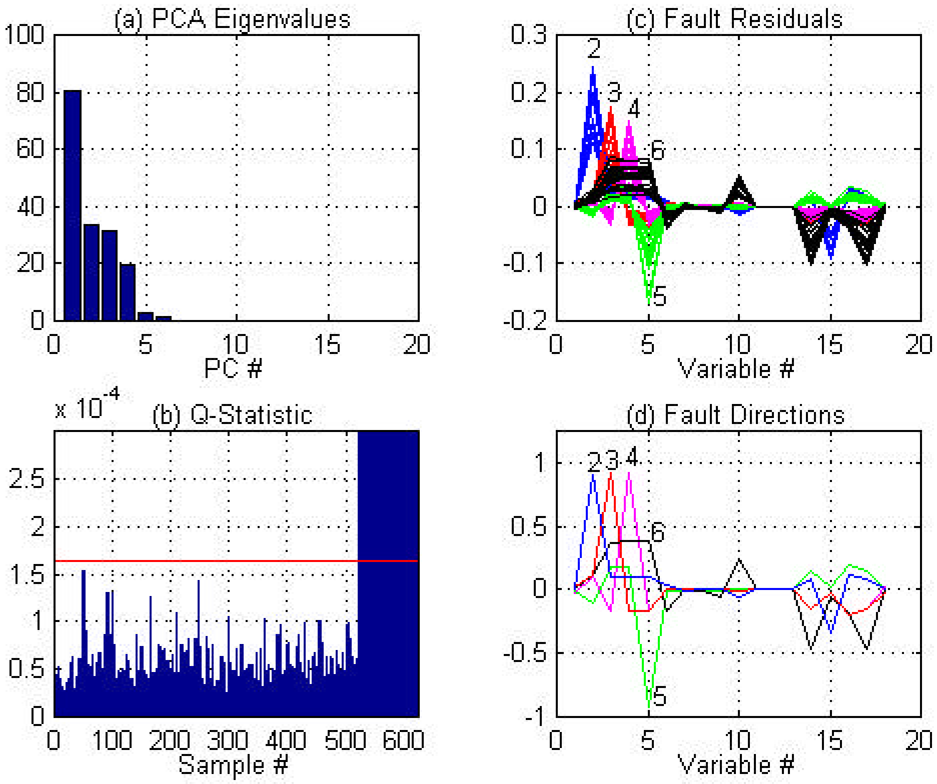


Figure 5.2. Results of principal component analysis of data from a U-tube steam generator system in a PWR plant.

6. CONCLUDING REMARKS AND FUTURE WORK

6.1 Summary and Concluding Remarks

An integrated method for incipient fault detection and isolation of sensors and field devices, using structured residuals, has been developed. The FDI system uses two approaches for characterizing system variables – states and control functions. These are (1) group method of data handling (GMDH) with rational function approximation, and (2) principal component analysis (PCA). The latter may be performed using both linear and nonlinear representation of the measurements. The isolation of device faults is performed using both a rule-based decision-making and a pattern classification of prediction error vectors in the *fault space*. This integrated approach enhances the fault diagnostics capability and provides a robust method for FDI. The integration of model predictive control [40, 41], *stateflow* graphical simulation environment [42], and the FDI module will be considered during future continuing work.

The methods being developed under this task for the DOE-NERI project have been tested and illustrated using measurements from a laboratory process control loop and simulation studies of a U-tube steam generator in a PWR. All the faults being simulated were detected successfully. The faults were primarily of the drift-type. Two manuscripts were prepared for presentations at scientific meetings that are sponsored by the American Nuclear Society during 2000.

6.2 Plans for Future Work

The following research tasks have been planned for completion during Phases 2 and 3 of this three-year project.

Phase 2 R&D Tasks

- Implementation of the FDI method for a UTSG system as part of a full-scope PWR plant. This would incorporate interactions among the various plant control loops.
- Development and extension of the current methods for the case of fault detection during plant transients.
- Interfacing the FDI module with the control system module through a system executive. A currently available virtual instrument platform will be used.
- Identification of realistic faults in a PWR and establishment of the characteristics of transient faults as compared with steady-state operation faults.

Phase 3 R&D Tasks

- Development of minimum requirements for application to an existing PWR plant.
- Fully automated implementation of the FDI system as part of the overall control architecture demonstration.
- Identification of issues in technology transfer to nuclear power industry.
- Resolution of algorithmic issues of concern in system implementation.
- **Deliverables:** Annual Reports and a Final Report.
  FDI software system and *User's Manual*.
  Conference and journal manuscripts.

REFERENCES

1. V.J. Van Doren, "Advanced Control Software Goes Beyond PID," Control Engineering, pp. 73-78, January 1998.

2. W. Gough, "Predictive Adaptive Control Aids Pulp Digestion," Control Engineering, p. 97, November 1997.

3. R.J. Patton, J. Chen, and S.B. Nielsen, "Model-Based Methods for Fault Diagnosis: Some Guidelines," *Transactions of the Institute of Measurements and Control,* Vol. **17**, pp. 73-83, 1995.

4. S.K. Yung, and D.W. Clarke, "Local Sensor Validation," *Measurement and Control,* Vol. **22**, pp. 132-141, 1989.

5. B.R. Upadhyaya, et al., *Development and Testing of an Integrated Signal Validation System for Nuclear Power Plants*, Vol. 1, 2, 3, Final Report prepared for the U.S. Department of Energy, DOE/NE/37959-34,35,36, September 1989.

6. K. Kavaklioglu and B.R. Upadhyaya, "Monitoring Feed water Flow Rate and Component Thermal Performance of Pressurized Water Reactors by Means of Artificial Neural Networks," *Nuclear Technology,* Vol. **107**, pp. 112-123, 1994.

7. A.S. Erbay and B.R. Upadhyaya, "A Personal Computer-Based On-Line Signal Validation System for Nuclear Power Plants," Nuclear Technology, Vol. 119, No. 1, pp. 63-75, July 1997.

8. X. Xu and J.W. Hines, "Sensor Validation and Fault Detection Using Neural Networks, *Proc. MARCON 99*, Vol. **2**, pp. 58.01-58.09, May 1999.

9. M. Kramer, "Nonlinear Principal Component Analysis Using Autoassociative Neural Networks," *AIChE Journal*, Vol. **37**, p. 233, 1991.

10. H. Tong and C.M. Crowe, "Detection of Gross Errors in Data Reconciliation by Principal Component Analysis," *AIChE Journal*, Vol. **41**, p. 1712, 1995.

11. S.J. Farlow, *Self-Organizing Methods in Modeling: GMDH Type Algorithms*, Marcel Dekker, New York, 1984.

12. P.B. Ferreira and B.R. Upadhyaya, *Incipient Fault Detection and Isolation of Sensors and Field Devices*, Research Report, Nuclear Engineering Department, The University of Tennessee, Knoxville, UTNE/BRU/99-02, December 1999.

13. S. Wold, K. Esbensen, and P. Geladi, "Principal Component Analysis," *Chemometrics and Intelligent Laboratory Systems*, Vol. **2**, p. 37, 1987.

14. B.R. Upadhyaya, O. Glockler, and J. Eklund, "Multivariate Statistical Signal Processing Technique for Fault Detection and Diagnostics," ISA Transactions, Vol. 29, No.4, pp. 79-95, 1990.

15. C. Hawman, "Intelligent Alarm management Saves Industry," InTech, Vol. 44, No. 10, pp. 34-36, October 1997.

16. C.M. Crowe, “Data Reconciliation-Progress and Challenges,” *Journal of Process Control*, Vol. **6**, p. 89, 1996.

17. C.M. Crowe, A. Hrymak, and Y.A. Garcia Campos, “Reconciliation of Process Flow Rates by Matrix Projection: The Linear Case,” *AIChE Journal*, Vol. **29**, p. 881, 1983.

18. B. Dayal, J.F. MacGregor, P.A. Taylor, R. Kildaw, and S. Marcikic, “Application of Feedforward Neural Networks and Partial Least Squares Regression for modeling Kappa Number in a Continuous Kamyr Digester,” *Pulp and Paper Canada*, Vol. **95**, p. 26, 1994.

19. D. Dong, and T.J. McAvoy, “Batch Tracking via Nonlinear Principal Component Analysis,” *AIChE Journal*, Vol. **42**, p. 2199, 1996.

20. D. Dong and T.J. McAvoy, “Nonlinear Principal Component Analysis- Based on Principal Curves and Neural Networks,” *Computers in Chemical Engineering*, Vol. **20**, p. 65, 1996.

21. R. Dunia, J. Qin, T.F. Edgar, and T.J. McAvoy, “Sensor Fault Identification and Reconstruction Using Principal Component Analysis,” *Proceedings of the 13th IFAC World Congress*, Vol. **N**, p. 259, 1996.

22. P. Fantoni and A. Mazzola, “Applications of Autoassociative Neural Networks for Signal Validation in Accident Management,” *Proceedings of the IAEA Specialist Meeting on Advanced Information Methods and Artificial Intelligence in Nuclear Power Plant Control Rooms*, 1994.

23. N.B. Gallagher, B.M. Wise, and C.W. Stewart, “Application of Multi-way Principal Component Analysis to Nuclear Waste Storage Tank Monitoring,” *Computers and Chemical Engineering*, Vol. **20,** Supplement S739, 1996.

24. J. Gertler, *Fault Detection and Diagnosis in Engineering Systems*, Marcel Dekker, New York, 1998.

25. J. Gertler and D. Singer, “Augmented Models for Statistical Fault Isolation in Complex Dynamic Systems,” *Proceedings of the American Control Conference*, 1985.

26. J. Gertler and D. Singer, “A New Structural Framework for Parity Equation Based Failure Detection and Isolation in Complex Plants,” *Automatica*, Vol. **26**, p. 381, 1990.

27. J. Gertler, W. Li, Y. Huang, and T.J. McAvoy, “Isolation Enhanced Principal Component Analysis,” *AIChE Journal*, Vol. **45**, p. 323, 1999.

28. D. Houdouin, J.F. MacGregor, M. Hou, and M. Franklin, “Multivariate Statistical Analysis of Mineral Processing Plant Data,” CIM Bulletin, *Mineral Processing*, Vol. **86**, p. 23, 1993.

29. J.E. Jackson and G. Mudholkar, “Control Procedures for residuals Associated with Principal Component Analysis,” *Technometrics*, Vol. **21**, p. 341, 1979.

30. J. Jolliffe, *Principal Component Analysis*, Springer Verlag, Berlin, 1996.

31. J. Keeler and B. Ferguson, “Commercial Applications of Soft Sensors: The Virtual Online Analyzer and the Software Cem,” *Proceedings of IFPAC Conference*, 1996.

32. E.B. Martin and A.J. Morris, “Overview of Multivariate Statistical Process Control in Continuous and Batch Process Performance Monitoring,” *Transactions of the Institute of Measurement and Control*, Vol. **18**, p. 51, 1996.

33. P. Miller, R.E. Swanson, and C.F. Heckler, “Contribution Plots: The Missing Link in Multivariate Quality Control,” *Proceedings of Fall Conference of the ASQC and ASA*, 1993.

34. M. Naghedolfeizi and B. R. Upadhyaya, *Dynamic Modeling of a Pressurized Water Reactor Plant for Diagnostics and Control*, Research Report, Nuclear Engineering Department, University of Tennessee, Knoxville, 1991.

35. P. Nomikos, and J.F. Macgregor, “Monitoring of Batch Process using Multiway Principal Component Analysis,” *AIChE Journal*, Vol. **40**, p. 1361, 1994.

36. S.J. Qin and W. Li, “Detection, Identification, and reconstruction of Faulty Sensors with Maximized Sensitivity,” *AIChE Journal*, Vol. **45**, p. 1963, 1999.

37. D.K. Rollins and J.F. Davis, “Unbiased estimation of Gross Errors in Process Measurements,” *AIChE Journal*, Vol. **38**, p. 563, 1992.

38. B. Skagerberg, J.F. MacGregor, and C. Kiparissides, “Multivariate Data Analysis Applied to Low Density Polyethylene Reactors,” *Chemometrics and Intelligent Laboratory Systems*, Vol. **14**, p. 341, 1992.

39. G. Strang, *Introduction to Linear Algebra*, Chapter 7, Wellesley-Cambridge Press, 1993.
40. J.B. Rawlings, “Tutorial Overview of Model Predictive Control,” IEEE Control Systems Magazine, Vol. 20, No. 3, pp. 38-52, June 2000.

41. S. Piche, B. Sayyar-Rodsari, D. Johnson, and M. Gerules, “Nonlinear Model Predictive Control Using Neural Networks,” IEEE Control Systems Magazine, Vol. 20, No. 3, pp. 53-62, June 2000.

42. Stateflow: A Graphical Simulation Environment for Modeling and Designing Complex Systems, The MathWorks, Inc., June 2000.

Attachment II
Task 2 Phase 2 Report

UTNE/DOE/00-01

FAULT DETECTION AND ISOLATION OF NUCLEAR PLANT SYSTEM SENSORS AND FIELD DEVICES

Belle R. Upadhyaya
(Principal Investigator)

Baofu Lu
Ke Zhao
Nitin Kaistha

Nuclear Engineering Department
The University of Tennessee
Knoxville, TN 37996-2300
E-mail: bupadhya@utk.edu

Annual Report
Task 2 Phase 2

Prepared for the DOE-NERI Project
DE-FG03-99SF21906

September 2001

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

This report describes the tasks performed and the progress made by The University of Tennessee (UTK) during 2000-2001 on the DOE-NERI project entitled ***Automatic Development of Highly Reliable Control Architecture for Future Nuclear Power Plants.*** UTK is collaborating with the Instrumentation & Controls Division of ORNL (lead organization) and the North Carolina State University (NCSU). The objective of the UTK research task is to develop an on-line monitoring system for *Fault Detection and Isolation (FDI)* of sensors and field devices in a nuclear power plant. This phase of the research has been focused on sensors and field devices in a nuclear power plant and the detection of dual faults in a steam generator system. Sensors include temperature, pressure, flow, level transmitters, and measurements of control functions, and field devices include valve actuators, spray and heater systems, pumps, and other similar equipment. The goal of this task is to provide diagnostics information to a central executive for enhanced decision-making by the plant control system.

The following R&D tasks have been accomplished during this reporting period:

- Development of data-driven models for the characterization of sub-system dynamics for predicting state variables, control functions, and expected control actions.
- Enhancement of the nonlinear system modeling approach called the *Group Method of Data Handling (GMDH)* with rational functions, to include non-linearity and temporal correlation (fault detection during plant transient).
- Enhancement of the *Principal Component Analysis (PCA)* to non-linear PCA.
- Development of a fault detection and isolation module based on pattern classification of fault residuals.
- Development of a technique for the isolation of simultaneous dual faults in the steam generator system.
- Development of an FDI demonstration module using the Halden *PICASSO* platform.

The above techniques have been applied to a laboratory process control loop and to data from a full-scope pressurized water reactor (PWR) simulation system.

During Phase-3 of this project (FY 2001), the FDI algorithms will be fine-tuned with the extension of the technique to the whole PWR plant. This phase will also include the development of minimum requirements for application to an existing PWR, and the limitations imposed by the measurements. The information generated by the FDI module will be interfaced with the control design system. A paper was presented at the ANS Topical Meeting on NPIC&HMIT, November 2000, and another paper will be presented at the American Nuclear Society Winter Meeting, November 2001. A journal publication in Nuclear Technology is due for November 2001.

1. INTRODUCTION

1.1 Background and Motivation

Existing and new generation of nuclear power plants have economic and reliability concerns as addressed by overall plant performance, unscheduled downtime and the long-term management of critical assets. The key to achieving these needs is to develop an integrated approach for monitoring, control, fault detection and diagnosis of plant components such as sensors, actuators, control devices and other equipment. Several methods developed by industry and academia, for monitoring isolated sensors and system components were reported [1-8]. Model-based local sensor validation and fault diagnosis methods were developed for specific applications [3,8]. These approaches assume that a system fault being monitored occurs in a specific plant component and in an isolated fashion. Fault detection and isolation (FDI) of sensors and field devices is an important step towards the implementation of an automated and intelligent process control strategy [12].

A large-scale system, such as a nuclear power plant, has several feedback control loops. This makes the identification and isolation of faults in these interconnected systems highly complex. Even when a sensor used for set point control is faulty, the control system through feedback, tries to vary the actuating signals until the error in the set point is eliminated. The *sensor-alone* type validation will fail in this situation. It is therefore necessary to consider fault detection and isolation at the system level rather than at the device level. The objective of this R&D task is to develop an on-line sensor and field device monitoring and fault detection system, when simultaneous faults may occur in two or more of these devices. This goal will be achieved by a two-step approach: (1) Development of data-driven models for predicting multiple variables, using rational function approximation and group method of data handling; (2) A decision-making module that uses system functional knowledge base and pattern classification algorithms, that will be deployed in a distributed configuration. High priority will be given to the computational efficiency of these techniques, with the capability to change the module structure with changing plant conditions. The intrinsic merit of the project lies in the development of an autonomous global monitoring and fault detection approach that would be executed with minimal human interaction.

1.2 Objectives of R&D and Definition of Tasks

The objective of this research task is to develop an on-line monitoring system for *Fault Detection and Isolation* (FDI) of sensors and field devices in a nuclear power plant. The sensor suite consists of major process variables in a plant, such as temperature, pressure, flow, level, and control functions. Field devices in a power plant include, but are not limited to, valve actuators, control modules, spray and heater systems, pumps, and similar equipment. Figure 1.1 shows an integrated plant monitoring, diagnosis, and prognosis system, with the ability to estimate the remaining useful life of plant devices.

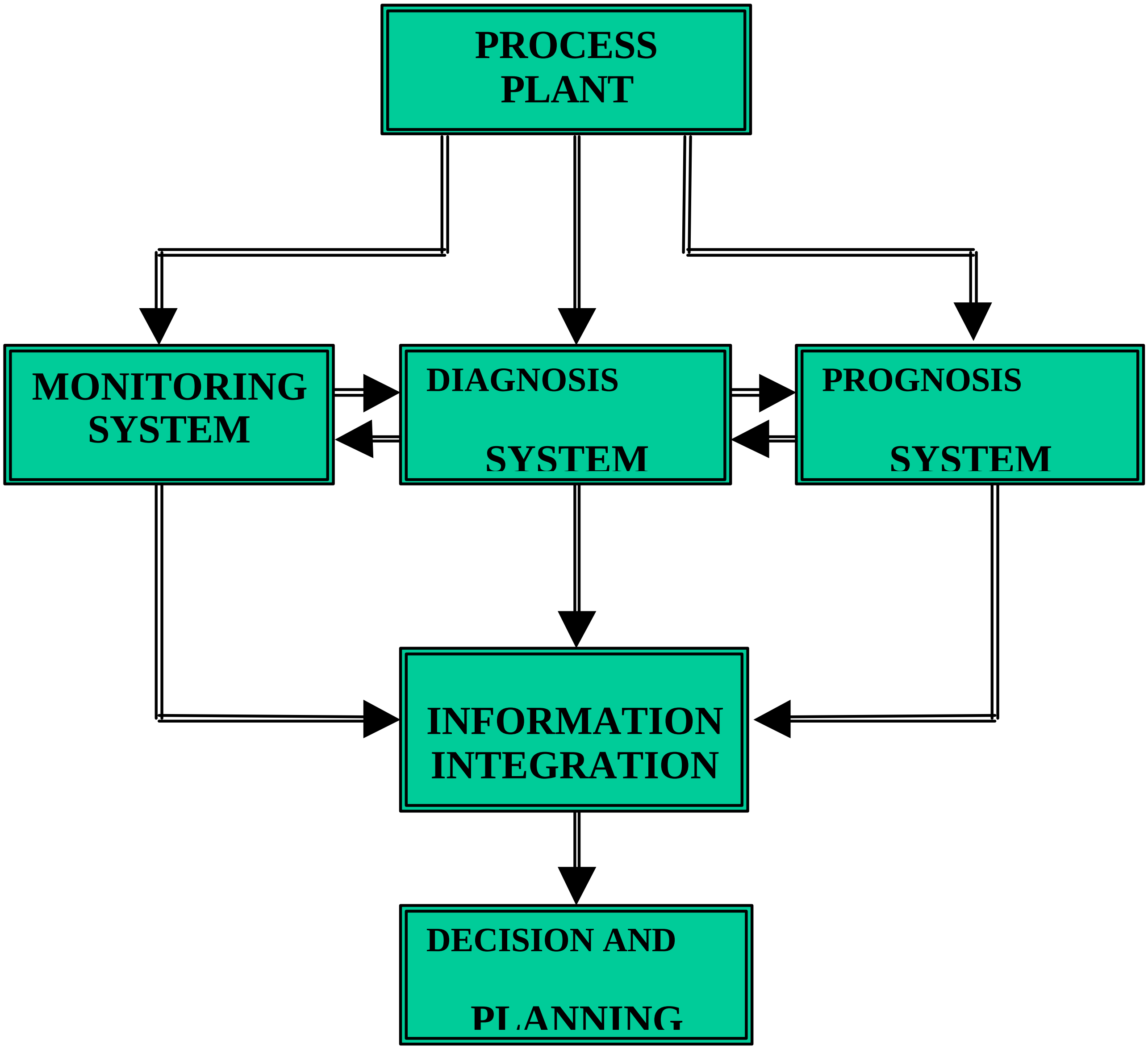


Figure 1.1 Integrated plant monitoring, diagnosis and prognosis system

The objectives of this R&D are being accomplished through the completion of the following technical tasks:

- Review of literature and previous work.
- Characterization of sub-system dynamics using data-driven models for predicting state variables, control functions, and expected control actions.
- Development of a *Group Method of Data Handling (GMDH)* modeling algorithm with rational function approximation.
- Development of a *Principal Component Analysis (PCA)* algorithm with linear and nonlinear mapping.

- Development of an FDI module that combines system operational knowledge and a rule-based logic for both single and dual faults in dissimilar sensors and field devices.
- Development of a complimentary module that quantifies the prediction error using a fault pattern classification technique.
- Demonstration of the FDI system with application to an experimental process control loop.
- Demonstration of the FDI system with application to a U-tube steam generator (UTSG) in a full-scope simulation model of a 1,300 MWe PWR.
- Development of minimum requirements for FDI system implementation.
- Extension of the techniques for the case of fault detection during plant transients.
- Identification of realistic faults in a PWR and establish the characteristics of transient faults as compared with steady-state faults.
- Interfacing the FDI module with control system module via the system executive and development of a graphical user interface (GUI) for the *FDI* system demonstration.
- Identification of issues in technology transfer to nuclear power industry.
- **Deliverables:** Annual Reports and a Final Report.
  FDI software system and *User's Manual.*
  Conference and journal manuscripts.

1.3 Summary of Significant Accomplishments During 2000-2001

The following major milestones were accomplished during this reporting period:

- Development and testing of time-dependent ***GMDH*** modeling module for state and control function prediction.

- Development and testing of linear and non-linear ***PCA*** method for system characterization.

- Generation of an extensive simulation database for normal and fault operation for a 1,300 MWe PWR plant in the power range 20% - 100%.

- Demonstration of the FDI module for both single and dual/simultaneous faults and includes the following highlights:

  - Rule-based decision making.

  - Fault isolation using fault residuals and pattern classification.

  - Steady state and transient plant operation conditions.

  - Combination of sensors and valve actuators.

  - Use of both Simulink™ and NCSU-PWR simulation codes.

- Demonstration of the FDI module using the Halden *PICASSO* platform.

- Preparation of the following manuscripts for publication.

  - *Fault Detection and Isolation of Nuclear Power Plant Sensors and Field Devices,* ANS Topical Meeting on NPIC & HMIT, November 2000.

  - *Incipient Fault Detection and Isolation in a Nuclear Power Plant Using Principal Component Analysis*, American Control Conference, Washington, D.C., June 2001.

  - Summary submitted to the ANS Winter Meeting, November 2001.

  - *Incipient Fault detection and Isolation of Field Devices in Nuclear Power Systems Using Principal Component Analysis*, to appear in Nuclear Technology, November 2001.

1.4 FDI Architecture and Issues in Developing a Robust FDI Algorithm

Figure 1.2 shows the functional modules of the FDI system being developed in this project. *GMDH, PCA* and *Artificial Neural Network (ANN)* modeling of process measurements are considered. This provides a crosschecking of prediction techniques applied to the measurements. Fault isolation is based on either a rule-based algorithm or a pattern classification algorithm. The following issues must be considered in developing a robust FDI algorithm.

- Sensor faults may not be detected in a closed-loop control system.

- Redundancies in sensors and controllers are used in nuclear power plants (NPPs).

- Separation of process variations from sensor/field-device faults must be considered.

- Noise levels in measurements can increase false alarms. It may be necessary to pre-process signals to eliminate this effect at different sub-bands.

- The use of physics models and data-driven models to understand and characterize the process dynamics.

1.5 Organization of the Report

The principal component analysis *(PCA)* and its applications are described in Section 2. The group method of data handling *(GMDH)* algorithm and its extensions to transient data are presented in Section 3. Applications to the FDI of dual faults through multiple observers are discussed in Sections 4. Section 5 describes the development of an FDI demonstration module for a PWR using the Halden *PICASSO* platform. Concluding remarks and plans for Phase-3 are given in Section 6.

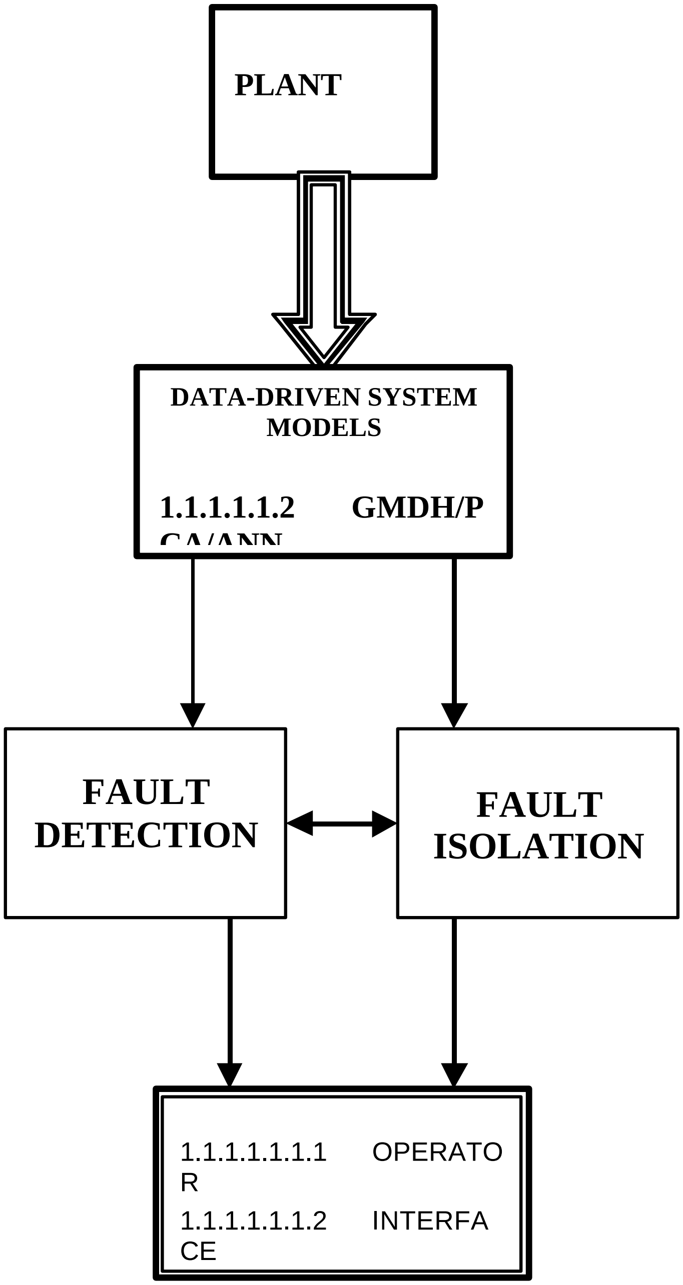


Figure 1.2 The schematic shows the FDI system functional modules.

2. PRINCIPAL COMPONENT ANALYSIS (PCA) AND ITS GENERALIZATION FOR INCIPIENT FAULT DETECTION

2.1 Introduction

Over the past few decades, industrial process systems have moved towards automation, and most processes are very well instrumented with various sensors providing both control and safety-related measurements. The automation is achieved through control systems that use the sensor measurements to make adjustments to the manipulated variables such as fluid and energy flows to maintain the process at the desired set points. Field devices such as sensors for measurements and actuators (valves, heaters, control rod drives, etc.) for process adjustment are thus central to any control system and their reliable functioning is crucial for optimal process operation. Incipient faults in the field devices such as a drift in the sensor or change in the valve characteristics can occur, adversely affecting the plant performance. Considerable research in the recent past has therefore focused on the development of methods for incipient fault detection and isolation (FDI), especially in the sensors.

The underlying philosophy in any fault detection and isolation approach is to exploit the relationships that exist between the various measurements due to constraints imposed by the process. As a simple example, the temperature and pressure in a steam generator are related through the liquid-vapor equilibrium. This nominal relationship would not be followed in case one of the sensors fails. Basic principles or data based methods can be used to model these relationships and then use the model to capture the breakdown in the relationships due to incipient fault conditions.

Given the availability of information in historical databases in modern computer controlled industry, data-based empirical modeling techniques such as GMDH[13], principal component analysis (PCA)[14], and artificial neural networks[15-17] have been applied to capture the relationships between the various measurements for fault detection and isolation (FDI). Matrix projection equations to eliminate unmeasured variables in the balance equations have been used for data reconciliation.[18] An unbiased fault magnitude estimation technique has also been proposed.[19] Other data reconciliation approaches include the use of recurrent neural networks and model based filters.[20]

Multivariate statistical process control using empirical techniques, such as PCA and partial least-squares for data characterization, have been used successfully in the process industry.[21,22] Contribution plots[23] can be used as a tool for diagnosing the variables that numerically contribute to abnormal statistics on the SPC (statistical process control) charts. Structured residuals for isolation of faults have also been proposed.[24,25] Gertler[26] provides a good overview of common model-based techniques for fault detection and isolation in engineering systems.

More recently, the use of the squared prediction errors for fault detection, an optimization approach for reconstructing the fault, and isolation by reconstructing each sensor in turn has been proposed. The method uses PCA or PLS to build the process model. Qin and Li[27] have extended the work of Gertler et al.[28] for sensor fault identification and reconstruction with maximized sensitivity. The main criticism of the work is that the methodology is tested using artificial drifts imposed on nominal data available offline. However, in real systems, the control system uses the sensor reading and makes adjustments so that the drift is transformed to the manipulated variable. This aspect of transformation of variability, which is characteristic of any industrial process, is thus completely ignored. This FDI approach, as is the case with most other reported applications, is therefore of limited utility.

Realistic methods for FDI must take into account the plant-wide nature of the problem and the flow of information, including feedback through a complex plant and non-linearities associated with plant components. The research reported here develops a quantitative framework for plant-wide fault detection and isolation in field devices.

The approach is based on the fact that data for normal operation lies on a low dimension surface in the measurement space. Faulty conditions in one or more of the field devices lead to deviations from the surface. These deviations from the surface can be used for fault detection. Fault isolation is possible when the deviations are in different directions for different faults. In other words, different fault conditions lead to different patterns in the measurement space. We refer to the various prediction error directions as the fault directions, and the particular fault may be isolated as the one with the maximum projection on the enumerated set of fault direction.

The method thus consists of modeling the normal operation plant database using empirical methods, proposing statistical limits on deviations from the surface for fault detection and extracting systematic directions of deviations from the surface as observed in the historical faulty data. The basis for FDI is the exploitation of the redundancy that exists between the process measurements. Principal component analysis (PCA) is a technique especially suitable for this purpose and is used in this work for empirical modeling of the data.

In the following sections, PCA as a *data-modeling* tool is briefly described. A fault detection algorithm using statistical limits on the residuals, and fault isolation using projections on the systematic fault directions, is formulated. A technique based on singular value decomposition (SVD)[29] for extracting the fault directions given the samples in the different fault clusters is presented. Two case studies are used to demonstrate the application of the integrated approach: (a) a PWR plant steam generator and (b) a laboratory process control loop. The latter example also illustrates augmented PCA using nonlinear terms in the data matrix for enhanced FDI capabilities. A comparison with earlier work[30] on this system, using GMDH for empirical modeling, shows the efficacy of augmented PCA with a better fault isolation. Finally, the various engineering issues and some concluding remarks are presented.

2.2 Principal Component Analysis (PCA)

PCA is an exploratory data analysis technique that has become very popular among researchers for the analysis of ill-conditioned data. Consider an (*m x n*) data matrix **X,** with *n* samples along the rows and each sample consisting of m measurements. PCA decomposes **X** into a product of scores (**T**) and orthogonal loadings (**P**) as

$$\mathbf{X} = \mathbf{T}\mathbf{P}^{\mathbf{T}} + \mathbf{E} \qquad (2.1)$$

where **E** contains the residuals. The principal components (PCs) in the successive columns of **P** are obtained such that maximum variance in **X** is explained. Thus in case the data are highly collinear, the first few PCs explain most of the variability in the data and are retained. The residuals in **E** constitute the unexplained variation in the data and contain the higher PCs that are rejected. PCA is thus a very efficient data compression technique. The scores so obtained are uncorrelated, that is, $\mathbf{T}^{\mathbf{T}}\mathbf{T}$ is a diagonal matrix. The PCA represents the rotation of the laboratory reference frame to the principal component frame in which the data appear uncorrelated. Another interpretation of PCA for data centered around the origin is that the PCs are in the direction of the major and minor axes of the ellipse (or hyper-ellipse) that contains the data. The variances of the scores represent the lengths of the axes. For our purpose, it is sufficient to note that PCA represents a low dimensional hyperplane that best fits the data in the highly redundant measurement space. The PCs can easily be obtained as the right singular vectors of **X** using the singular value decomposition (SVD).[29]

A new sample **x** can be projected on an existing PCA model to obtain the scores

$$\mathbf{t} = \mathbf{x}\mathbf{P}$$

and the residuals as

$$\begin{aligned} \mathbf{e} &= \mathbf{x} - \mathbf{t}\mathbf{P}^{\mathbf{T}} \\ &= \mathbf{x} - \mathbf{x}\mathbf{P}\mathbf{P}^{\mathbf{T}} \\ \mathbf{e} &= \mathbf{x}(\mathbf{I} - \mathbf{P}\mathbf{P}^{\mathbf{T}}) \qquad (2.2) \end{aligned}$$

2.3 FDI Methodology

The FDI methodology consists of building a model to characterize the relationships between the various measurements. A breakdown in the relationships is indicated as abnormal prediction residuals or lack of fit with the model and may be used for fault detection. Characteristic patterns in the residuals most probably correspond to faults of a particular type and may be used for fault isolation. It is emphasized that the above is true irrespective of the type of model that is used. More formally, given a model, the detection and isolation of faults may be achieved using the residuals that signify the mismatch between the model and the actual data as follows.

2.3.1 Fault Detection

The residuals **e** obtained from a model can be combined into a squared prediction error statistic Q as

$$\mathbf{Q} = \mathbf{ee^T} \qquad (2.3)$$

The Q statistic quantifies the lack of fit between the sample and the model and denotes the distance of the sample from the nominal operation surface. Under the standard assumptions of a multivariate normal distribution for **e**, **e** ~ N(**0**, **1**), control limits at confidence level a may be obtained using the ?^{2} distribution as[30]

$$Q_a = T_1\left[\frac{c_a\sqrt{2T_2h_0^2}}{T_1} + 1 + \frac{T_2h_0\left(h_0 - 1\right)}{T_1^2}\right]^{\frac{1}{h_0}} \qquad (2.4)$$

where

$$\Theta_i = \mathrm{trace}\left(\frac{\mathbf{EE^T}}{I-1}\right)^{i} \text{ for i} = 1, 2, 3 \qquad (2.5)$$

$$h_0 = 1 - \frac{2\Theta_1\Theta_3}{3\Theta_2^2} \qquad (2.6)$$

and c_a is the standard normal deviate (2.57 for α = 0.01). In the above equations, I is the number of samples (rows) in the normal operation set and Θ_i is the sum of the ith power of the singular values of **E**/(I-1)$^{0.5}$. For normal operation, the Q-statistic would be very small. Detection of faults is accomplished as violations of the control limit, that is, when Q exceeds Q_α.

2.3.2 Fault Isolation

Let $\mathbf{F} = [\mathbf{f}_1\ \mathbf{f}_2\ \ldots\ \mathbf{f}_R]$, where $\mathbf{f}_1\ \mathbf{f}_2\ \ldots\ \mathbf{f}_R$ are *(m x 1)* column vectors, denote the fault directions for the various fault scenarios that are observed in the database. Clustering techniques on an existing database can be used to automatically obtain these fault directions. In case the fault is of a particular type j, then the projection of the residuals on $\mathbf{f}_j$ would be very high. Fault isolation is accomplished by calculating the projections onto **F** and classifying the fault as the one with the maximum projection norm. For example, a fault isolation index for the i^th fault is defined as

$$FI_i = 1 - Q_i/Q \qquad (2.7)$$

where

$$Q_i = \mathbf{e}(\mathbf{I}-\mathbf{f}_i\mathbf{f}_i^{\mathbf{T}})(\mathbf{I}-\mathbf{f}_i\mathbf{f}_i^{\mathbf{T}})\mathbf{e}^{\mathbf{T}} \qquad (2.8)$$

In the above equations, Q_i denotes the distance of the sample from the origin after subtracting the projection of the residuals on the fault direction $\mathbf{f}_i$. It represents the sum of squares of residuals remaining after removing the contribution from the i^th fault direction. The fault isolation index quantifies the fraction of Q that is due to $\mathbf{f}_i$. If the j^th fault scenario occurs, FI_i would be the largest and nearly 1 for i = j where i varies from 1 to R. This results in the isolation of the fault from the various existing scenarios. The fault matrix **F** can be extracted from the historical data as described in the next section.

2.4 Establishing Fault Directions

The fault direction $\mathbf{f}_i$ in the matrix $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_i, \ldots, \mathbf{f}_R]$ represents the direction in the residuals space for the i^th fault such that the samples corresponding to the fault have the maximum projection on $\mathbf{f}_i$. In other words, if $\mathbf{E}_i$ denotes the residuals for samples corresponding to the i^th fault, the optimization problem is

$$\mathbf{J} = \max_{\mathbf{f}_i} \mathbf{f}_i^{\mathbf{T}}\mathbf{E}_i^{\mathbf{T}}\mathbf{E}_i\mathbf{f}i \qquad (2.9)$$

subject to the constraint

$$\mathbf{f}_i^{\mathbf{T}}\mathbf{f}_i = 1$$

Using the Lagrangian multiplier and differentiating J with respect to $\mathbf{f}_i$ and setting the derivative to zero for maximization, we get

$$2EiTEifi - 2\sigma fi = 0$$
$$\mathbf{E}_i^{\mathbf{T}}\mathbf{E}_i\mathbf{f}_i = \sigma\mathbf{f}_i \qquad (2.10)$$

The fault direction $\mathbf{f}_i$ is thus obtained as the first eigenvector of $\mathbf{E}_i^{\mathbf{T}}\mathbf{E}_i$. The singular value decomposition (SVD)[29] may be used to obtain the eigenvector.

In historical databases, the number of fault scenarios and the correspondence of the fault samples to the particular fault scenarios are not known *apriori*. Classification of the residuals must then be used to establish the number of fault scenarios and also the correspondence of the samples to the particular scenarios. Existing classification algorithms[24] can be suitably adapted for this purpose.

The overall method as described in the above sections thus consists of the following steps:

1. Building a model for historical normal operation data.
2. Subjecting the complete database to the monitoring scheme to identify samples that deviate abnormally from the model and are therefore faulty.
3. Classifying the abnormal (faulty) samples in the database to find the fault directions.

Once a reasonable model is built, the FDI scheme can be implemented for the detection and isolation of faults in new samples. The next section applies the methodology to two examples: 1. A simulated nuclear power plant steam generator system, and 2. A laboratory process control loop.

2.5 FDI Case Studies

2.5.1 A PWR Plant U-Tube Steam Generator (UTSG) System

The simulation developed by Naghedolfeizi and Upadhyaya[29] is used in this work to demonstrate the implementation of the FDI system. The focus of the study is the U-tube steam generator (UTSG) in a typical 1,140 MWe PWR plant and the propagation of faults in the sensors or valves associated with it. Schematic of a typical UTSG and the associated measurements are shown in Figure 2.1. In order to build the model for nominal operation, 500 cases are simulated for randomly selected power set points between 70 to 100% of the plant capacity and also in the level set point in the steam generator. The 16 measurements that are available are listed in Table 2.1. The fault cases, shown in Table 2.2, are simulated with randomly selected drifts. The range of the drifts is also given in the table. A total of 120 (20 per fault type) single fault cases are simulated.

A PCA model is built on the data for the nominal operation case. The nominal operation data matrix is pre-processed by auto-scaling the columns in the data matrix to zero mean and unit variance. This puts all the measurements with their different units on a common unit variance scale. The results of analysis are plotted in Figure 2.2. The eigenvalues are shown in subplot 2.2a. Seven PCs are retained, and they explain about 99.9% of the total variation in the data, indicating the high degree of redundancy in the measurements. The Q-statistic for the samples in the complete database is plotted in Figure 2.2b. All the samples with faults, except for the case of drifts in the level sensor, are detected on the SPC monitoring chart. This is an expected result and indicates that for the particular set of measurements, the correlation structure between the measurements remains the same as for normal operation even when a drift in the level sensor occurs. The reason is that there are no other measurements that are highly correlated with the level. The absence of such alternative measurements makes drifts in the steam generator inventory *unobservable*. For the other cases, alternate highly correlated measurements are available so that the faults are *observable*. The residuals for each of the observable fault scenarios are plotted in subplot 2.2c. Each of the fault scenarios has unique pattern. The corresponding fault directions obtained from an SVD on the faulty samples are plotted in 2.2d.

The fault isolation indices for each of the detected fault scenarios are plotted in Figure 2.3. Note that the fault index is close to unity for the particular fault that occurs and is smaller for the others. All the fault cases are detected correctly using the fault isolation approach described earlier. This demonstrates the effectiveness of the proposed method for fault detection and isolation in complex systems with several interacting units and feedback control loops.

2.5.2 Laboratory Process Control System

A schematic of the laboratory process loop with a tank water level control system is shown in Figure 2.4. The 12 measurements that are available for the system are listed in Table 2.3 and the various fault types are given in Table 2.4. The normal operation database consists of 500 samples at various randomly chosen set points for the water level, and positions of outlet and bypass motor operated valves (MOVs). Fifty samples each, for the various fault scenarios, are also simulated.

Using the linear PCA on the measurement data and retaining six PCs, the Q statistic detects five of the six fault types. The fault corresponding to a drift in the level sensor goes undetected. This is not reasonable since the tank level significantly affects the flow out of the tank. One possible reason is the non-linearity present in the system, and the linear PCA does not properly characterize the relationships among the various measurements. Indeed, in the previously reported application[12], GMDH (*group method of data handling*) was used with non-linear rational terms to achieve good prediction models, indicating the presence of significant non-linearity in the system. One simple modification to incorporate non-linearity in the PCA analysis is to augment the data matrix with non-linear terms such as higher order terms and rational functions of the measurements. See Section 3 for details of GMDH method.

In order to decide the terms to be used for augmenting the matrix, we note that a drift in the level sensor causes the actual level in the tank to be different than under nominal operation. Since level affects the flow out of the tank, a mismatch in the predicted versus measured flow out would occur when a level sensor drift is present. For nominal operation, good predictions of the flow out can be obtained using the level set point (x_7) and the measured outlet MOV position (x_3). The measurement matrix is thus augmented using the nonlinear terms x_3^2, x_7^2, $x_3.x_7$ and x_3/x_7. The last term is rational and is used since previous experience with GMDH had shown that the incorporation of such terms significantly enhanced the prediction accuracy.[12] Note that division by zero must be avoided by adequately scaling the data between 0.5 and 2.0. Once all the columns in the data matrix are defined, the nominal operation data is auto-scaled to obtain columns with unit variance and zero mean values.

A total of seven PCs are retained to model the augmented normal operation data. The Q-statistic is obtained and plotted for all the samples in Figure 2.5. All the six fault types are now detected. A single false alarm is observed while 50 faulty samples are not detected. These results are better than for the case of linear PCA.

As a comparison, GMDH models (described in earlier work[12]) with four predicted variables are developed. Using the mismatch between the measured and predicted values for the four variables, the residuals are obtained. The residual patterns for the fault cases are plotted in Figure 2.6. Only three unique patterns for the six scenarios are observed. The patterns for two pairs of faults (1 and 5; 2 and 4) are very similar so that isolation between the pairs is not possible as the fault directions are shaped similarly (collinear). Also, 78 faulty samples go undetected. The better performance of augmented PCA, both in terms of detection and isolation, is due to the fact that it is truly a multivariate technique that exploits the correlation structure of *all* the measurements rather than using only a few measurements for prediction, as was the case with GMDH. The augmented PCA is thus a more powerful FDI technique. This completes the demonstration of the FDI method for the example cases. Several issues relevant to the implementation of the method are discussed in the next section.

2.5.3 Engineering Issues

We described a method for fault detection and isolation of field devices using structured residuals from historical data and demonstrated using engineering systems as examples. The availability of a fault free database is implicitly assumed. In practice, even though a historical database may be available, the distinction between faulty and fault-free samples is generally not very clear. A few iterations in the model building exercise may be needed before the distinction becomes clear.

Once a nominal operation model is built, the various fault scenarios are easily obtained. The challenge, however, lies in pinpointing the physical cause for each of the scenarios. A substantial engineering judgement is involved in the exercise and the effort spent is probably the most useful for process improvements. Simulations that approximately represent the system behavior can be very instructive for this purpose. Most engineering systems have certain characteristic failure modes so that the particular signature of a fault in the measurement space may be obtained from the simulation. For practical applications, a very powerful hybrid approach for FDI would utilize historical data mining in conjunction with approximate simulations.

The FDI approach uses empirical data-based modeling technique for characterizing the system under study. It is therefore applicable only in the operating region for which data were used to build the model. Rich data sets with a wide operating region should therefore be used for building the model. New faults with a distinct signature that have not been characterized earlier can occur and would lead to low projections on all the fault directions. In such situations, the enumerated fault set can be updated to include the new fault direction so that the fault can be correctly isolated in the future. It is also noted that the fault isolation results indicate the most likely cause of a particular abnormal deviation based on past experience. Faults due to other physical reasons that have a similar effect on the measurements are also possible and the inferences from the isolation module are at best tentative and require verification.

We emphasize that the sensor-alone type validation, for which a significant body of literature exists, and the plant-wide FDI as demonstrated here are complimentary in nature. The former is useful at the device level in detecting device faults quickly. The latter is useful for anomalies that go undetected at the device level and propagate through the plant to be detected by the plant-wide system. The plant-wide FDI system is thus a useful tool for the detection of subtle faults.

For the plant-wide FDI problem, multivariate methods such as PCA that study the covariance structure of *all* the variables as a whole, are very powerful. The only issue, as was shown in the level control example, is that the technique is linear and therefore not adequate for highly non-linear systems. This research has shown that augmenting the measurement matrix with non-linear terms and then performing PCA on the augmented matrix is an effective way of enhancing the non-linear PCA. Alternatively, more rigorous methods such as principal curves[26] may be used and is a recommended goal for future research.

Lastly, the framework at present is applicable to steady state data. It can be applied only after the process settles down (or during slow drifts) and not during transients. One notable exception is that of redundant sensors where the correlation structure between the sensor measurements is valid even during transients. The present method can be extended to transient FDI by including time-lagged measurements in the data matrix to capture the normal dynamic signature of a process. This is referred to as dynamic PCA and should be an effective tool for FDI since the first sign of field device degradation is quite often a change in its dynamic response. The training data should be rich in dynamic information so that the transient relationships are captured well by the PCA model. Data from periods of startup, shutdown and operational changes are especially suitable for this purpose. A logical direction for future work would be the extension and demonstration of the current framework to transient FDI.

Table 2-1 Nuclear plant steam generator system variables

Index	Description	Index	Description
1	SG Level	9	T_{avg} Controller Output
2	SG Pressure	10	Pressurizer Level
3	SG Steam Flow	11	Pressurizer Pressure
4	SG FW Flow	12	Primary Side PC Output
5	SG LC Output	13	Pressurizer LC Output
6	Reactor Reactivity	14	Turbine Power
7	Hot Leg Temperature	15	Turbine RPM
8	Cold Leg Temperature	16	RPM Controller Output

SG: steam generator; FW: feed water; PC: pressure controller; LC: level controller.

Table 2.2. Steam generator fault cases

Index	Description	Drift Magnitude (Sign)
1	Level sensor	0.5 – 2% (-ve)
2	Pressure sensor	0.5 – 2% (-ve)
3	Steam flow sensor	0.5 – 2% (+ve)
4	FW flow sensor	0.5 – 2% (+ve)
5	FW valve signal	0.5 – 2% (+ve)
6	Steam valve signal	0.5 – 2% (+ve)

FW: feed water

Table 2-2 Laboratory process control system variables

Index	Description	Index	Description
1	Bypass MOV SP	7	Water level SP
2	Inlet MOV SP	8	Water level
3	Outlet MOV SP	9	Controller output
4	Bypass MOV position	10	Inlet flow (TFM)
5	Inlet MOV position	11	Inlet flow (RMT)
6	Outlet MOV position	12	Outlet flow (TFM)

MOV: motor operated valve; SP: set point; TFM: turbine flow meter;

RMT: Rosemount mass flow transmitter

Table 2.4. Laboratory tank level control system fault cases

Index	Description	*Drift Magnitude (Sign)*
1	Inlet MOV position	5 – 10% (+ve)
2	Level sensor	5 – 10% (+ve)
3	Inlet flow RMT sensor	5 – 10% (+ve)
4	Outlet MOV position	5 – 10% (+ve)
5	*Bypass MOV position*	5 – 10% (+ve)
6	Outlet flow TFM sensor	5 – 10% (+ve)

MOV: motor operated valve; SP: set point;

TFM: turbine flow meter;

RMT: Rosemount mass flow transmitter

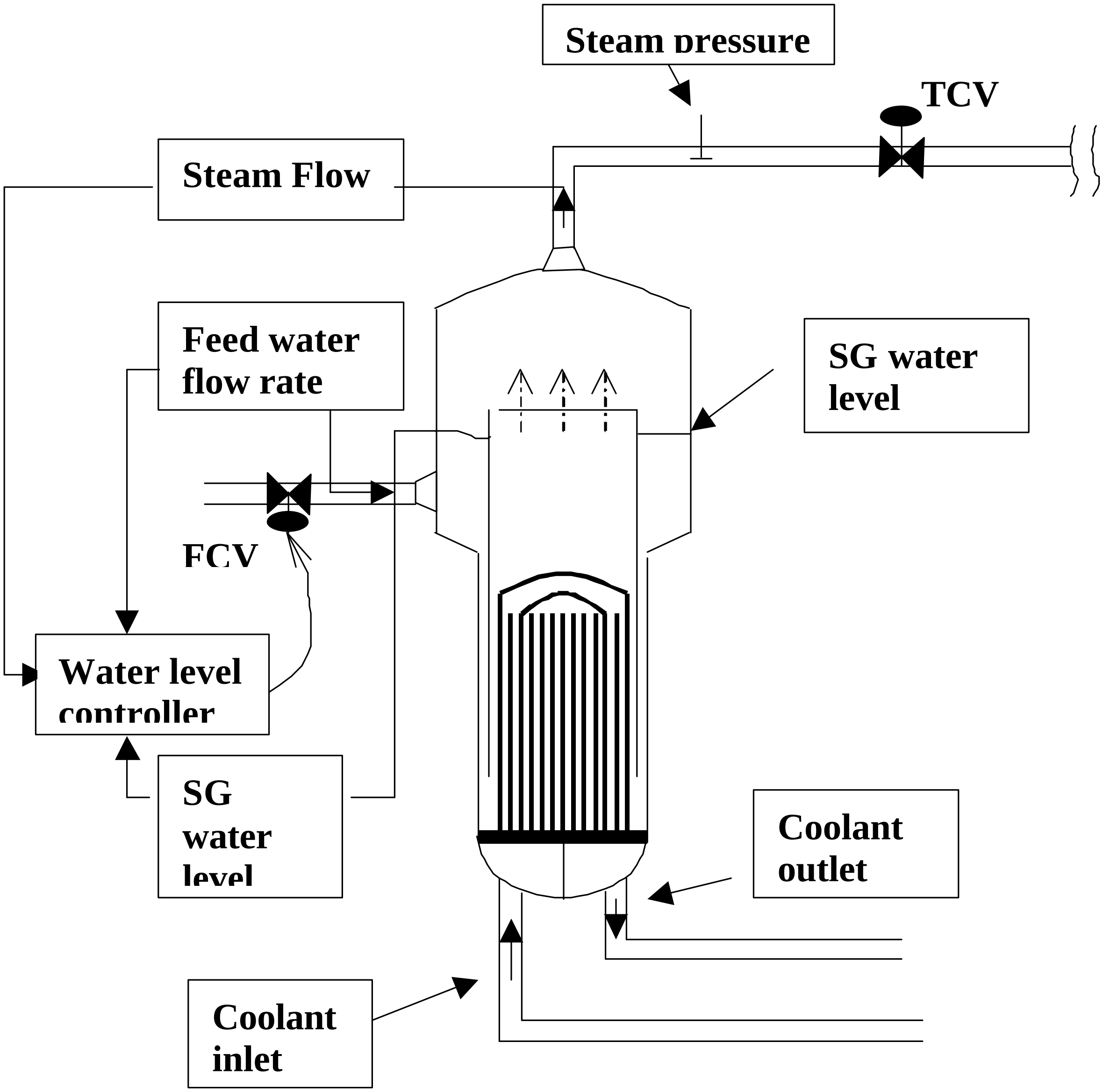


Figure 2.1 Schematic of a typical steam generator system (UTSG) showing the various

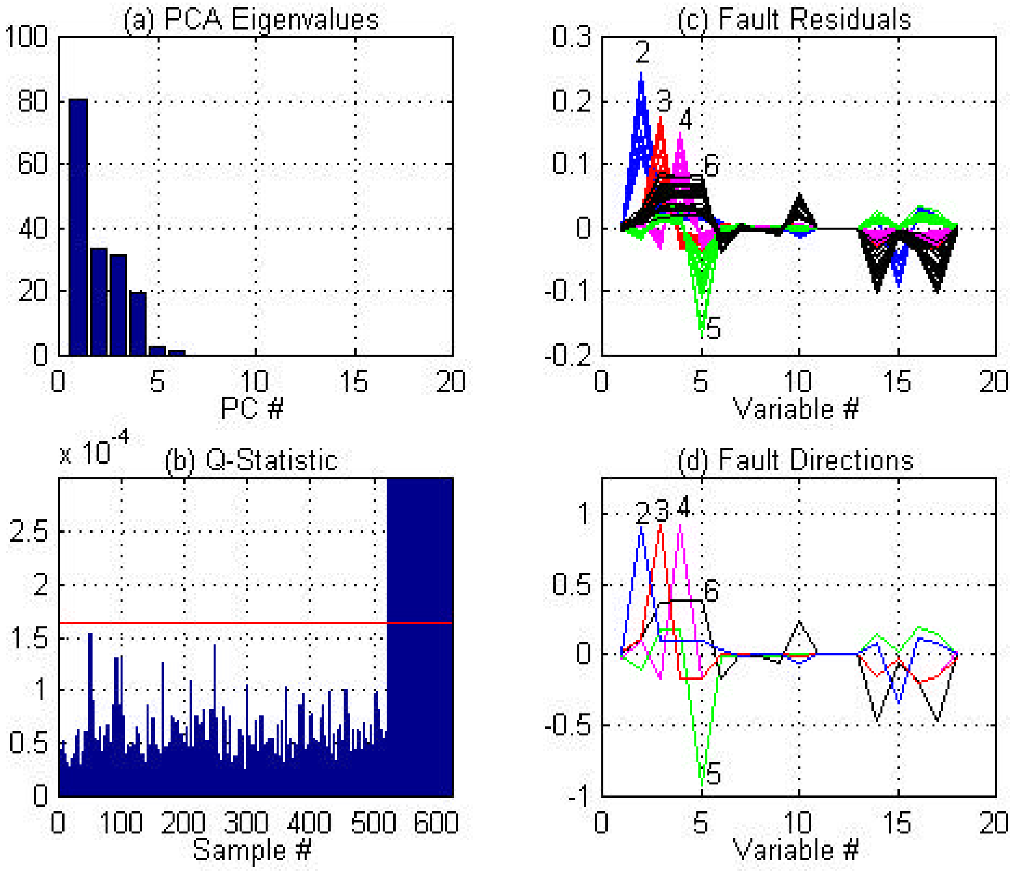


Figure 2.2 PCA results: A PWR U-tube steam generator

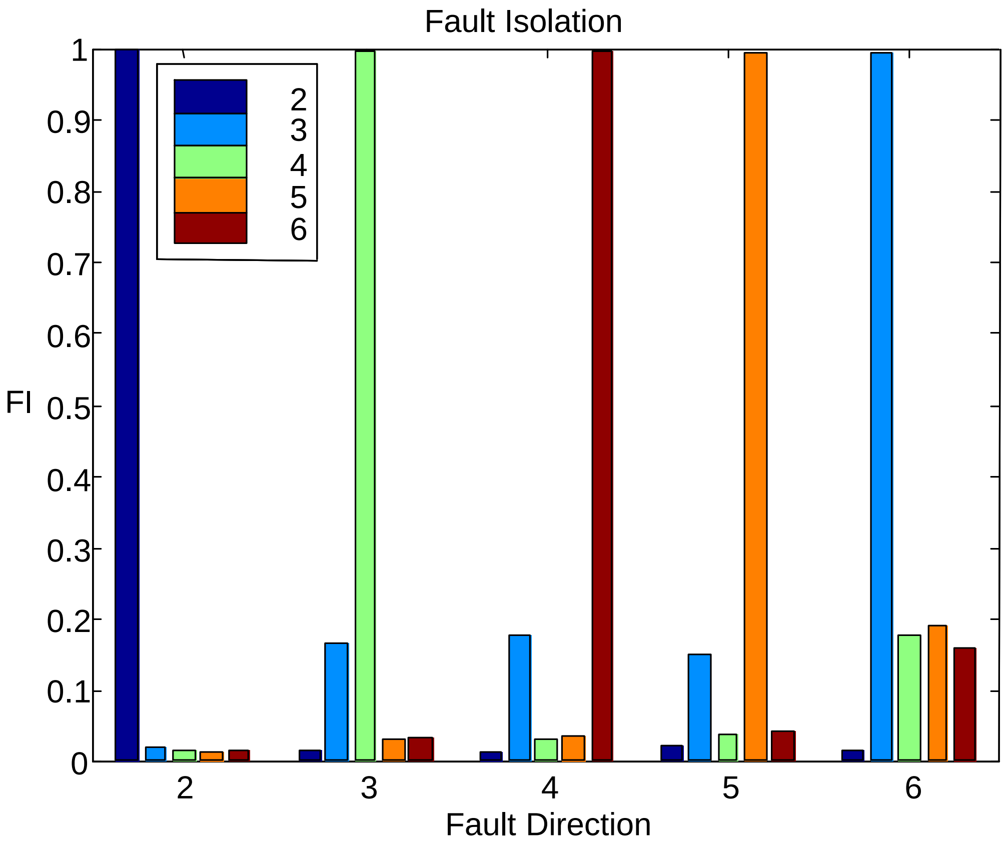


Figure 2.3 Fault isolation index for an example sample for each detectable fault case: steam generator example

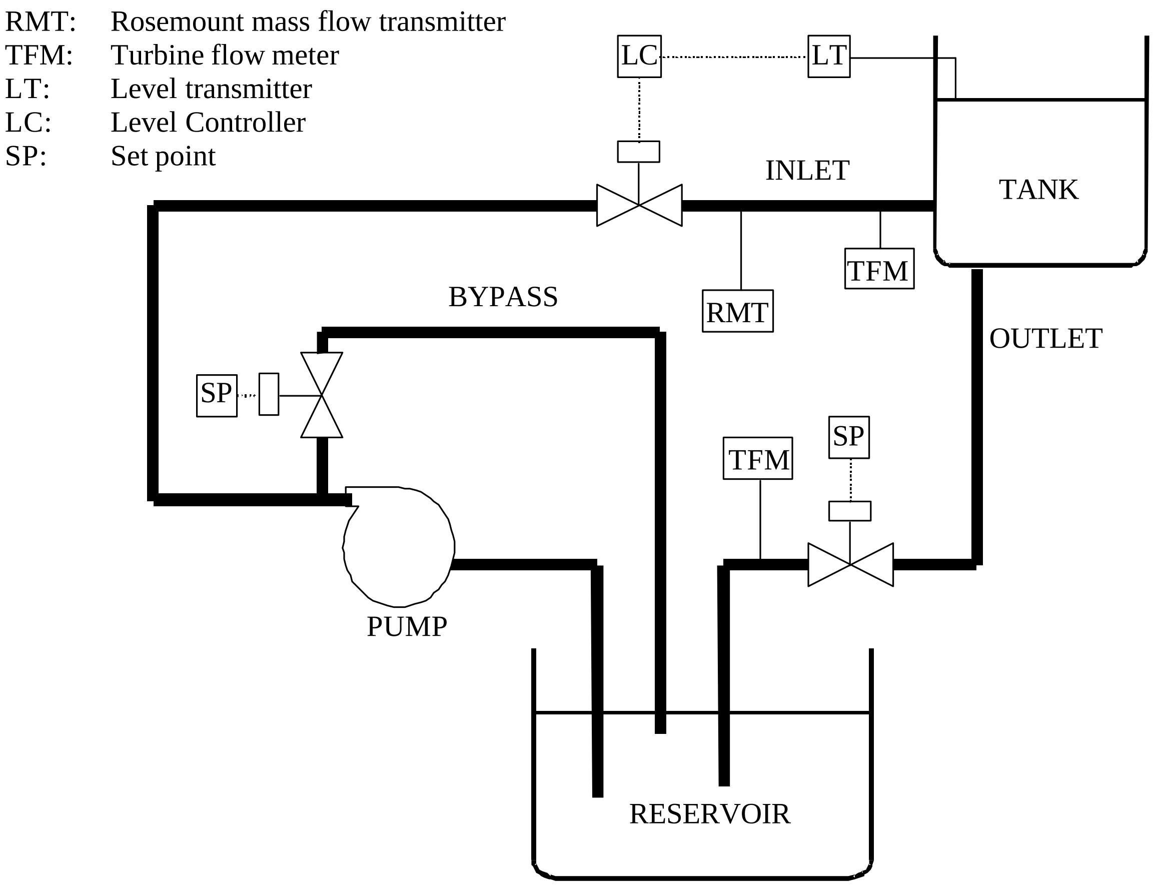


Figure 2.4 Schematic of a tank level control system

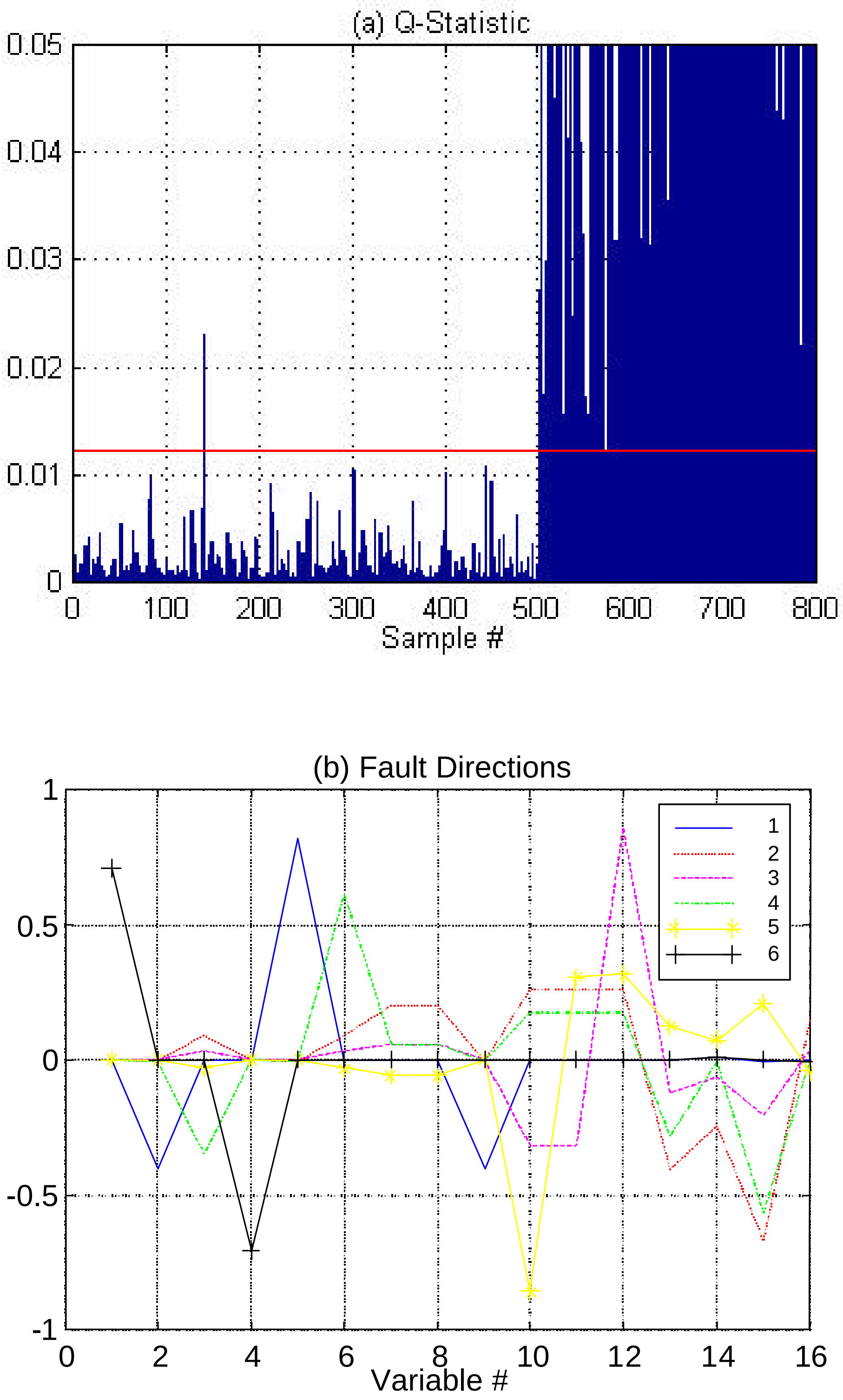


Figure 2.5 Augmented PCA results: Tank level control example

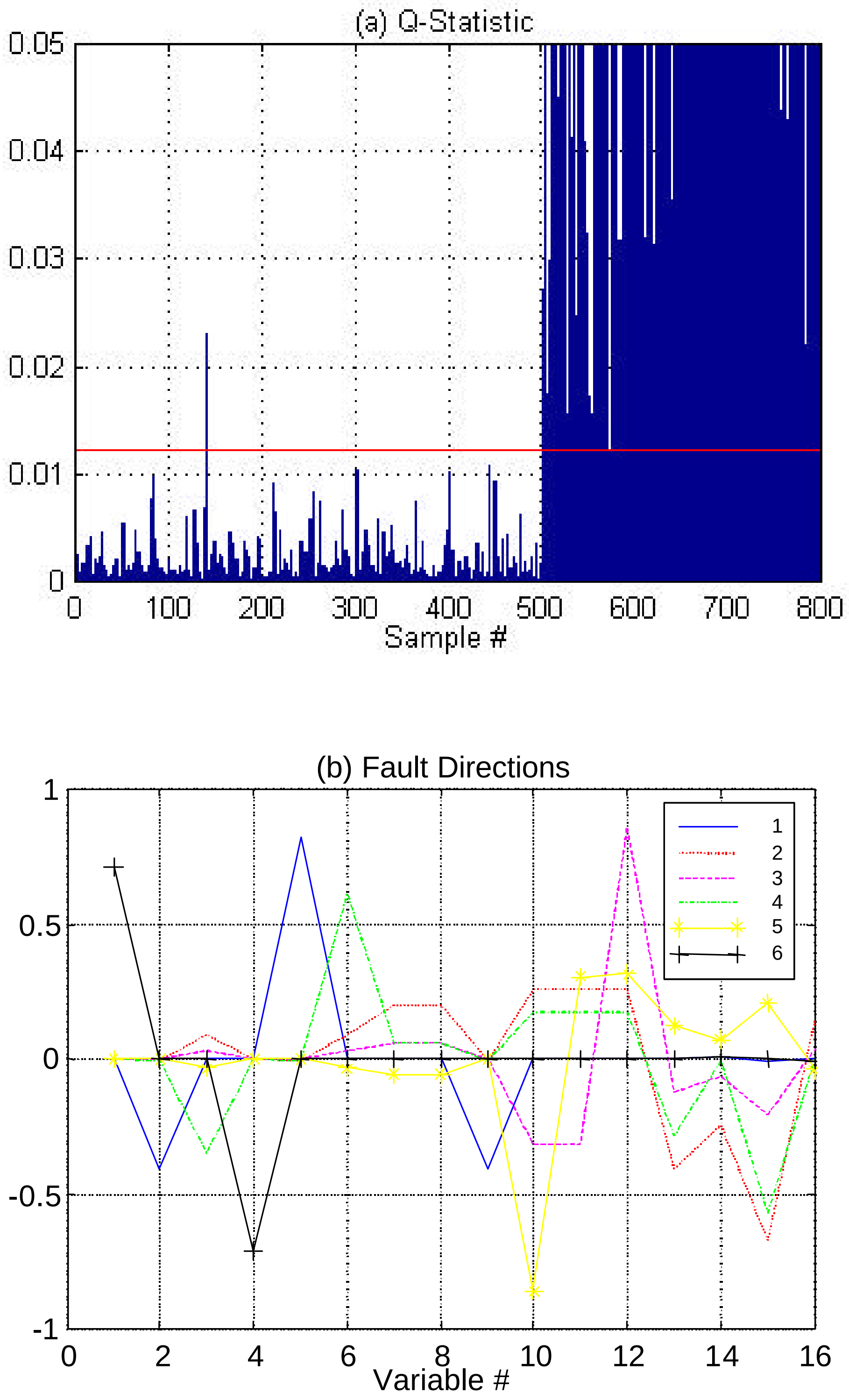


Figure 2.6 Augmented PCA results: Tank level control example

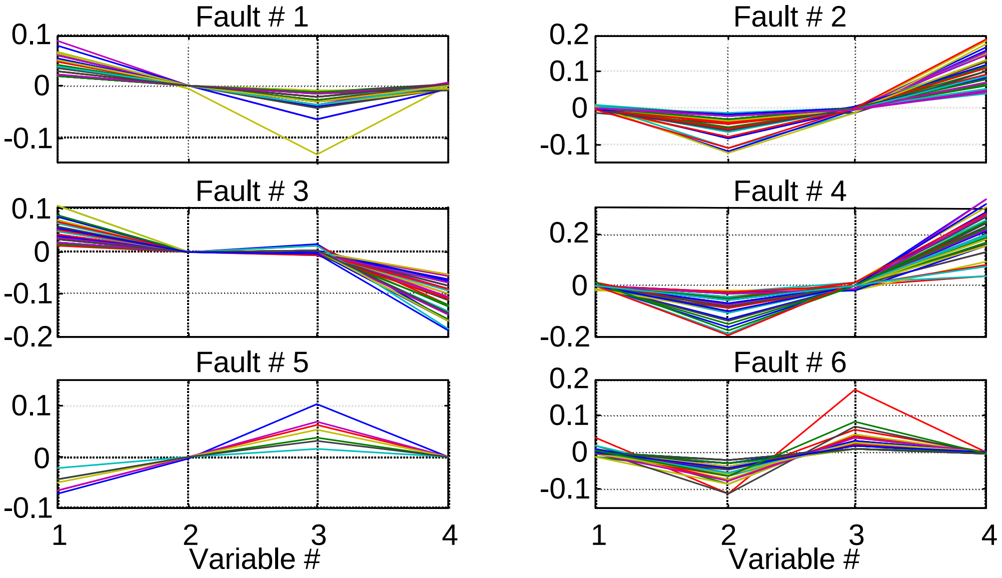


Figure 2.7 Fault residuals for GMDH: Tank level control example.

3. ENHANCED GROUP METHOD OF DATA HANDLING (GMDH) FOR TRANSIENT PROCESS CHARACTERIZATION

3.1 The GMDH Method with Rational Function Approximation

The objective of this sub-task is to characterize the mapping among process variables and control functions using self-organizing and data-driven modeling. The so-called *Group Method of Data Handling (GMDH)* is an algebraic method for predicting system states, controller and actuator functions. A new algorithm, that will create appropriate prediction models for different nuclear plant sub-systems, will be developed by a rational function approximation of the original GMDH algorithm [11,12]. The GMDH approach has the advantage over artificial neural networks in not requiring tedious network training procedures. It is also easy to update the prediction models during plant operation. Extension of the GMDH to fault detection during plant transient has been developed during Phase-2.

The GMDH constructs a model, of a desired output as a function of a set of related inputs from a subsystem, by a successive polynomial approximation. The general relationship has the form shown in Equation (2.1) where $\{x_1, x_2, \dots, x_m\}$ is a vector of input variables and y is the variable to be predicted. This formulation can be extended to the prediction of multiple outputs $\{y_1, y_2, \dots, y_n\}$ as well.

$$y = a + \sum_{i=1}^{m} b_i x_i + \sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij} x_i x_j + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m} d_{ijk} x_i x_j x_k + \cdots \tag{3.1}$$

A typical node of a GMDH modeling layer is a basic quadratic predictor using variables $[x_i, x_j]$. The model parameters such as {A, B, C, D, E, F}, are estimated from a least-squares fit using **N** observations of the input and output variables.

$$y = A + Bx_i + Cx_j + Dx_i^2 + Ex_j^2 + Fx_i x_j \tag{3.2}$$

Figure 3.1 illustrates that the predicted values of *y* are propagated to successively higher layers of the algorithm, with the approximation of y_{pred} improving at successive stages. At each stage of the approximation, y_{pred} is formed from pairs of input signals (to that layer), and new values of the predicted variable are propagated pair-wise to the next layer. The iteration is continued until the mean-squared error between the predicted and the measured values of the output variable attains a desired value.

Parsimony in model fitting is achieved by comparing the fractional prediction errors from one generation to the next, and by terminating the algorithm when the error is a minimum or when the difference between errors from successive approximation stages is less than a preset limit [12].

The GMDH approach described above uses polynomial approximation. This polynomial set may be satisfactory in establishing some of the relationships of interest. In characterizing the subsystems in a nuclear power plant it may be necessary to use terms containing rational functions (for example, ratios of polynomials in x_1 and x_2). The expression (2.3) represents a set of such terms that forms a *complete* set of terms in a given domain.

$$\left\{\begin{array}{l} 1,(x_1,x_2),(x_1^2,x_2^2),(x_1x_2),(\frac{1}{x_1},\frac{1}{x_2}),(\frac{1}{x_1^2},\frac{1}{x_2^2}),(\frac{1}{x_1+x_2},\frac{1}{x_1x_2}),(\frac{x_1}{x_2},\frac{x_2}{x_1}), \\ (\frac{x_1}{x_1+x_2},\frac{x_2}{x_1+x_2}),(\frac{x_1+x_2}{x_1},\frac{x_1+x_2}{x_2}),\ldots \end{array}\right\} \qquad (3.3)$$

The new set should facilitate the development of prediction models with a minimum number of terms. The computational efficiency of establishing these models will be enhanced by a systematic choice of the terms in the set shown in Expression (2.3).

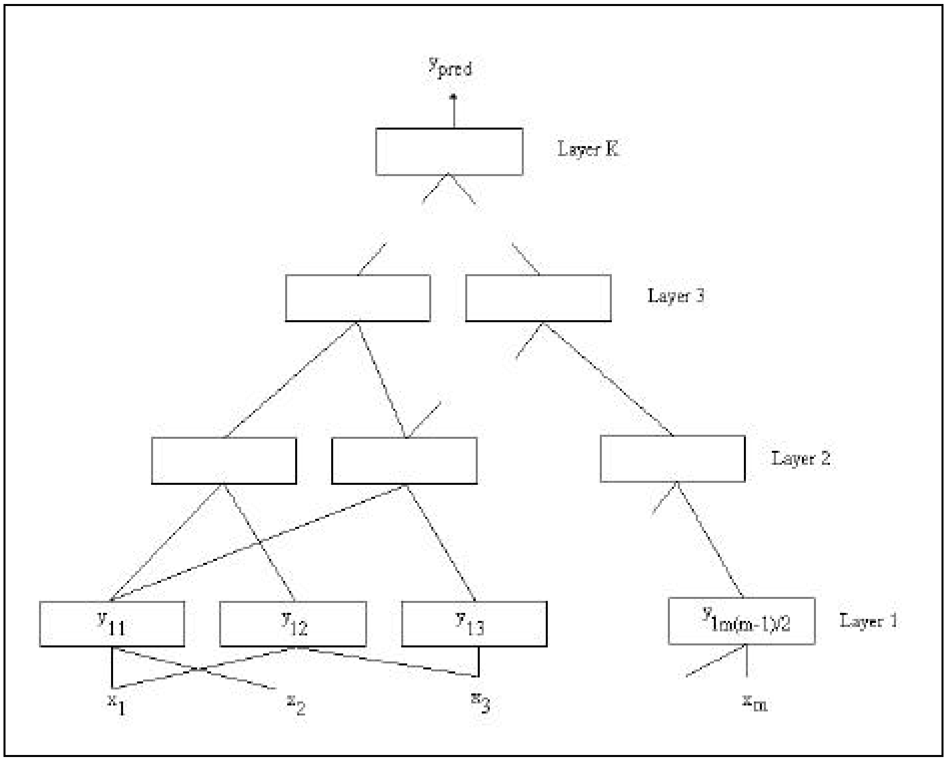


Figure 3.1 GMDH network showing m inputs and K layers

3.2 The GMDH Algorithm

The following steps explain the procedure used in developing data-driven models using the group method of data handling.

- Consider *N* observations of *m* variables $X \equiv \{x_1, x_2, \ldots, x_m\}$ and the measurements of the variable to be estimated, $Y \equiv \{y_1, y_2, \ldots, y_N\}$.

- Divide the data into a training set (*nt*) and a test set (*N*-*nt*).

- For each pair $\{x_i, x_j\}$ and Y, compute the regression polynomial

$$y = A + Bx_i + Cx_j + Dx_i^2 + Ex_j^2 + Fx_i x_j$$

A total of m(m-1)/2 polynomials are computed.

- Create new observations, Z, for each of the new m(m-1)/2 variables.
- Screening out the least effective variables: Compute the SSE

$$r_j^2 = \frac{\sum_{i=1}^{nt} (y_i - z_{ij})^2}{\sum_{i=1}^{nt} y_i^2} \tag{3.4}$$

- Pick those new *inputs* for which $r_j < R$ (choice of the user).

- Repeat the stage-wise computation until the method starts over-fitting the data. Plot the smallest of $\{r_j\}$ at each stage and look for a *minimum*. This is called the minimum *Ivakhnenko* polynomial.

- Using the best-fit model, compute the prediction errors using the test data of length (*N*-*nt*). Check if the error r_{best} is satisfactory.

3.3 Enhancement of the GMDH Algorithm

To improve model building with a minimum number of layers, the set of terms in the regression model is generalized to include rational functions of $\{x_1, x_2, \ldots, x_m\}$.

- The choice of terms in the regression is made according to a binary selector: For example, for k=8, the binary number is between 0 and 255 (a total 256 input vectors).

- *Example:* model number 179 has the terms [1 0 1 1 0 0 1 1]
- Choose ~ ten best-fit models. From this set, choose the model with the least number of terms!

- To avoid unlimited increase in the number of nodes in a higher GMDH layer, use the best m nodes for the succeeding layer. *All layers have the same number of nodes, m.*

- Make sure that the number of input variables in the first layer is $m > 2$, in order to avoid the termination of GMDH after the first (input) layer.

- To avoid long training times, limit the maximum number of layers for a single model (30 was suggested in this study, since no improvement was observed beyond this level).

- Transient models were developed by incorporating time-correlated data as part of the measurement set.

3.4 Summary of Accomplishments during Phase-2

The following key tasks were accomplished in the development and application of GMDH models.

- Generation of PWR database using the full-scope PWR simulator developed by NCSU. Steady-state data cover 20% - 100% power level.
- Transient data were generated between 80% and 100% power levels.
- Development of temporal GMDH models for transient data.
- Detection and isolation of faults in sensors, feed water and turbine control valve position, and valve actuation time constant.
- Integration of rule-base and pattern classification techniques for fault isolation.

The method was demonstrated by application to the UTSG system. The following variables were used for developing GMDH models:

x_1 = SG water level
x_2 = Feed water flow rate
x_3 = Feed control valve position
x_4 = Steam flow rate (for one steam generator)
x_5 = TCV1 flow rate
x_6 = TCV2 flow rate
x_7 = TCV3 flow rate
x_8 = TCV4 flow rate
x_9 = Reactor power level
TCV: Turbine Control Valve

3.5 Summary of Results of Application to the UTSG System

3.5.1 Fault Detection During Steady-State Operation

The following single device faults were diagnosed:

- TCV1 fault during valve open stroke (increased time constant and dead band).
- TCV1 fault during valve close stroke.
- SG level sensor (offset and drift).
- Feed water flow meter (offset and drift).
- FCV dead-band error.
- Steam pressure sensor (offset).
- Steam flow meter (offset).

The following GMDH models were developed in order to predict the device degradation.

- Narrow range level sensor = f(x_9, x_4, x_2)
- Feed water flow rate = f(x_9, x_4, x_3)
- FCV position = f(x_9, x_4, x_1, TCV1 position)
- Steam flow rate = f(x_9, x_3)
- TCVi flow rate = f(x_9, x_4, x_2, x_1, TCVi position), i = 1, 2, 3, 4.

Figure 3.2 shows a comparison of the feed control valve (FCV) position and the GMDH model prediction. Model error is less than 0.5% for this case. The following eight fault cases were demonstrated.

Fault type 1: One TCV under degradation under valve opening process (time constant)
Fault type 2: One TCV under degradation under valve closing process (time constant)
Fault type 3: Water level sensor (Narrow range) fault (offset or drifting).
Fault type 4: Flow meter sensor fault (offset)
Fault type 5: FCV stuck at 70% position.
Fault type 6: FCV dead band error.
Fault type 7: Steam pressure sensor drifting (5% per hour)
Fault type 8: Steam flow rate sensor drifting.

Figure 3.3 shows the prediction errors of the seven variables for the eight different faults.

variable 1: error between NR model prediction and measurement
variable 2: error between Feed water flow rate model prediction and measurement
variable 3: error between steam flow rate model prediction and measurement
variable 4: error between TCV1 flow rate model prediction and measurement
variable 5: error between TCV2 flow rate model prediction and measurement
variable 6: error between TCV3 flow rate model prediction and measurement
variable 7: error between TCV4 flow rate model prediction and measurement.

The residual pattern due to FCV dead band change is plotted separately in Figure 3.4.

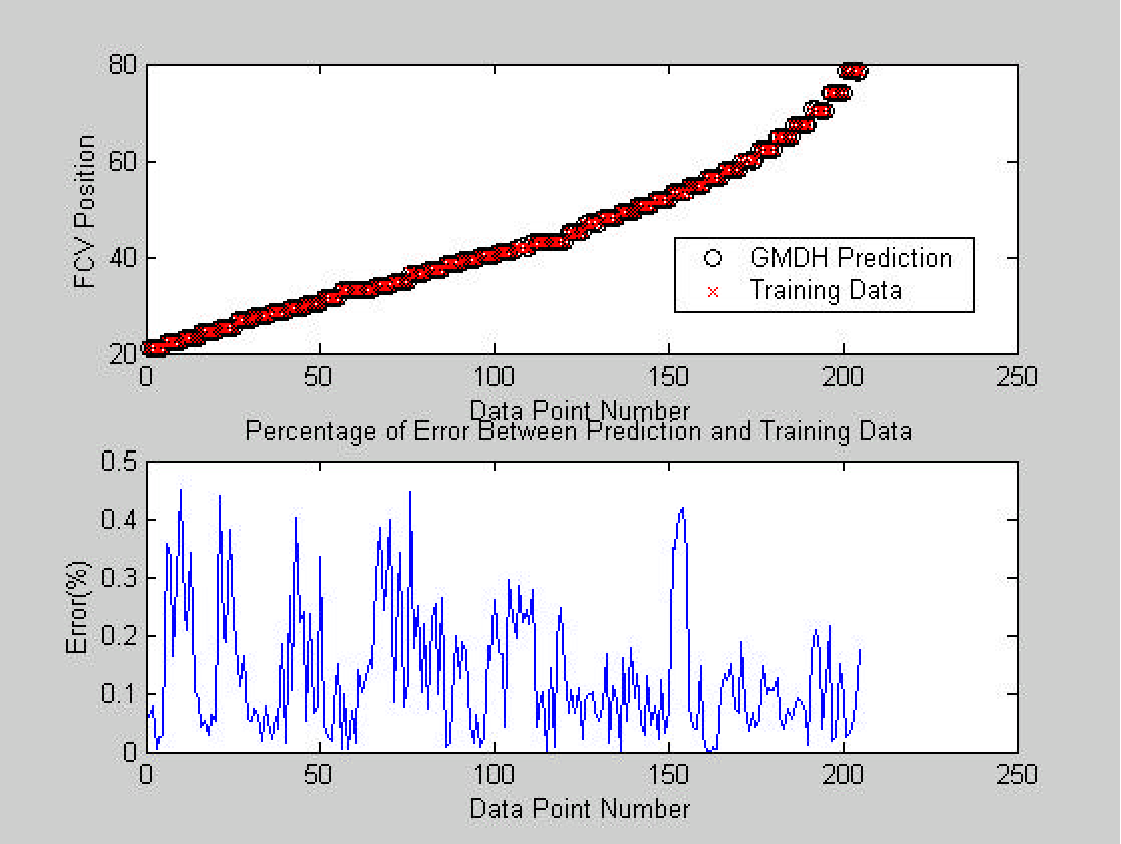


Figure 3.2 Static GMDH model training results for feed control valve (FCV)

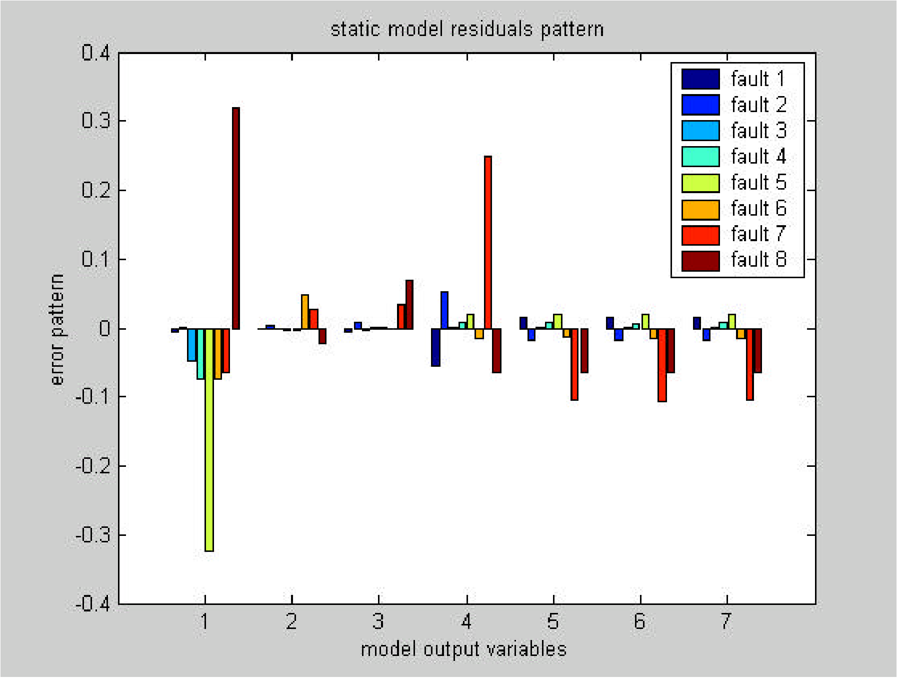


Figure 3.3 Plots of residual errors for the eight different device faults

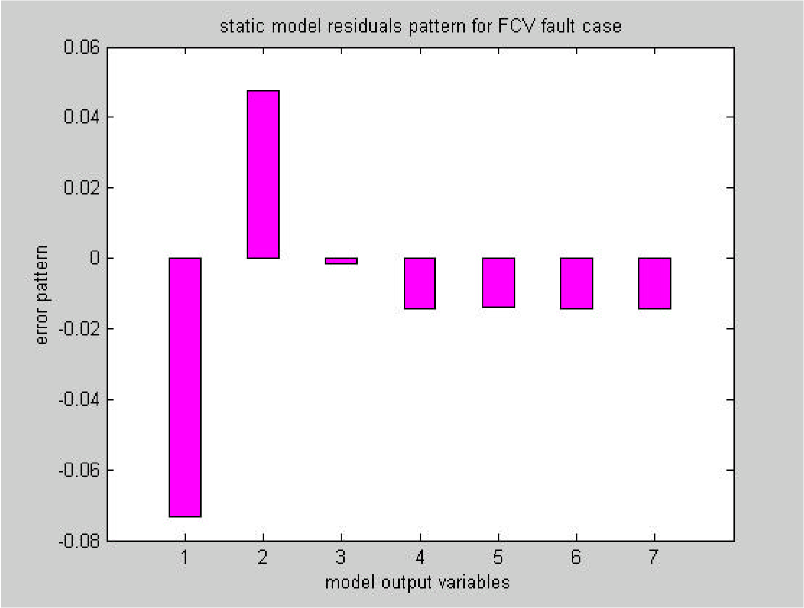


Figure 3.4 Plots of prediction errors of the seven variables for the case of feed control

3.5.2 Fault Detection During Plant Transient

The degradation of the turbine control valve (TCV) flow rate during a power transient from 100% to 95% was simulated. Figure 3.5 shows the comparison of the actual and indicated steam flow rates. Note that the prediction error changes with time, and is typical of patterns of faults during a plant transient.

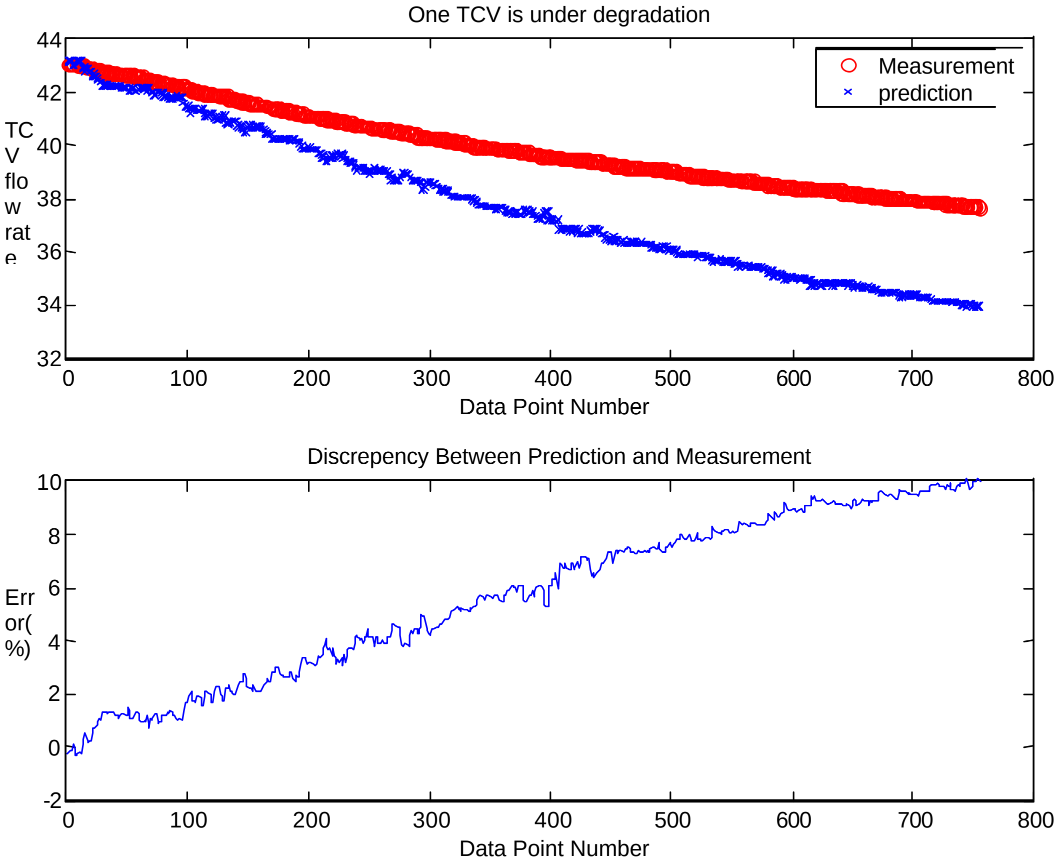


Figure 3.5 TCV flow rate showing the actual and predicted values for the case of TCV degradation.

3.6 Remarks

All the single fault cases, for both stationary and transient plant operation cases have been successfully isolated. Future work will include the diagnosis of controller degradation and certain equipment degradation, such as pump performance. Both a rule-based expert system and a pattern classification algorithm have been used for fault isolation.

4. DETECTION AND ISOLATION OF DUAL FAULTS IN THE PWR STEAM GENERATOR SYSTEM

4.1 System Description

The multi-model observer approach, for the detection of dual faults in the steam generator system, is presented in this section. The current available previous work in simulataneous multiple fault detection is limited.[31,32] The research results presented in this section are original contributions to this area of fault diagnosis. The steam generator (SG) water level control system is a typical multiple-input multiple-output (MIMO) feednback system. This sub-system has the following inputs and measured outputs:

a) Control variables:
Feed water control valve(FCV) position.
b) Reference signals:
Steam generator water level set point.
c) Measured outputs:
Steam generator narrow range water level.
Steam generator wide range water level.
Feed water flow rate.
Steam flow rate.
Steam generator pressure.
Steam generator temperature.

4.2 Selected Faults

The following single faults are defined in the designed FDI scheme:

a) SG narrow range level sensor offset
b) SG pressure sensor offset
c) FCV position offset
d) Steam flow meter offest
e) Feed water flow meter offset
f) FCV controller offset fault.

The following dual faults are considered:

a) Feed water flow meter offset and steam flow meter offset
b) Feed water flow meter offset and SG narrow range level sensor offset
c) Steam water flow meter offset and SG narrow range level sensor offset
d) Feed water flow meter offset and SG pressure sensor offset
e) Steam water flow meter offset and SG pressure sensor offset
f) SG narrow range level sensor offset and SG pressure sensor offset
g) Feed water flow meter offset and FCV position offset
h) Steam water flow meter offset and FCV position offset
i) FCV controller offset fault and FCV valve position offset
j) SG pressure sensor offset and FCV controller offset fault.

The above dual faults include all the four types of possible combinations of single component faults:

a) sensor + sensor
b) sensor + controller
c) sensor + actuator
d) actuator + controller.

4.3 Development of Observers

4.3.1 Model Structure

A process system can be characterized by a set of plant models, control models and measurement models. These models constitute a block diagram showing the relationship among plant parameters, state variables and measurement variables. Figure 4.1 shows such a diagram for the steam generator water level system in a PWR power plant. If all the inputs and the outputs of each model in the diagram are measured variables, these models will provide a reliable means to predict the output measurements under normal operation. Hence, each model can be considered as a constraint imposed on the measurements during normal operation. Any significant difference between the measured value and the predicted value will signify that the inherent relationship among the measurements has been violated due to faulty measeuements or process faults. Assuming different faults will result in different patterns of models being violated, the patterns may be used as signatures to isolate the faults.

Some advantages of using this approach to build input-output models are summarized as follows:

a) Simplify the FDI scheme.

Separate sets of models are built for different subsystems after the interaction between them is properly considered and the method can be easily extended to a large scale process plant.

b) Helpful to evaluate the detectability and isolatability of the faults.

Based on whether a faulty signal is involved in a model, the model consistency can be performed without the need to actually build a model. This makes it easy to determine whether two faults are isolatable.

4.3.2 General Considerations in Building Models

The process models can be derived either from physical laws or from historical data. In most cases, it would be too complicated to build analytical models based on physical laws, not only due to the interactions of systems and the interactions of physical phenomina but also due to nonlinear system behavior. However, the advantage of either of the two types of models can be taken advantage of by extracting qualitative relationships among the measured variables based on phyisical laws while extracting quatitative relationships using data-driven models.

In order to achieve the goal of isolating the specified faults in an FDI system, an ideal situation is that one model is only sensitive to one fault and insenstive to all the other faults. In order to describe a process more accurately and make the model applicable over a wider normal operation condition or even fault conditions, more variables are generally needed. For this reason, a model should be as simple as possible as long as it is able to accurately estimate the output under both normal and faulty conditions.

Since dynamic models are able to capture more information than static models, dynamic models are usually used in an FDI system. For instance, the steam generator level sensor offset cannot be detected using a static model because the controller can always bring the water level back to the indicated level. However, if a dynamic model is used, an immediate inconsistency between the measured level and the estimated level will occur. The residual between the measured level and the estimated level will eventually be brought back to zero when a new steady state is reached. The dynamic feature can thus be used to characterize SG level sensor fault. Moreover, a dynamic model tends to be more robust than a static model when there is disturbance in the process. In addition, a static model cannot be used to detect controller faults.

4.4 Model Development

4.4.1 4.4.1 Searching for Qualitative Relationships

The qualitative input-output relationship can be easily determined by using the block diagram shown in Figure 4.1. If the system is assumed to be of first order, the model structures can then be determined.

The model structure with FCV flow rate as output:

FCV flow rate(t) = f(FCV valve position(t), SG pressure(t))

The model structure with FCV valve position as output:

FCV valve position(t+1) = f(controller output(t),FCV valve position(t))

The model structure with SG pressure as output:

SG pressure(t) = f(SG temperature(t))

The model structure with Steam flow rate as output:

Steam flow rate(t+1) = f(Steam flow rate(t),feed water temperature(t), SG pressure(t), hot leg temperature(t), cold leg temperature(t))

The model structure with SG level as output:

SG level(t+1) = f(SG level(t),Feed water flow rate(t), steam flow rate(t))

The model structure with controller output as output:

Controller output(t+1) = f(SG level(t), SG reference level(t), steam flow rate(t), feed water flow rate(t))

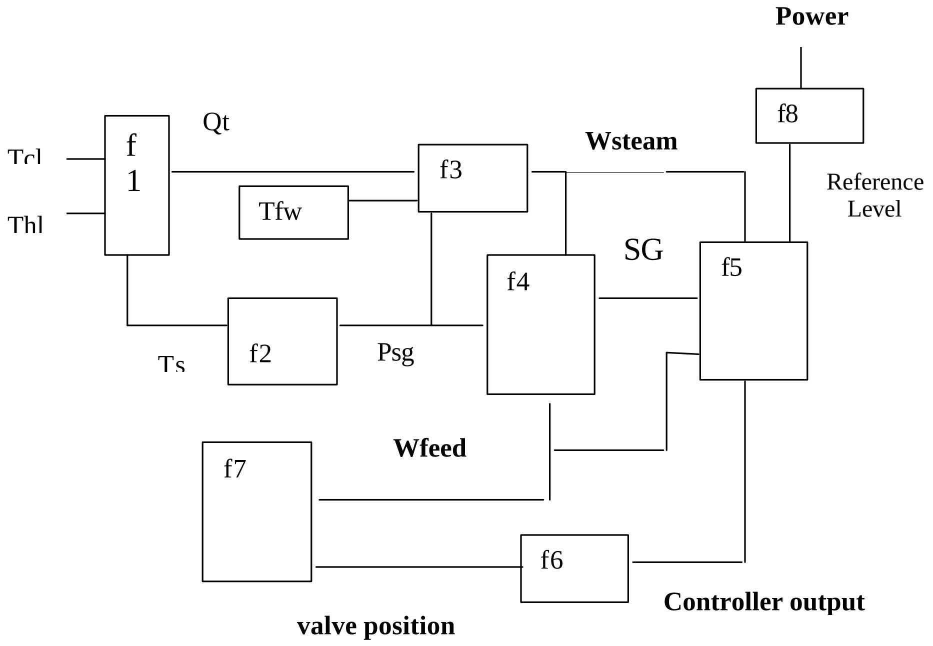


Figure 4.1 Block diagram of the structural models of nuclear steam generator water level control system

4.4.2 Collecting Data

In order to build data driven models to describe the dynamic behavior of the SG water level system under normal operation, the data being used must cover the entire possible opeartion range. Otherwise, the data driven models would try to perform extrapolation and the prediction would not be reliable.

The following operation conditions are considered in order to collect the data:
(1) slow power transient beginning from 20% to 100% at an interval of 2%.
(2) large power transient from 20% to 100% power level.
(3) large power transient from 100% to 20% power level.

The data collected during large power transients are used to build data driven models to predict controller output and valve position.

4.4.3 ANFIS Model and Model Validation

Adaptive Network-based Fuzzy Inference System (ANFIS) is selected to build system models during nominal operation. ANFIS is a fuzzy inference system implemented in the framework of neural network. It combines the reasoning capability of fuzzy logic with the learning capability of neural networks.

Six ANFIS models were built to estimate the feed control valve (FCV) flow rate, steam flow rate, SG water level, SG pressure, FCV position and controller output. The six models were validated by reconstructing a dynamic process for a power transient that the models have not learned before.

4.5 FDI Results Using Local Models

Table 4.1 shows the residual patterns and the distinguishability of the 16 faults defined in the FDI scheme. In the table, the sign 0 indicates that the model to predict the specific variable is not violated while the sign 1 indicates that the model to predict the specific variable is violated. The threashold to distinguish whether a model is violated or not is determined by the model accuracy and the level of plant disturbance. Careful inspection of the model inconsistences shows that a model is violated only if there are faulty inputs or fault outputs in the model. It thus indirectly proves that the selected model structure and the ANFIS models are correct.

It is interesting to notice the sign **?** in the table representing that residual is not stable for different fault magnitudes. The residuals to predict SG level are not stable for the following combination of faults:

a) Steam flow meter offset plus feed flow meter offset.
b) SG pressure sensor offset and SG level sensor offset.
c) FCV valve position offset and Feed water flow meter offset.

The reason why the residual is not stable is that either of the element faults has the possibility of cancelling the contribution of the other fault to the SG level for these three dual faults. From table 4.1, it is found that some of the dual faults are not distinguishable from their element faults.

a) Feed flow meter offset plus SG level sensor offset cannot be separated from feed flow meter offset.
b) Steam flow meter offset plus SG level sensor offset cannot be separated from steam flow meter offset.
c) SG pressure sensor fault plus Feed flow meter offset cannot be separated from steam SG pressure sensor fault.
d) SG pressure sensor fault plus SG level sensor offset cannot be separated from SG pressure sensor fault plus SG steam flow meter offset.

The reason why some of the dual faults are not separable from their element faults is that the fault signatures of the two elements are not independent. For instance, feed flow meter offset will result in all the fault signatures that SG level sensor offset will create, so feed flow meter offset plus SG level sensor offset will exhibit the same fault signatures as will the feed flow meter offset. Therefore, feed flow meter offset plus SG level sensor offset cannot be separated from feed flow meter offset.

In conclusion, checking the consistency of the constraint models imposed only on the dynamics of a feedback control loop is sometimes not enough to isolate the dual faults, such as two sensor faults or a sensor fault and an actuator fault in the control system. The reason is that some dual faults will result in violating the same pattern of constraint models as one of their element faults. In addition, the consistency of some constraint equations may not be stable for dual faults with different combination of fault magnitudes.

Table 4-1Checking the consistency of local mode

functional Model Faults	FCV flow rate	Steam flow rate	SG pressure	FCV valve position	SG level	Controller output
Feed Flow Meter Offset	1	0	0	0	1	0
Steam Flow Meter Offset	0	1	0	0	1	0
Steam Flow Meter and Feed Flow meter Offset	1	0	0	0	?	0
SG NR Level Sensor Offset	0	0	0	0	1	0
Feed Flow Meter offset and SG Level Sensor Offset	1	1	0	0	1	0
Steam Flow Meter Offset and SG Level *4.5.1.1.1.1.1.1.1 Se nsor Offset*	0	1	0	0	1	0
SG Pressure Sensor offset	1	1	1	0	1	0
Feed Flow Meter offset and SG Pressure Sensor Offset	1	1	1	0	1	0
Steam Flow Meter Offset and SG Pressure Sensor Offset	1	1	1	1	1	0
SG Level Sensor Offset and SG Pressure Sensor Offset	1	1	1	1	?	0
Feed water Flow Meter Offset and FCV Position Offset	1	0	0	1	?	1
Steam Flow Meter Offset and FCV Position Offset	0	1	0	1	1	1
FCV Position Offset	0	0	0	1	1	1
4.5.1.2 FCV Controller Gain Offset	0	0	0	0	?	1
FCV Controller Gain Offset and FCV Valve Position Bias	0	0	0	1	1	1
FCV Controller Gain Offset and SG Pressure Sensor Offset	1	1	1	0	1	1

4.6 FDI Results Using Global Models

It is found that dual faults are strongly distinguishable from single faults on the condition that either of the element faults can be isolated whatever the other fault it is combined with. Considering this, appropriate constraint models can be built following some empirical rules. For a sensor fault, other than the controlled variable related sensor (SG level in the case of SG water level control system), the constraint equation should relate the variable measured by the sensor and some variables which will not be affected by any faults in the considered control loop. A dynamic model based on controller output needs to be built in order to isolate an actuator fault from some other sensor fault in the control loop. In order to isolate the controlled variable related sensor fault from its combination with some other sensor fault, a redundant measurement is needed. For instance, the measurement of wide range level sensor needs to be used in order to check if the narrow range level sensor is in good condition.

In order to achieve the goal of strong separation of dual faults from their element faults for the SG level system, six ANFIS models were built based on the following qualitative relationships:

FCV valve position(t+1) = f(controller output(t),FCV valve position(t));
SG pressure(t) = f(SG temperature(t))
FCV flow rate(t) = f(Power(t))
Steam flow rate(t+1) = f(Power(t))
SG NR level(t) = f(SG WR level)
Controller output(t+1) = f(SG level(t), SG reference level(t), steam flow rate(t), feed water flow rate(t)).

Table 4.2 shows the residual patterns and the distinguishability of the 16 faults defined in the FDI scheme. In the table, 0 indicates that the model to predict the specific variable is not violated while 1 indicates that the model to predict the specific variable is violated. It is found that all the sixteen faults are separable. A careful inspection of the model inconsistencies shows that a model is specifically designed to isolate one fault. Such a model is senstive to the fault it is designated to isolate and insensitive to all the other faults that are not designated to isolate.

Table 4.2. Consistency Checking of Global Models

functional Model Faults	FCV flow rate	steam flow rate	SG pressure	FCV valve position	SG level	Controller output
Feed Flow Meter Offset	1	0	0	0	0	0
Steam Flow Meter Offset	0	1	0	0	0	0
Steam Flow Meter and Feed Flow meter Offset	1	1	0	0	0	0
SG Level Sensor Offset	0	0	0	0	1	0
Feed Flow Meter offset and SG Level Sensor Offset	1	0	0	0	1	0
Steam Flow Meter Offset and SG Level *4.6.1.1.1.1.1.1.1 Sensor Offset*	0	1	0	0	1	0
SG Pressure Sensor offset	0	0	1	0	0	0
Feed Flow Meter offset and SG Pressure Sensor Offset	1	0	1	0	0	0
Steam Flow Meter Offset and SG Pressure Sensor Offset	0	1	1	1	0	0
Steam Level Sensor Offset and SG Pressure Sensor Offset	0	1	1	1	1	0
Feed water Flow Meter Offset and FCV Position Offset	1	0	0	1	0	1
Steam Flow Meter Offset and FCV Position Offset	0	1	0	1	0	1
FCV Position Offset	0	0	0	1	0	1
4.6.1.2 FCV Controller Gain Offset	0	0	0	0	0	1
FCV Controller Gain Offset and FCV Valve Position Bias	0	0	0	1	0	1
FCV Controller Gain Offset and SG Pressure Sensor Offset	1	0	1	0	0	1

4.7 Remarks

Checking the model consistency in a structural sense provides a systematic way to design an FDI system. It makes it easy to extend the design scheme from subsystems to the whole plant. Based on whether a faulty signal is involved in a model, the model consistency can be performed without the need to actually build a model. This makes it easy to determine whether two faults could be isolated before the models are actually built.

This approach has been implemeted to diagnose six single faults and ten dual faults for the SG water level system of a typiocal PWR power plant.

Checking the consistency of the models imposed locally on the dynamics of a feedback control loop is sometimes not enough to isolate some of the dual faults. The major reason is that the fault signatures of the two device faults are not independent.

Dual faults are strongly distinguishable from the single faults on the condition that either of the element faults can be isolated whatever the other fault it is combined with. To achieve this goal, some empirical rules have been found for selecting model structures. For a sensor fault, other than the controlled variable related sensor, the constraint equation may relate the variable measured by the sensor and some variables which will not be affected by any faults in the considered control loop. A dynamic model based on controller output may be built in order to isolate an actuator fault from some other sensor fault in the control loop. In order to isolate the controlled variable related sensor fault from its combination with some other sensor fault, a redundant measurement is usually needed. Using six models specifically designed for separating each fault, all the dual faults have been successfully isoalted for the SG water level system.

5. DEVELOPMENT OF A GRAPHICAL USER INTERFACE FOR THE DEMONSTRATION OF THE FDI SYSTEM

5.1 Introduction

In order to demonstrate the effectiveness of the method of fault detection and isolation (FDI) for nuclear power plants, a graphical user interface software was developed under using Halden's *PICASSO-3* platform which is a user interface management system[33,34]. The software is able to create a fault by changing the fault characteristic parameters, display key parameters on a flowchart of the reactor system, exhibit the residual patterns specific to the fault, trend the process variables relevant to the fault, and echo the FDI results. The software integrates the NCSU-PWR simulator (a reactor system analysis code in FORTRAN) and the FDI code in MATLAB, and the C++ code to control the graphical user interface. NCSU code is the driving code in the software, which makes it possible to advance simulation time without interrupting the simulation after the data are flushed to the user interface. The Picasso Real Time Manager is controlled by a C-code that keeps running using multithread mode while the NCSU simulator is running so that the performance of the user interface display does not degrade due to possible time delay before the C code can get data from the simulation code.

It is a very important task to evaluate the overall performance of a newly developed FDI system. Although quite a few FDI methods have been available, all of these methods have their inherent weakness as compared with the others. This is mainly due to the great challenges to the comprehensive requirements of an FDI system such as early detection and diagnosis, isolatability, robustness, novelty identification, multiple fault identifiability, explanation facility, adaptability, etc. None of a single FDI method is able to have all these preferred characteristics.

Development of a graphical user interface (GUI) provides a convenient and cost effective way to make the evaluation. A simulation code can be used to study the process behavior under normal and faulty conditions. A fault can be created without much effort by changing some parameters related to the fault. Some noise can be easily added to the input and the output of the simulation codes in order that the robustness of the FDI method can be tested. With regard to testing the adaptability of the FDI method, the operational power levels can be modified or some disturbances such as steam generator tube fouling factor can be changed on the user interface. In addition, a GUI can also facilitate the evaluation of the FDI performance in detecting and isolating a single fault or multiple faults during a transient.

5.2 PICASSO-3 Development Environment

Picasso-3 is a User interface Management Systems (UIMS) developed as part of the OECD Halden Reactor Project.

Picasso-3 has three components. Graphics Editor (GED) is a tool for designing the user interface. GED can be used to design some user interface components, draw some pictures, set up some dialogues and define dynamic attributes to some user interface components. User interface database is where the GED saves the information containing the complete specification of the user interface. Run-time manager (RTM) actually realizes the application's user interface. Application process is the C-code written by the user to guide how RTM is to generate the user interface as desired by the user. Application Programmer's Interface (API) is a library of C-functions that is linked to the application process to enable it to communicate with RTM.

When an application is started, the application process calls functions in the API library in order to connect to the RTM. RTM responds by loading the application's user interface from the UI database and displaying it on the screen. By calling API functions periodically, RTM will continue to handle incoming events generated by the end user or by the processes.

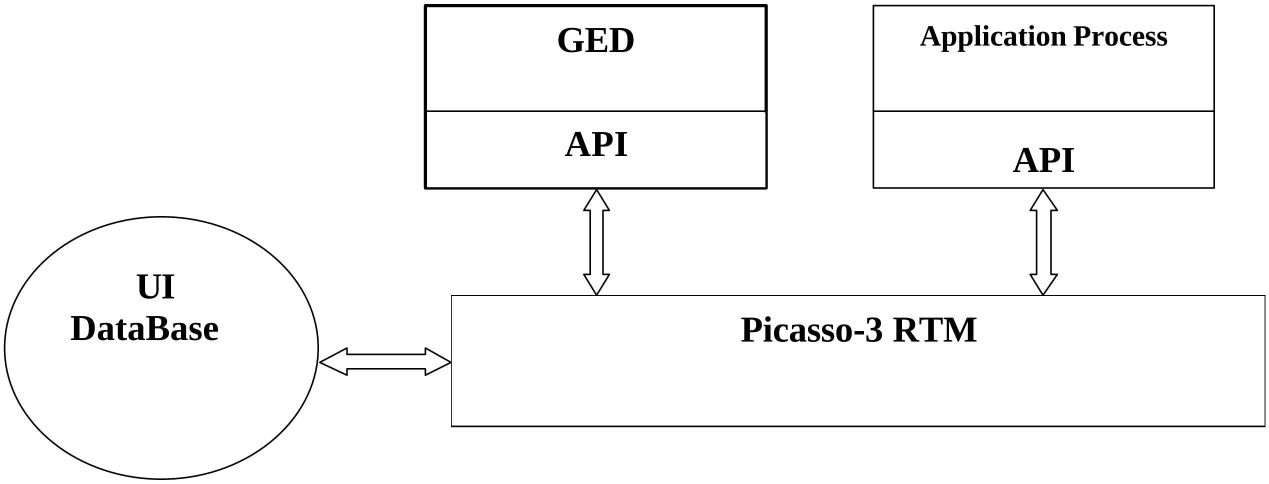


Figure 5.1 Block diagram of Picasso-3 system

5.3 Application Process Design

Figure 5.2 shows the block diagram of the application process design.

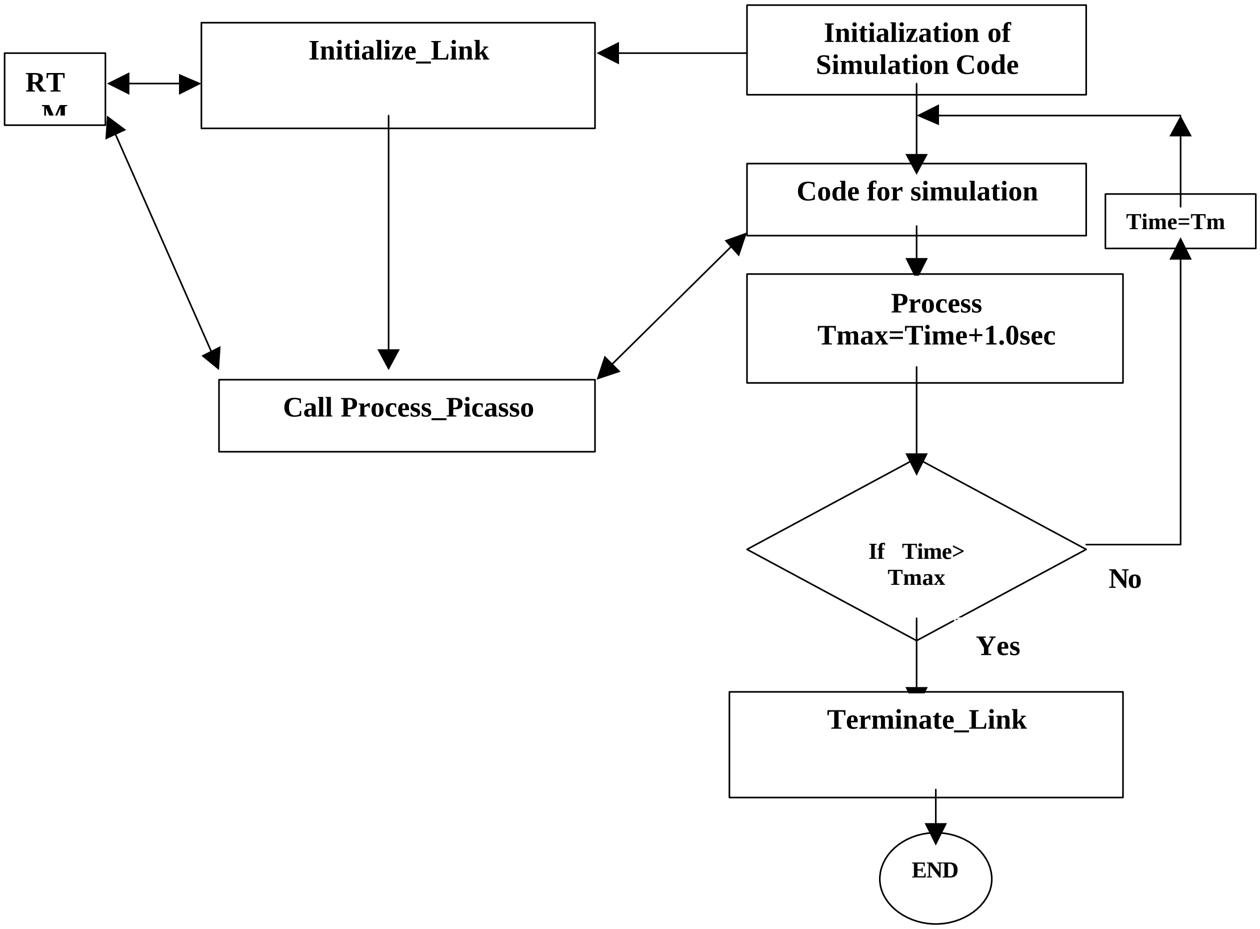


Figure 5.2 Flowchart of the application process.

The NCSU code is the main program. Before entering the main body of the computation, initial_link is called in order to set up a connection with the Picasso RTM. Meanwhile, some initialization data for NCSU calculation is transferred to RTM. The main body of NCSU code is run in a loop. At the end of the loop, the simulation time advances one second. The loop keeps running until the simulation time exceeds the specified maximum time. When the NCSU code steps out of the loop, terminate_link is called in order to end Picasso gracefully.

The application process program is written in multithread operation mode. The application process code is connected with the Picasso RTM. The function process_picasso is called periodically. On the one hand, it flushes the results calculated by NCSU to Picasso-3 RTM so that they can be displayed on the flowchart of the reactor system and can be trended on the trending plots. On the other hand, it detects whether some parameters defined on the screen to create some faults have been changed by the end user. If so, the process_picasso will transfer the changed parameters to NCSU for a new NCSU simulation. If process_picasso detects the request from the end user to perform fault detection and fault diagnosis, it will call the FDI module and compute the residuals due to the fault and send the residuals to RTM for display. The FDI diagnostic results will also be transferred to the FDI diagnostic information window indicating what is the fault according to the FDI algorithm.

The data exchange between the Picasso-3 application process and the NCSU simulation code is through global variables. These global variables exist as a structure in the C++ part of the application process and as a common block in the Fortran part of the application process. The data exchange between the Picasso-3 RTM and the Picasso-3 application process is through process structures and process variables. Both these data exchanges are two-way due to their global attributes.

Two remote functions, stopApplication() and datMount(), are defined. These two remote functions can be called directly by the user interface. The function stopApplication() can help the API code end its task gracefully. The function datMount enables the user to input the samples and sampling interval for the NCSU code to return adequate amount of data for fault diagnosis.

5.4 Descriptions of the Major Functions

A list of user interface files is given in Table 5.1. A description of the major functions is in this section.

1) **Header**

The header files for Picasso API, MFC socket, FDI module, MATLAB as well as the C++ application process itself are included in this part of the code. The header file for the C++ application process declares the function prototypes, structures for sensor characteristics, valve characteristics, controller characteristics, simulation data for display, and residuals. It also defines the global variables or structures used in the application process. In addition, the global variables used to access the common blocks in the NCSU FORTRAN code are also declared here.

2) int Initialize_link()

The functions of this link are as follows:

a) initialization of the variables for display on the screen of end user

b) calling PfInitialize to connect the application process to the RTM.

3) int process_picasso

This is the kernel code to be executed periodically. It is this code that calls PfSend and PfFlush to update the variables in all user windows.

The functions of this code are as follows:

a) calling some functions to transfer data from RTM to NCSU in order to follow the recent changes in the parameters by end users;

b) sending the most recent NCSU simulation results to RTM;

c) sending data to the FDI module or extract data from FDI module if FDI is requested.

4) terminate_link

This function calls PfEndLoop to end the picasso application process.

5) int32 createRecords()

It calls PfReadScript to create records and variables according to specification in RecordDefs.pdat.

6) int32 createVariables()

It calls PfCreateVar to create variables locally in API and puts the information into a local buffer to be used by PfFlushCreateVar.

7) void whenRtmConnects()

PfInitialize calls this function. It establishes connection with RTM and calls createRecords, createVariables and registerFunctions to let both RTM and the application process know the declarations of some process variables, structures and functions.

8) int32 registerFunctions();

This calls PfRegisterFunction to Register the function stopApplication to terminate the API and the function datMount to receive the user's input of samples and sampling interval from NCSU code.

9) int32 stopApplication()

This is a function defined in API code but available to RTM as a remote function. Its function is to end the application.

10) int32 datMount()

This is a function defined in API code but available to RTM as a remote function. Its function is to receive the user's input of samples and sampling interval from NCSU code.

11) void whenRtmDisconnects()

It is a function to give a message if connection has been lost with RTM.

12) RESD class_conversion

It is a function to convert the residual array to the structure type RESD.

13) void Pushdata()

It is a function to convert a double matrix into a mxArray data structure used as input of Matlab function.

14) void Extractdata()

It is a function to convert mxArray data structure used as output of Matlab function to a one-dimensional array.

15) char* faultType()

It is a function to determine the type of faults according to the residuals.

5.5 User Interface Design

The graphic user interface consists of five main windows.

The main window is designed to facilitate switching between functional windows. This provides the following options:

a) switch to the simulation window;
b) switch to the trending plot window;
c) switch to the FDI diagnostic results window;
d) switch to the fault creation window;
e) end task.

The fault creation window is designed to create faults by changing the parameters of the sensors, controllers, and actuators. The following parameters can be changed on this window:

a) FCV1 valve stuck position;
b) FCV1 offset;
c) FCV1 time constant;
d) FCV2 valve stuck position;
e) FCV2 offset;
f) FCV2 time constant;
g) TCV1 valve stuck position;
h) TCV1 offset;
i) TCV1 time constant;
j) TCV2 valve stuck position;
k) TCV2 offset;
l) TCV2 time constant;
m) TCV3 valve stuck position;
n) TCV3 offset;
o) TCV3 time constant;
p) TCV4 valve stuck position;
q) TCV4 offset;
r) TCV4 time constant;
s) Reactor power;
t) FCV1 controller offset;
u) Proportional gain of FCV1 controller;
u) Integral gain of FCV1 controller;
v) FCV2 controller offset;
w) Proportional gain of FCV2 controller;
x) Integral gain of FCV2 controller;
y) SG1 narrow range level sensor drifting rate;
z) SG2 narrow range level sensor drifting rate;
aa) SG1 flow meter drifting rate;
bb) SG2 flow meter drifting rate;

The trending plot window trends the following plots, which are important to represent the reactor system responses to the created faults:

a) Reactor nuclear power;
b) Reactor power output;
c) SG 1 water level;

d) SG 2 water level;
e) Hot leg temperature;
f) Cold leg temperature;
g) Feed water temperature;
h) Feed water flow rate;
i) SG 1 steam flow rate;
j) TCV 1 flow rate;
k) TCV 2 flow rate;
l) TCV 3 flow rate;
m) TCV 4 flow rate.

The fault diagnostic result windows shows the residual patterns of the following variables:

(a) SG1 narrow range water level;
(b) SG2 narrow range water level;
(c) FCV1 flow rate;
(d) FCV2 flow rate;
(e) SG 1 steam flow rate;
(f) TCV1 flow rate;
(g) TCV2 flow rate;
(h) TCV3 flow rate;
(i) TCV4 flow rate;
(j) hot leg temperature;
(k) cold leg temperature;
(l) FCV1 valve position;
(m) FCV2 valve position;
(n) feed water temperature(lumped loop);
(o) pressurizer temperature;
(p) presurizer level;

The simulation window shows the following variables on the schematic of the reactor system:
(a) reactor nuclear power;
(b) hot leg temperature;
(c) cold leg temperature;
(d) pressure in the pressurizer;
(e) water level in the pressurizer;
(f) steam generator water level;
(g) feed water flow rate to SG1;
(h) feed water temperature;
(i) steam flow rate from SG1;

5.6 How to Use the FDI User Interface

1) Configuration of Picasso-3 environment

It is important to define a unique Picasso-3 Environment variable in the operating system in order to make copying and system commands easier. This variable is PICASSOPATH referred to by putting % before and after the variable.

In addition, the Windows environment variable PATH should include where Picasso3 is installed such as C:\Picasso3\bin\win32.

2) Copying files

For convenience, all the files necessary for Picasso, NCSU code, application process code would better be placed in one directory. A complete list of the required files is shown in table 1.

3) Starting the Picasso-3 Program

It is required to start two main programs, GED and RTM before any other application programs. GED can then be used to load the application process NERI. Finally, all the pictures should be loaded using GED before the FDI application program SimPWRP is run.

4) Running the FDI application program

The FDI application program SimPWRP can be run under DOS by typing:
SimPWRP<input.txt

5) Using the interface program

As soon as SimPWRP is running, the user is able to switch between the five windows back and forth. In most cases, the user may firstly go to the fault creation window and create a fault. Secondly, the user may go to the fault diagnostic window, click either transient or steady state button to find out the diagnostic results.

Figure 5.3 – 5.7 shows the various windows created by the FDI demonstration system. A PWR plant was simulated using the full-scope simulator. All the screen displays are obtained from the *PICASSO* interface.

Table 5-1 A list of files for FDI user interface

File Name	File Size(Bytes)
ANNt_PBK.c	15,182
ANNt_PBK.h	3,531
ANNt_PBK.m	1,018
BOP.dat	13,541,837
brent.f	683
ButtonLib.plib	9,164
close.c	62,525
close.h	3,537
Comp3DLib.plib	163,512

compLib.plib	**76,112**
Components.plib	**64,188**
convert.c	**10,696**
convert.h	**3,508**
convert.m	**157**
convert2.c	**7,735**
convert2.h	**3,476**
convert2.m	**81**
Faults.ppic	**53,212**
Faults.Tdoc	**44,373**
faulttest.m	**7,428**
fault_det.c	**75,237**
fault_det.h	**3,436**
fault_det1.c	**62,319**
fault_det1.h	**3,491**
fault_det1.m	**6,983**
fbInterface.h	**1,062**
FCV_Pos_Coef.output967	**12,690**
FCV_Pos_LayerOrder.output967	**505**
FDI.ppic	**21,568**
FDI.Tdoc	**29,181**
FDImain.c	**28,244**
FDIres.ppic	**28,100**
FDIres.Tdoc	**26,915**
Feed_Flow_Coef.output967	**774**
Feed_Flow_LayerOrder.output967	**63**
fun3.c	**18,022**
fun3.h	**3,463**
fun3.m	**1,309**
gcbf.c	**10,040**
gcbf.h	**3,377**
Gmdh_residual.dat	**70**
input.txt	**38**
InputFields.plib	**70,064**
Logics.plib	**43,112**
MotifCtrLib.plib	**68,172**
MotifLib.plib	**91,212**

MyBinaryGenerator.c	17,225
MyBinaryGenerator.h	3,561
MyBinaryGenerator.m	885
myVars.pdat	295
NewJimSimPWRP.f	581,055
NewJimSimulatorInterface_test2.cpp	29,312
NewSimPWRP.f	581,604
new_database.txt	250,100
NR_level_Coef.output967	5,355
NR_level_LayerOrder.output967	284
Pressurizer.dat	2,733,945
REACTOR.plib	25,408
RECORD.pdat	2,447
RecordDefs.pdat	2,532
restart.dat	4,878
S1zscore.c	9,773
S1zscore.h	3,517
S1zscore.m	614
S2zscore.c	10,162
S2zscore.h	3,508
S2zscore.m	192
SGData.f	2,472
SGdatabase2.dat	18,480
SGdatabase2.txt	18,480
SGlevel.f	7,289
SimCntrl.exe	2,781,246
SimPWRP.dsp	5,546
SimPWRP.dsw	537
SimPWRP.exe	2,170,926
SimPWRP.ncb	214,016
SimPWRP.opt	51,712
SimPWRP.plg	4,703
SimPWRP_Tmax.f	577,261
SimulatorInterface.h	10,415
Steam_Flow_Coef.output967	1,161
SYSTEM.ppic	37,768
SYSTEM.Tdoc	41,872

TCV_Flow_Coef.output967	6,345
TCV_Flow_LayerOrder.output967	255
TloopLib.plib	5,352
trendLib.plib	18,092
Trips.dat	113
UTSG.plib	212
Valves.plib	29,948
we4bop.dat	1,221
we4bop2.dat	1,217
we4core.boc.dat	1,085
we4core.boc2.dat	1,083
we4core.eoc.dat	1,081
we4core.eoc2.dat	1,079
we4geom.dat	1,592
we4geom2.dat	1,590
we4init.boc.dat	5,292
we4init.boc2.dat	5,224
we4init.eoc.dat	5,292
we4init.eoc2.dat	5,224
we4init0.boc.dat	5,292
we4init0.boc2.dat	5,224
we4init0.eoc.dat	5,292
we4init0.eoc2.dat	5,172
we4init20.boc.dat	5,292
we4init20.boc2.dat	5,224
we4init20.eoc.dat	5,292
we4init20.eoc2.dat	5,224
we4init40.boc.dat	5,292
we4init40.boc2.dat	5,224
we4init40.eoc.dat	5,292
we4init40.eoc2.dat	5,224
we4init60.boc.dat	5,292
we4init60.boc2.dat	5,224
we4init60.eoc.dat	5,292
we4init60.eoc2.dat	5,224
we4init80.boc.dat	5,292
we4init80.boc2.dat	5,224

we4init80.eoc.dat	5,292
we4init80.eoc2.dat	5,224
we4prz.dat	239
we4prz2.dat	213
we4utsg.dat	1,074
we4utsg2.dat	1,024
zscore2.c	12,061
zscore2.h	3,495
zscore2.m	812

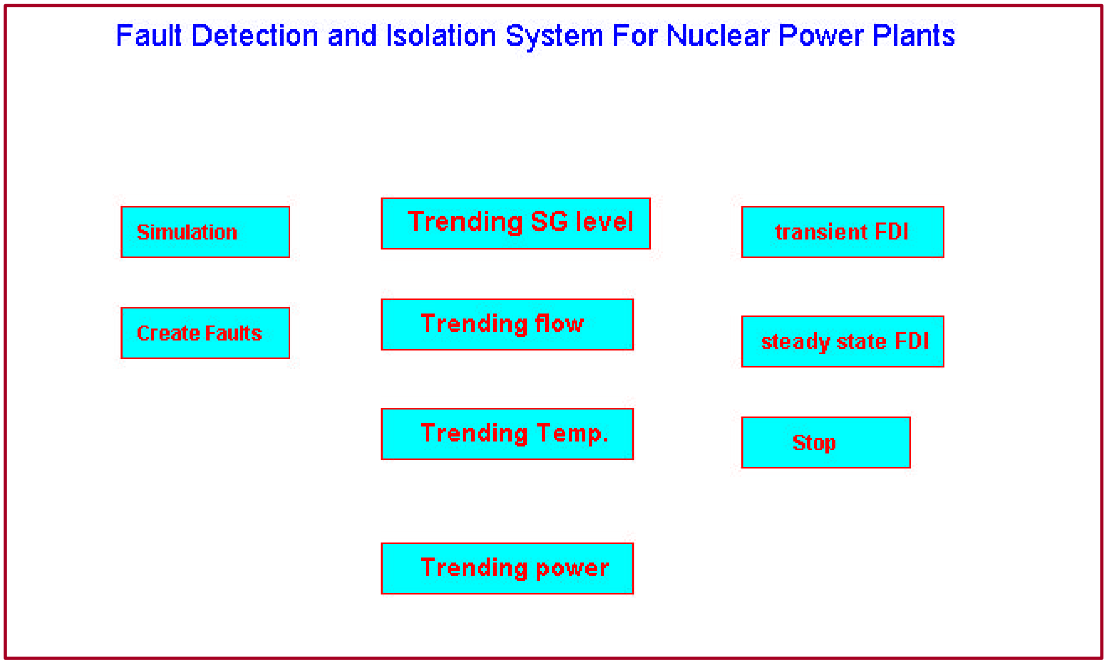
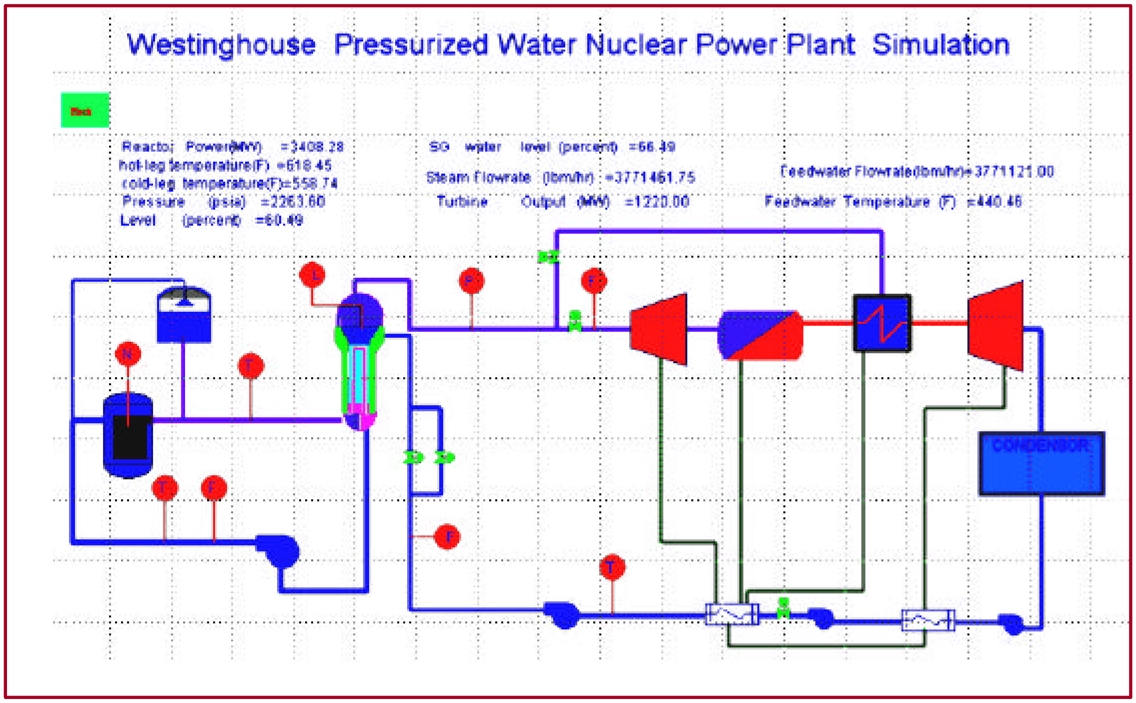


Figure 5.3 . PICASSO GUI main window



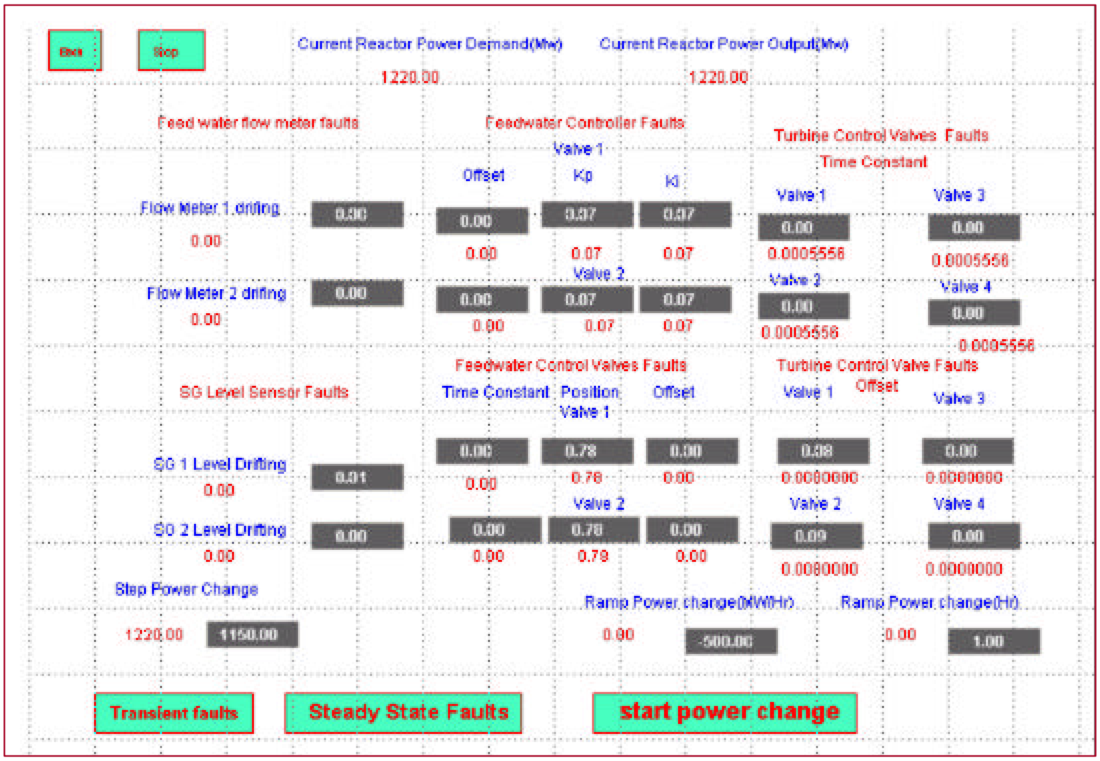
Westinghouse Pressurized Water Nuclear Power Plant Simulation

Reacto; Power(MW) =3408.28
hot-leg temperature(F) =618.45
cold-leg temperature(F)=558.74
Pressure (psia) =2263.60
Level (percent) =60.49

SG water level (percent) =66.49
Steam Flowrate (lbm/hr) =3771461.75
Turbine Output (MW) =1220.00

Feedwater Flowrate(lbm/hr)=3771121.00
Feedwater Temperature (F) =440.46

Figure 5.4 PWR plant simulation window



Current Reactor Power Demand(Mw) 1220.00
Current Reactor Power Output(Mw) 1220.00

Feed water flow meter faults
Flow Meter 1 drifting 0.00 [0.00]
Flow Meter 2 drifting 0.00 [0.00]

Feedwater Controller Faults

	Offset	Kp	Ki
Valve 1	0.00 0.00	0.07 0.07	0.07 0.07
Valve 2	0.00 0.00	0.07 0.07	0.07 0.07

Turbine Control Valves Faults
Time Constant
Valve 1 [0.00] 0.0005556
Valve 3 [0.00] 0.0005556
Valve 2 [0.00] 0.0005556
Valve 4 [0.00] 0.0005556

SG Level Sensor Faults
SG 1 Level Drifting 0.00 [0.01]
SG 2 Level Drifting 0.00 [0.00]

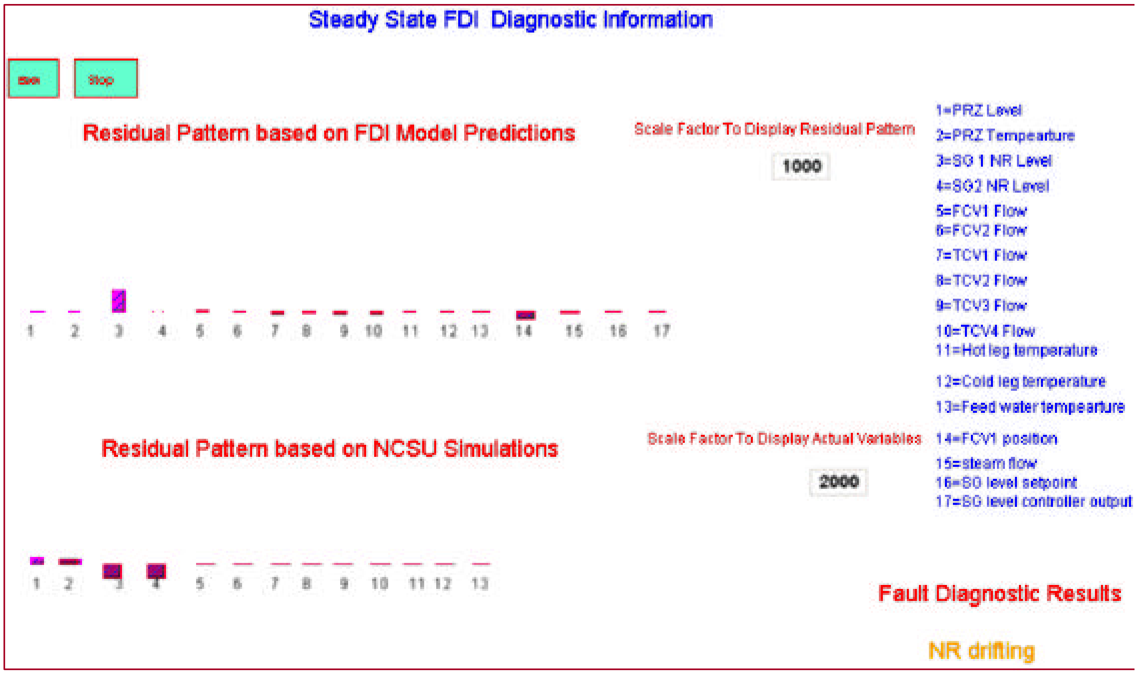
Feedwater Control Valves Faults

	Time Constant	Position	Offset
Valve 1	0.00 0.00	0.78 0.78	0.00 0.00
Valve 2	0.00 0.00	0.78 0.79	0.00 0.00

Turbine Control Valve Faults
Offset
Valve 1 [0.08] 0.0080000
Valve 3 [0.00] 0.0000000
Valve 2 [0.09] 0.0080000
Valve 4 [0.00] 0.0000000

Step Power Change 1220.00 [1150.00]
Ramp Power change(MW/Hr) 0.00 [-500.00]
Ramp Power change(Hr) 0.00 [1.00]

Figure 5.5 Device fault cre ation window



Steady State FDI Diagnostic Information

Residual Pattern based on FDI Model Predictions

Scale Factor To Display Residual Pattern: 1000

Residual Pattern based on NCSU Simulations

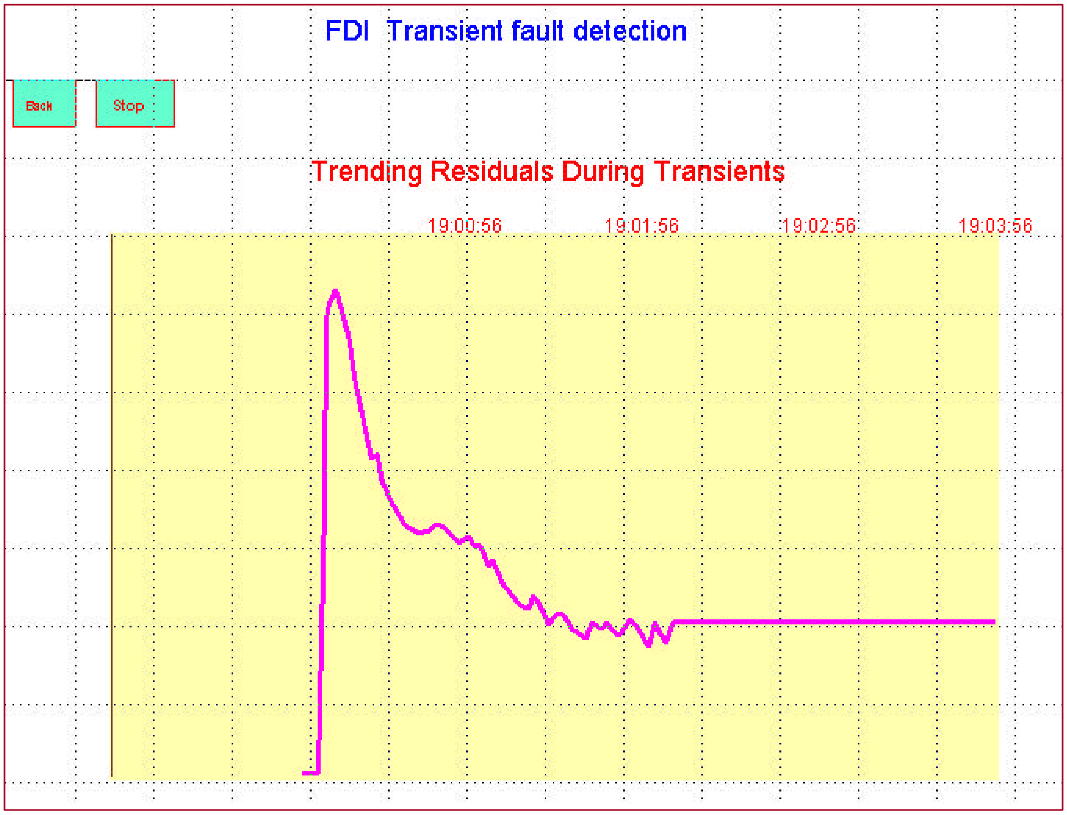
Scale Factor To Display Actual Variables: 2000

1=PRZ Level
2=PRZ Tempearture
3=SG 1 NR Level
4=SG2 NR Level
5=FCV1 Flow
6=FCV2 Flow
7=TCV1 Flow
8=TCV2 Flow
9=TCV3 Flow
10=TCV4 Flow
11=Hot leg temperature
12=Cold leg temperature
13=Feed water tempearture
14=FCV1 position
15=steam flow
16=SG level setpoint
17=SG level controller output

Fault Diagnostic Results

NR drifting

Figure 5.6 FDI window showing fault diagnostics results.



FDI Transient fault detection

Trending Residuals During Transients

Figure 5.7 Illustration of fault detection during transient operation: stuck feed control

6. CONCLUDING REMARKS AND FUTURE WORK

6.1 6.1 Summary and Concluding Remarks

An integrated method for incipient fault detection and isolation of sensors and field devices, using structured residuals, has been developed. The FDI system uses three approaches for characterizing system variables – states and control functions. These are (1) group method of data handling (GMDH) with rational function approximation, (2) principal component analysis (PCA) with nonlinear extension, (3) artificial neural network models. The isolation of device faults is performed using both a rule-based decision-making and a pattern classification of prediction error vectors in the *fault space*. This integrated approach enhances the fault diagnostics capability and provides a robust method for FDI. The enhancements of the methods for the detection and isolation of dual faults in the steam generator system of a PWR plant and fault detection during plant transients are presented. A stand-alone demonstration of the FDI system has been developed using the Halden's *PICASSO* GUI platform.

The methods being developed under this task for the DOE-NERI project have been tested and illustrated using process measurements from a laboratory process control loop and a U-tube steam generator in a PWR. All the faults being simulated were detected successfully. The faults were primarily of the drift-type. Two manuscripts were prepared for presentations at scientific meetings during 2000-2001 and a journal publication is due in November 2001.

6.2 Plans for Future Work

Phase-3 R&D Tasks

- Fine tuning of the FDI algorithms.
- Extension of the FDI method by increasing the system boundary to include equipment faults beyond the UTSG system.
- Development of minimum requirements for application to an existing PWR plant.
- Fully automated implementation of the FDI system as part of the overall control architecture demonstration.
- Identification of issues in technology transfer to nuclear power industry, including implementation requirements.
- Completion of the *FDI* module demonstration system.
- **Deliverables:** Annual Reports and a Final Report.
  FDI software system and *User's Manual.*
  Conference and journal manuscripts for dissemination.

REFERENCES

1. V.J. Van Doren, “Advanced Control Software Goes Beyond PID,” Control Engineering, pp. 73-78, January 1998.

2. W. Gough, “Predictive Adaptive Control Aids Pulp Digestion,” Control Engineering, p. 97, November 1997.

3. R.J. Patton, J. Chen, and S.B. Nielsen, “Model-Based Methods for Fault Diagnosis: Some Guidelines,” *Transactions of the Institute of Measurements and Control,* Vol. **17**, pp. 73-83, 1995.

4. S.K. Yung, and D.W. Clarke, “Local Sensor Validation,” *Measurement and Control,* Vol. **22**, pp. 132-141, 1989.

5. B.R. Upadhyaya, et al., *Development and Testing of an Integrated Signal Validation System for Nuclear Power Plants,* Vol. 1, 2, 3, Final Report prepared for the U.S. Department of Energy, DOE/NE/37959-34,35,36, September 1989.

6. K. Kavaklioglu and B.R. Upadhyaya, “Monitoring Feed water Flow Rate and Component Thermal Performance of Pressurized Water Reactors by Means of Artificial Neural Networks,” *Nuclear Technology,* Vol. **107**, pp. 112-123, 1994.

7. A.S. Erbay and B.R. Upadhyaya, “A Personal Computer-Based On-Line Signal Validation System for Nuclear Power Plants,” Nuclear Technology, Vol. 119, No. 1, pp. 63-75, July 1997.

8. X. Xu and J.W. Hines, “Sensor Validation and Fault Detection Using Neural Networks, *Proc. MARCON 99,* Vol. **2**, pp. 58.01-58.09, May 1999.

9. M. Kramer, “Nonlinear Principal Component Analysis Using Autoassociative Neural Networks,” *AIChE Journal*, Vol. **37**, p. 233, 1991.

10. H. Tong and C.M. Crowe, “Detection of Gross Errors in Data Reconciliation by Principal Component Analysis,” *AIChE Journal*, Vol. **41**, p. 1712, 1995.

11. S.J. Farlow, *Self-Organizing Methods in Modeling: GMDH Type Algorithms*, Marcel Dekker, New York, 1984.

12. P.B. Ferreira and B.R. Upadhyaya, *Incipient Fault Detection and Isolation of Sensors and Field Devices*, Research Report, Nuclear Engineering Department, The University of Tennessee, Knoxville, UTNE/BRU/99-02, December 1999.

13. H. Tong, and C. M. Crowe, “Detection of Gross Errors in Data Reconciliation by Principal Component Analysis,” *AIChE Journal*, Vol. 41, p1712, 1995.

14. J. Keeler, and B. Ferguson, “Commercial Applications of Soft Sensors: The Virtual Online Analyzer”, *Proceedings of IFPAC Conference*, 1996.

15. M. Kramer, “Nonlinear Principal Component Analysis Using Autoassociative Neural Networks,” *AIChE Journal*, Vol. 37, p233, 1991.

16. P. Fantoni, and A. Mazzola, “Applications of Autoassociative Neural Networks for Signal Validation in Accident Management,” *Proceedings of the IAEA Specialist Meeting on Advanced Information Methods and Artificial Intelligence in Nuclear Power Plant Control Rooms*, 1994.

17. C. M. Crowe, A. Hrymak, and Y. A. Garcia Campos, “Reconciliation of Process Flow Rates by Matrix Projection: The Linear Case,” *AIChE Journal*, Vol. 29, p881, 1983.

18. D. K. Rollins, and J. F. Davis, “Unbiased estimation of Gross Errors in Process Measurements,” *AIChE Journal*, Vol. 38, p563, 1992.

19. T.W. Karjala and D.M. Himmelblau, “Dynamic Rectification of Data via Recurrent Neural Nets and the Extended Kalman Filter,” AIChE Journal, Vol. 42, p225, 1996.

20. E. B. Martin, and A. J. Morris, “Overview of Multivariate Statistical Process Control in Continuous and Batch Process Performance Monitoring,” *Transactions of the Institute of Measurement and Control*, Vol. 18, p51, 1996.

21. B. M. Wise and N. B. Gallagher, “The Process Chemometrics Approach to Process Monitoring and Fault Detection”, *Journal of Process Control*, Vol. 6, p329, 1996.

22. P. Miller, R. E. Swanson, and C. F. Heckler, “Contribution Plots: The Missing Link in Multivariate Quality Control,” *Proceedings of Fall Conference of the ASQC and ASA*, 1993.

23. J. Gertler, and D. Singer, “Augmented Models for Statistical Fault Isolation in Complex Dynamic Systems,” *Proceedings of the American Control Conference*, 1985.

24. J. Gertler, and D. Singer, “A New Structural Framework for Parity Equation Based Failure Detection and Isolation in Complex Plants,” *Automatica*, Vol. 26, p381, 1990.

25. J. Gertler, *Fault Detection and Diagnosis in Engineering Systems*, New York: Marcel-Dekker, 1998.

26. S. J. Qin, and W. Li, “Detection, Identification, and Reconstruction of Faulty Sensors with Maximized Sensitivity,” *AIChE Journal*, 45, p1963, 1999.

27. J. Gertler, W. Li, Y. Huang, and T. J. McAvoy, “Isolation Enhanced Principal Component Analysis,” *AIChE Journal*, Vol. 45, p323, 1999.

28. T. W. Wang, S. Vedula and A. Khettry, “Geometric Interpretation of SVD, rank, mean centering and scaling in applying multivariate statistical analysis methods”, *Fifth International Conference on Process Control*, p267, 1994.

29. M. Naghedolfeizi, and B. R. Upadhyaya, *Dynamic Modeling of a Pressurized Water Reactor Plant for Diagnostics and Control*, Research Report, Nuclear Engineering Department, The University of Tennessee, 1991.

30. E.L. Russell, L.H. Chiang and R.D. Braatz, *Data-Driven Techniques for Fault Detection and Diagnosis in Chemical Processes*, Springer-Verlag, London, 2000.

31. K. Watanabe, S. Hirota and L. Hou, "Diagnosis of Multiple Simultaneous Faults via Hierarchical Artificial Neural Networks," AIChE Journal, 1994.

32. G. Lee, "Multiple Fault Diagnosis Under Uncertain Conditions by Quantification of Qualitative Relations," American Chemical Society, 1999.

33. K.A. Barmsnes, “Picasso: A User Interface Management System for Real Time Applications,” OECD Halden Reactor Project, 1992.

34. O. Jakobsen, “The Picasso-3 User Interface Management System,” 1994.

Attachment III
Task 2 Phase 3 Report

UTNE/DOE/01-02

FAULT DETECTION AND ISOLATION OF NUCLEAR PLANT SYSTEM SENSORS AND FIELD DEVICES

Belle R. Upadhyaya
(Principal Investigator)

Baofu Lu
Ke Zhao

Nuclear Engineering Department
The University of Tennessee
Knoxville, TN 37996-2300
E-mail: bupadhya@utk.edu

Phase 3 Report

Prepared for the DOE-NERI Project
DE-FG03-99SF21906

September 2002

SUMMARY

This report presents a summary of the overall project and describes the tasks performed and the progress made by The University of Tennessee (UTK) during Phase-3 (2001-2002) on the DOE-NERI project entitled ***Automatic Development of Highly Reliable Control Architecture for Future Nuclear Power Plants.*** UTK is collaborating with the Instrumentation & Controls Division of ORNL (lead organization) and the North Carolina State University (NCSU). The objective of the UTK research task is to develop an on-line monitoring system for *Fault Detection and Isolation (FDI)* of sensors and field devices in a nuclear power plant. The focus of this phase of the research has been on the refinement of fault isolation techniques, fault tracking during transient operation, and the development of a new multi-observer method for detecting and isolating dual device faults with applications to a nuclear power plant steam generator system. Sensors include temperature, pressure, flow, level transmitters, and measurements of control functions, and field devices include valve actuators, spray and heater systems, pumps, and other similar equipment. The Phase-3 sub-tasks include the refinement of software systems and the completion of two demonstration modules.

The following R&D tasks have been accomplished during Phase-3 of the project:

- Development of data-driven models for characterizing transient operation and for predicting state variables, control functions, and expected control actions for FDI in the steam generator system.
- Development of multiple observers using the adaptive network fuzzy inference system (ANFIS) modeling approach. This also provides a systematic approach for physical interpretation of system behavior during a fault condition.
- Development of a directional graph (digraph) approach for detecting and isolating dual faults in the steam generator system.
- Extension of the *Principal Component Analysis (PCA)* of model prediction residuals for establishing fault directions. This approach has been very effective in fault isolation.
- Development of a stand-alone demonstration module that cycles through the various steps of the FDI system.
- Development of additional software modules and complete refinement of all the codes developed under the project.

All the software systems and the data used for analysis are available on a CD-R, included as part of the deliverables of the project.

Several publications were generated during Phase 3. These include papers presented at SMORN-8 (Conference on Nuclear Reactor Surveillance and Diagnostics), Goteborg, Sweden; MARCON-2002, Maintenance and Reliability Conference; Conference on Statistical Data Mining and Knowledge Discovery, June 2002; American Nuclear Society Annual Meeting, June 2002; publication in a journal by Springer-Verlag.

Acknowledgments

The research and development reported here has been sponsored by the U.S. Department of Energy Nuclear Energy Research Initiative (NERI) program with The University of Tennessee, Knoxville, under Grant Number **DE-FG03-99SF21906**. We gratefully acknowledge this support. We want to thank Professor Michael Doster of North Carolina State University for the use of a PWR plant simulation code, and for his assistance during the course of the project.

We appreciate the help given by Dr. Jose March-Leuba and Dr. Richard Wood of the Oak Ridge National Laboratory during all phases of this project.

TABLE OF CONTENTS

1. INTRODUCTION

(neriFinalReport02-02)

1.1. Background and Motivation

Existing and new generation of nuclear power plants have economic and reliability concerns as addressed by overall plant performance, unscheduled downtime and the long-term management of critical assets. The key to achieving these needs is to develop an integrated approach for monitoring, control, fault detection and diagnosis of plant components such as sensors, actuators, control devices and other equipment. Model-based local sensor validation and fault diagnosis methods were developed for specific applications. These approaches assume that a system fault being monitored occurs in a specific plant component and in an isolated fashion. Fault detection and isolation (FDI) of sensors and field devices is an important step towards the implementation of an automated and intelligent process control strategy.

A large-scale system, such as a nuclear power plant, has several feedback control loops. This makes the identification and isolation of faults in these interconnected systems highly complex. Even when a measurement used for set point control is faulty, the control system through feedback, tries to vary the actuating signals until the error in the set point is eliminated. The *sensor-alone* type validation will fail in this situation. It is therefore necessary to consider fault detection and isolation at the system level rather than at the device level. The overall objective of this R&D task is to develop an on-line sensor and field device monitoring and fault detection system, when one or more faults may occur in two or more of these devices. This goal has been achieved by a two-step approach: (1) System characterization using data-driven models; (2) A decision-making module that uses system functional knowledge base and pattern classification algorithms, that is deployed in a distributed configuration. High priority will be given to the computational efficiency of these techniques, with the capability to change the module structure with changing plant conditions. The intrinsic merit of this high-level project task lies in the development of an autonomous global monitoring and fault detection approach that would be executed with minimal human interaction.

1.2. Objectives of R&D and Definition of Tasks

The objective of this research task is to develop an on-line monitoring system for *Fault Detection and Isolation* (FDI) of sensors and field devices in a nuclear power plant. The sensor suite consists of major process variables in a plant, such as temperature, pressure, flow, level, and control functions. Field devices in a power plant include, but are not limited to, valve actuators, control modules, spray and heater systems, pumps, and similar equipment. Figure 1.1 shows an integrated plant monitoring, diagnosis, and prognosis system, with the ability to estimate the remaining useful life of plant devices.

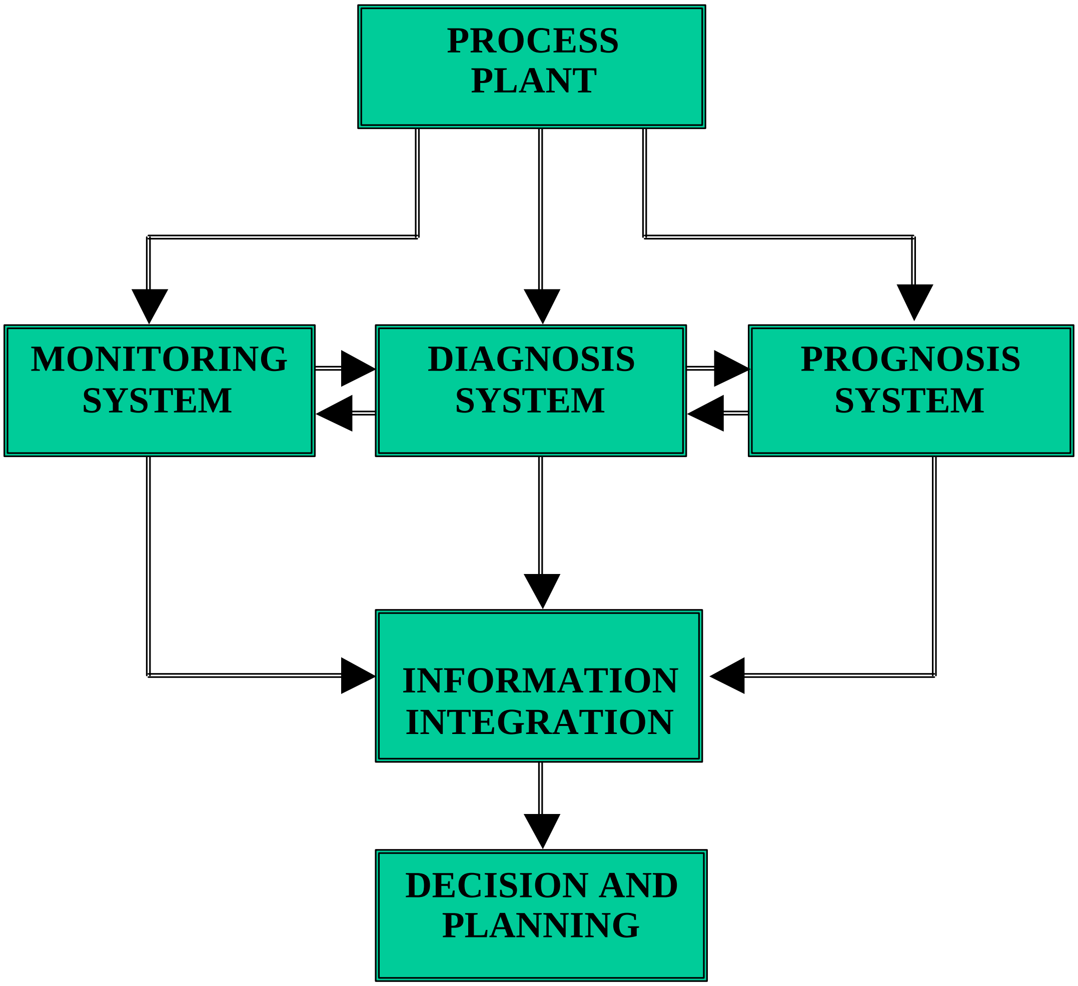


Figure 1.1. Integrated plant monitoring, diagnosis and prognosis system.

The objectives of this R&D have been accomplished through the completion of the following technical tasks during the three phases of the project:

- Review of literature and previous work (see bibliography).
- Characterization of sub-system dynamics using data-driven models for predicting state variables, control functions, and expected control actions.
- Development of a *Group Method of Data Handling (GMDH)* modeling algorithm with rational function approximation.
- Development of a *Principal Component Analysis (PCA)* algorithm with linear and nonlinear mapping.
- Development of multiple observers using the *adaptive network fuzzy inference system* (ANFIS).
- Development of an FDI module that combines system operational knowledge and a rule-based logic for both single and dual faults in dissimilar sensors and field devices. An automated pattern classification technique, based on the PCA of model residuals, has been developed and applied successfully.
- Development of a *directional graph (digraph)* approach for fault isolation with minimum prior knowledge of the system.
- Demonstration of the FDI system with application to an experimental process control loop.
- Demonstration of the FDI system with application to a U-tube steam generator (UTSG) in a full-scope simulation model of a 1,300 MWe PWR.
- Development of minimum requirements for FDI system implementation.
- Extension of the techniques for the case of fault detection during plant transients.
- Identification of realistic faults in a PWR and establish the characteristics of transient faults as compared with steady-state faults.
- Extension of the FDI method to systems beyond the steam generator.
- **Deliverables:**
    - Annual Reports and the Final Report.
    - FDI software system and *User's Manual*. A CD-R, containing the various computer codes, *readme* files, and plant data, is attached.
    - Conference and journal manuscripts.

1.3. Summary of Significant Accomplishments During 2001-2002

The following major milestones were accomplished during Phase-3 of the project:

- Development of data-driven models for characterizing transient operation and for predicting state variables, control functions, and expected control actions for FDI in the steam generator system.
- Development of multiple observers using the adaptive network fuzzy inference system (ANFIS) modeling approach. This also provides a systematic approach for physical interpretation of system behavior during a fault condition.
- Development of a directional graph (digraph) approach for detecting and isolating dual faults in the steam generator system.

- Extension of the *Principal Component Analysis (PCA)* of model prediction residuals for establishing fault directions. This approach has been very effective in fault isolation.
- Development of a stand-alone demonstration module that cycles through the various steps of the FDI system.
- Development of additional software modules and complete refinement of all the codes developed under the project.
- Demonstration of the FDI module for both single and dual/simultaneous faults:

  - Rule-based decision making.
  - Fault isolation using PCA of fault residuals and pattern classification.
  - Steady state and transient plant operation conditions.
  - Combination of sensors and valve actuators.
  - Use of both Simulink™ and NCSU-PWR simulation codes.

- List of Publications during Phase-3:
  - N. Kaistha and B.R. Upadhyaya, *Incipient Fault Detection and Isolation of Field Devices in Nuclear Power Systems Using Principal Component Analysis*, Nuclear Technology, Vol. 136, No. 2, pp. 221-230, November 2001.
  - B.R. Upadhyaya, K. Zhao, B. Lu, and M. Doster, *Fault Detection and Isolation of Sensors and Actuators in a Nuclear Plant Steam Generator,* Transactions of the American Nuclear Society, Vol. 85, pp. 350-351, November 2001.
  - B.R. Upadhyaya, B. Lu, K. Zhao, and J.M. Doster, *Equipment Monitoring During Process Transients and Multiple Fault Conditions,* Proceedings of MARCON 2002, Knoxville, TN, May 2002.
  - B.R. Upadhyaya, K. Zhao, and B. Lu, *Fault Monitoring of Nuclear Power Plant Sensors and Field Devices,* Proceedings of SMORN-8, Symposium on Nuclear Power Plant Surveillance and Diagnostics, Goteborg, Sweden, May 2002.
  - B.R. Upadhyaya, K. Zhao, B. Lu, J.M. Doster, M.G. Na, Y.R. Sim, and K.H. Park, *Nuclear Plant System Monitoring Under Process Transients and Multiple Fault Conditions,* Transactions of the American Nuclear Society, Vol. 86, pp. 482-484, June 2002.
  - B.R. Upadhyaya and B. Lu, *Data Mining for Monitoring Plant Devices Using Group Method and Pattern Classification,* Conference on Statistical Data Mining and Knowledge Discovery, June 2002.

1.4. FDI Architecture and Issues in Developing a Robust FDI Algorithm

Figure 1.2 shows the functional modules of the FDI system being developed in this project. *GMDH, PCA* and *Adaptive Network Fuzzy Inference System (ANFIS)* modeling of process measurements are considered. This provides a crosschecking of prediction techniques applied to the measurements. Fault isolation is based on either a rule-based algorithm or a pattern classification algorithm, or both. The following issues must be considered in developing a robust FDI algorithm.

- Sensor faults may not be detected in a closed-loop control system.
- Redundancies in sensors and controllers are used in nuclear power plants (NPPs).
- Separation of process variations from sensor/field-device faults must be considered.
- Noise levels in measurements can increase false alarms. It may be necessary to pre-process signals to eliminate this effect at different sub-bands.
- The use of physics models and data-driven models to understand and characterize the process dynamics.

The following are some additional issues to be reviewed before developing an FDI system.

1.4.1. Multi-operational regimes

A nuclear power plant may operate at numerous operational points due to the change in power demand, the evolution of fuel cycle, performance change of components throughout its lifetime, the change in system configuration to meet safety requirements, and others. The designed FDI system must be adaptable to all these operational points. For instance, the FDI system should be able to correctly distinguish a fault under all these operational conditions. A normal operational transient such as a power change, a chemical volume control system startup or shutdown, a steam generator blow down system startup or shutdown, must not trigger a false alarm. This requires that the developed FDI system be able to adaptively adjust its models at all operational points.

1.4.2. Dynamic process behavior

A nuclear power plant always experiences internal disturbances such as the vibration of machinery components and turbulence induced fluctuation and some external disturbances such as the change in power demand. Therefore, dynamic models need to be used to capture the system characteristics. In addition, all the state variables and/or the measured variables are random variables due to measurement disturbances or process disturbances.

Unlike a process, electric circuit exhibits static behavior. Once an electric circuit is around its operation point, a set of algebraic equations can always be found to characterize the relationship of the voltage, the current and the resistance among certain nodes. Through systematically checking the consistency of all the algebraic equations, it is not difficult to detect a faulty component and isolate it within the circuit.

For a dynamic process, a set of algebraic equations may not be able to characterize the relationship among process variables. Different initial conditions may result in different sets of relationships. A group of differential equations is usually required to characterize a dynamic system.

Non-linearity results in additional difficulties in modeling the behavior of a dynamic system especially for a nuclear power plant where many nonlinear components such as valves, pumps, and controllers with dead band and saturation limits are utilized.

The existence of non-linearity would also result in certain features of many FDI design schemes unavailable.

1.4.3. Controller feedback effects

Feedback control loops are common in nuclear power plants in order to maintain the operation within the designed operation regimes. Power control system controls the reactor power such that the power generation from the core matches the power output of the plant. Steam generator (SG) water level control system controls the feed water control valve position such that the SG level is maintained at the set point. Pressurizer level and pressure control system manipulates the power of electric heaters and spray flow rate such that the level and pressure are maintained at desired set points.

Because of feedback controller, a sensor fault or an actuator fault will propagate throughout the system. The fault propagation will result in challenges in designing an effective FDI system in the following aspects (Dash and Venkatasubramanian, 2000):

- Data reconciliation approach is not applicable.
- A minor fault is harder to be detected and isolated.
- A fault may propagate from one subsystem to another subsystem through a control system bridging them.
- A comparison between set points and measured values after a new steady state cannot reveal the occurrence of a sensor fault that is involved in the feedback control loop.

1.4.4. Complexity of faults

In a large process like NPPs, the natures of possible faults are very complicated because many different components may be involved. From the FDI methodology point of view, these components may be categorized as sensor fault, actuator fault, controller fault and process fault. With regards to fault effects on the measurements, a fault can be classified as additive fault and multiplicative fault. The time dependence of fault magnitude allows categorizing a fault as abrupt fault, drift fault and intermittent fault. None of the available FDI approaches has acceptable performance for all the different types of faults.

1.4.5. Multiple faults

The importance of multiple fault diagnosis in the entire FDI picture should not be underestimated simply because its probability is much lower than single faults. In practice, in a facility such as a nuclear power plant where safety is always placed at first place, multiple fault diagnosis plays a role as important as single fault diagnosis because the risk contribution due to multiple faults is much higher than single faults. The major reason for multiple faults is common cause failure.

Multiple fault diagnostics is challenging because of the interactive nature of most faults (Dash and Venkatasubramanian, 2000). In a complicated process such as a nuclear power plant, the control systems will be involved in the process. The interaction of different faults through a closed control loop will make the symptoms more complicated. The non-linearity makes it even impossible to develop analytical methods to infer multiple faults simply based on the information contained in single faults.

1.4.6. Complex systems

When most FDI techniques are applied to a complex system such as a nuclear power plant, some serious difficulties may occur. These difficulties are:

- Many input variables may be involved in a model that makes the model accuracy deteriorate significantly.
- Faults in many subsystems may have the same symptoms.
- The system interaction or controller interaction among the subsystems may make the cause-effect relationship highly complicated.

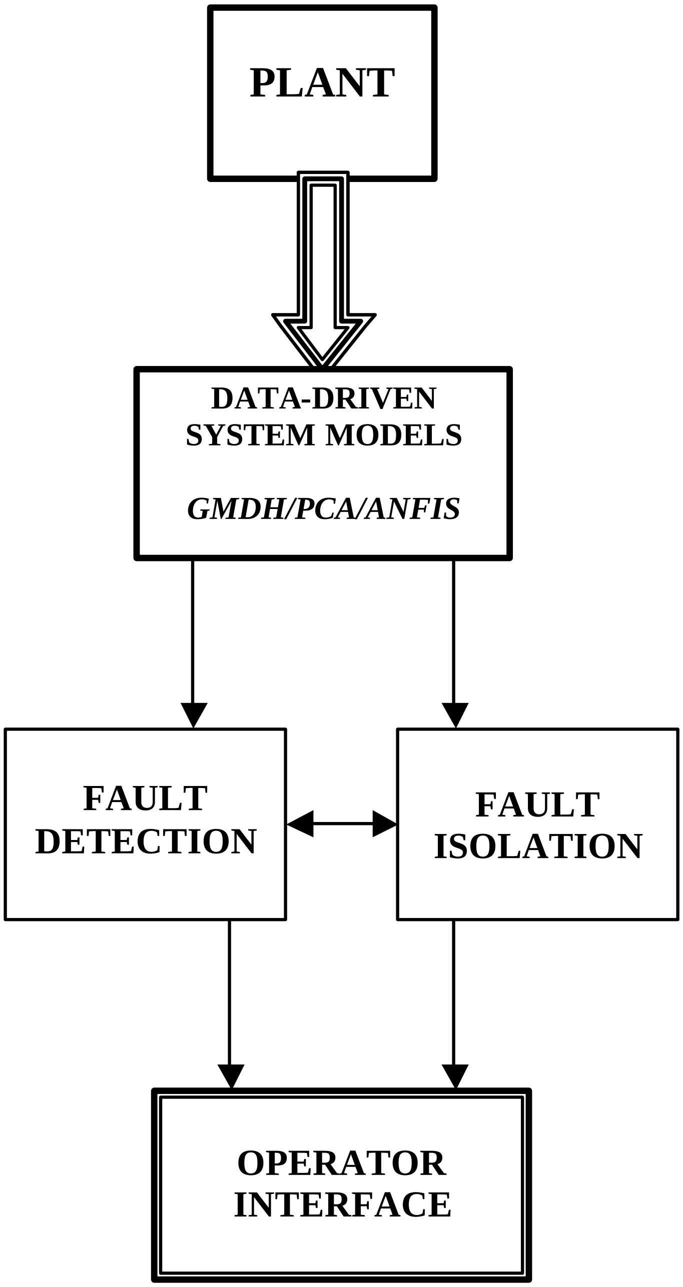


Figure 1.2. The schematic shows the FDI system functional modules.

1.5. Organization of the Report

Pattern classification techniques for fault isolation are presented in Section 2. The development of data-driven models for the UTSG is described in Section 3. The results of applications of FDI for single faults in the UTSG system are described in Section 4. Detailed PCA-based approaches for FDI and comparison with parity space method are described in Section 5. The ANFIS based approach and multiple observer digraph method for FDI are presented in Sections 6 and 7, respectively. The results of applications are also presented in Sections 6 and 7. Section 8 describes the application to the extended system and the development of the demonstration module. Concluding remarks and suggestions for future research are given in Section 9. A complete bibliography appears at the end of the report. A summary of FDI computer codes and guidance for executing these codes are described in Appendix A. A CD-R, with all the computer codes developed during the project, plant data, and appropriate *readme* files, is attached to the report.

2. PATTERN CLASSIFICATION TECHNIQUES FOR FAULT ISOLATION

(neriFinalReport02-03)

2.1. Introduction

Pattern recognition techniques implemented in this research form a key step towards the successful application of FDI algorithms to a nuclear power plant SG system. Since all types of data driven models describe the system operating status, the residual patterns are formed through the comparison of system measurements with model predictions. The residual pattern generally indicates the existence of any possible fault in a postulated device.

In this research, pattern classification techniques are used for fault device isolation from the coupled system measurements. This is based on the premise that each device fault results in a unique pattern of the residual vector. Otherwise, it is impossible to perform fault isolation. Since different pattern classification techniques accommodate different data structures, several different kinds of pattern recognition methods, such as rule-based expert system, data clustering, and PCA of the residual vectors, are developed and implemented.

Pattern recognition methods are concerned with the identification of pattern classes from among a set of multiple patterns or behaviors, and are based on the comparison of known patterns to those derived from process measurements and the derived signatures [10]. Patterns, also referred to as signatures, are derived from different types of transformations of measurements. Raw data can also be used directly in pattern recognition in order to find abnormal changes in the system. However, it is very difficult to do so due to the huge number of variables, extreme complexity and overlapping of system variables in measurement data space. In fact, feature selection and extraction is always a problem that must be considered by every system designer. The feature selection, of course, is closely related to other problems of pattern recognition. From this point of view, GMDH/PCA serves as a data representation and feature formulation approach and is developed in this research.

In the process dynamics considered in this thesis, the interaction between system variables causes the overlapping of model residuals patterns, especially during transient case. An example is that narrow range (NR) level sensor in a steam generator, feed flow sensor and steam flow sensor measurements are strongly coupled through the feedback of the water level controller. Thus, efforts have been put in appropriate feature extraction and selection of pattern recognition techniques.

Uncertainty, such as the random noise, is another problem in dealing with industrial systems. Therefore, the robustness of the pattern recognition, which will generate out reliable output, is very important for the successful implementation of FDI in a nuclear power plant. Rule based expert system, PCA, and distance comparison in residual space are utilized and evaluated in this study. In addition, several other pattern classification methods such as Fuzzy C-means and discriminant function [27] are also studied so that suitable methods can be selected for the current application.

2.2 Pattern Recognition Techniques

Pattern recognition often includes the process of feature extraction and feature classification into clusters. Many techniques are focused on finding optimal objective functions for all clusters of data. Cluster analysis is the search for structure in data sets or samples [10, 12]. A mathematical model of the prediction residuals is a vehicle for organization, recognition, and representation of the information included. Two main types of pattern classifier models exist: supervised learning (parametric and non-parametric models) and unsupervised clustering.

Supervised learning uses data groups from which classifiers can learn how to organize data into indicated groups. This information about the clusters is supplied from a knowledge base. The most fundamental principle for the problem of classification is the Bayesian decision theory [27]. Many models based on this theory are postulated in parametric form, and the data are used to identify the parameters, such as the maximum a-posteriori probability. The basic principle of Bayesian decision theory is summarized below.

Suppose we have two classes C1 and C2 in the sample space with *a priori* probabilities of P1 and P2, where P1+P2 = 1. Our objective is to partition all sample observations into class C1 and class C2. For all the samples in the space, the probability densities for both classes are greater than zero, thus misclassification exists.
The overall probability of correctly classifying samples from class C1 is:

$$P(d1/C1) = \int_{S1} p(x/C1)dx \qquad (2.1)$$

p(x/C1): the conditional probability density for measurement x from class C1.
S1: part of sample space, patterns that are classified to C1.
d1: the decision that samples belong to class C1.

The probability of correctly classifying samples from class C2 is

$$P(d2/C2) = \int_{S2} p(x/C2)dx \tag{2.2}$$

p(x/C2): the conditional probability density for measurement x from class C2.
d2: the decision that samples belong to class 2.
S2: part of sample space, patterns that are classified to C2.
S1+S2 = S, the sample space.

The probability of incorrectly classifying samples from class C1 is given by

$$P(d2/C1) = \int_{S2} p(x/C1)dx \tag{2.3}$$

And the probability of misclassifying samples from class C2 is given by

$$P(d1/C2) = \int_{S1} p(x/C2)dx \tag{2.4}$$

The cost function from misclassification is

$$P_1 L(C_1, d_2)P(d_2/C1) + P_2 L(C_2, d_1)P(d_1/C2) \tag{2.5}$$

L(C1,d2) denotes the cost caused by the decision d2 when samples are from class C1.
L(C2,d1) denotes the cost caused by the decision d2 when samples are from class C2.
The Bayesian decision procedure is to minimize the cost function.

The theory of Bayesian decision is fundamental to the development of the discriminant function. Unfortunately, Bayesian decision theory assumes that the forms of the underlying density functions are known and all of the parametric densities are unimodal. However, in practice, many problems involve multi-modal densities and we usually are not able to estimate their distributions.

Fortunately, non-parametric methods are designed to deal with arbitrary distributions without the assumption that the forms of the underlying densities are known [27], such as the PCA and rule base expert systems. The discriminant function is also one type of non parametric learning algorithm. The main goal in discriminant analysis is to construct a linear composite of the observed variables so as to maximize the difference between groups.

The new variables constructed from the linear combination of input variables:

$$Y = a_1 X_1 + a_2 X_2 + ... + a_n X_n \qquad (2.6)$$

Where n is the number of variables used to separate the groups. Define B as the between-groups sum of squares and Cross-Products matrices) and W as the within-groups SSCP. Then

$$B = (m_1 - m_2)(m_1 - m_2)^T$$

$$W = \sum_{j=1}^{2} \sum_{x \in D_j} (x - m_j)(x - m_j)^T \qquad (2.7)$$

$$\frac{B}{W} \propto \frac{a' Ba}{a' Wa} = l$$

Where: $a = [a_1, a_2, ..., a_n]$

m_1 and m_2 are the mean vectors for group 1 and group 2, respectively.
D_j represent the j^{th} data group.

The well-known Fisher discriminant analysis is the procedure to find general eigenvalues and eigenvectors of the problem: $Ba = l Wa$.

Neither parametric nor non-parametric methods can be universally effective because the extensive change of data structure, especially for the complex SG system. It is obvious that no clustering criterion or measure of similarity will be applicable to all situations, and selection of a particular criterion is at least partially subjective, and should always be open to question. Therefore, we compare several pattern recognition algorithms in order to find the optimum.

2.3. Feature Extraction

Feature extraction is usually the first step in patter recognition, and is very critical in correctly classifying test samples into groups. An ideal feature extractor would yield a representation that makes the job of the classifier easy [27]. The traditional goal of the feature extractor is to characterize an object to be recognized by measurements whose values are very similar for objects in the same category, and very different for objects in different categories. Although raw data from an industrial plant can represent the characteristics of current plant status, the data from different plant conditions overlap with each other and thus are very difficult to classify. By the careful selection of input and output variables, GMDH modules are able to extract the system abnormal characteristic changes into an error space and discard the unimportant part of the information for FDI. These error vectors are easier to be isolated with proper classification algorithms compared with original data from plant measurements. A rule-based expert system is very useful for its robustness and easy implementation in analyzing the properties of the residuals. The PCA method is also very powerful when a specific error vector changes toward a certain direction due to a fault in a specific device. Discriminant functions can be applied if there is no special shape found in patterns.

Different data features will dictate how to arrange effective classification methods. The following methods describe the main purposes of these classification methods in general.

- Hierarchical methods

  This group of methods had their origin in taxonomic studies [10,12]. The connections between groups are calculated using the Mahalanobis distance, which is a type of weighted distance between data points considering the covariance of vectors $E * \sum * E^T$. Data groups are divided into subgroups according to the distance between groups.

- Graph-theoretic Method

  This technique is suited for data with *chains*, or pseudo-linear structure [10]. The criterion for clustering is typically some measure of connectivity or bonding between groups of nodes. The PCA method is useful when these chains are not parallel, otherwise Fisher's discriminant function can transfer them into more distinct scale space. Breaking edges in a minimal spanning tree to form sub-graphs is often the clustering strategy.

- Objective Function Methods

  These methods ordinarily allow the most precise formulation of clustering criterion. For each cluster, a criterion or objective function measures the *desirability* of clustering candidates. If one uses the Euclidean distance as the similarity measure of data and as a measure of cluster quality, the objective function is the sum of squared errors. This clustering criterion is called a minimum variance objective. This algorithm is regarded as the most applicable method if the clusters are basically hyper-spherical and of roughly equal in proportion.

2.4. Rule Based Expert System

Clustering of numerical data forms the basis of many classification and system modeling algorithms. The purpose of clustering is to identify natural groupings of data from a large data set to produce a concise representation of a system's behavior [27]. A rule-based expert system is specific to a system. Its robustness makes it the most commonly used diagnostics tool for operators in power and process plants [1].

The recognition system of human beings is a complex rule based expert system in some sense. Basic rules for identifying a pattern are established to recognize the specific type of data character. A simple example is to distinguish a passenger car from a truck. The rules may be simple or complex.

It is a typical deductive process. The correct recognition of certain patterns depends on the how accurate the rules are. The rules are also based on the collection of experience. The rules could be added or deleted as the environment changes. Therefore, a rule-based expert system can always be revised and thus, evolves with time.

The robustness of rule-based expert systems compared with other methods, such as PCA, comes from its capability to include all distinguished characters into it, without being limited to a specific data structure.

2.5. Nearest-Neighbor Classification

The nearest-neighbor classifier relies on a metric or distance function between patterns. The test data are grouped into a class that is nearest to the data set. This is decided by a distance metric. The metric must have the following four properties [27].

i. Nonnegativity: $D(a,b) \geq 0$;
ii. Reflexivity: $D(a,b) = 0 \Leftrightarrow a = b$;
iii. Symmetry: $D(a,b) = D(b,a)$;
iv. Triangle inequality: $D(a,b) + D(b,c) \geq D(a,c)$.

Let C_i (i=1,2, …) represent the reference points for the ith class. C_i may be, for example, the mean value of a number of samples from the ith class. The Euclidean distance from point x to point C_i is defined as

$$d(x, C_i) = \|x - C_i\| = \sqrt{(x - C_i)^T (x - C_i)} \qquad (2.8)$$

Other general or weighted distances such as the Mahalanobis distance can also be used in this classification method, especially for special type of pattern classification such as number-alpha recognition, where Euclidean distance is of little use.

The Mahalanobis distance is a multivariate measure of the separation of a data set from a point in space. It is a universally applied metric in statistical classification study. It is also the criterion minimized in linear discriminant analysis. However, singular data matrices are not suitable for this type of distance calculation.

The minimum distance classifier is a subclass of linear decision functions. The measurement x is assigned to the class which has the smallest distance $\|x - C_i\|$ to it. However, since multi-dimensional residual patterns from data driven models have the same scale unit, when we assume those error patterns are separable in the residual space, it is convenient to do classification according to the nearest neighbor rule. The residual pattern due to a specific component abnormality may concentrate in a certain area in the multivariate residual space, whose scattering relies on severity, random noise, and other uncertainties.

The advantages of the nearest neighbor classifier over the PCA method include: PCA classifier depends very heavily on the first principal component of residual patterns, if the residual vectors do not concentrate heavily in one direction, vector projection is not as attractive as other methods.

2.6. PCA Method for Data Clustering

PCA is a method used to reduce a set of observed variables into a relatively small number of components that account for most of the observed variance. This is accomplished by mathematical linear transformations of the observed variables under two conditions. The first condition is that the first component accounts for the maximum amount of variance possible, and the others following the decreasing order of importance. The second condition is that all components are uncorrelated with each other.

The PCA algorithm is also a powerful modeling tool for industrial systems. The transformed data sets represent the variance directions arranged in descending order. PCA is especially useful when the high dimensional data sets could be represented by the first several PCs if heavy covariance redundancy exists in the data sets. Since a few number of dimensions (less than 3) is typically sufficient to represent the data, the PCA algorithm has a computational advantage in data clustering. The important prerequisite is that the data in groups must have different main variance direction and can be summarized by one or two PCs.

Principle of PCA algorithm in fault isolation:

Suppose we have several multi-dimensional data groups, g1, g2, ... gm. We can manipulate PCA on every group and thus get:

[PC_{11}, PC_{21}, PC_{31}.. PC_{n1}] and $[l_{11}, l_{21}, l_{31} \ldots l_{41}]$

[PC_{12}, PC_{22}, PC_{32}.. PC_{n2}] and $[l_{12}, l_{22}, l_{32} \ldots l_{42}]$

[PC_{13}, PC_{23}, PC_{33}.. PC_{n3}] and $\left[l_{13}, l_{23}, l_{33}...l_{43}\right]$

[PC_{14}, PC_{24}, PC_{34}.. PC_{n4}] and $\left[l_{14}, l_{24}, l_{34}...l_{44}\right]$ (3.9)

.

.

.

where λ_{ij} is the ith eigenvalue for data group j.
If $\lambda_{11} >> \Sigma\lambda_{1i}$, $d_1 = PC_{11}$ can be used to represent g1.

The same process is applied to other groups generates $d_2 = PC_{12}$, $d_3 = PC_{13}$,… Therefore a matrix representing all group variation directions is $[d_1, d_2,..., d_m]$.

When a test data set comes in, it will be projected onto each direction or PCA module. The residuals (distance from test point to the ith direction) of using this data direction to fit test data is given by

$$r_i = X(I - d_i d_i^T)X^T \qquad (2.10)$$

The closeness of test data to the ith data group is measured by

$$\frac{r_i}{r} \qquad (2.11)$$

where $r = X^T X$ is the Euclidean length of the test data vector.

Before the test data enter into the group isolation engine, it is necessary to scale the test data to the center of every group. The scaling method selection is flexible, but the training data must be scaled the same as the test data.

Since we could assume that there are changes only in residual magnitude during a transient process, it is feasible to find changing direction of the residuals for one specific type of fault without caring too much about the changing process. In other words, as long as the residual direction remains constant for a transient fault case, and different types of faults give rise to different directions in the hyperplane, we can utilize the projection of residuals (from our system simulation module) onto these vectors representing fault directions to find the most likely type of fault.

2.7. Selection of Different Models for Pattern Classification

No one pattern classification algorithm is universally effective and generally useful. Therefore, different models of the same process are usually utilized and compared with each other. Since different models may compete, and they may also supplement one another, it is possible to find an optimum solution through comparison. This section compares the performance of several pattern classification methods in order to decide which ones are more suitable for the robust on-line FDI process implemented in this research.

- Discriminant Function Algorithm:

For the SG system studied, multivariate variance analysis was introduced here to perform error space data cluster analysis. The results are shown in Figures 2.1 and 2.2. The first Figure 4.1 shows the pattern classification results for training data collected under 100% nominal power level, we can find it performs very well. However, this classifier performs very bad for the test data as shown in Figure 2.2, which is collected under 80% nominal power level.

It is obvious that discriminant function method is very vulnerable to the subtle changes in residual vectors, even if the vectors are very similar, as shown in the figures. Although the residual vectors from test data are actually very close to the vector form training data, the ANOVA gives out large different scale in the new transformed analysis space. Therefore, the discriminant function algorithm will not be adopted in the final FDI structures in this research.

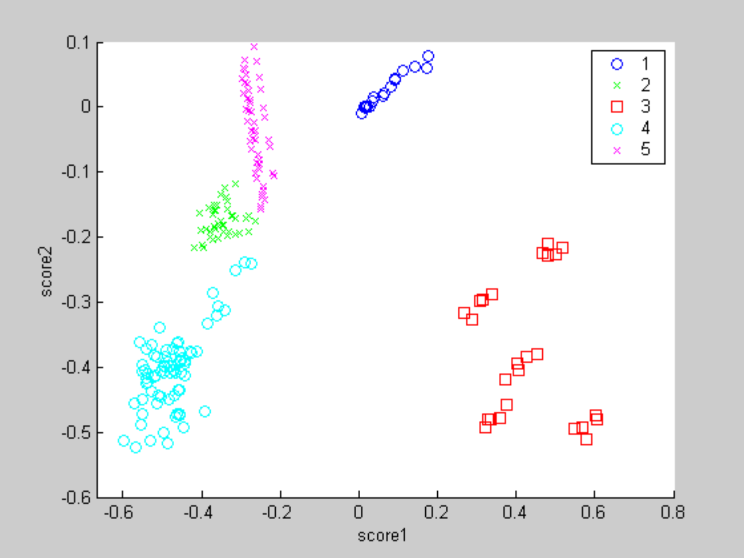


Figure 2.1. Pattern classification results for training data sets using discriminant function.

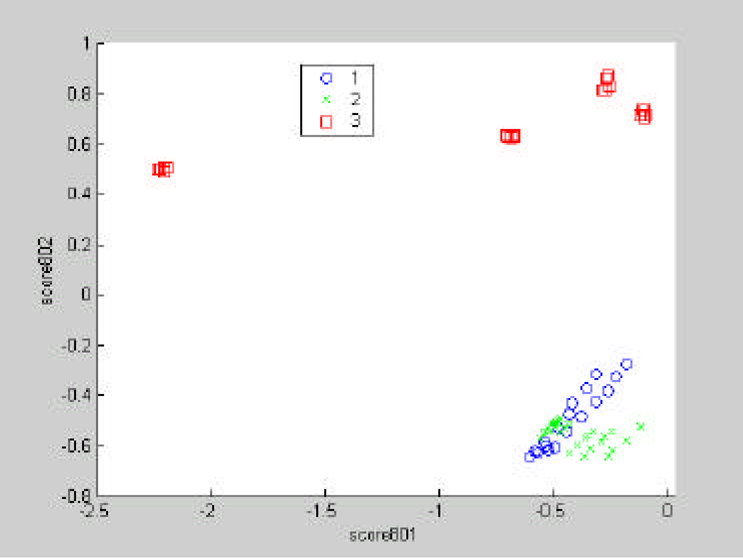


Figure 2.2. Pattern classification results for test data sets using discriminant function.

The definitions of various fault types are given below.

Fault 1: NR degradation
Fault 2: Feed water flow sensor degradation
Fault 3: TCV1 degradation
Fault 4: Steam flow sensor degradation
Fault 5: Steam pressure sensor degradation

- FCM (fuzzy C means) for Residual Space Data Clustering

The FCM method is substantial to the variance of data group. This method may not provide satisfactory results if a large difference exists between the data distribution groups. Therefore, the FCM algorithm will not be used in the final FDI method developed in this research, even though it has good properties in considering uncertainty in the data.

- Rule Based Expert System

By comparison, the rule-based expert system is the most robust, intuitive, and easily applicable pattern classification method in this research. This algorithm can be applied in both static and transient study for the SG system. The difficulty is in extracting useful properties from the large amounts of collected information, which is also experience based. The results of applying rule-based expert system in FDI will be discussed in detail in Section 4.

- Principal Component Analysis (PCA)

The PCA method for data classification performs very well in dealing with the data with a few main directions or principal components. The result of application of PCA for residual pattern classification is described in Section 4. Good performance is found when PCA is applied to transient case residuals analysis. However, PCA algorithm will perform badly if data are uniformly distributed in groups.

- Minimum Distance Cluster

Minimum distance analysis is very useful when the data groups are distributed uniformly. The implementation of this algorithm is easier than other methods. Disadvantages include the lack of a confidence level, and the lack of ability to detect unknown faults. The implementation results will be discussed in Section 4.

Multiple pattern classification techniques have been utilized in residual space analysis in order to find the faulty device in a steam generator system. Different pattern recognition modules transmit different kinds of information or perspective of properties stored in residual space. They extract different amounts of information about the process they represent, even though the same residual data is processed. Through the comparison of their performance in the SG system, we finally select three types of methods to be included in the FDI structure, for the purpose of cross checking and compensation. They are: rule based expert system, which is applied to both static and transient analysis; PCA, which is applied to both static and transient analysis, but performs better for transient conditions; and the nearest-neighbor pattern classification method, which is only applied to static system analysis because the residual cluster center are not constant during transient processes.

Based on the methods of pattern classification discussed here, the modeling residuals will be analyzed in order to isolate the faulty component.

3. DEVELOPMENT OF DATA-DRIVEN MODELS FOR A U-TUBE STEAM GENERATOR

(neriFinalReport02-04)

3.1 Introduction

A nuclear plant steam generator is one of the key components in a nuclear power plant. Depending on the complexity of the model about 20 state variables are used to describe its response characteristics. The causal relations between these variables are so complicated that it is a tedious task to explore accurate mathematical formulas to express all the interactions among them. Fortunately, the data driven model provides us a practical approach in reaching this goal. The development of these models requires an extensive database. This database could be a combination of plant measurements and data generated from a physics model simulation. With the aid of GMDH, ANN, PCA or other modeling algorithms, process measurements from an industrial system are used to generate accurate models that can simulate system behavior. New operational data were then used to monitoring changing conditions of the process or field devices.

Different forms of data-driven models are developed for the U-Tube Steam Generator (UTSG) system, for the reason that they may transmit different kinds and amount of information about the process they represent, even if they model the same process. In general, different models may compete and supplement each other.

Traditional GMDH models have been used in many areas such as industry monitoring, market prediction, ocean wave study, and others. GMDH modeling is developed in this research to map relationship between a desired output and a set of related inputs. Models developed using steady-state or quasi steady-state operational data are well-suited to detect and isolate incipient faults under stationary operation conditions. However, some faults may occur during the transient operation, such as reactor start-up, and it is necessary to monitor the plant at these conditions. Moreover, some fault types may not be detected during steady-state operation, especially in devices that contain moving elements, such as valve actuators, control rod drives, and some transmitters. An example is the case of four turbine control valves (TCVs) degrading simultaneously, in which case static models do not provide any early warning of the impending failure. But transient data analysis would provide this information, since actuators respond during plant maneuvers.

3.2. GMDH Algorithm for Modeling the UTSG System

The *Group Method of Data Handling* (GMDH) approach is a self-organizing modeling method, that may be utilized for automatically mapping the complicated relationships among a set of process variables and control functions. The GMDH algorithm (described in Phase-2 Report) is extensively used in this research due to the advantages of this algorithm over other modeling methods:

- The capability of mapping nonlinear relations between model input and output variables.
- The flexible selection of GMDH network structure during the training phase, instead of pre-estimated structure.
- GMDH can be designed to be free from the problem of over-fitting. The GMDH method divides the input data into training and checking sets in order to select an optimal model structure.

The input-output relationship for a specific cell in the GMDH structure is often described by a second order polynomial, although other necessary functional terms can also be included into this function.

3.2.1 Database generation

In this research the necessary database has been generated using a full-scope simulation code for a typical 1,300 MWe PWR, developed by the North Carolina State University. The code is able to simulate PWR normal operations under different power levels from zero power to 100% power. In addition, reactor operation under different fault conditions can be simulated. Furthermore, many specific types of abnormal cases can be studied such as sensor fault (drift, offset, noise), actuator fault (time constant, dead band), controller fault (proportional gain and integral gain), heat transfer capacity change, and others. Schematic of a typical UTSG with measurement points is shown in Figure 3.1.

In order to generate a good data-driven model, a large representative database must be prepared before the modeling process. One complete database is created for the conditions of BOL (Beginning of Life) in PWR, the power level changes from zero power to 100% power level. The data is collected at every 2% power level interval. A total of 480 stationary data sets are generated.

A separate steady state database is also generated for EOL (End-of-Life) condition in a PWR system considering the system characteristics change along with the change of fuel enrichment and boron concentration.

Another separate database is generated for transient study in a PWR system. The power increase and power decrease conditions were studied individually for evaluating the performance of some devices, such as the turbine control valve during these processes. A summary of data information will be given in next section.

The database under different types of faults is also generated. The details of

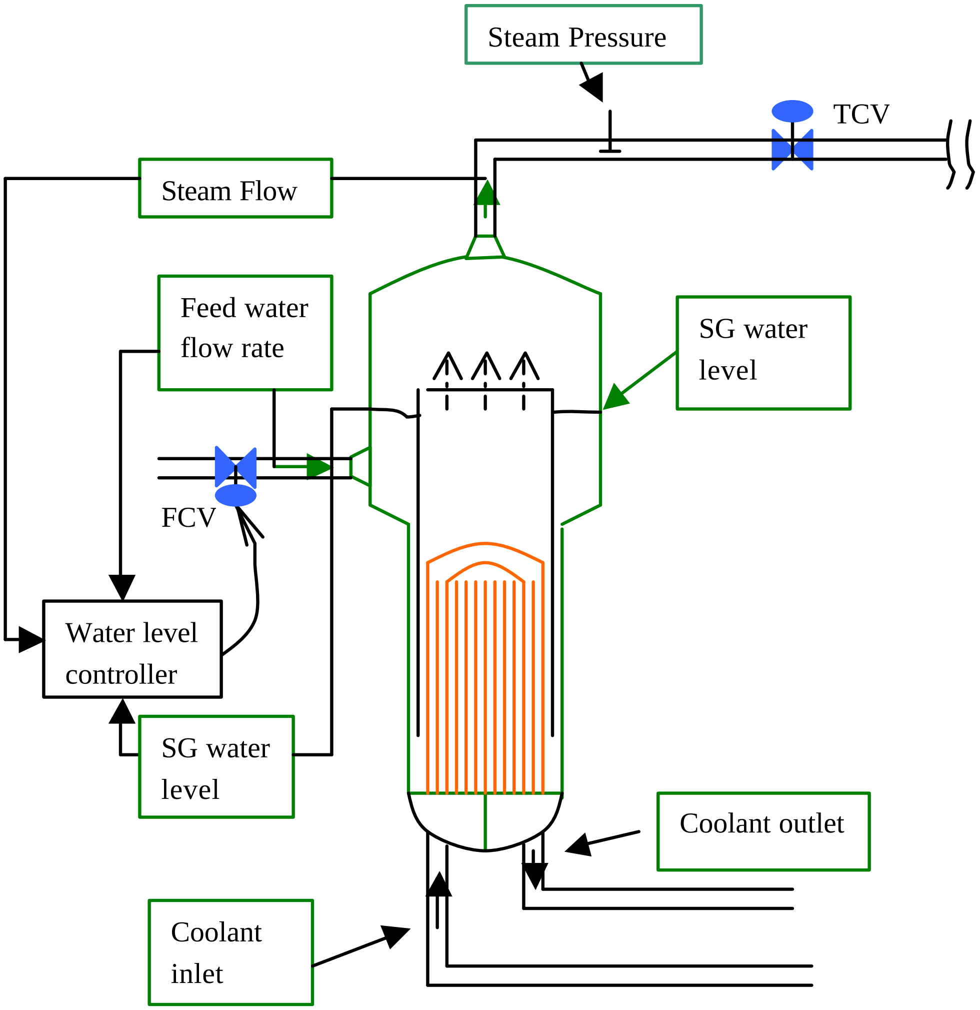


Figure 3.1. Schematic of a UTSG showing measurement points.

these are given in Section 4. About 0.2% system noise are imposed on all of the sensors related to SG system in a nuclear power plant. Thus the noise filtering becomes one property in evaluating the performance of the models established.

3.2.2 Static system model

Static GMDH models are developed for nine variables of the UTSG system in order to perform redundant analysis for this complex system. The model outputs are compared with system measurements to check whether the models perform well. The residuals are evaluated using the percentage difference between model simulation and the actual values.

The models are created under the philosophy that there are constant parity relations among the variables in a complex system, such as the UTSG. The relationship can be expressed in a functional form:

$\Pr edicted \quad Variable \quad Y = f\left(x_1, x_2, \ldots x_n\right)$, where x_1 x_2 ,,, x_n are predictor variables such as NR water level, TCV position, etc. Furthermore, the relationship can be approximated by higher order polynomials. Table 3.1 shows the input terms selected and the corresponding modeling errors. Note that most of the models include thermal power level as one of the input regressors, this is because we are trying to include the set point information into the models. The input selection of the referential models is based on correlations between output and input and engineering experience.

For the models with only two input variables such as FCV position and steam pressure, one additional multiplication term of those two input variables must be generated so that GMDH training process can continue and will not stop at the first layer. The various model responses are shown in Figures 3.2 – 3.13.

The model residuals show that GMDH algorithm works great in mapping relations among system variables. The largest prediction error is less that 1%. The prediction error includes two parts, the error due to noise in measurement and the modeling error. When the training process begins to learn the noise, the GMDH algorithm stops the training process automatically.

Table 3.1 List of GMDH models for PWR SG system under static conditions

Predicted variable	Input variables	Figures	Maximum error (%)
NR	Thermal power Level, Steam flow rate, Feed water flow	5.2&5.3	0.7
Feed water flow rate	Thermal power level, Steam flow rate	5.4&5.5	0.6
FCV position	Thermal power level, Steam flow rate, TCV1 flow rate	5.12&5.13	<0.6
Steam flow Rate	Thermal power level, FCV position	5.6&5.7	<0.35
Steam Pressure	Hot leg temperature, Steam flow rate	5.8&5.9	<0.45
TCV1 flow	Thermal power level, Steam flow rate, FCV position, NR level for SG1, TCV1 position	5.10&5.11	<0.25
TCV2 flow	Thermal power level, Steam flow rate, FCV position, NR level for SG1, TCV2 position	5.10&5.11	<0.25
TCV3 flow	Thermal power level, Steam flow rate, FCV position, NR level for SG1, TCV3 position	5.10&5.11	<0.25
TCV4 flow	Thermal power level, Steam flow rate, FCV position, NR level for SG1, TCV4 position	5.10&5.11	<0.25

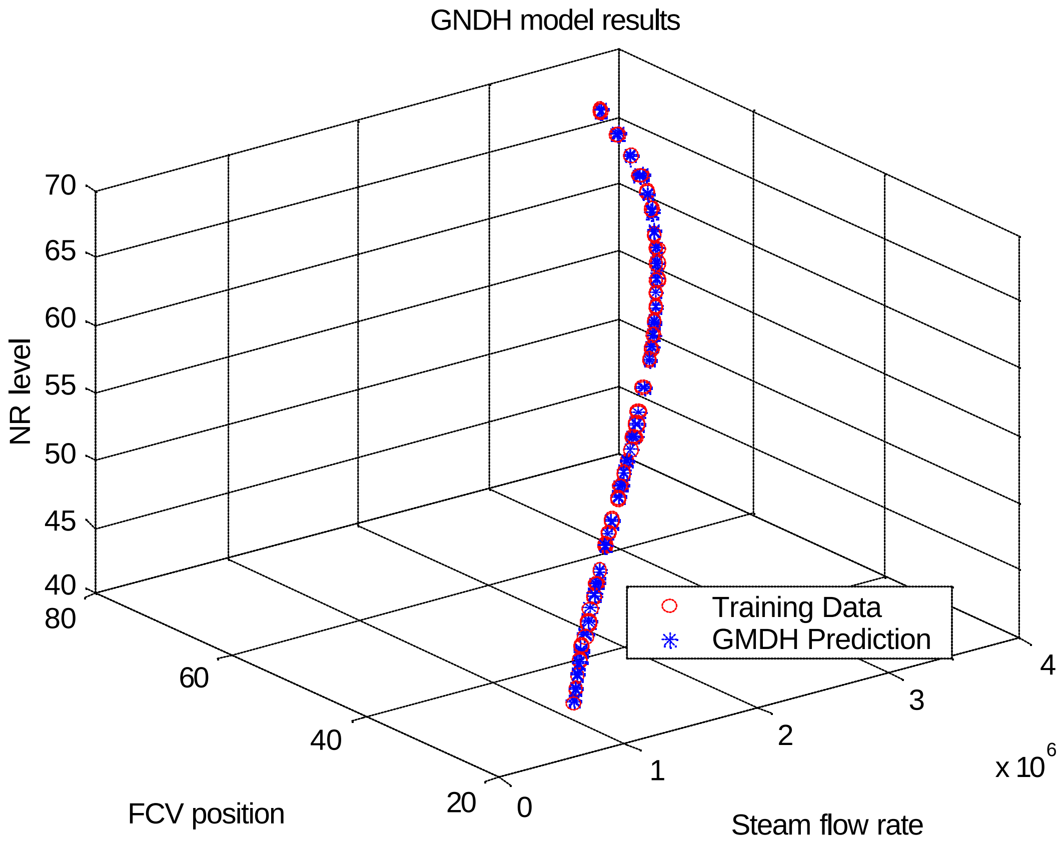


Figure 3.2. Narrow range steam generator level model with FCV position and Steam flow rate as model inputs.

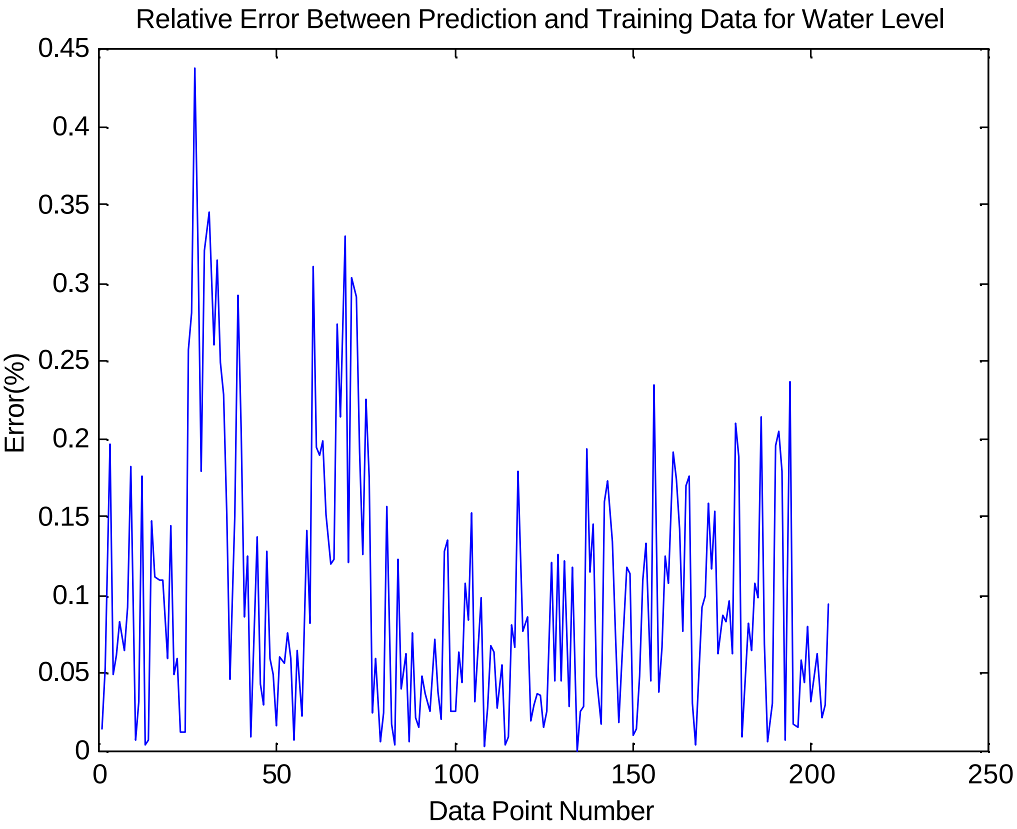


Figure 3.3 Error between training data and model prediction for NR level model.

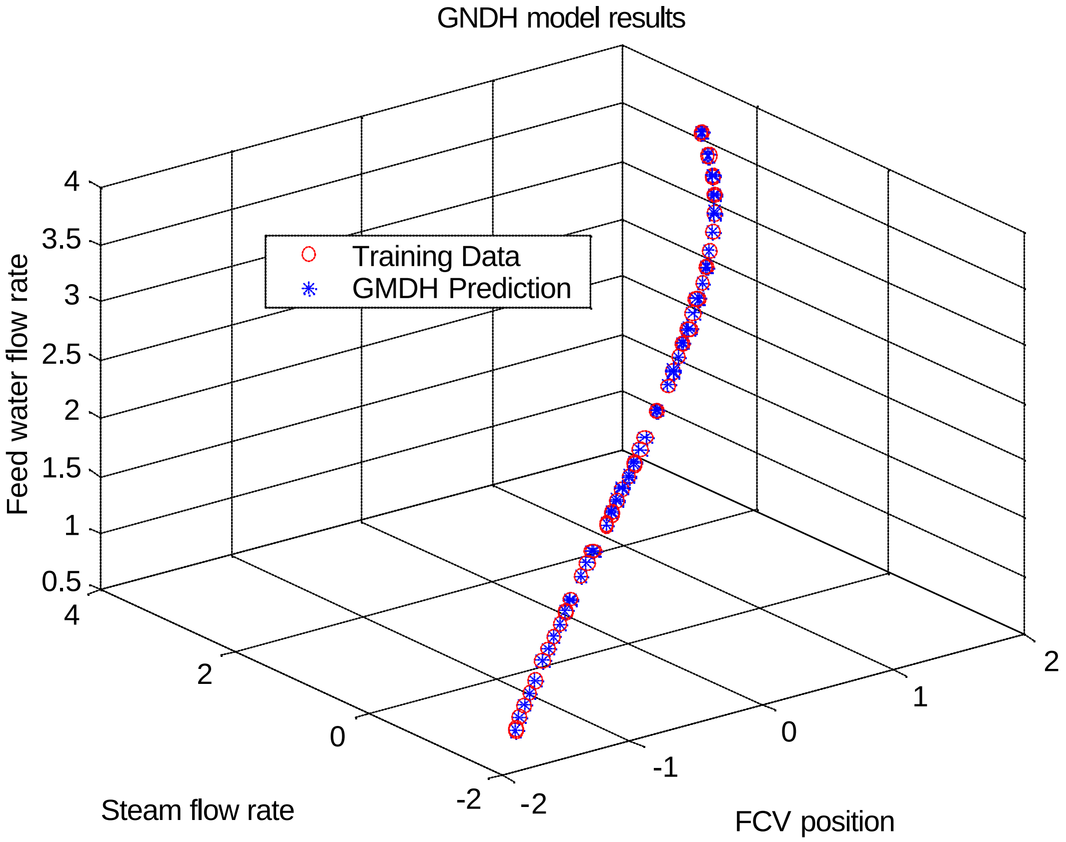


Figure 3.4. Feed water flow model with FCV position, Steam flow rate and power level as model inputs.

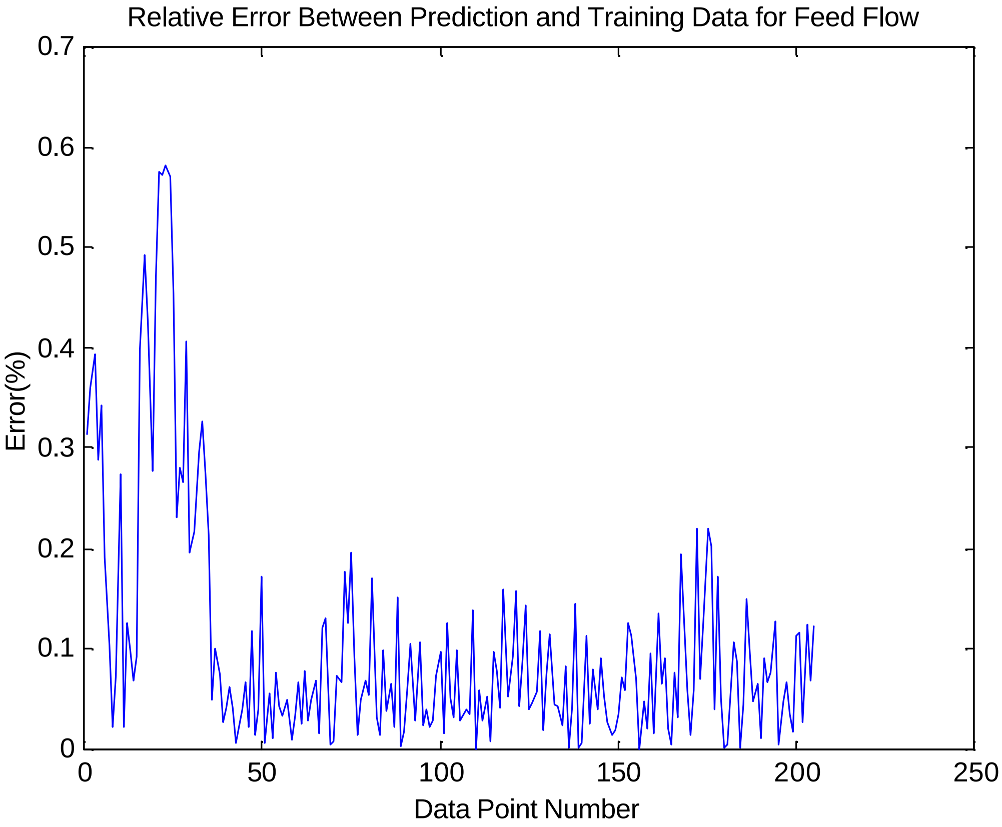


Figure 3.5. Error between training data and model prediction for feed water flow model.

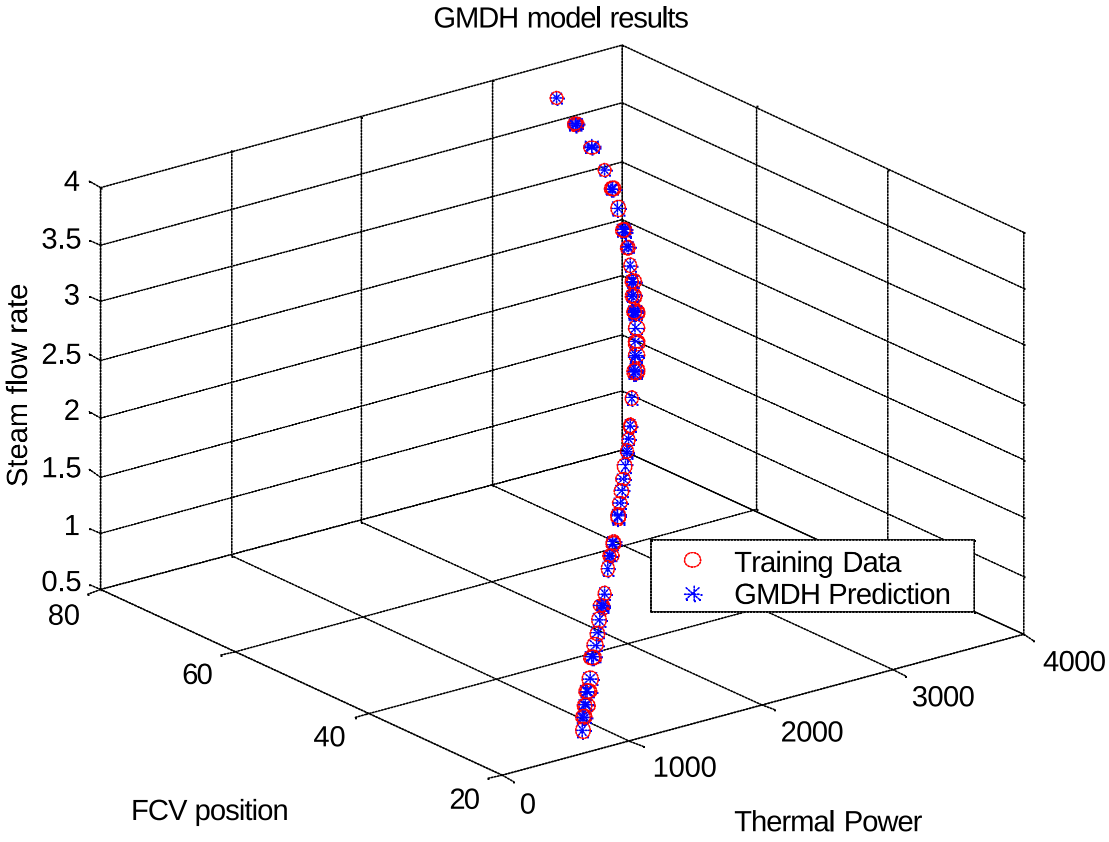


Figure 3.6. Steam flow model with FCV position and thermal power level as model inputs.

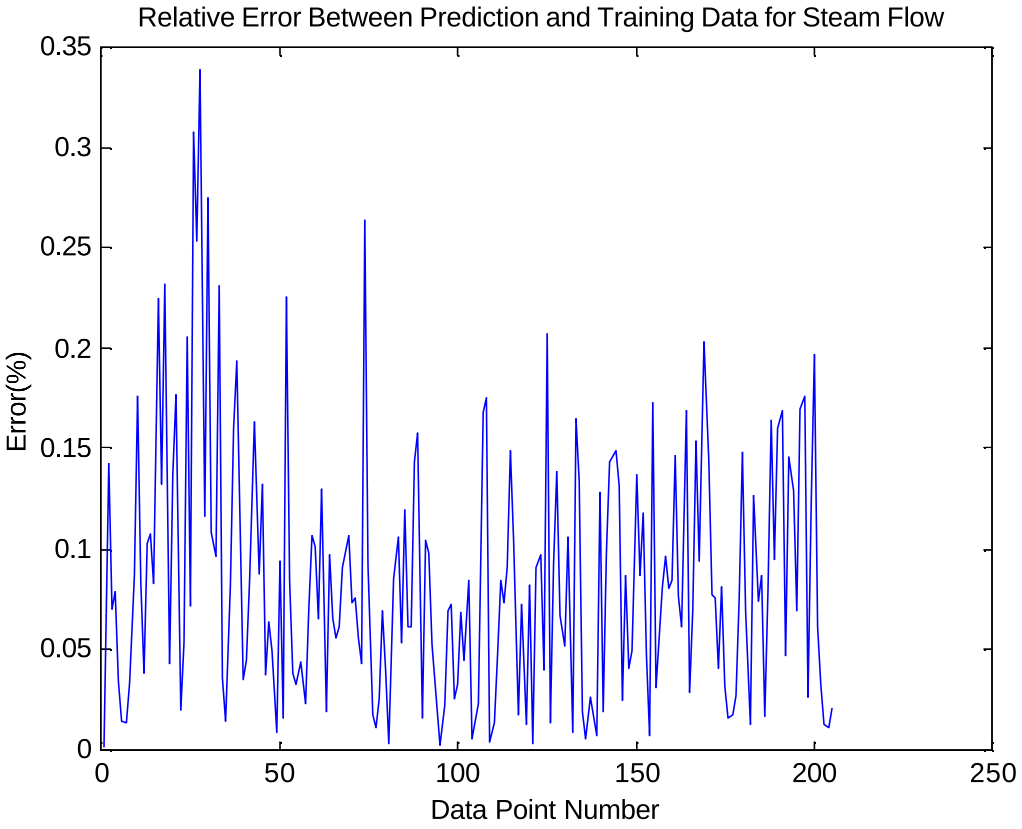


Figure 3.7. Error between training data and model prediction for SG steam flow model.

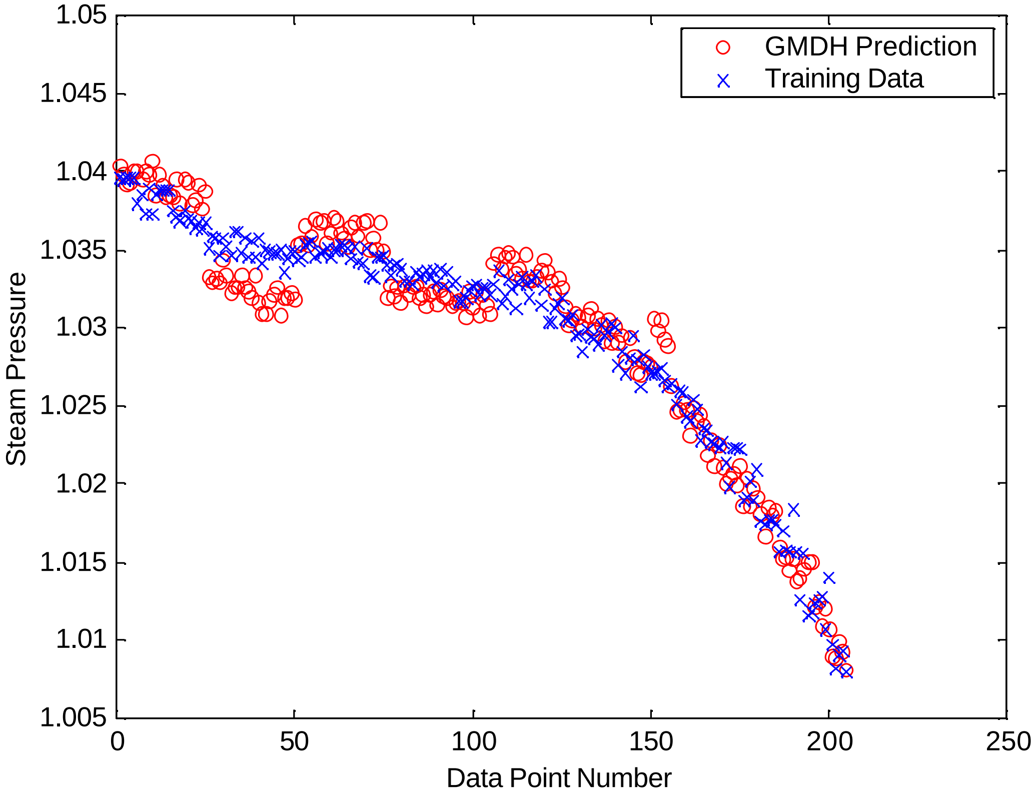


Figure 3.8. Steam pressure model with steam flow and hot leg temperature as model inputs.

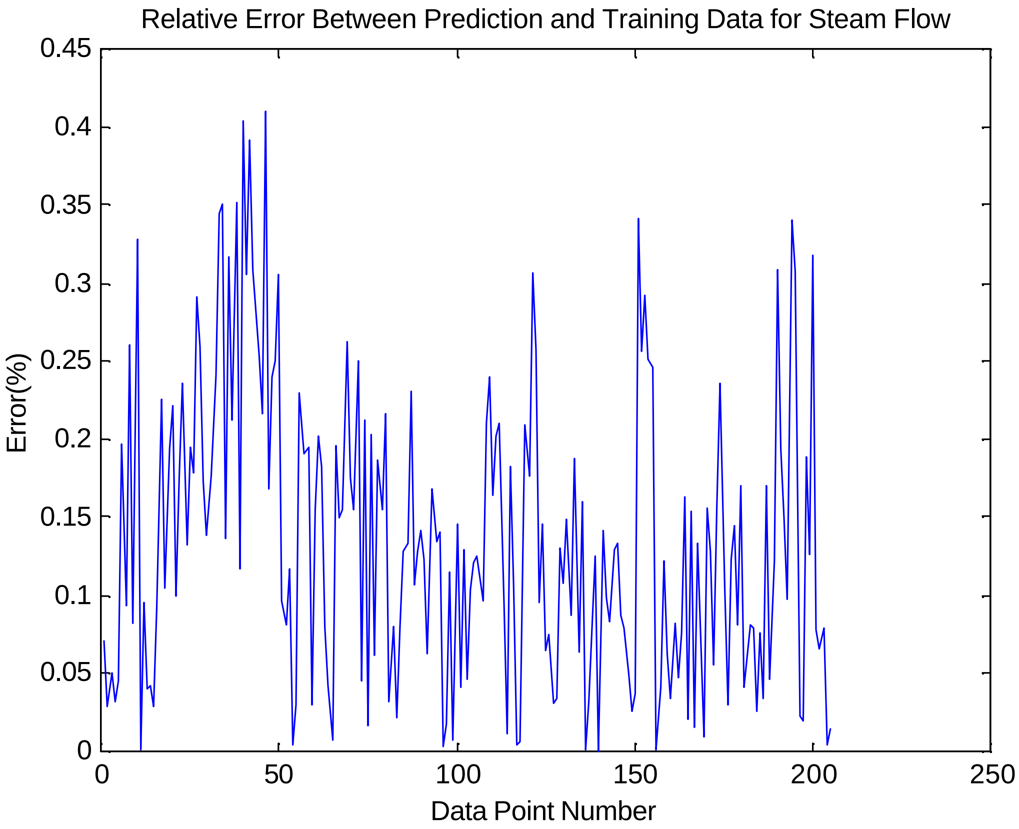


Figure 3.9. Error between training data and model prediction for SG steam pressure model.

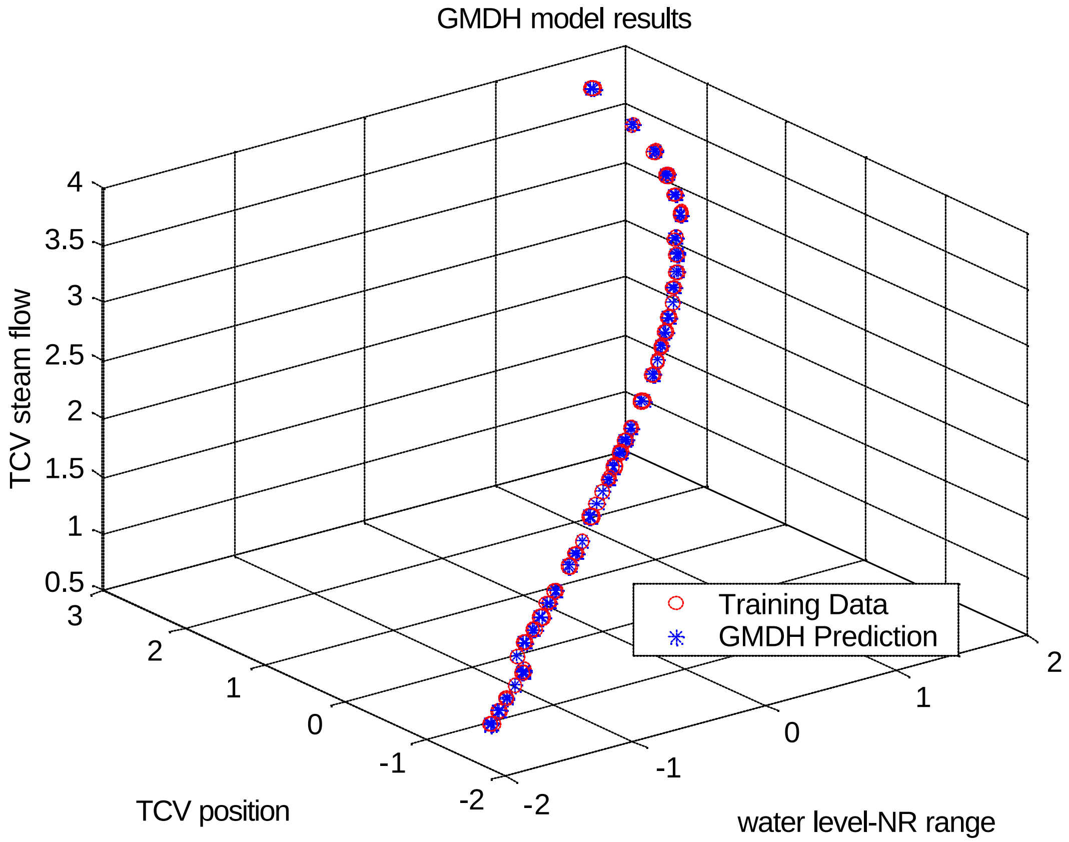


Figure 3.10. TCV flow model with steam flow, hot leg temperature and three more variables as model inputs.

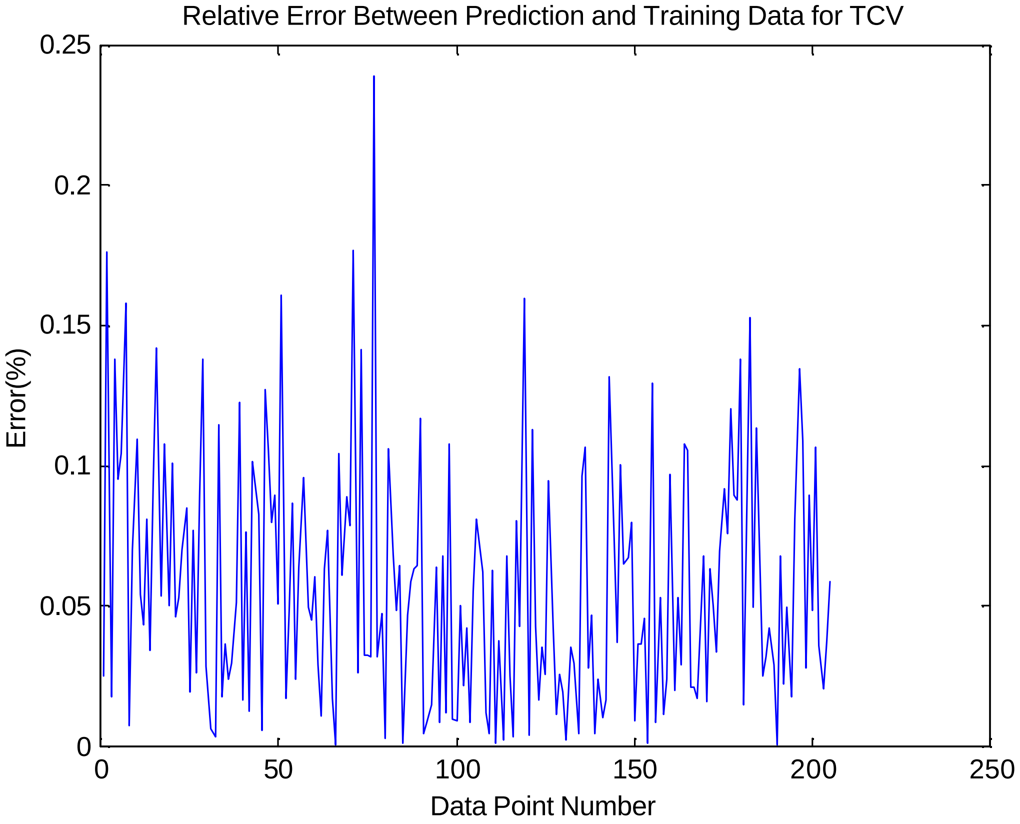


Figure 3.11. Error between training data and model prediction for SG TCV steam flow model.

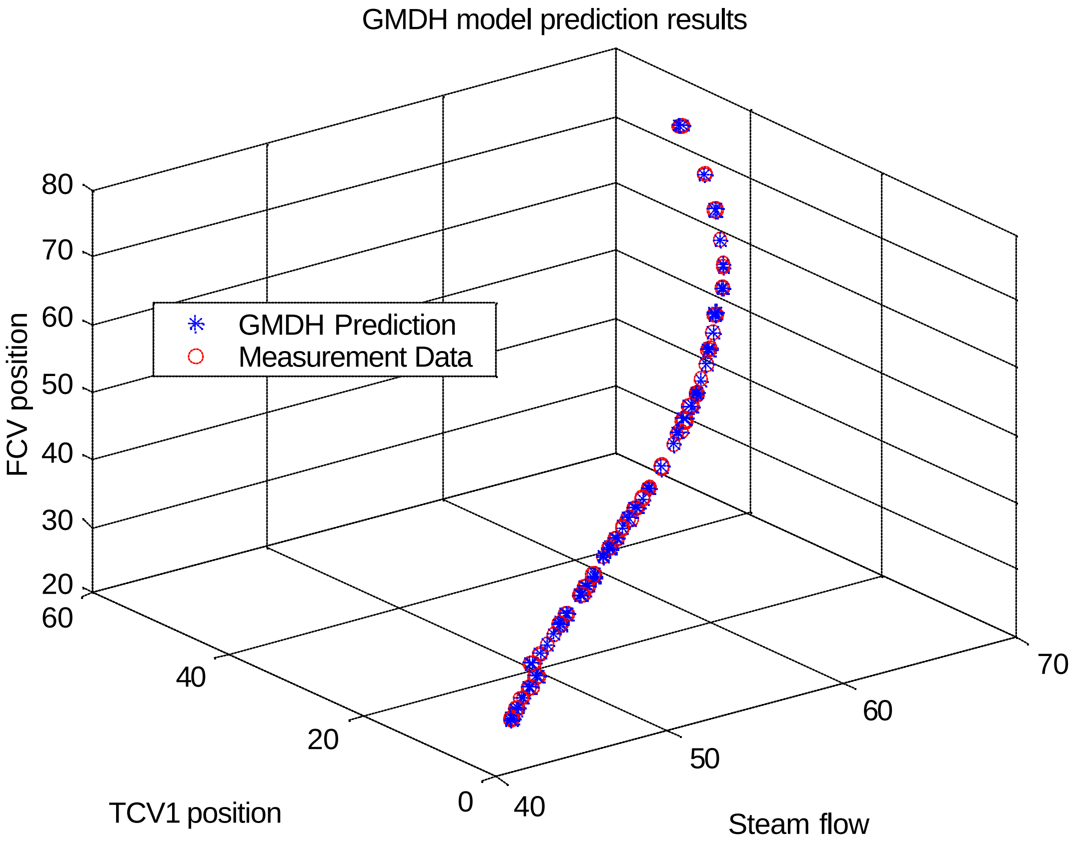


Figure 3.12. FCV Position model with steam flow, TCV position and power level as model inputs.

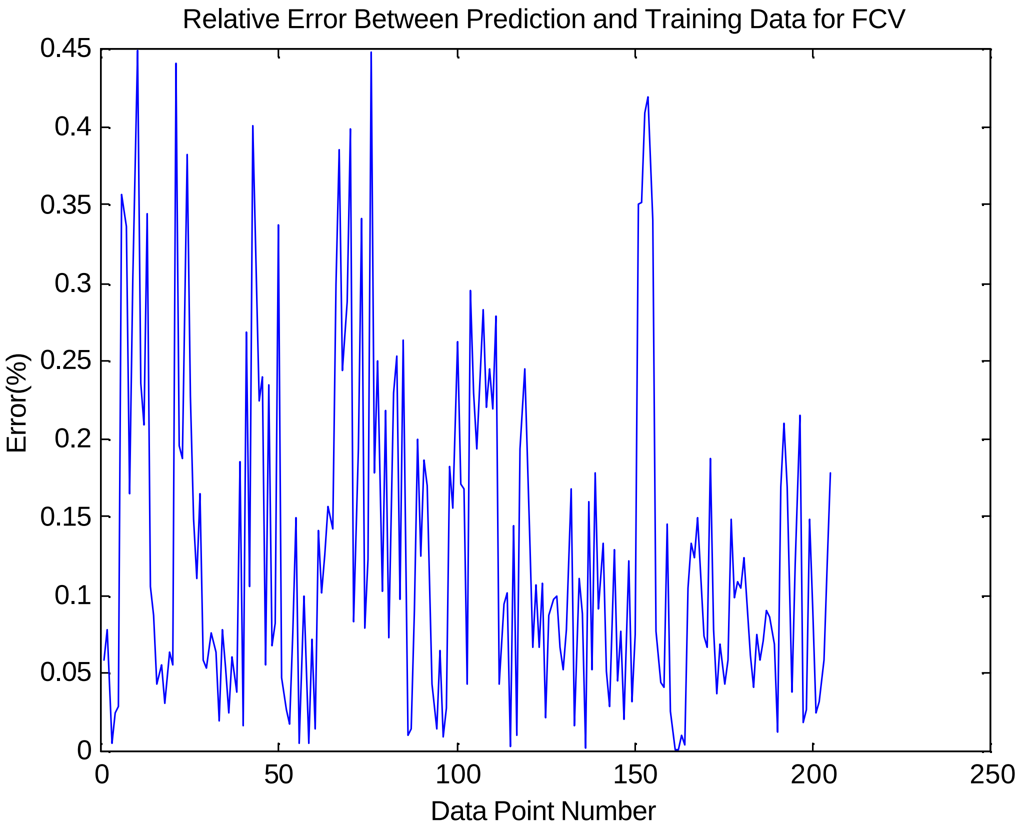


Figure 3.13. Error between training data and model prediction for SG FCV position model.

3.2.3 Transient modeling of steam generator system

Considering the fact that system faults may occur during transient PWR operation, it is necessary to develop transient system models. Another reason is that some fault types may not reveal their signature during the static condition. As an example, consider the case when four TCVs in a PWR are under similar degradation simultaneously (for example, time constant change), all the relationships among the variables will be the same when the system reaches steady state. This does not indicate an anomaly in any device. However, if we follow the transient dynamics of these TCVs during opening or closing conditions, we can identify the abnormal behavior (Figure 3.13). Therefore, transient models become absolutely necessary in this environment. It will increase our confidence of the judgment of whether there is any abnormal scenario in the system.

For time series to be included in the input data for the GMDH transient models, time lag terms need to be added. Thus the functional relationship between an output variable and the input variables becomes:

$$\mathrm{Pr}\,edicted \quad Variable \quad Y(t) = f\left(x1, x2, \ldots xn, x1_{t-1}, x2_{t-1}, \ldots xn_{t-1}, \cdots x1_{t-k}, x2_{t-k}, \ldots xn_{t-k}\right)$$

Where x_1 x_2 ,,, x_i are the input variables at time t, x_1(t-1) x_2 (t-1),,, x_i(t-1) are the independent variables at time t-1; x_1(t-k) x_2 (t-k),,, x_i(t-k) are the input variable values tracing back k time steps. The lag k can be obtained from the autocorrelation analysis of model residuals. Usually, if the model residuals change regularly, it indicates that time lag effects exist. This can be established using the MATLAB Statistics toolbox or other Statistical Software (SAS). Through our study, we have found that the first two time steps (two seconds) may have a significant effect on the distribution of the residuals. Hence, up to two time lags may be added to the PWR SG GMDH transient models.

However, as we can see in the results, sometimes time lag terms have little effect in reducing the magnitude of model prediction error, especially when the transient dynamics process is slow. And usually PWR operation procedures require the power changing process as steady as possible due to safety considerations. Therefore, time-lag terms will not be included in transient models unless a fast transient is present, such as shown in Figure 3.14. In the transient modeling process of this research, the principle about time lag terms is: time lag terms are added if they can improve model prediction significantly, otherwise, no such terms will be included in the model.

Figure 3.15 shows the training results for narrow range level transient model. More transient models and residual patterns are presented in Section 4.

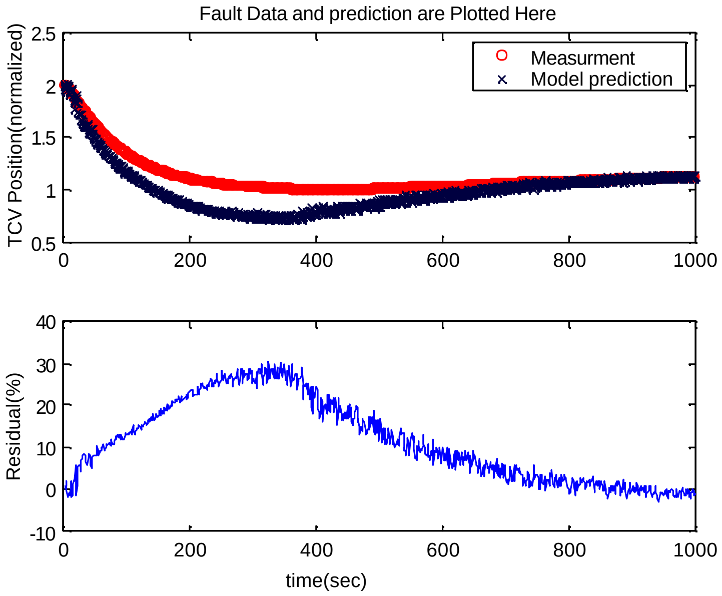


Figure 3.14. Fast transient process modeling of a TCV and residual generation.

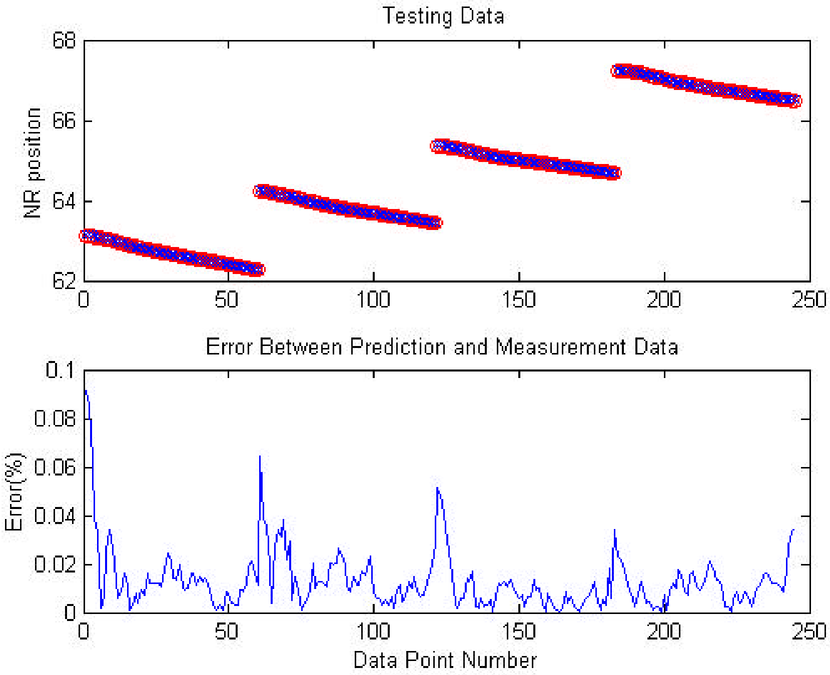


Figure 3.15. NR level model under transient dynamics.

3.3. PCA Algorithm for Modeling the PWR Steam Generator

Another important data-driven method, the PCA algorithm, is tested in this study. PCA modeling performs very well in extracting useful information in the full scope of variables. All input variables have contributions to the new constructed vectors, which free us from processing the redundant information in order to select useful input data. PCA assumes that a set of orthogonal vectors from the original data construct a new hyper- plane, where normal operation is located. Any significant deviation from this plane indicates device or system faults. Table 3.2 lists 38 variables included in the PCA model in this thesis.

Table 3.2. Process variables in the database

Variable #	1	2	3	4	5	6	7	8
Variable name:	Qthnew	Ref. Tave	Tinlet	T CL 1	T CL 2	T HL	Tsat	Qhtr
Variable #	9	10	11	12	13	14	15	16
Variable name	Psg 1	Psg 2	SG 1 Tsat	SG 2 Tsat	Feed Temp.	SG 1 Flow	Feed Flow 1	FCV 1 Pos.
Variable #	17	18	19	20	21	22	23	24
Variable name	SG 1 WR	SG 1 NR	SG 1 WR Ind	SG 1 NR Ind	SG 1 Mass	SG Ref WR	SG Ref NR	SG 2 Flow
Variable #	25	26	27	28	29	30	31	32
Variable name	Feed Flow 2	FCV 2 Pos.	SG 2 WR	SG 2 NR	SG 2 WR Ind	SG 2 NR Ind	TCV 1 Pos.	TCV 1 Flow
Variable #	33	34	35	36	37	38		
Variable name	TCV 2 Pos.	TCV 2 Flow	TCV 3 Pos.	TCV 3 Flow	TCV 4 Pos.	TCV 4 Flow		

Qthnew: Heat generated in Core.

Ref. Tave: Reference Average Temperature.

Tinlet: Temperature of inlet Plentum.

T CL 1 and T CL2: Cold leg Temperature:

T HL: Hot leg temperature

Tsat: Saturate Temperature

Qhtr: Heat transferred in SG.

WR: Wide Range of SG water level

NR: Narrow Range of SG water level

FCV: Feed Water Valve

TCV: Turbine Control Valve

Ind : Indicated

The Principal Component Analysis (PCA) model for the variables shows that the first 12 significant PCs, where the 12th PC is a breakpoint, should be retained in the model. The other 26 PCs will be discarded because of their insignificant contribution in state prediction. Figure 3.16 shows the percentage of information explained by each principal component. T-square and Q -statistics with 95% confidence level show the significance of the PCA model (Figure 3.17 and 3.18), though several outliers exist in Figure 3.17.

However, we should still keep in mind that a complicated system such as a PWR always has nonlinear relationship among its process variables. This part of information is discarded by PCA model as noise. This may cause problems in fault diagnosis and will be discussed in Section 4..

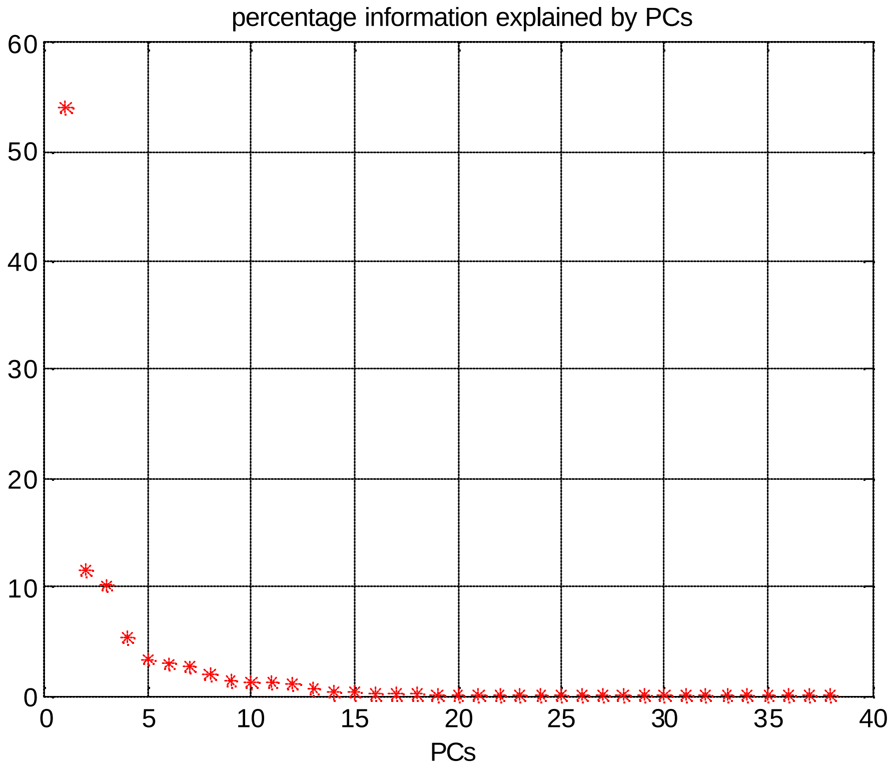


Figure 3.16. Percentage of variance explained by principal components.

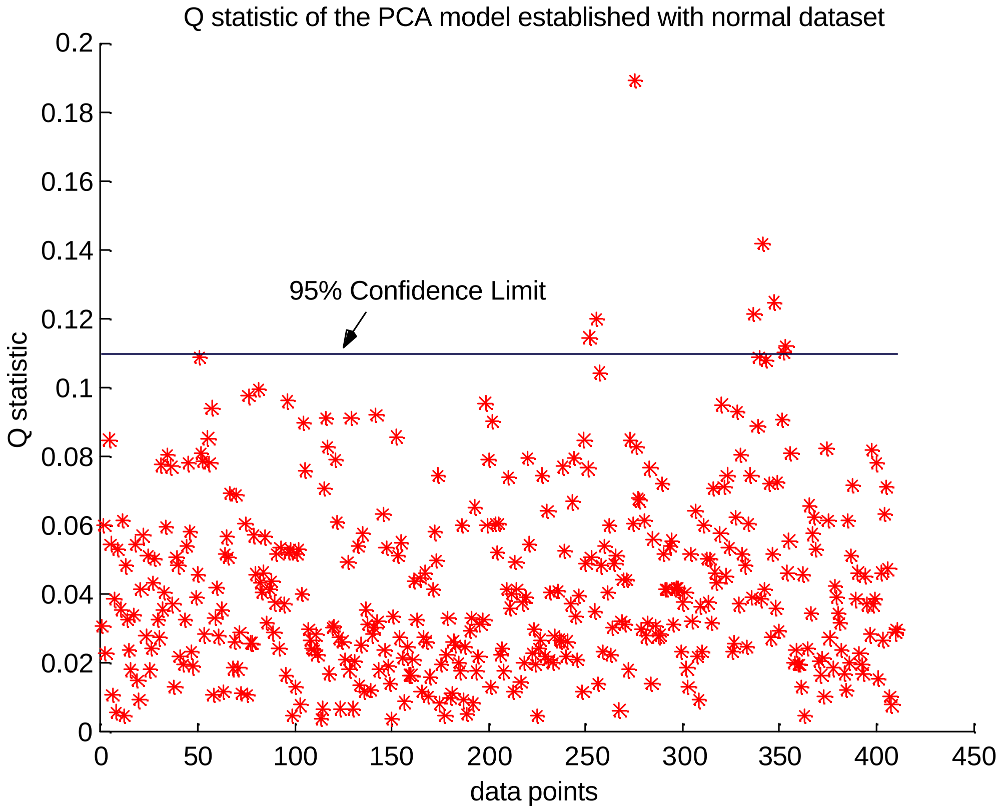


Figure 3.17. Q -statistic test for PWR SG PCA model.

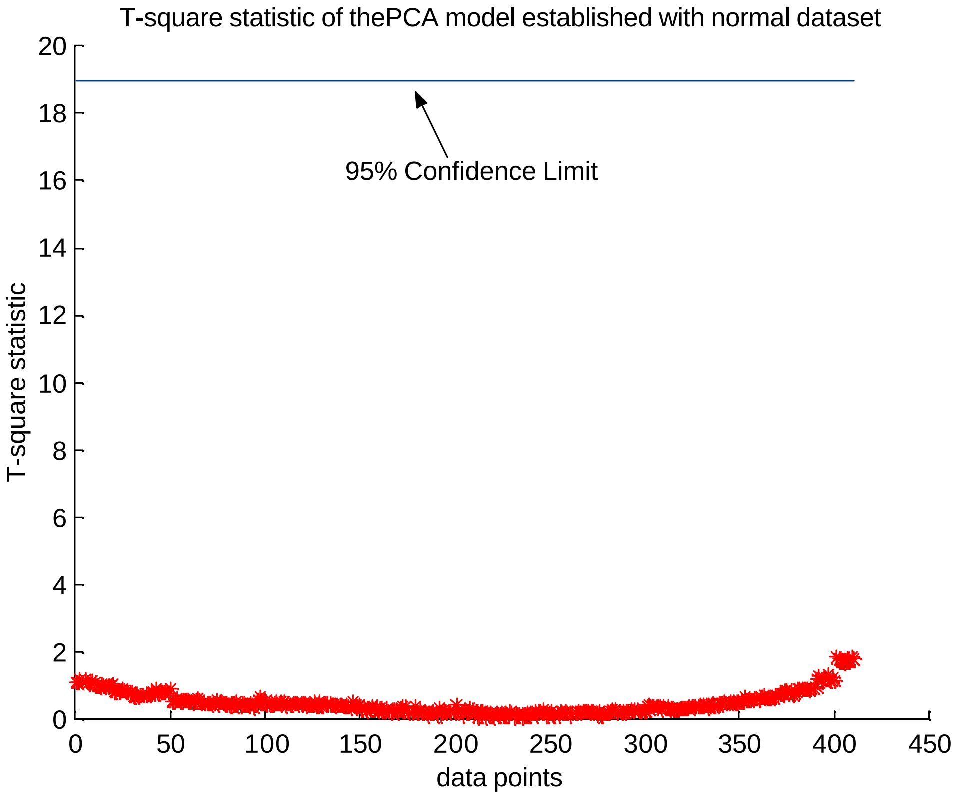


Figure 3.18. T^2 statistic test for PWR SG PCA model.

3.4. Evaluation of Data-Driven Models

The modeling results show the good performance of GMDH in capturing fixed input and output relationships in polynomial forms, especially in mapping the nonlinear causal behavior. The great benefit of this character will be found in directional residual analysis described in Chapter 6. A careful selection of input parameters for GMDH models will benefit us in saving computer memory, calculation time, and avoiding collinearity problems. The selection of input regressors for each model is based on the physical relationships between the output (predicted) variables and the input variables. In addition, correlation coefficients, as shown in appendix C, are also used as measures of selection.

Transient GMDH models have performed excellently in simulating SG transient dynamics. Although time delay terms work very well in considering the effect of time series, they are not always necessary for transient analysis. Actually when a PWR system is under slow dynamics, time lags are not included in the model inputs in order to save memory and time. However, they cannot be neglected during fast transient operation.

The PCA model successfully reduces the data dimension from 38 to 10 in this study. The Q-statistic and the T^2 -statistic indicate the quality of this model. However, a potential problem exists due to the loss of nonlinear information. This may create problems when the PCA model residuals are used for fault monitoring. In spite of this limitation of PCA, another exciting application is found when it is combined with the GMDH method for fault device isolation. This is described in detail in Section 4.

4. RESULTS OF APPLICATIONS TO A PWR U-TUBE STEAM GENERATOR

(neriFinalReport02-05)

4.1. Introduction

For a complicated system such as a PWR, the on line monitoring of components is very important in improving plant performance through the reduction of unscheduled shutdowns and maintenance costs. Hardware redundancy is one of the traditional methods usually adopted to ensure the availability of sensor data. The technique of analytical redundancy performs a similar task and is cost-effective if implemented properly. Many analytical techniques, which predict parameter values by using the information in other correlated sensors, have been developed in recent years. However, the application of these algorithms is still a challenging task, especially when the nonlinear measurements propagate through feedback control loops and disseminates the effect of a faulty component onto other variables.

In order to solve the problem of system level fault detection and isolation, data-driven models are established for generating residual signatures and to simulate the signature patterns under faulty device operation conditions. Two approaches for building data-driven models, GMDH and PCA, are used in this study to capture any abnormal system changes. However, the residuals from GMDH models have been found more useful in due to the capability of GMDH in mapping nonlinear relationships among process variables. By comparison, PCA model residuals are not able to distinguish between the faulty variables closely related through controller such as SG water level and SG feed water flow. In this research, the PCA algorithm is successfully applied in classifying the residual directions derived from nonlinear models.

Nearest neighbor classification and rule-based expert system are also implemented and compared to find a robust algorithm suitable for fault monitoring in the UTSG system. Different pattern recognition techniques are evaluated based on their ability to identify the faulty devices.

4.2. Fault Detection Using System Models

As described in Chapter 5, three different types of data driven models, static GMDH, transient GMDH, and PCA models, which can simulate system process accurately, are developed for the PWR SG system. Section 3 results illustrate the successful performance of these models.

Under normal conditions, small residuals would be generated and limited to a certain range. In the test cases when one or several components in the system are under degradation, the so-called causal relations among these variables will be violated. As a result, the mapping of residuals from residual generators or system models will increase in a specific direction. As an example, Figure 4.1 shows the residual pattern from GMDH models when one of the TCVs is under degradation. The residuals are the difference between the measurement values and their GMDH predictions. Figure 4.2 shows a significant increase in the Q -statistic of the PCA model when the narrow range SG level sensor is drifting. These illustrate the capability of data-driven models in detecting system anomalies.

It is clear that the residuals reflect not only whether there is an abnormal component, but also the severity of the fault, which is very important in helping the operator or the automatic controller to select the correct strategy in order to avoid severe negative effect caused by faulty devices.

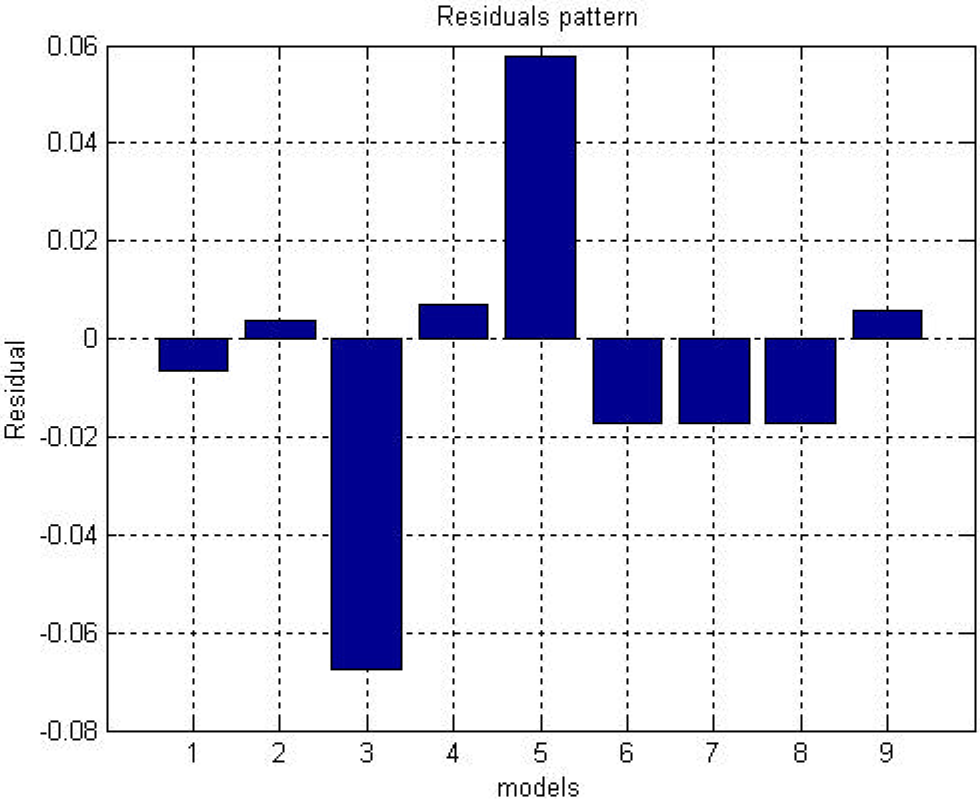


Figure 4.1. Residual pattern for the hysteresis change of one of the turbine control valves (TCVs).

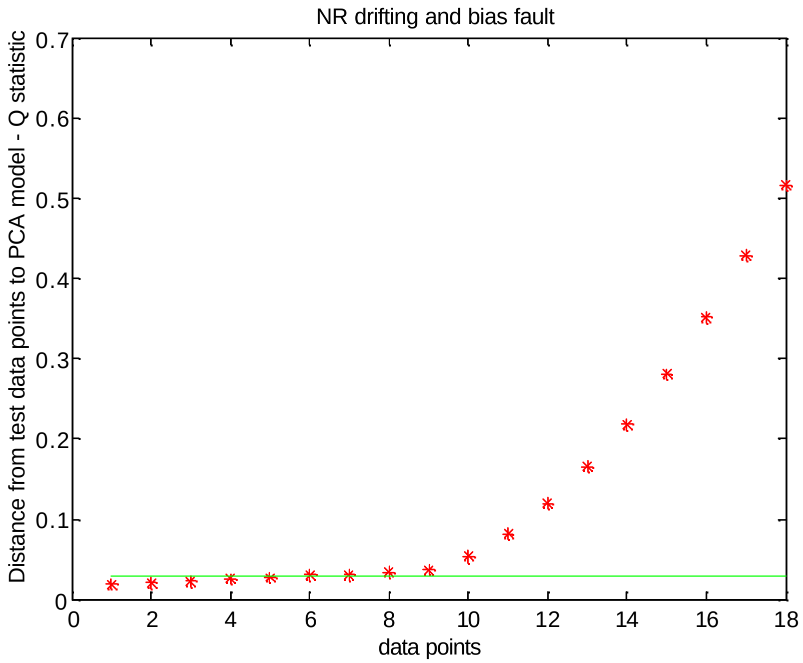


Figure 4.2. PCA model residuals indicating a drift in the SG narrow range level sensor.

4.3. Results of Isolating Typical Device Faults

The following is a summary of UTSG device faults simulated in this study. These include both sensor and actuator faults.

1. Narrow range (NR) SG level sensor drifting from 1% to 6% of full range. The offset change of sensor covers from 1.5% to 5.5% of the full range. In order to study the effect of the set point, the test data are collected under different power levels. The first part of data was collected when the nomi nal power decreases from 100% to 96%. The second part of data was generated during the nominal power increases from 80% up to 84%. The transient part of the data are processed and used in transient fault analysis, and the data during stationary phase are used in static fault study.

2. Feed water flow rate sensor drifting from 1% to 6% of full range. Offset change of sensor covers 1.5% to 5.5%. The first part of data was collected when nominal power decreases from 100% to 96%. The second part of the data was generated during nominal power increase from 80% up to 84%. The

transient part of the data is sampled one point per second and used in transient fault analysis, and the data during stationary phase are used in static fault study.

3. Steam flow rate sensor drifting from 1% to 6% of the full range. Sensor offset covers 1.5% to 5.5%. The first part of data was collected when the nominal power decreases from 100% to 96%. The second part of data was generated during the nominal power increases from 80% to 84%. The transient part of the data is sampled one point per second and used in transient fault diagnosis, and the data during stationary phase are used in static fault study.

4. Steam pressure sensor drifting from 1% to 6% of full range. Sensor offset covers 1.5% to 5.5% of the full range. The first part of the data is collected when the nominal power decreases from 100% to 96%. The second part of the data is generated during the nominal power increases from 80% up to 84%. The transient part of the data is sampled one point per second and used for transient fault analysis, and the data during stationary phase are used for static fault study.

Actuator faults related to the SG system include the following.

1. Feed water control valve degradation. The time constant changes from 2 seconds to 4 seconds, valve dead band increases up to 3% of its full position. The first part of data is collected when the nominal power decreases from 100% to 96%. The second part of data is generated during nominal power increases from 80% to 84%. Note that when the FCV dead band is over 3%, PWR safety protection system will be triggered and causes the plant to trip.

2. Turbine control valve (TCV) degradation. The time constant changes from 3 seconds to 6 seconds, valve dead band increases from 3% up to 6% of the full position. The first part of data is collected when the nominal power decreases from 100% to 96%. The second part of data is generated during nominal power increases from 80% to 84%. The transient part of data are processed and used in transient fault analysis, and the data during stationary phase are used in static fault study.

3. Feed water control valve stuck at a fixed position during its opening or closing process. In this study, the case of FCV stuck at 70% and 72% are observed and summarized.

Although controller faults are not included in the study of fault isolation, they can be detected and isolated easily with our FDI modules. The reason for not including these faults is that the NCSU code does not provide controller signal output as a parameter that may be modified. However, since controller signals are generally available in power and process plants, FDI algorithm would be able to diagnose controller error.

4.4. Isolation of Faulty Devices

Fault isolation is one of the most important steps in FDI as we move towards an advanced nuclear power plant automatic control algorithm. Several techniques are developed and compared in order to find the most suitable method(s) to solve the problems pertinent to the steam generator system. The performance of rule-based expert systems, PCA, and the nearest-neighbor pattern classification approach are compared and evaluated according to their performance under different plant conditions, for both steady state and transient processes.

4.3.1 Rule based expert system

As a commonly utilized diagnostic technique in nuclear power plants, rule based expert system uses knowledge and experience. The warning levels usually come from the observation of plant performance. The decision about a degraded device or process anomaly is made by comparing the patterns of residual signatures of the current test case with those of known fault cases. The latter is established either from actual process measurements or from a combination of simulation and plant data.

The residuals from all system modules, such as for example NR level GMDH module, are checked with corresponding threshold values. This value is set as 1% (percent prediction error) in this research, based on the noise level and modeling accuracy. When the residuals exceed this threshold value, it is declared that the parity relationship described by this module is violated, or the rule established in this module is broken. A group of binary code (0 for no violation, 1 for violation of a rule) can be established for one specific type of fault. If necessary, another number –1 may be included for description of residuals direction in order to increase the analytical capability. However, it is not used here because of the cost of losing robustness. For example, it would generate opposite residual patterns for valve opening and closing strokes if a TCV is under degradation. Thus, we need to consider them as two different faults if –1 is used in describing the change in residual directions.

Table 4.1 gives a summary of residual patterns for different types of faults under stationary conditions. Table 4.2 gives a summary of residual patterns for different types of faults under transient conditions. There are totally about 40 data points collected for each type of static component anomaly, and about 100 data points for each type of transient fault from the corresponding transient process.

Table 4.1. GMDH model residual patterns for PWR SG system under stationary conditions

Fault types	Residual patterns and base rules	Figures	Threshold (%)
NR sensor(drifting and bias)	Residual 1> threshold; Residual 5> threshold; Residual 6> threshold; Residual 7> threshold; Residual 8> threshold	6.3	1
Feed water flow rate sensor(drifting and offset)	Residual 1> threshold; Residual 2> threshold; Residual 3> threshold;	6.4	1
Steam flow rate sensor (bias and drifting)	Residual 1> threshold; Residual 2> threshold; Residual 3> threshold; Residual 4> threshold; Residual 5> threshold; Residual 6> threshold; Residual 7> threshold; Residual 8> threshold; Residual 9> threshold;	6.5	1
Steam Pressure sensor (bias and drifting)	Residual 9> threshold;	6.6	1
TCV degradation (hysteresis change)	Residual 3> threshold; Residual 5> threshold; Residual 6> threshold; Residual 7> threshold; Residual 8> threshold; Residual 5 > (Residual 6 = Residual 7 = Residual 8)	6.7	1
2 TCV degradation (hysteresis change)	Residual 3> threshold; Residual 5> threshold; Residual 6> threshold; Residual 7> threshold; Residual 8> threshold; Residual 5 = Residual 6 > 0; Residual 7 = Residual 8 < 0;	6.8	1
FCV degradation (hysteresis change)	Complicated, discussed specially.	6.9	1

Note about the residuals:

Residual 1: NR level static GMDH module residual
Residual 2: Feed water flow rate static GMDH module residual
Residual 3: FCV static GMDH position module residual
Residual 4: Steam flow rate static GMDH module residual
Residual 5: TCV1 static GMDH module residual
Residual 6: TCV2 static GMDH module residual
Residual 7: TCV3 static GMDH module residual
Residual 8: TCV4 static GMDH module residual
Residual 9: Steam pressure static GMDH module residual

Discussion: When the feed control valve (FCV) hysteresis changes, the response of the system is very complicated. If there is only a time constant change, the system will not produce significant responses; therefore, we are not able to detect FCV degradation under this case, because our FDI is based on the assumption that incipient faults will cause abnormal changes in system. If the dead band exceeds 3% of full range, the safety protection system would cause a plant trip. When the dead band is within 3% of its full range, the residuals from NR level, feed water flow, FCV position, and 4 TCV position modules would exceed the threshold.

Table 4.2. GMDH model residual patterns for PWR SG system under transient conditions

Fault types	Residual patterns and base rules	Figures	Threshold (%)
NR sensor(drifting and bias)	Residual 1> threshold; Residual 2> threshold; Residual 3> threshold;	6.10	1
Feed water flow rate sensor(drifting and offset)	Residual 2> threshold; Residual 3> threshold;	6.11	1
Steam flow rate sensor (bias and drifting)	Residual 2> threshold; Residual 3> threshold; Residual 4> threshold; Residual 5> threshold; Residual 6> threshold; Residual 7> threshold; Residual 8> threshold; Residual 9> threshold;	6.12	1
Steam Pressure sensor (bias and drifting)	Residual 9> threshold;	6.13	1
TCV degradation (hysteresis change)	Residual 1> threshold; Residual 3> threshold; Residual 5> threshold; Residual 6> threshold; Residual 7> threshold; Residual 8> threshold; Residual 5 > (Residual 6 = Residual 7 = Residual 8)	6.14	1

Note about the residuals:

Residual 1: NR level transient GMDH module residual
Residual 2: Feed water flow rate transient GMDH module residual
Residual 3: FCV position transient GMDH module residual
Residual 4: Steam flow rate transient GMDH module residual
Residual 5: TCV1 transient GMDH module residual
Residual 6: TCV2 transient GMDH module residual
Residual 7: TCV3 transient GMDH module residual
Residual 8: TCV4 transient GMDH module residual
Residual 9: Steam pressure transient GMDH module residual.

During a plant transient with a device fault, the residuals from the GMDH modules increase with time. The residuals may not exceed the threshold at the beginning of the transient process. From the residual patterns for stationary and transient conditions, we observe that the rule-based expert system is suitable for the fault component isolation under both environments. A set of *If_Then* rules may be easily established.

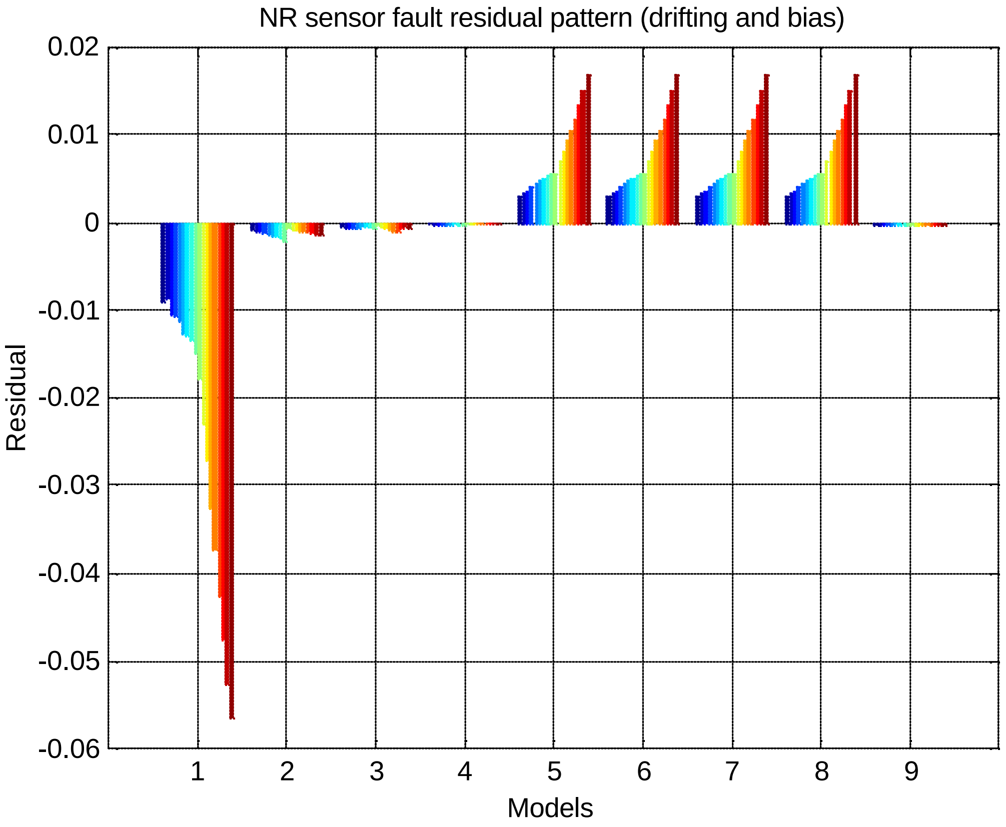


Figure 4.3. GMDH model residuals when NR level transmitter is under degradation.

Figure 4.3 shows the residual pattern for the case when there is a drift or bias fault in the narrow range SG level sensor. About 40 different fault operating conditions under different severity of NR fault are illustrated. It is noticed that when the sensor fault is not severe, the magnitude of residuals is small (blue part in the plot). When the sensor fault is severe, the magnitude of residuals becomes large (red color in the plot). Similar conditions exist for other residual pattern plots shown in Figures 4.4 - 4.14. These indicate that the residual plots also reflect the severity of faulty devices.

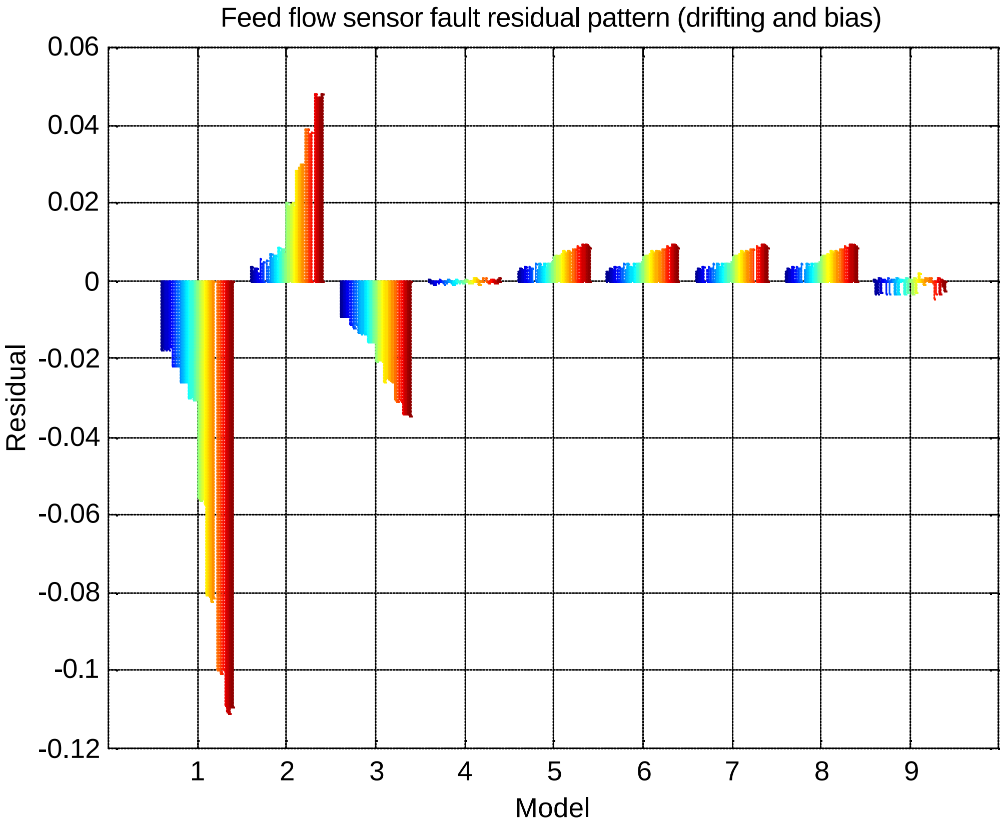


Figure 4.4. GMDH model residuals when feed water flow sensor is under degradation.

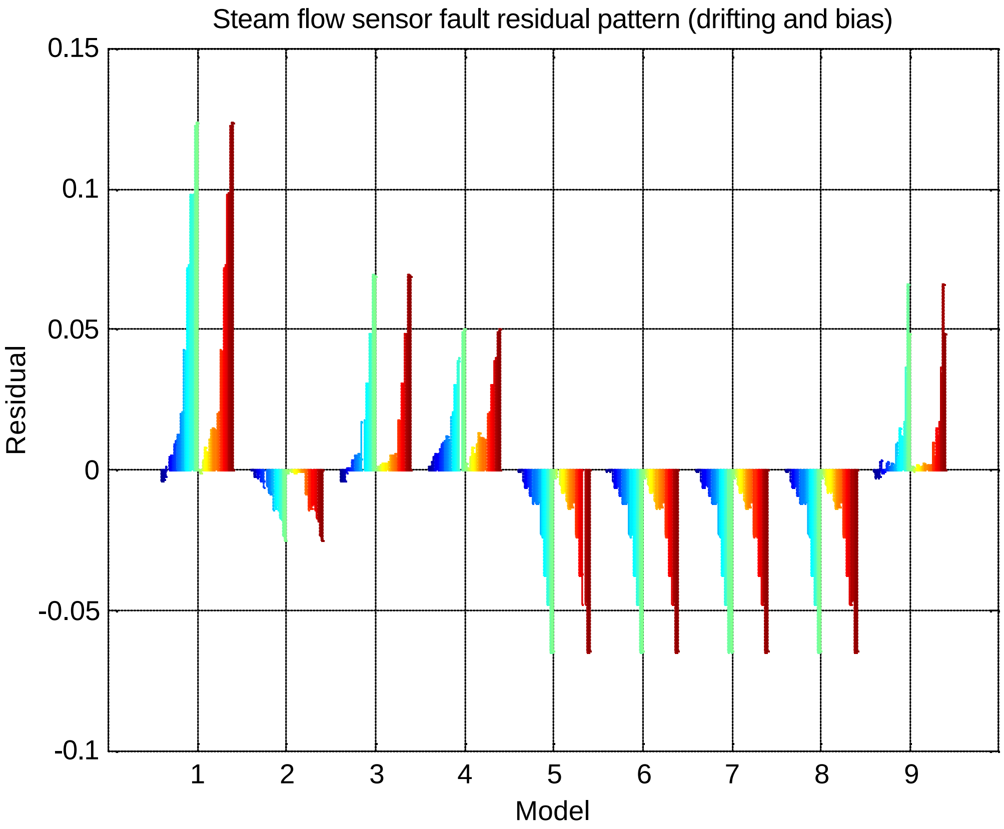


Figure 4.5. GMDH model residuals when steam flow sensor is under degradation.

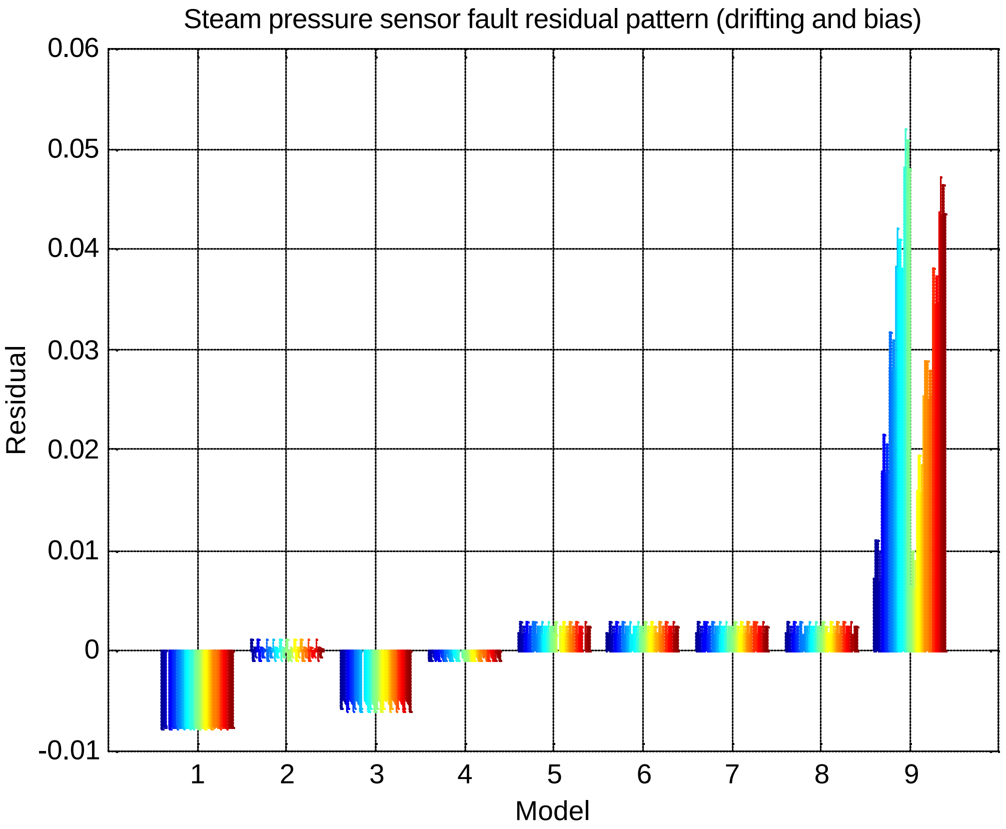


Figure 4.6. GMDH model residuals when steam pressure sensor is under degradation.

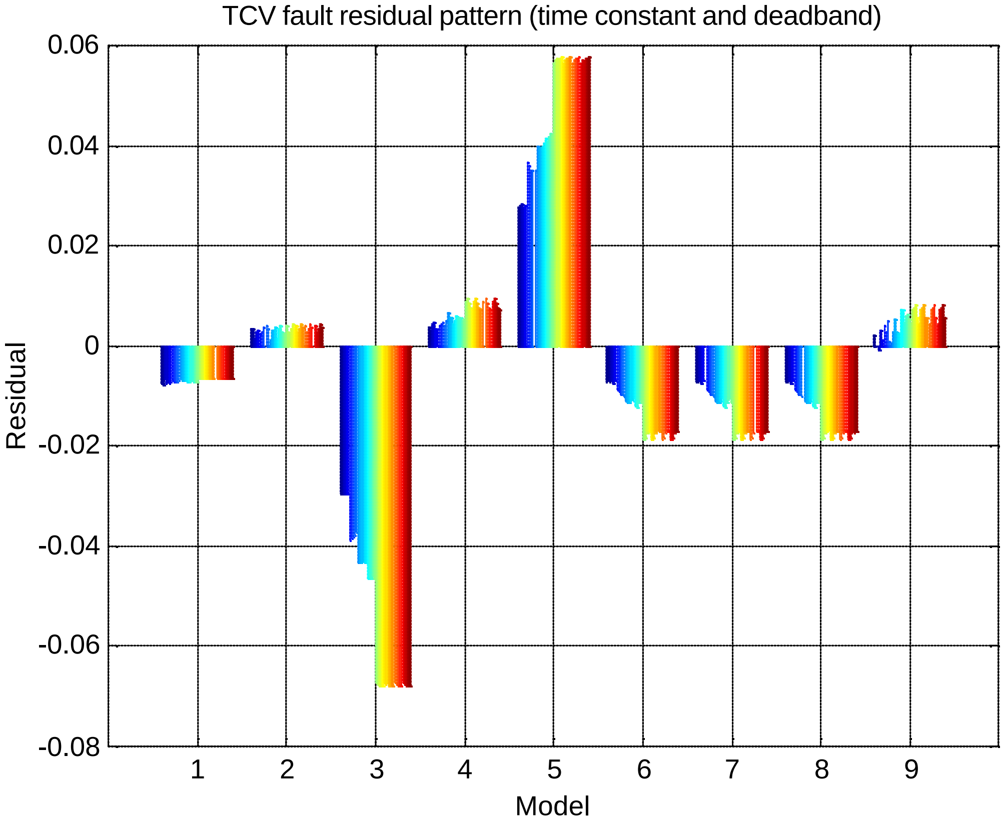


Figure 4.7. GMDH model residuals when turbine control valve (TCV1) is under degradation

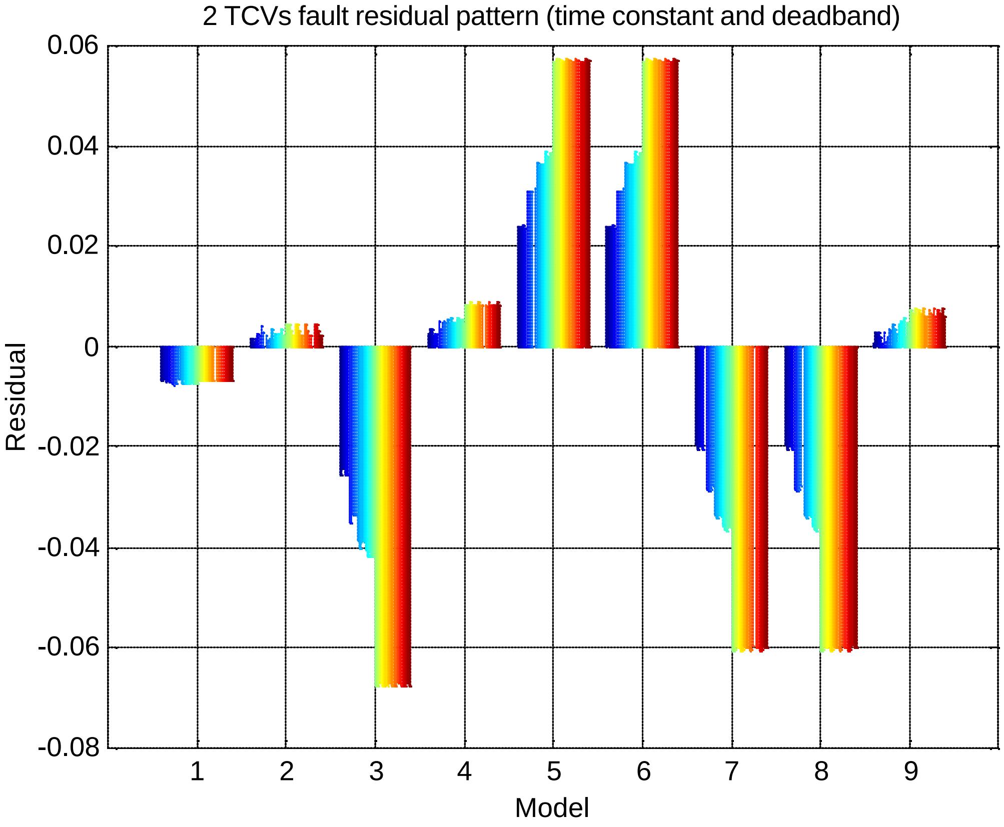


Figure 4.8. GMDH model residuals when TCV1 and TCV2 are under degradation

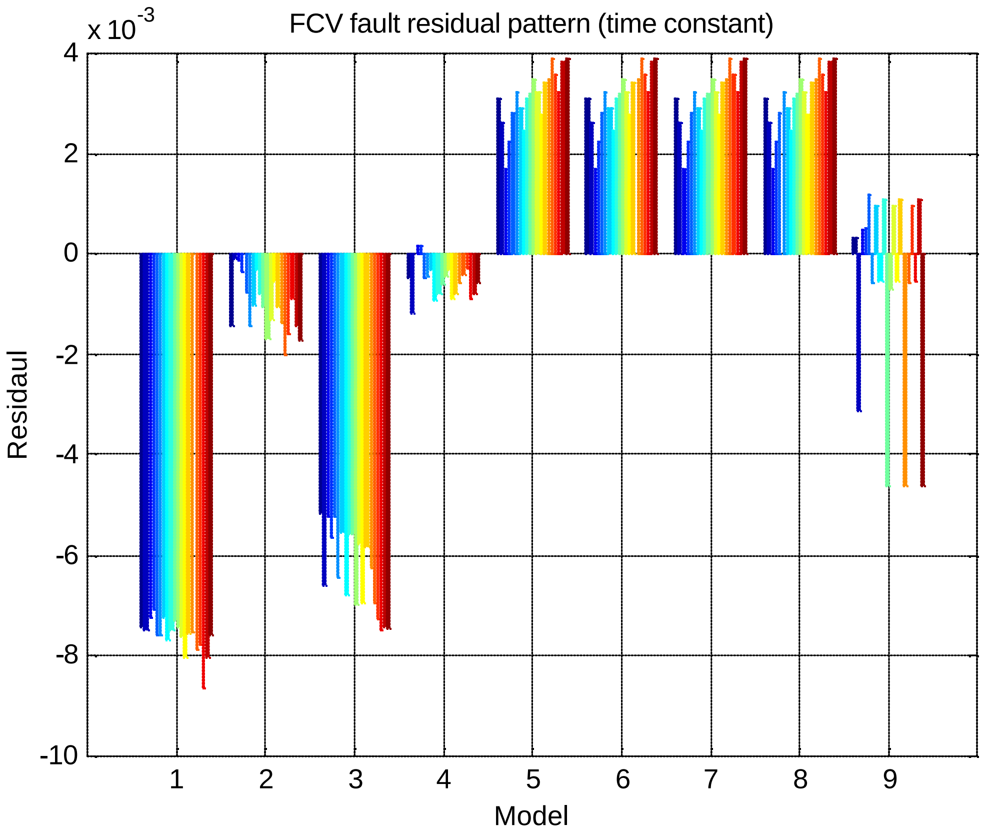


Figure 4.9. GMDH model residuals when feed control valve (FCV1) is under degradation.

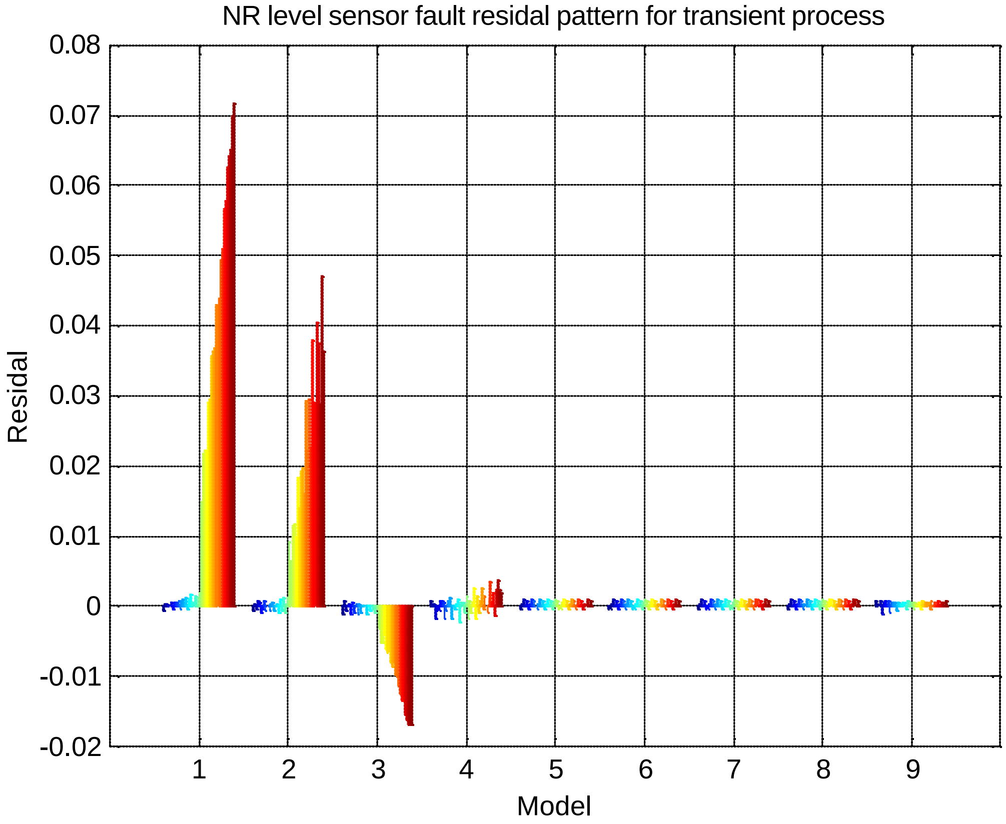


Figure 4.10. GMDH transient model residuals when NR level sensor is under degradation

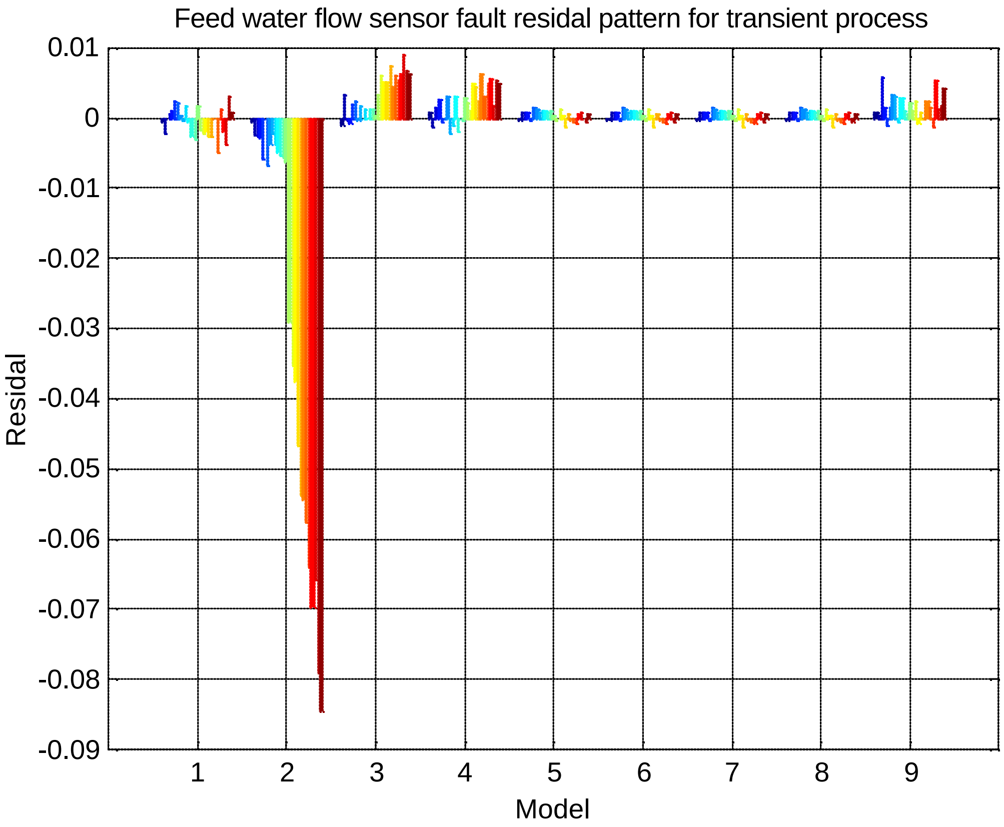


Figure 4.11. GMDH transient model residuals when Feed flow rate sensor is under degradation.

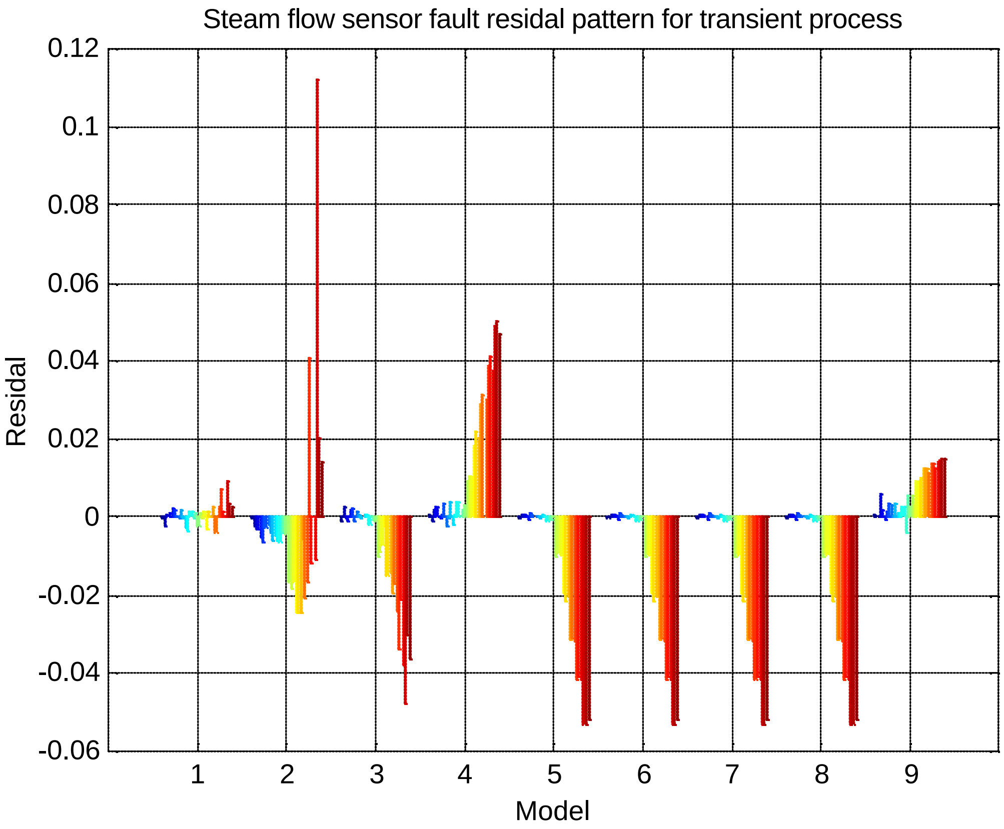


Figure 4.12. GMDH transient model residuals when steam flow rate sensor is under degradation.

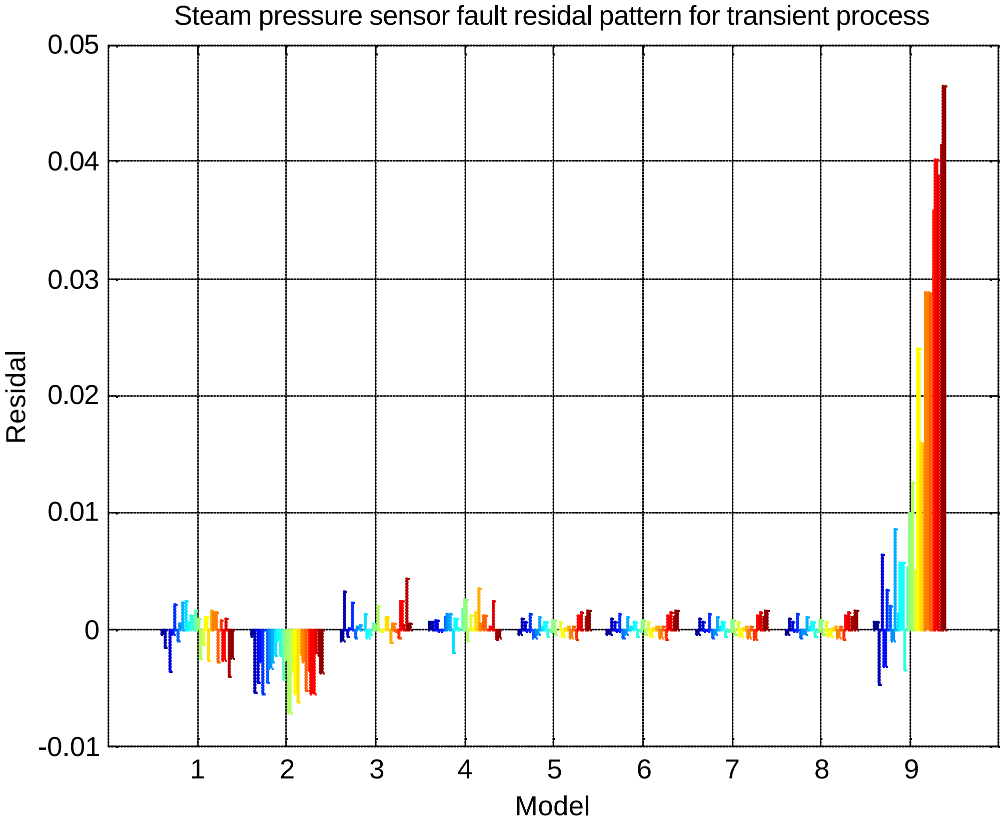


Figure 4.13. GMDH transient model residuals when steam pressure sensor is under degradation.

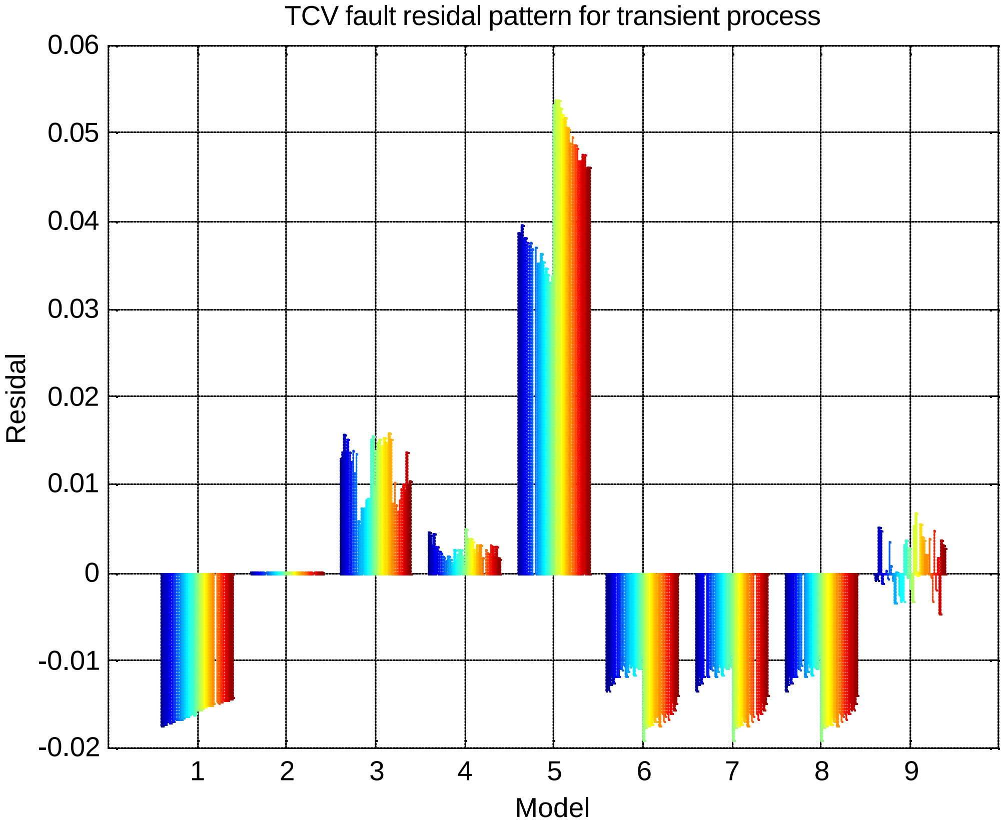


Figure 4.14. GMDH transient model residuals when TCV1 is under degradation.

4.3.2 PCA algorithm for fault isolation

PCA is powerful in reducing the dimensionality of redundant data. This may become very useful if the fixed directions exist for different types of system fault conditions. As mentioned in Section 3, PCA works well in modeling PWR SG system and detecting abnormal changes in the system. However, its performance is not highly satisfactory when using the residuals from the PCA models to isolate faulty devices. The residual patterns have similar behavior for the cases of abnormal NR level and feed water flow transmitters. This is due to 1) PCA modeling considers only the linear relationship among the variables; 2) PCA includes all the system variables that are listed in table 3.1 as inputs. Therefore NR level and feed flow transmitter faults may generate the same residual patterns due to the interaction between the level controller and the feed control valve actuation. Furthermore, the level set point information is not explicitly included in decision making.

Therefore the residuals from the PCA model are not used directly for fault isolation. The PCA tool is used for processing the residuals generated by the GMDH modules. The PCA analysis is especially useful for transient case, because the residuals from GMDH transient models change with time, but the direction in which they propagate remains unchanged. The schematic of combining the GMDH module with the PCA is shown in Figure 4.15.

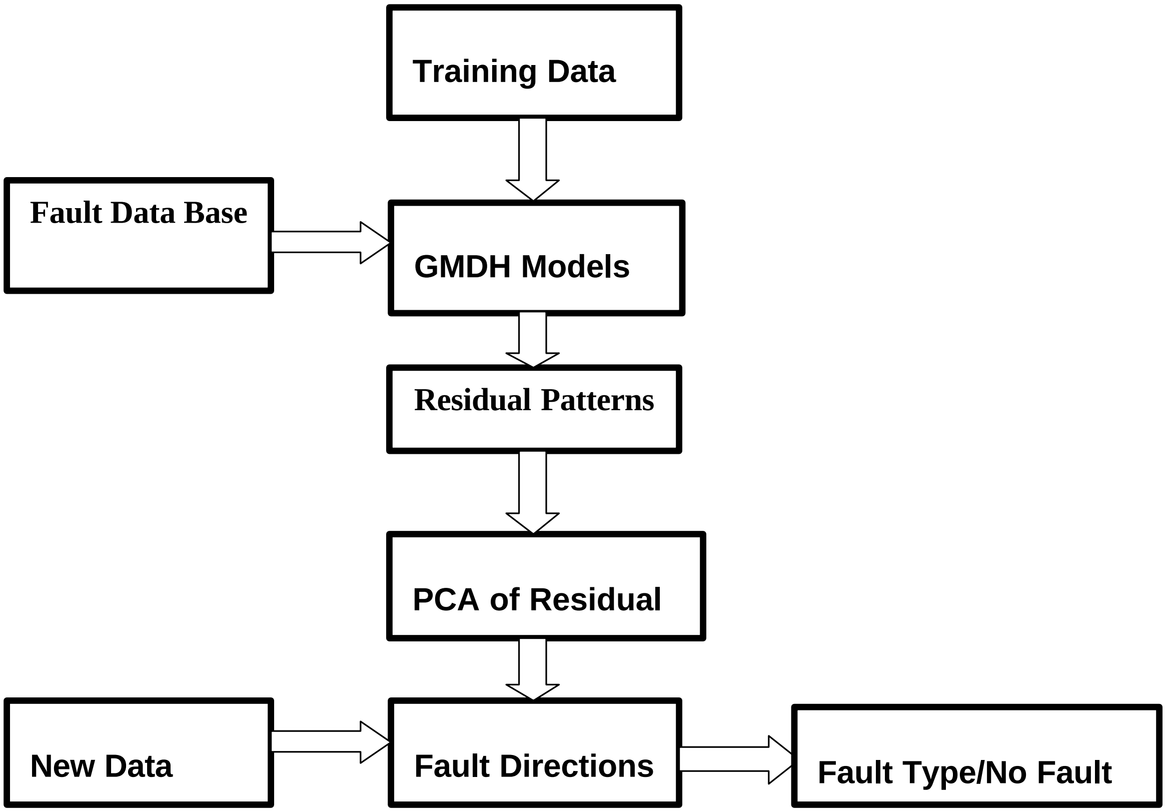


Figure 4.15. Schematic representation for combining GMDH and PCA methods.

Tables 4.3 and 6.4 list the results of using PCA to perform directional analysis on GMDH residuals under stationary and transient dynamics.

Table 4.3 PCA residual analysis for PWR SG system under steady state conditions

Fault type	Detected fault direction	Figures	Percentage of residual variance expressed by the first PC
NR level sensor(drifting and bias)	Fault direction 1	6.16	93.98%
Feed water flow rate sensor(bias and drifting)	Fault direction 2	6.17	88.63%
Steam flow rate sensor (bias and drifting)	Fault direction 3	6.18	82.91%
Steam Pressure sensor (bias and drifting)	Fault direction 4	6.19	83.68%
TCV degradation (hysteresis change)	Fault direction 5	6.20	92.16%
2 TCV degradation (hysteresis change)	Fault direction 6	6.21	64.18%

Fault direction 1: NR level sensor degradation
Fault direction 2: Feed flow rate sensor degradation
Fault direction 3: Steam flow rate sensor degradation
Fault direction 4: Steam pressure sensor degradation
Fault direction 5: TCV1 degradation
Fault direction 6: TCV1 and TCV2 degradation.

Table 4.4. PCA residual analysis for PWR SG system under transient conditions

Fault type	Detected fault direction	Figures	Percentage of residual variance expressed by the first PC
NR sensor(drifting and bias)	Fault direction 1	6.22	85.96%
Feed water flow rate sensor(bias and drifting)	Fault direction 2	6.23	72.71%
Steam flow rate sensor (bias and drifting)	Fault direction 3	6.24	65.02%
Steam Pressure sensor (bias and drifting)	Fault direction 4	6.25	71.58%
TCV degradation (hysteresis change)	Fault direction 5	6.26	83.11%

Fault direction 1: NR level sensor degradation
Fault direction 2: Feed flow rate sensor degradation
Fault direction 3: Steam flow rate sensor degradation
Fault direction 4: Steam pressure sensor degradation
Fault direction 5: TCV1 degradation.

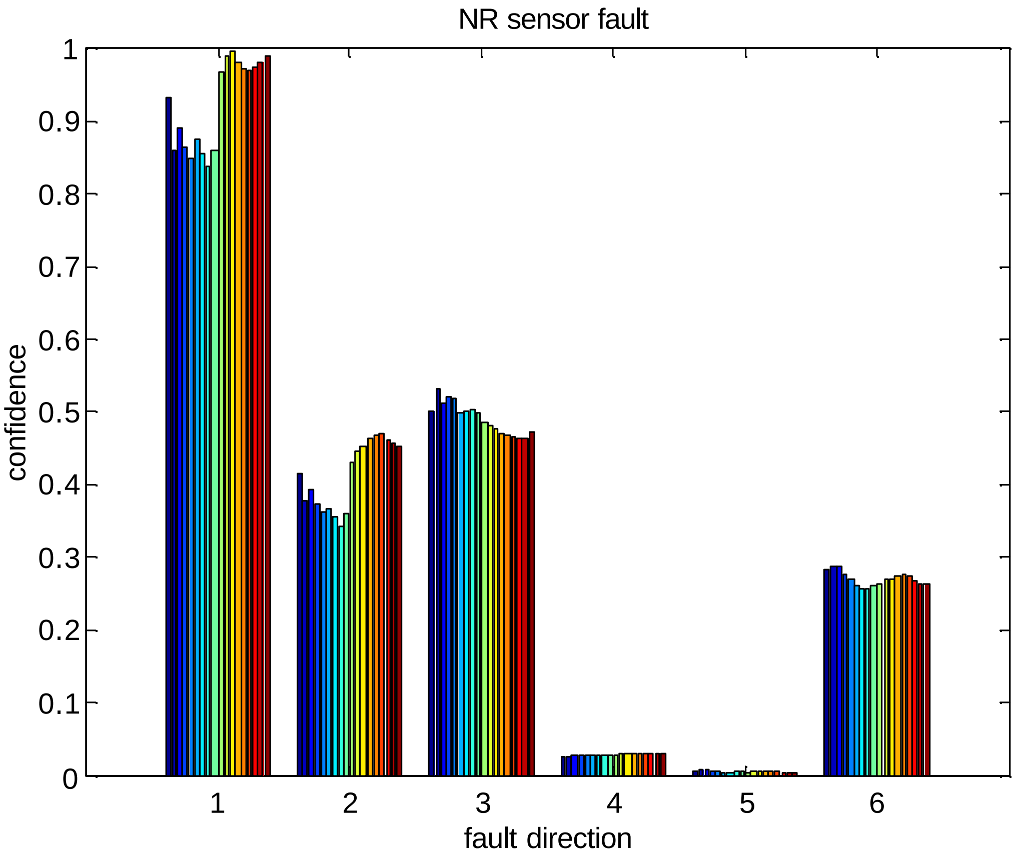


Figure 4.16. Static fault direction indicated by PCA residual model, when NR level sensor is under degradation.

Figure 4.16 shows the plots of the residual directions of the measurements for the case when there is a drift or bias fault in the narrow range SG level sensor. The NR directional signature has a maximum value (about 0.9). The directional signatures for the steam flow and feed flow are not insignificant, primarily because their settings change because of error in the NR sensor and the resulting feedback. Each of the fault direction plots illustrates about twenty steady-state operating conditions.

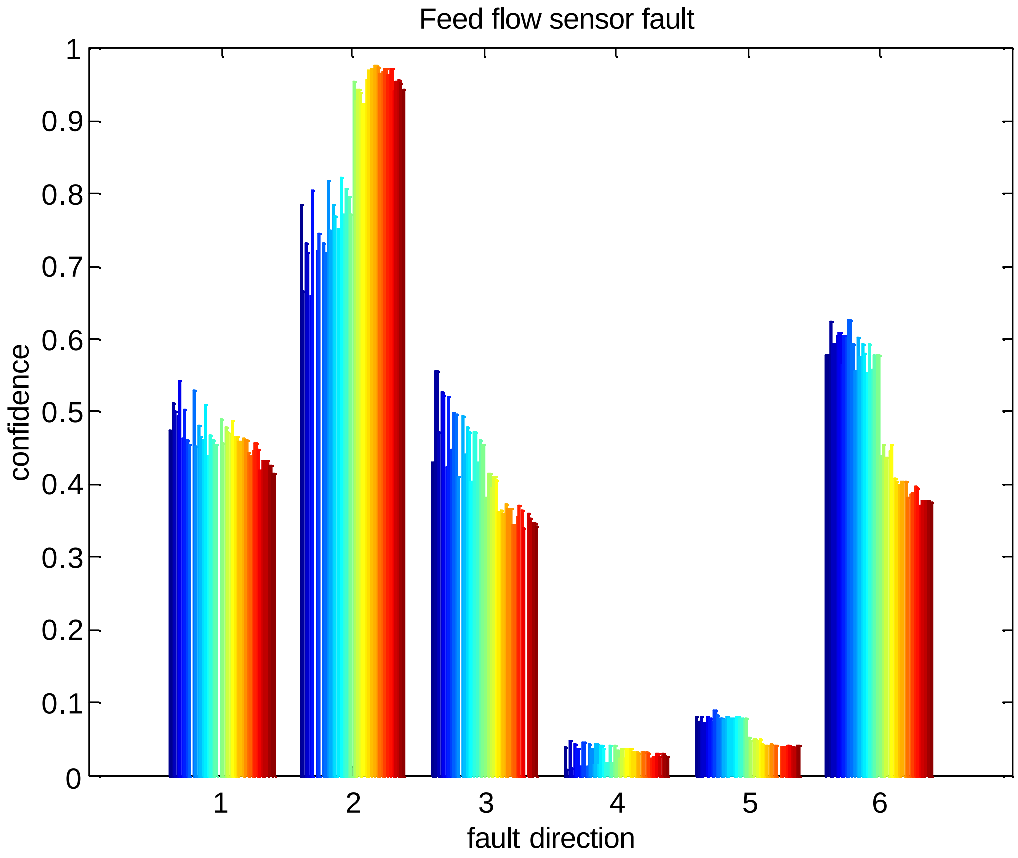


Figure 4.17. Static fault direction indicated by PCA residual model When the feed water flow sensor is under degradation.

Figure 4.17 shows the plots of the residual directions of the measurements for the case when there is a drift or bias fault in the feed flow sensor. The feed flow sensor directional signature has a maximum value compared to the others. The directional signatures for the NR and steam flow are not insignificant, primarily due to the feedback from feed flow controller. Each of the fault direction plots illustrates about forty steady-state operating conditions.

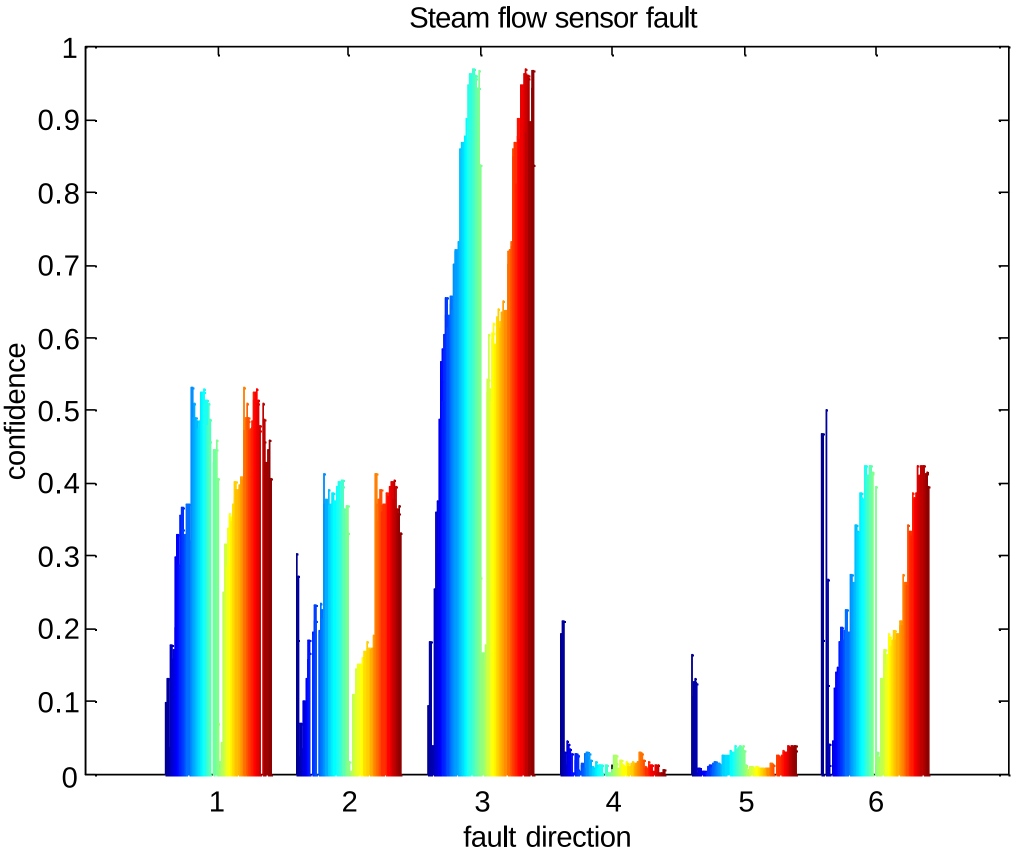


Figure 4.18. Static fault direction indicated by PCA residual model when the steam flow sensor is under degradation.

Figure 4.18 shows the plots of the residual directions of the measurements for the case when there is a drift or bias fault in the steam flow sensor. The steam flow sensor directional signature has a maximum value. The directional signatures for the NR and steam flow are not insignificant. It is very important to notice that when the fault is not severe (blue color in the plot) the confidence levels for all fault directions are less significant. Each of the fault directional plots illustrates about forty steady-state operating conditions.

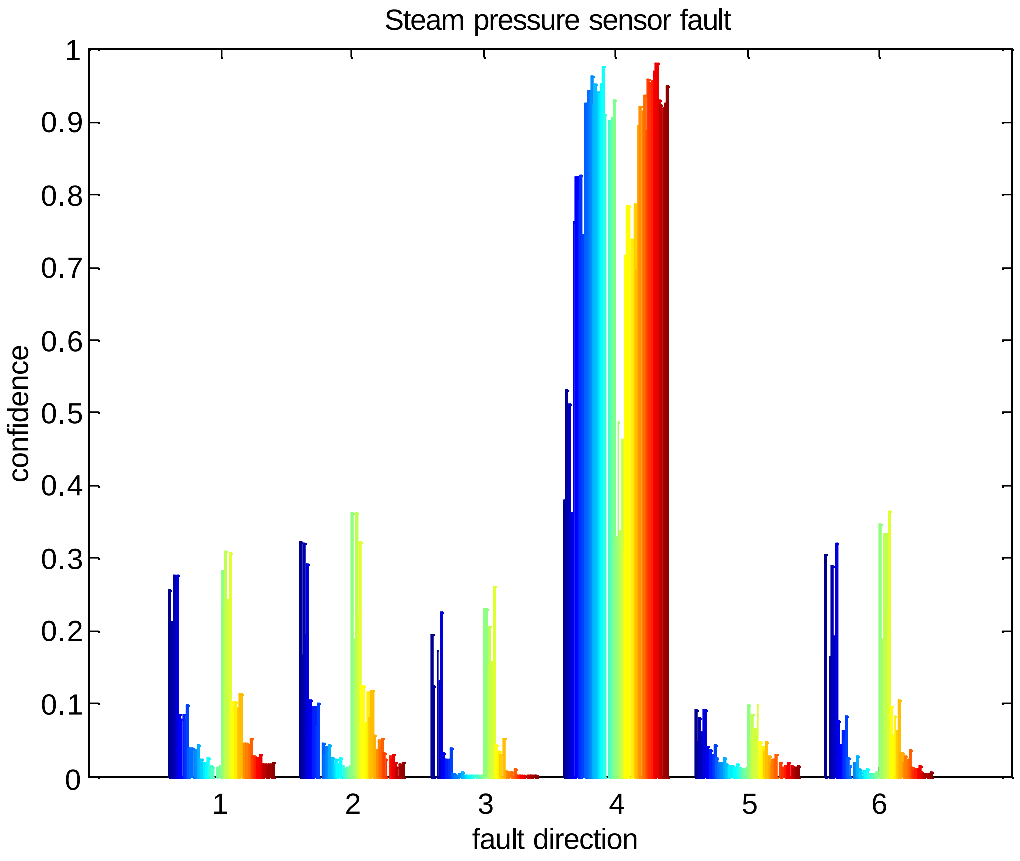


Figure 4.19. Static fault direction indicated by PCA residual analysis when the steam pressure sensor is under degradation.

Each of the fault direction plots illustrates about forty steady-state operating conditions in Figure 4.19. Similar condition hold true for confidence plots shown in Figures 4.20 and 4.21 for different type of fault cases.

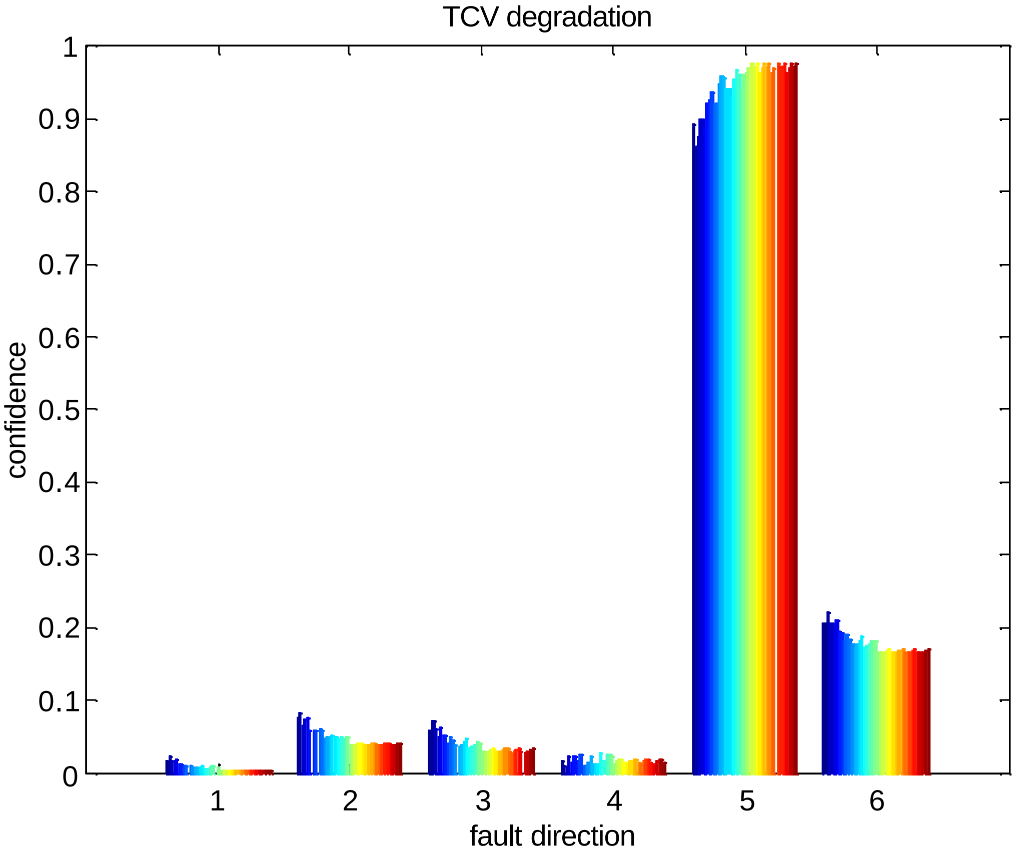


Figure 4.20. Static fault direction indicated by PCA residual analysis when one of TCVs is under degradation.

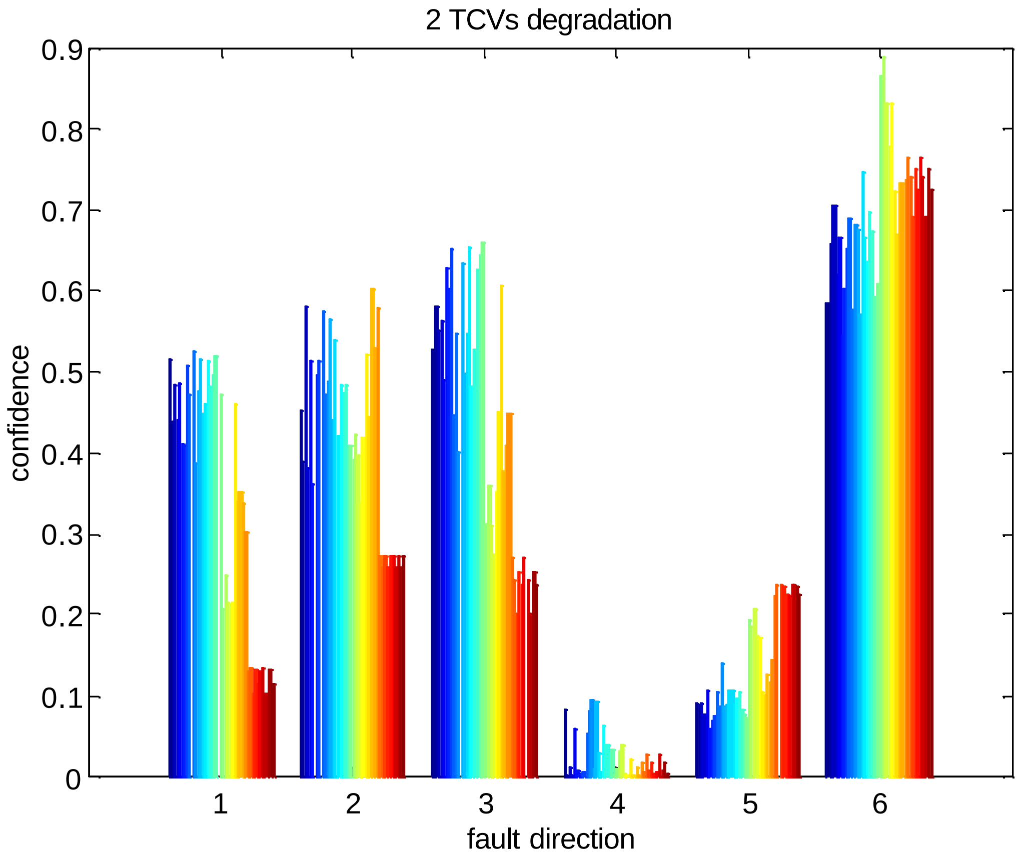


Figure 4.21. Static fault direction indicated by PCA residual analysis when two of the TCVs are under degradation.

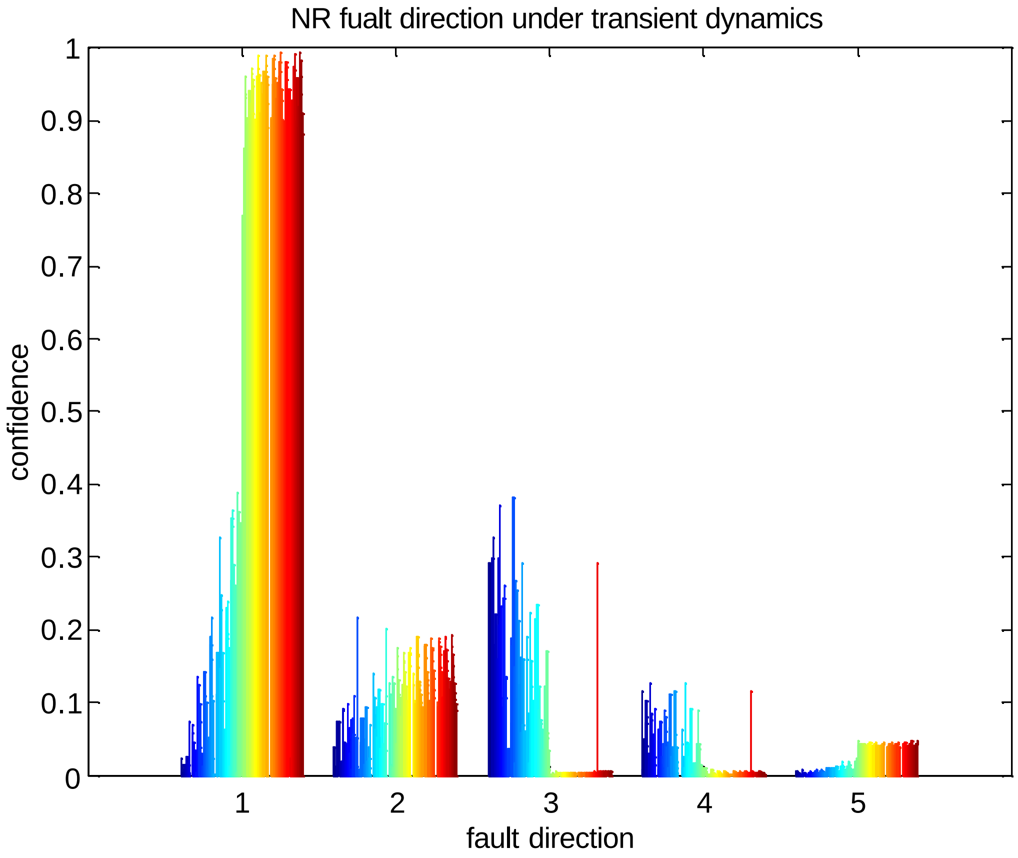


Figure 4.22. Fault direction indicated by PCA residual analysis when the NR level sensor is under degradation during transient process.

Each of the fault direction plots illustrates about fifty data points from the transient process shown in Figure 4.22 – 4.26 under different type of fault. At the beginning of fault transient process, the confidence levels for all the directions are low because the fault has not caused any significant changes in the system.

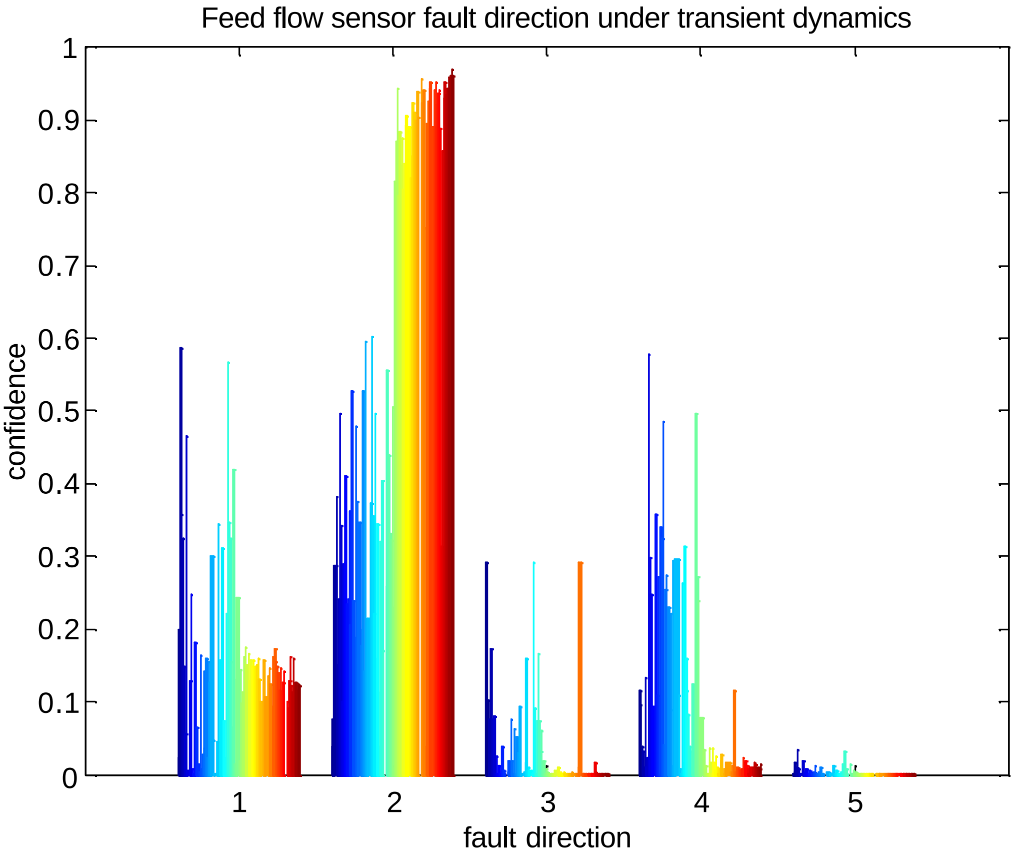


Figure 4.23. Fault direction indicated by PCA residual analysis when the feed water flow sensor is under degradation during transient process.

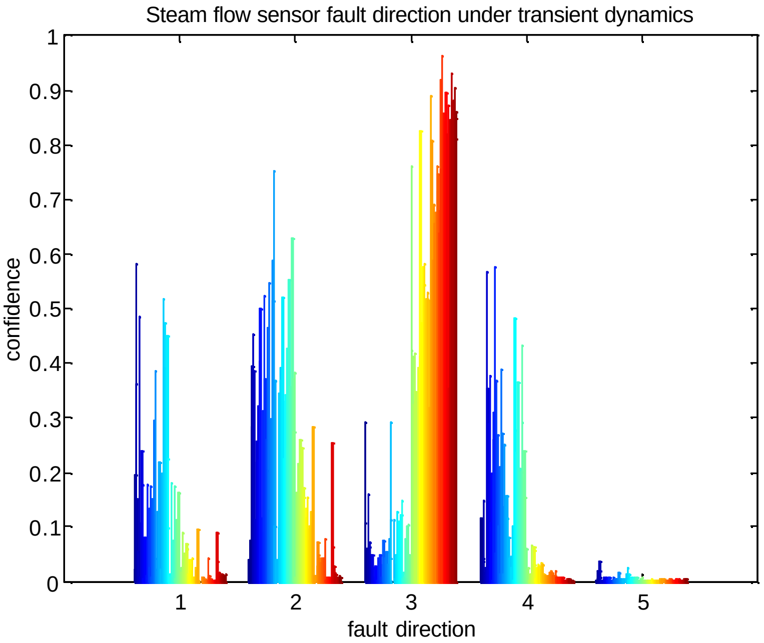


Figure 4.24. Fault direction indicated by PCA residual analysis when the steam flow sensor is under degradation during transient process.

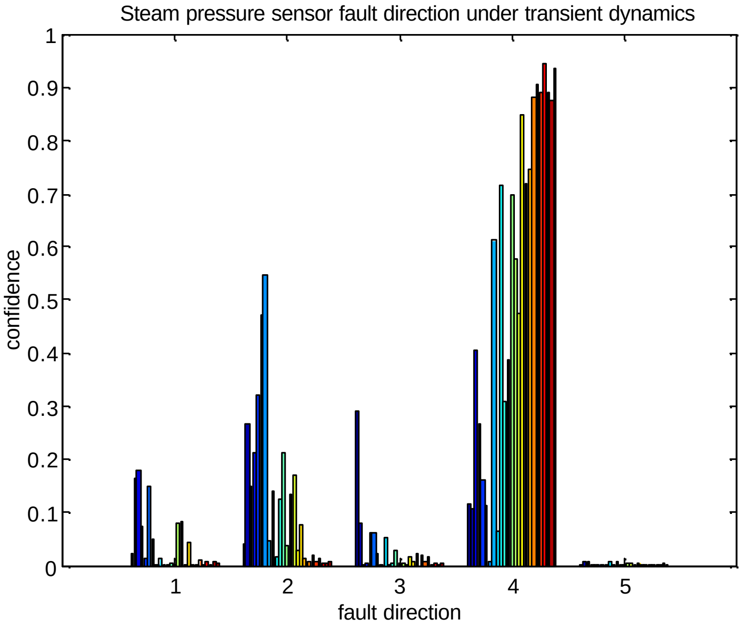


Figure 4.25. Fault direction indicated by PCA residual analysis when the steam pressure sensor is under degradation during transient process.

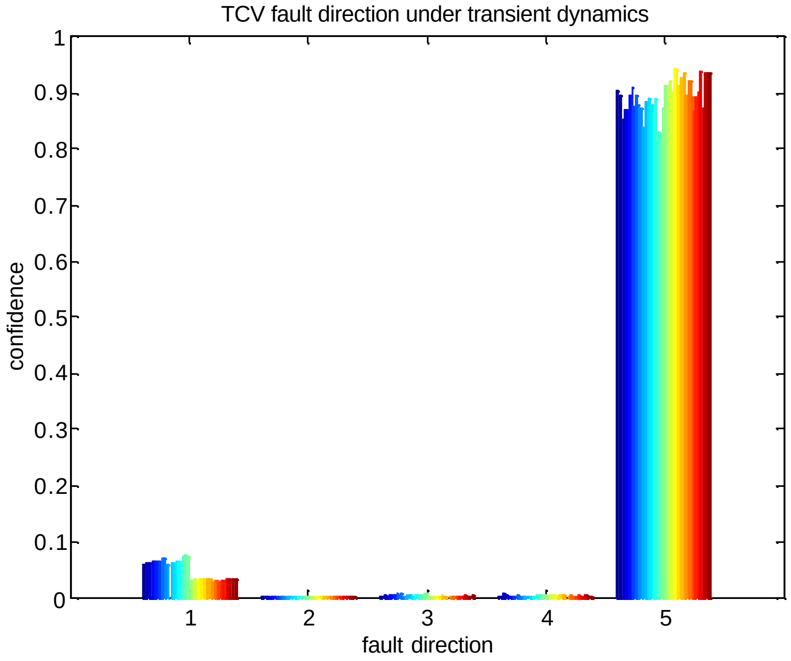


Figure 4.26. Fault direction indicated by PCA residual analysis when one of the TCVs is under degradation during transient process.

4.3.3. Nearest-neighbor pattern recognition for SG fault isolation

The nearest-neighbor classifier relies on the metric or distance between patterns. The test data are grouped into the class that is closest in terms of a distance measure. A mean distance between the test point and data group corresponding to a specific fault is used as a decision criterion. The mean distance is used in order to decrease the effect of outliers in the test data.

The following steps describe the procedure for implementing this method.

1. If there are n groups of fault data known as X1, X2, ... Xn, then calculate the mean distance among the data in the group Xi={x1,x2...xm}, 1<i<n. Totally m average distances are generated for group Xi. Every point in Xi will have m-1 Euclidean distances to other points, $d(x_j, x_k) = \left| x_j - x_k \right| = \sqrt{(x_j - x_k)(x_j - x_k)^T}$. Therefore the average distance between x1 and other data points is $dave1 = \dfrac{\sum_{k=2}^{m} d(x_1, x_k)}{m-1}$. Maximum and minimum value of data set $dave1, dave2, \cdots davem$ will be found and denoted as dmax$_i$ and dmin$_i$.

2. For a test data point, calculate the average distance between this point and the points in group Xi, denoted as di. There are n average distances between this point to the n groups, $d1, d2, \cdots dn$.

3. Scale the distance with $d\max, d\min$, using formula:
$$di\_ind = 1 - \left(\frac{di - d\min i}{d\max i - d\min i}\right)$$

4. Classify the test data to the group that has the smallest distance to this point.

Table 4.5 summarizes the result of applying the nearest-neighbor pattern classification technique to the SG data.

Figures 4.27 to 4.33 illustrate the results when the nearest neighbor is used in residual space classification. The test data are classified into the fault cluster whose average distance to test data is nearer zero than other fault clusters. For example, the fault is classified as TCV degradation in figure 6.28 because the test data have the smallest average distance to the TCV fault cluster.

Note that the distance to the steam flow sensor fault is always the cluster that is near to the true fault data cluster, this is because that all system models have included steam flow as input term. Thus, an improvement should be made in selecting system model input terms if the nearest neighbor is selected as part of FDI algorithm, that is, two models with different input terms should be established for a predicted variable for the purpose of cross checking.

Table 4.5. Nearest neighbor algorithm for PWR SG system under stationary conditions

Fault type	Training data	Figure
Feed water flow rate sensor(offset and drifting), 96% nominal power level	Data groups describing all types of faulty device under 96% power level	6.27
TCV degradation (hysteresis change), 96% nominal power level	Same as above	6.28
NR sensor (drifting and bias) , 96% nominal power level	Same as above	6.29
NR sensor (drifting and bias), 84% nominal power level	Same as above	6.30
Steam flow sensor (drifting and bias), 96% nominal power level	Same as above	6.31
Steam pressure sensor (drifting and bias), 96% nominal power level	Same as above	6.32
TCV1 and TCV2 degradation (hysteresis change), 96% nominal power level	Same as above	6.33

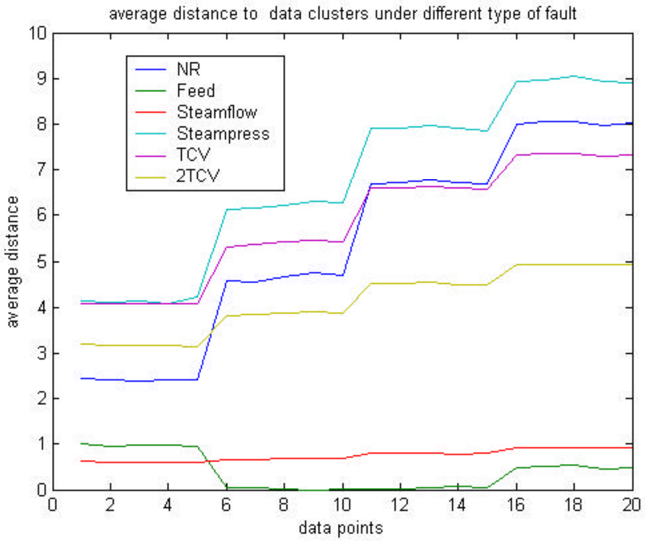


Figure 4.27. The fault direction indicated by the nearest neighbor method when the Feed flow sensor is abnormal.

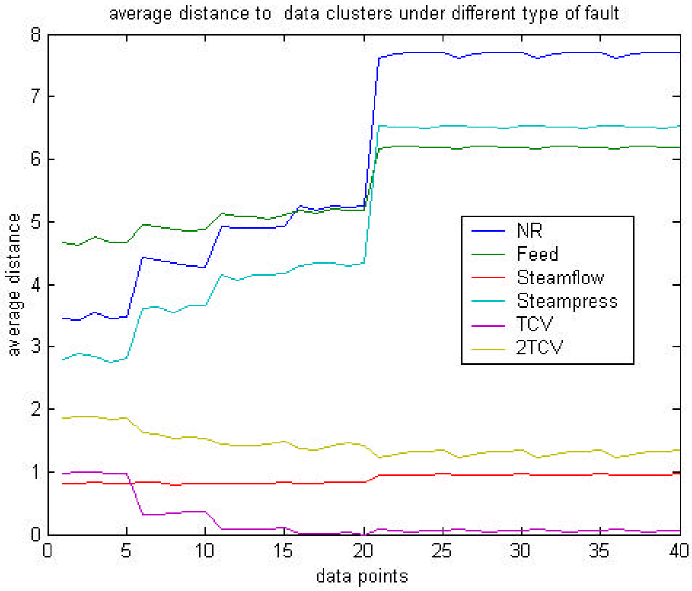


Figure 4.28. Fault direction indicated by the nearest neighbor method when one of TCVs is under degradation.

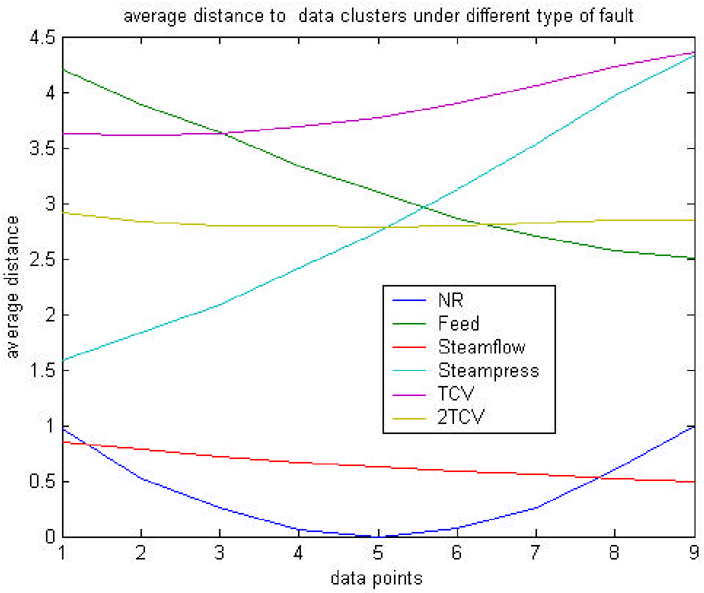


Figure 4.29. Fault direction indicated by the nearest neighbor method when the NR sensor is under degradation.

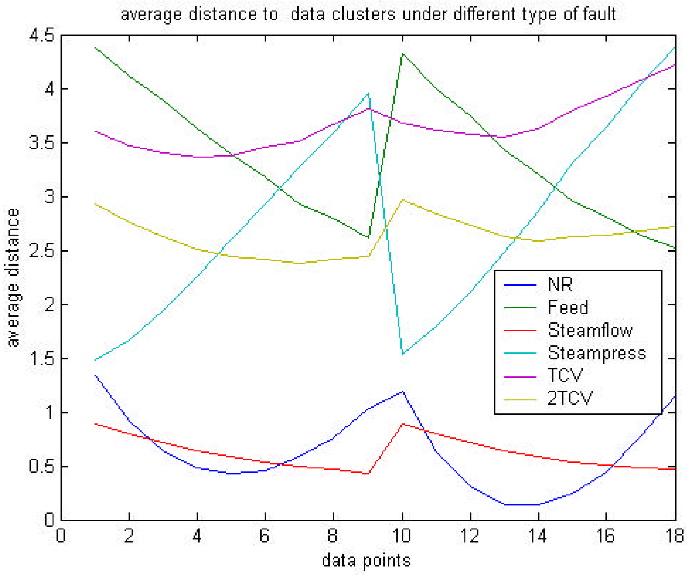


Figure 4.30. Fault direction indicated by the nearest neighbor method when the NR sensor is abnormal, system is under 84% of nominal power level.

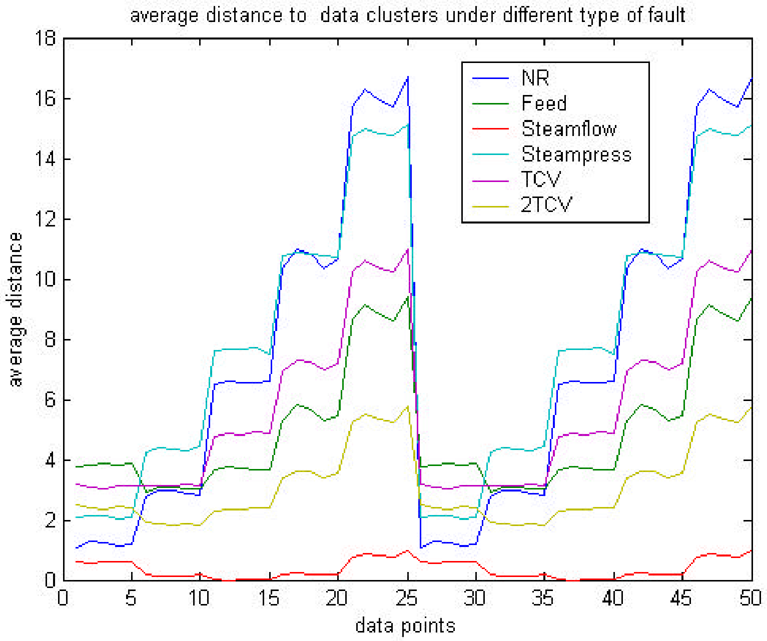


Figure 4.31. Fault direction indicated by the nearest neighbor method when the steam flow sensor is abnormal, system is under 96% nominal power level.

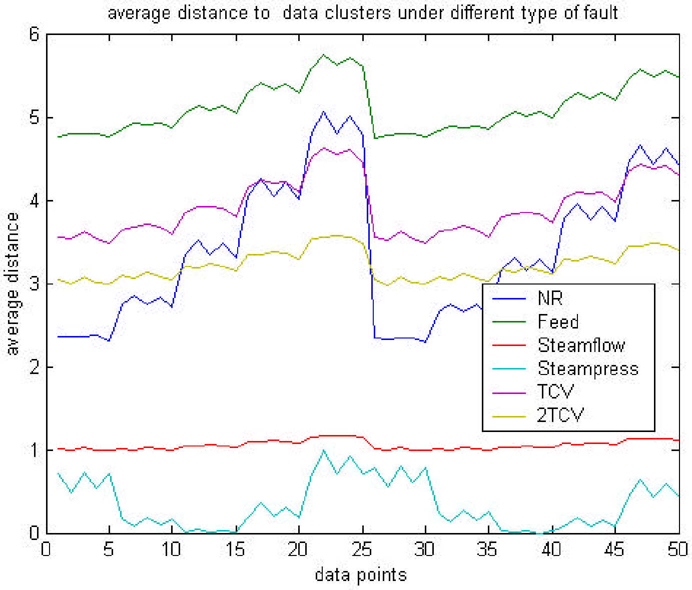


Figure 4.32. Fault direction indicated by the nearest neighbor method when the steam pressure sensor is abnormal, system is under 96% nominal power level.

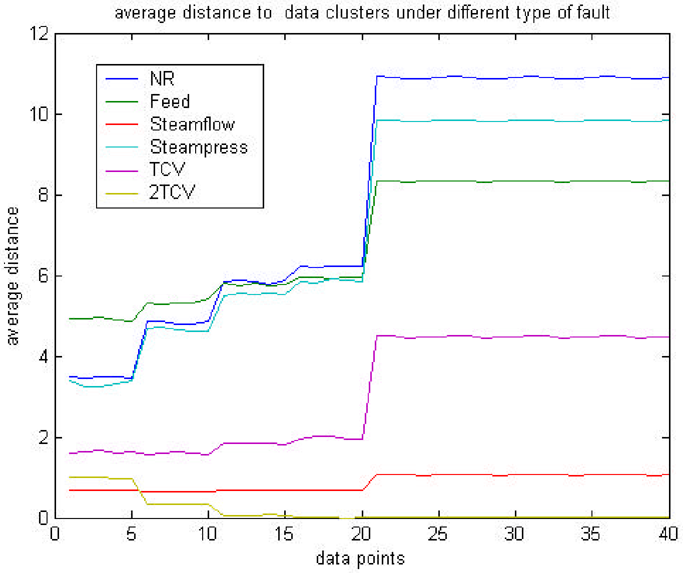


Figure 4.33. Fault direction indicated by the nearest neighbor method when two of the TCVs are abnormal, system is under 96% nominal power level.

4.5. Evaluation of Fault Isolation Methods

Three fault classification techniques are tested and compared in this chapter. In general, the rule-based method is robust in detecting and isolating the fault device under different operation conditions. It uses the knowledge base and operator experience about the behavior of residual patterns. It is obvious that the residual pattern vectors not only reflect the direction of system fault component, but also the severity of anomaly. But the fault is usually difficult to be isolated when the magnitude of the residuals is in the neighborhood of 1%, which is the threshold value.

PCA provides another convenient method in dealing with directional residuals from GMDH modules. The important assumption of PCA application is that the model residuals can be projected onto the first direction without missing the direction. In other words, the first PC provides the information about the faulty device more than the other PCs. We should be cautious that this assumption is not always satisfied, especially at the beginning of a fault transient. This algorithm is not highly sensitive to small anomalies in the system. In conclusion, PCA algorithm is an effective way to help in reaching the FDI objective, and deserves further study.

The third pattern classification technique uses the nearest neighbor criterion. The simple principle makes this method easy for implementation. In practice, this method is more suitable for data clusters that have multi-dimensional Gaussian distribution. One shortcoming of this method is that it is limited by the training database.

The parallel approaches, implemented in this chapter for fault detection and isolation of SG system devices, are effective in increasing the overall confidence level of making decision about the system status. Such a procedure is especially useful when the system dynamics is made complex because of changing system characteristics, nonlinearities, and temporal behavior. The demonstrated independent research in this thesis is step towards achieving this goal. The extension of FDI method to other subsystems is also very important and will be discussed in next chapter. In addition, a graphic user interface is also developed to demonstrate the performance of FDI.

5. PRINCIPAL COMPONENT ANALYSIS (PCA) BASED APPROACH FOR MULTIPLE FAULTS

(neriFinalReport02-06)

5.1. PCA Algorithms

From the point of view of data, the Principal Component Analysis (PCA) is a dimensional reduction method. The original data can be represented by a lower dimensional space without significant loss of the variability. From the modeling point of view, PCA transforms correlated variables into uncorrelated ones and determines the linear combinations with large and low variability (Flury, 1988).

Before the original data are transformed into a lower dimensional space, they are mean centered because only the variability of the data is of interest. The data are standardized with unit variance so that equal weights are given to all the variables as far as their variability is concerned. A complete description of the PCA is given in the Phase-2 report.

5.2. PCA for Fault Detection

During normal operation, the sample covariance of the measured data is governed by the physics in the process. Its structure will change if a fault occurs in the process. If a PCA model is used to describe the covariance structure of the measured data for the fault free condition, a fault can be detected when the model cannot explain the new observed data. Two cases may make the PCA model fail to explain the new data. The first case is that the new observation deviates from the mean of the normal operation defined by the effective region of the PCA model in the score space. The other case is that the residual of the model has changed significantly. The model residual represents the noise and the redundant information of a system. If a fault occurs, the characteristics of the noise and the redundancy are expected to change.

5.2.1. T^2 Statistics

If the score of a new observation is significantly outside the region defined by the scores of the fault free data, a fault may have occurred. If the scores of the fault free data satisfy multivariate normal distribution, the decision ellipse can be given by:

$$T'\Sigma^{-2}T < T_a^2 \tag{5.1}$$

where

$$T_a{}^2 = \frac{p(n-1)}{n-p} F_{p,n-p,a} \tag{5.2}$$

p = number of variables

n = number of observations

a = significance level.

The disadvantage of T^2 statistics is that it may be oversensitive to the small elements of Σ_a and result in high false alarm rate.

5.2.2. Q statistics

Q statistics can be used to test whether the principal component model can still explain a new observation. The random variable used for this testing is the sum squared error R of the original PCA model defined by the following equation:

$$R = r'r \tag{5.3}$$

where

$$r = (I - PP')x$$

If the sum squared error measuring the total sum of the variation in the residual space exceeds the Q threshold, it indicates that the original PCA model cannot explain the new data. The threshold of Q statistics Q_a is defined as follows (Jackson and Mudholkar, 1979)

$$Q_a = a(b + cz)^d \tag{5.4}$$

where

z= the critical value for standard normal distribution at a given significance level

$$a = \sum_{i=k+1}^{p} l_i$$

$$b = 1 + [q_2 h_0 (h_0 - 1)] / a^2$$

$$c = (\sqrt{2q_2} h_0) / a$$

$$d = 1 / h_0$$

$$q_2 = \sum_{i=k+1}^{p} l_i^{2}$$

$$q_3 = \sum_{i=k+1}^{p} l_i^{3}$$

$$h_0 = (1 - 2aq_3)/(3q_2^{2})$$

5.3. PCA for Fault Identification

The task of fault identification is to identify what are the most affected variables once a fault happens. These variables are usually most relevant to fault diagnosis. Fault identification is useful because it can help operators focus their attention on a reduced number of variables. The out-of-status score can be approximated by:

$$(\frac{t_i}{s_i})^2 > \frac{T_a^2}{k} \qquad i=1,...r$$

$r =$ the number of scores considered to be responsible for the out-of-control status.

$k =$ the number of principal components.

The contribution of one original variable to one of the out-of-control scores can be expressed as follows (Russell and Chiang, 2000):

$$C_{i,j} = \frac{(t_i)(\Delta x_j P_{ji})}{s_i^{\;2}}$$

where

$C_{i,j}$ = the contribution of variable x_j to the out-of-control scores t_i.

The total contribution of the j^{th} variable x_j to the out-of-control status can be given by:

$$\mathrm{C_j} = \sum_{i=1}^{r} C_{i,j}$$

C_i = the contribution of variable x_j to the out-of-control status.

The fault identification measure can also be defined based on the normalized error R_j (Russell and Chiang, 2000), which is given by:

$$R_j = \frac{r_j}{\hat{s}_j}$$

where

$$r_j = x_j - xP_j{}'P_j$$

$$\hat{s}_j^{\;2} = \sum_{i=k+1}^{m} P_{ij} s_i^{\;2} \qquad (5.5)$$

5.4. PCA Fault Isolation Versus Parity Space Approach

Gertler et al (1999) reported that there is inherent consistency between PCA approach and parity space approach when used for fault isolation. A linear static system can be described as follows:

$$y(t) = Au(t) + B\begin{pmatrix} \Delta u \\ \Delta y \end{pmatrix}$$

where

$y(t)$ = the observed outputs

$u(t)$ = the controlled inputs or the measured inputs

Δu = the disturbances or the unknown faults related to $u(t)$

Δy = the disturbances or unknown faults related to $y(t)$

A and B= known system matrices.

If we combine all the measured variables $u(t)$ and $y(t)$ as a column vector denoted as $x(t)$, a set of residuals can then be defined as:

$$o(t) = [-A, I]x(t) = B\Delta x \tag{5.6}$$

The residual vector $o(t)$ can be used as a parity vector for fault isolation since it is only influenced by the measurement error. When PCA is performed, the residual vector is given by

$$\begin{aligned} o(t) &= (x + \Delta x) - PP'(x + \Delta x) = (I - PP')(x + \Delta x) = QQ'(x + \Delta x(t)) = QQ'\Delta x \\ o(t) &= D\Delta x \end{aligned} \tag{5.7}$$

where

P = eigenvectors that span the principal component space

Q = eigenvectors that span the residual space

$D = QQ'$

In the above derivation, we have used the property of orthonormal matrices:

$$PP' + QQ' = 1$$

In addition, we have assumed that the variances of the scores on the eigenvectors corresponding to trivial components are approximately zero:

$$\text{var}(t_i) = \text{var}(Q'x) \approx 0$$

Correspondingly, if the original data are mean centered, then

$$Q'x = 0$$

In other words, the linear relationship between the variables can be represented by the eigenvectors corresponding to the trivial components.

To achieve fault isolation using PCA, structured residuals r(t) can be derived by a transformation as follows:

$$r(t) = Vo(t) = VD\Delta x(t)$$

where

V = a rotation matrix to achieve desired residual structure.

If the residual structure is obtained this way, the PCA based FDI approach will be in full agreement with the parity space approach.

5.5. PCA Fault Isolation Based on Fault Direction

Yoon and MacGregor (2001) reported that the fault directions both in the model space and in the residual space should be used in order to isolate a complex fault.

If a fault occurs in a control loop, the fault effects may propagate within the control loop after a new steady state is reached. Therefore, the developed PCA model from fault free conditions cannot be used to characterize the new relationship. This has twofold implications. The first one is that the linear redundant relationships between the variables have changed. The second one is that the system status has changed. The former can be represented by the residual change in the residual space and the latter can be represented by the score change in the model space.

Combining the system status change and the model structure change, a fault vector can be characterized by the superposition of two fault vectors defined in the model space and in the residual space as follows:

$$\vec{f} = f_1\hat{u} + f_2\hat{v} \tag{5.8}$$

where

$f_1\hat{u}$ = the fault vector defined in the principal component space;

$f_2\hat{v}$ = the fault vector defined in the residual space.

The developed PCA model for fault free conditions can be used to decompose a measurement vector x into two spaces, one component $x_1\hat{u}$ in the model space and the other one $x_2\hat{v}$ in the residual space, that is

$$x = x_1\hat{u} + x_2\hat{v} \tag{5.9}$$

Therefore, the fault direction in the residual space can be defined as:

$$\hat{v} = \frac{x_2^{post} - x_2^{initial}}{\| x_2^{post} - x_2^{initial} \|} \tag{5.10}$$

where

$x_2^{post} = (I - PP')x^{post}$
$x_2^{initial} = (I - PP')x^{initial}$
P =the loading matrix of the developed PCA model for fault free conditions.
x^{post}=the measurement obtained after a new steady state has reached since a fault
$x^{initial}$ =the measurement before a fault.

Since $x_2^{initial} \approx 0$, then

$$\hat{v} = \frac{x_2^{post}}{\| x_2^{post} \|} \tag{5.11}$$

The direction defined in the residual space characterizes the change of the model structure after a fault. However, the fault direction defined in the residual space may not be sufficient for fault isolation. The system status change before and after a fault also provides significant information to characterize a fault. The direction starting from the initial plant condition before a fault and pointing to the condition after a fault in the principal component space can be used to define the fault direction in the model space:

$$\hat{u} = \frac{x_1^{post} - x_1^{initial}}{\| x_1^{post} - x_1^{initial} \|} \tag{5.12}$$

where

$x_1^{post} = PP' x^{post}$
$x_1^{initial} = PP' x^{initial}$

After the fault signatures have been defined by fault directions, fault isolation can be achieved based on the angle of the fault vector both in the model space and in the residual space between a detected fault and some reference faults. A fault is isolated as one defined in a reference fault dictionary whose fault direction is most collinear with that of the detected fault both in the model space and in the residual space.

5.6. Determination of the Number of Constraint Equations of a System

Since PCA is in full agreement with the parity space approach when used for fault detection and isolation, it is crucial for a successful FDI system to find out all the constraints inherent in the process system. In the context of PCA based FDI, the constraint equations are implicitly represented by the eigenvectors spanning the residual space. Therefore, the correct choice of the number of principal components is important for PCA based FDI.

The most commonly used criteria are cumulative percent variance, Scree plot, average eigenvalue, and cross validation. Cumulative percent variance method selects the number of principal components by setting a subjective threshold of cumulative percent variance so that the model fitness and the parsimony in using principal components are balanced. Scree plot method is based on the plot of the fraction of variance explained by each principal component. The plot orders the principal components from the one that gives the largest amount of explanation to the one that gives the least amount of explanation. This method considers the beginning point of the Scree as the most reasonable number of principal components. Average eigenvalue method assumes that all the principal components whose corresponding eigenvalues are less than the average value should be discarded. Cross validation is recommended when used for model prediction (Wold, 1978).

5.7. Recommended PCA Based FDI Procedure

The procedure to implement a PCA based FDI is proposed as follows:

(1) Become familiar with the system.
(2) Get information on the operation history of the system and collect the operation experiences of similar plants.
(3) Select faults of interest from engineering point of view. The reliability data of the components, the environment of the components, the consequences of the component failure etc. should be taken into account.
As far as dual faults are concerned, the selection is mainly safety oriented.
(4) Study the fault responses of the selected faults.
(5) Collect data and evaluate its quality for fault free conditions.
(6) Build a PCA model able to characterize the system dynamics and the static relationships among the measured variables.
(7) Quantitatively define the fault directions for all the faults and save them in a fault dictionary. In effect, only one experiment or one simulation needs to be performed in order to determine the fault direction for each fault.
(8) Implement PCA fault detection using both Q statistics and T^2 statistics.
(9) Implement PCA fault isolation based on fault directions defined both in the model space and in the residual space.

5.8. Applications to Nuclear Plant SG System

The Principle Component Analysis based FDI algorithm has been implemented for PWR steam generator system.

5.8.1. Development of PCA models

A good model to characterize the relationships between the measured variables plays an important role in PCA based FDI algorithm.

Table 5.1 lists the fifteen measured variables used to develop the PCA model for the SG system. Before the simulated data are used to build a model, some noises in Gaussian distribution are added to the data based on the measurement errors of the corresponding sensors.

Figure 5.1 shows the fractions of the variance contained in the data explained by the 15 eigenvectors. It can be seen that the first three principal components can almost explain all the linear information, so the number of principal components may be chosen to be three. However, since there are nonlinear relationships among the variables, more principal components are necessary. Through cross validation, it is found eight principal components are the most parsimonious model that can give good prediction. It should be noted the choice of too many principal components would result in an inappropriately complicated model. A complicated model is able to reduce the training error, but it will lose the capability of generalization because some of the degrees of freedoms are only used for modeling the noise.

Figure 5.2 shows the predicted SG narrow range level and the actual values. It can be seen that the model can predict the trend of the actual data. The choice of more principal components may increase the accuracy of predicting the training data, but it may result in overfitting.

The nine eigenvectors to define the model space are as follow:

-0.2706	-0.0013	-0.0970	-0.0946	-0.0818	-0.2759	0.1669	0.3104
-0.2673	0.0053	-0.2591	0.3724	-0.3567	0.6139	-0.2662	0.3839
-0.2701	-0.0006	-0.1373	-0.0628	-0.0795	-0.1317	0.0112	-0.0808
0.2499	-0.0442	-0.6176	0.1526	0.0334	-0.1397	0.0593	-0.0636
-0.2666	-0.0229	-0.2695	-0.4125	0.7342	0.3625	-0.0562	0.0975
-0.2709	0.0074	-0.0580	0.0030	-0.1170	0.1376	0.2739	-0.3698
-0.2709	0.0064	-0.0576	0.0014	-0.1221	0.1314	0.3564	-0.3026
0.0268	0.9976	-0.0610	-0.0125	0.0149	-0.0075	-0.0000	0.0017
-0.2693	0.0166	0.1404	0.7813	0.5015	-0.2066	0.0428	-0.0255
-0.2709	0.0025	-0.0661	-0.0614	-0.0998	-0.2263	-0.5195	-0.2465
-0.2706	-0.0006	-0.0990	-0.0795	-0.0403	-0.2202	-0.5298	-0.2459
-0.2706	-0.0014	-0.0968	-0.0956	-0.0829	-0.2842	0.1756	0.3432
-0.2706	-0.0014	-0.0968	-0.0956	-0.0829	-0.2848	0.1756	0.3427
-0.2709	0.0074	-0.0580	0.0030	-0.1171	0.1380	0.2736	-0.3691
0.2497	-0.0441	-0.6198	0.1270	0.0154	-0.1386	0.0584	-0.1204

The seven eigenvectors to define the residual space are as follow:

0.0142	0.1209	0.0868	-0.5418	0.0572	0.6171	-0.0379
0.0165	-0.0024	0.0097	-0.0023	-0.0003	0.0020	-0.0004
-0.0437	-0.9318	0.0606	0.0029	-0.0031	-0.0036	-0.0002

-0.7060	0.0635	-0.0159	0.0004	0.0008	0.0001	-0.0002
0.0059	0.0185	-0.0457	-0.0030	0.0006	0.0031	-0.0001
-0.0116	0.1161	0.0205	0.0127	-0.3934	0.0088	-0.7140
-0.0099	0.1070	-0.0597	-0.0433	-0.4195	0.0017	0.6925
-0.0001	0.0011	-0.0002	-0.0000	-0.0004	0.0000	0.0006
0.0196	-0.0074	-0.0060	-0.0004	0.0001	0.0006	0.0000
-0.0077	0.1015	-0.7152	-0.0456	0.0102	0.0513	-0.0007
-0.0051	0.1931	0.6822	0.0506	-0.0591	-0.0618	0.0824
0.0165	0.1168	-0.0467	0.7910	-0.0176	0.1729	0.0018
0.0160	0.1173	-0.0453	-0.2680	0.0117	-0.7629	-0.0445
-0.0112	0.1127	0.0315	0.0470	0.8136	-0.0286	0.0212
0.7056	0.0058	-0.0143	-0.0006	-0.0002	0.0004	0.0004

Table 5.1. Measured variables used to develop PCA model

Variable number	Variable Description
1	Thermal power
2	Cold leg 1 temperature
3	Hot leg 1 temperature
4	SG1 pressure
5	Feed water temperature
6	Feed water flow rate to SG1
7	SG1 steam flow rate
8	FCV1 position
9	FCV1 controller output
10	SG1 WR indicated level
11	SG 1 NR indicated level
12	SG WR reference
13	SG NR reference
14	TCV1 flow rate
15	SG1 temperature

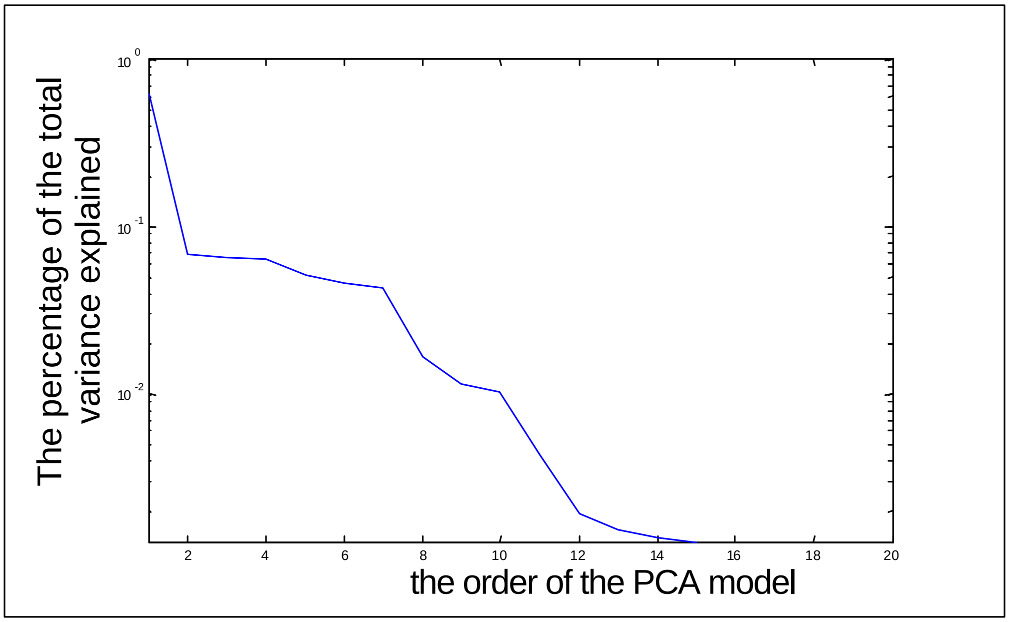


Figure 5.1. The fractions of the variance explained by different PC components

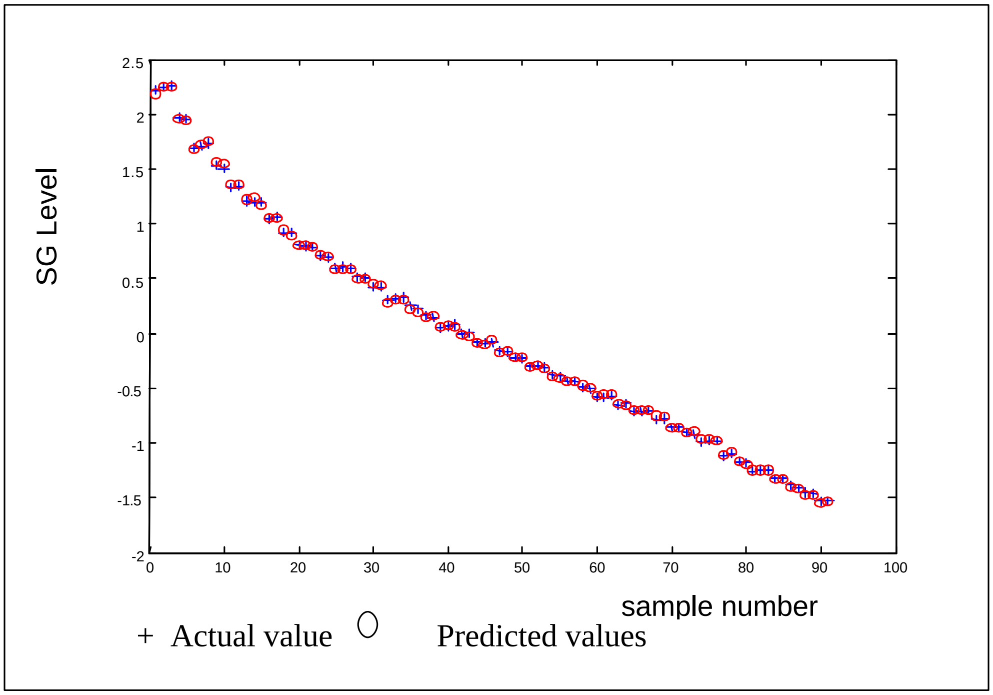


Figure 5.2. Comparison between the predicted SG NR level and the actual values.

From the eigenvectors in the residual space, the following approximate linear relationships among the measured variables can be derived:

-0.7140*steam flow+0.6925*feed flow=0;

0.6171*power+0.1729*WR reference level -0.7629*NR reference level=0;

-0.3934*steam flow-0.4195*feed flow+0.836*TCV flow=0.0;

-0.5418*power+0.7910*WR reference level-0.2680*NR reference level+0.047*TCV flow=0.0;

0.0868*power+0.0606*hot leg temperature-0.0597*Feed Flow rate-0.7152*WR level+0.6822*NR level=0.0;

0.1209*power-0.9318*hot leg temperature-0.0635*SG pressure+0.1161*Steam Flow+0.1070*Feed water flow+0.1015*WR level+0.1931*NR level+0.1168*WR reference level+0.1173*NR reference level+0.1127*TCV flow=0.0;

-0.7060*SG pressure + 0.7065*SG temperature=0.0;

These equations may be used to reveal the linear relationship among the variables. The corresponding physical relations can be written as follow:

Steam flow rate = feed water flow rate;

SG wide range reference level =f(SG narrow range reference level, power);

Steam flow rate + feed water flow rate = TCV flow rate;

SG reference level = f (power, TCV flow rate);

SG narrow range level =f(SG wide range level, power, hot leg temperature, feed water flow rate);

Hot leg temperature = f (power, SG pressure, SG NR level, SG flow rate);

SG temperature =f (SG pressure);

As can be seen, all the above equations have clear physical meanings. However, PCA model cannot capture the nonlinear relationship among variables. For example, the PCA model cannot reveal the relationship between FCV valve position and FCV flow rate. Another point that should be emphasized in using PCA for FDI is that the measurements must be carefully selected before a PCA model is to be built. If the available measurements do not allow finding out some relations among variables that are the basis to isolate some faults, these faults will hence not be able to be isolated.

5.8.2. Fault detection

The PCA models built in section 5.7 are used for fault detection.

Figures 5.2 and 5.3 show the T-Square and Q-statistics for the fault free data, respectively. The red lines in the two figures are the T square or the Q statistical limits corresponding to 99% confidence level. If the corresponding statistics exceeds the limit, the confidence to state that the fault free model cannot explain the data is at a level greater than 99 %. The two figures illustrate that all the fault free data are well below the limit lines. The probability of false alarms due to process disturbance is low.

Figures 5.4 and 5.5 show the T-square and Q-statistics based fault detection for feed water flow meter and steam flow meter drift faults. If the confidence level is chosen to be 99%, the false alarm rate and missing detection rate is shown as follow:

```
Detecting Fault: Normal Operation
 PCA detection
 False alarm rates by T2+Q testing  = 0.04
 False alarm rate by T2 testing  = 0.03
 False alarm rate by Q testing  = 0.01
Detecting Fault: Feed Water Flow Meter Drift Fault
 PCA detection
 Missing detection rate by T2+Q testing  = 0.000000
 Missing detection rate by T2 testing  = 0.012547
 Missing detection rate by Q testing  = 0.095358
Detecting Fault: Steam Flow Meter Drift fault
 PCA detection
 Missing detection rate by T2+Q testing  = 0.000000
 Missing detection rate by T2 testing  = 0.011292
 Missing detection rate by Q testing  = 0.100376
Detecting Fault: Steam Flow Meter Feed Flow Meter Drift Faults
PCA detection
 Missing detection rate by T2+Q testing  = 0.000000
 Missing detection rate by T2 testing  = 0.010038
 Missing detection rate by Q testing  = 0.115433
Detecting Fault: Feed Flow Meter Drift Fault and SG Level Sensor Drift Fault
 PCA detection
 Missing detection rate by T2+Q testing  = 0.000000
 Missing detection rate by T2 testing  = 0.013802
 Missing detection rate by Q testing  = 0.077792
Detecting Fault: Steam Flow Meter Drift Fault and SG Level Sensor Drift Fault
 PCA detection
 Missing detection rate by T2+Q testing  = 0.000000
 Missing detection rate by T2 testing  = 0.006274
 Missing detection rate by Q testing  = 0.042660
Detecting Fault: SG Pressure Sensor Drift Fault
 PCA detection
 Missing detection rate by T2+Q testing  = 0.000000
 Missing detection rate by T2 testing  = 0.002789
 Missing detection rate by Q testing  = 0.068042
Detecting Fault: Feed Water Flow Meter Drift fault & SG Pressure Sensor Drift Fault
 PCA detection
 Missing detection rate by T2+Q testing  = 0.000000
 Missing detection rate by T2 testing  = 0.001255
 Missing detection rate by Q testing  = 0.097867
Detecting Fault: SG Level Sensor Drift Fault & SG Pressure Sensor Drift Fault
 PCA detection
```

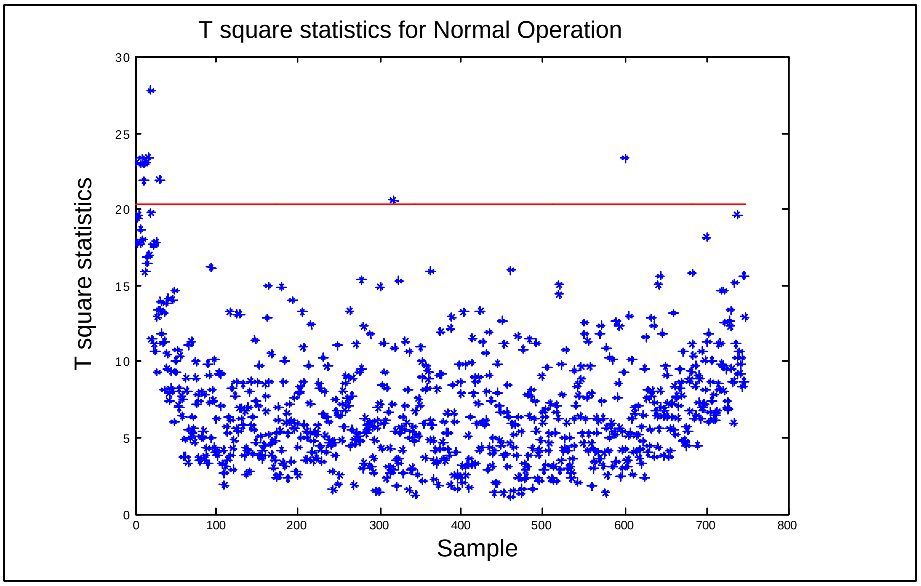


Figure 5.2. T square statistics for the normal data

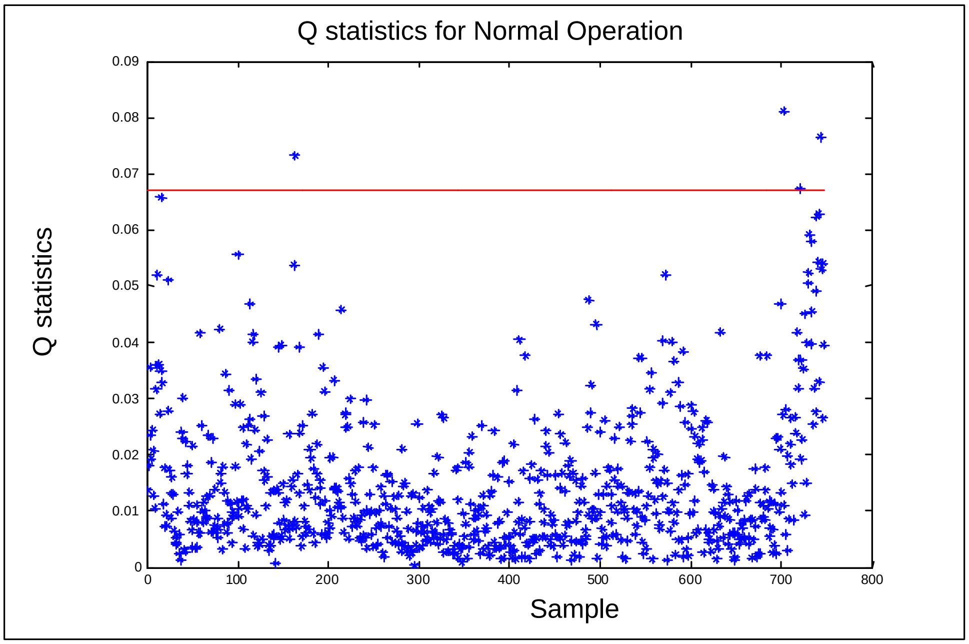


Figure 5.3. Q statistics for the normal data.

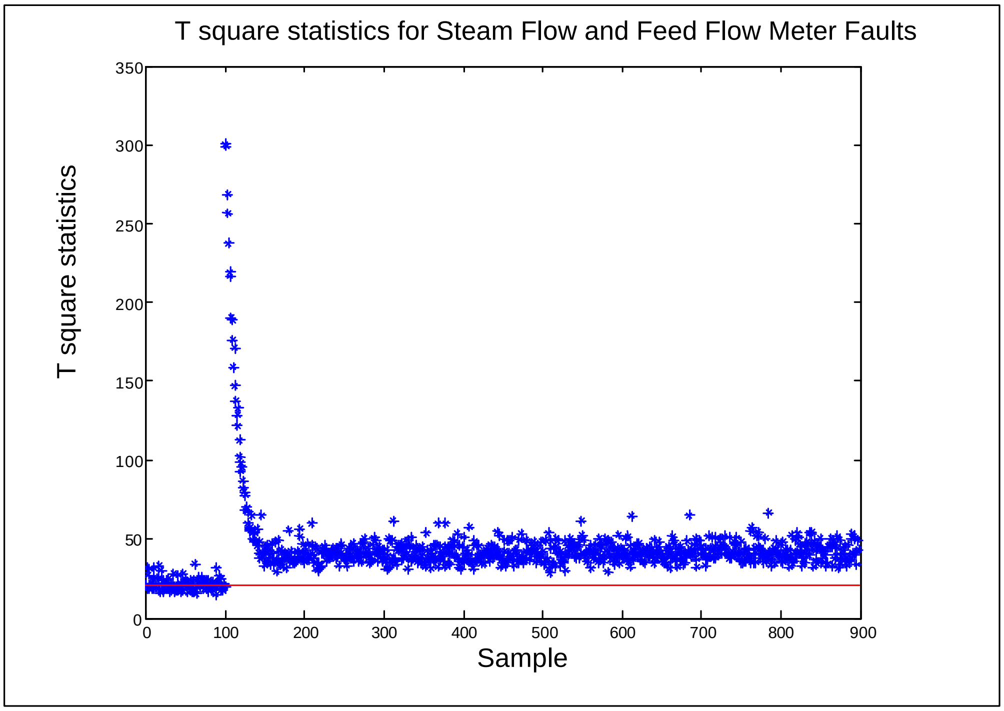


Figure 5.4. T square statistics to detect steam flow meter and feed water flow meter drift faults

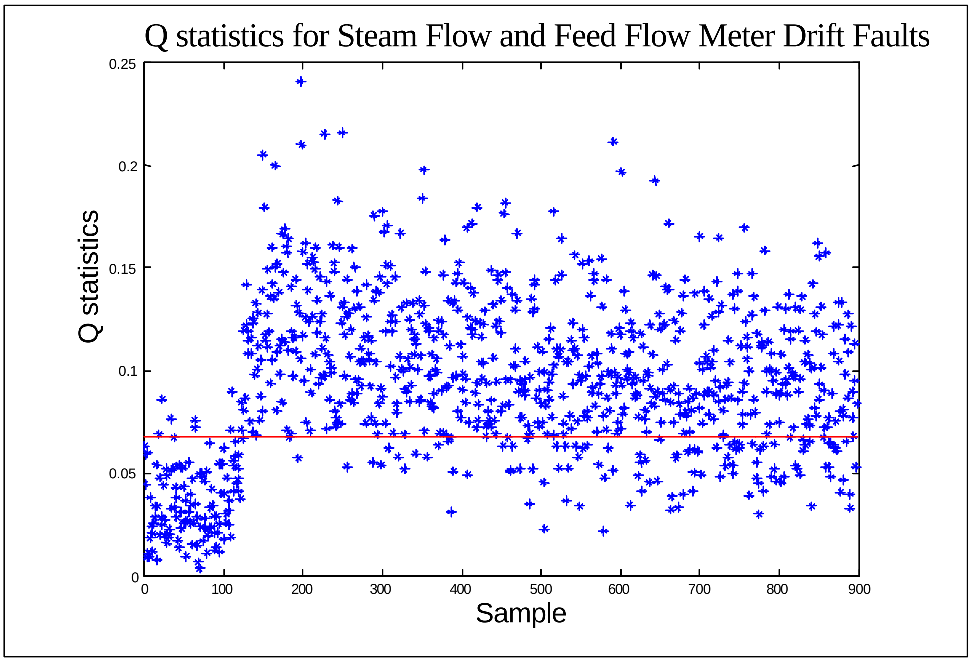


Figure 5.1. Q statistics for steam flow meter drift fault and steam flow meter drift fault.

```
Missing detection rate by T2+Q testing  = 0.000000
Missing detection rate by T2 testing  = 0.006274
Missing detection rate by Q testing  = 0.096612
Detecting Fault: SG Level Sensor Drift Fault.
```

It is seen that the missing detection rate is small for all the selected faults. It should be kept in mind that both T^2 and Q statistics must be used for fault detection. Either statistics being violated will signify that a fault has happened. The violation of T^2 statistics represents that the system operates at an abnormal state beyond the model space. The violation of Q statistics represents that some of the constraint equations defined in the residual space are violated and the system is abnormal.

PCA can only deal with steady state condition or a slow dynamic process. The algorithm to perform PCA based fault detection is only applicable to steady state conditions. When the false alarm rate and the missing detection rate are carefully examined, the false alarm rate and the missing detection rates are not equal to the expected value of one percent. The reason is that the probabilistic distribution underlying the data used to build the model is not normal. Therefore, it is reasonable that the false alarm rate and the missing detection rate are not equal to the specified significance level. The significance level should be determined using experiences obtained from testing the FDI design on the process system.

It should also be noted that the confidence level would affect the false alarm rate and the missing detection rate. A higher confidence level tends to result in a smaller false alarm rate but a higher missing detection rate. In a real application, the confidence level needs to be adjusted according to the operation requirements.

5.8.3. Fault identification

Figures 5.6 and 5.7 show the contribution plots of the abnormal scores and the abnormal residuals for the feed water flow meter drift fault, respectively. The contribution plots show that the most affected variables for the feed water flow meter drift fault as follows:

- reactor power;
- feed water temperature;
- feed water flow rate;
- steam flow rate;
- SG NR level;
- SG temperature.

All the identified variables are in agreement with the analysis of the fault responses. The feed water flow rate has been successfully identified as an important variables of concern.

The fault identification does not give immediate results to isolate faults. It only provides information about what variables significantly contribute to the residuals. This is true especially in the case that a feed back controller is involved since all the measurements within the control loop may be affected by a fault in the closed loop.

5.8.4. Fault isolation

The objective of fault isolation is to determine whether the fault is known in the fault dictionary and to determine which fault is the most likely one after the fault has been detected.

The fault direction jointly defined in the model space and in the residual space has been used as fault signature for fault isolation. By the way, fault signature is defined by fault direction instead of distance. The distance-based residual classification algorithm should not be used for fault isolation. Such pattern classification can be used only under the following conditions:

- The fault directions of the concerned faults differ from each other significantly.
- The magnitudes of the faults are predefined within a certain range.

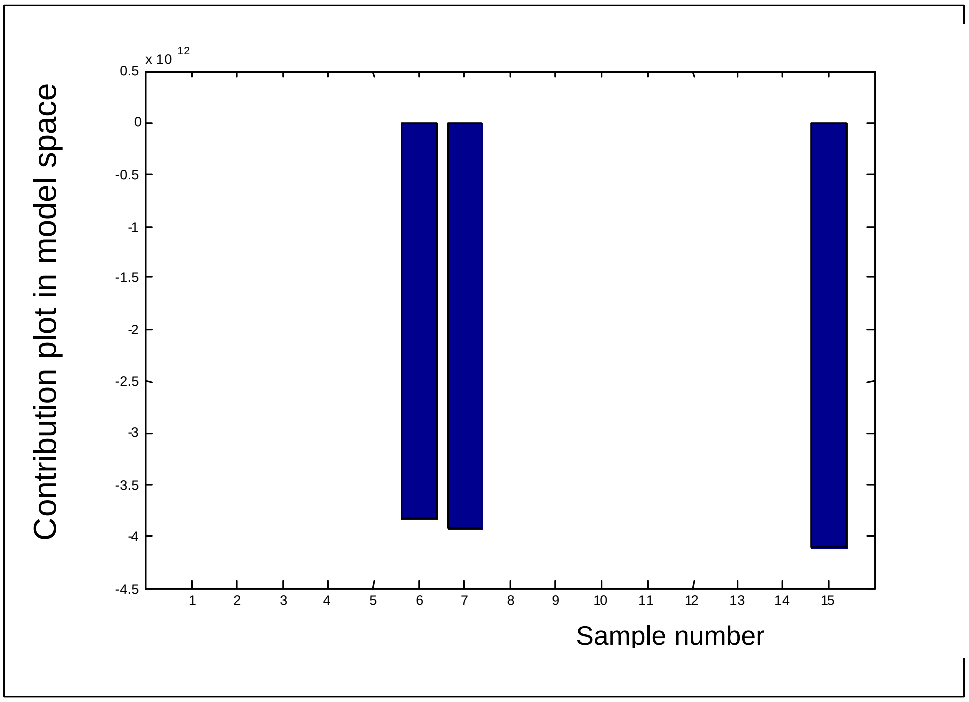


Figure 5.6. Contribution plot in the model space for feed water flow meter fault

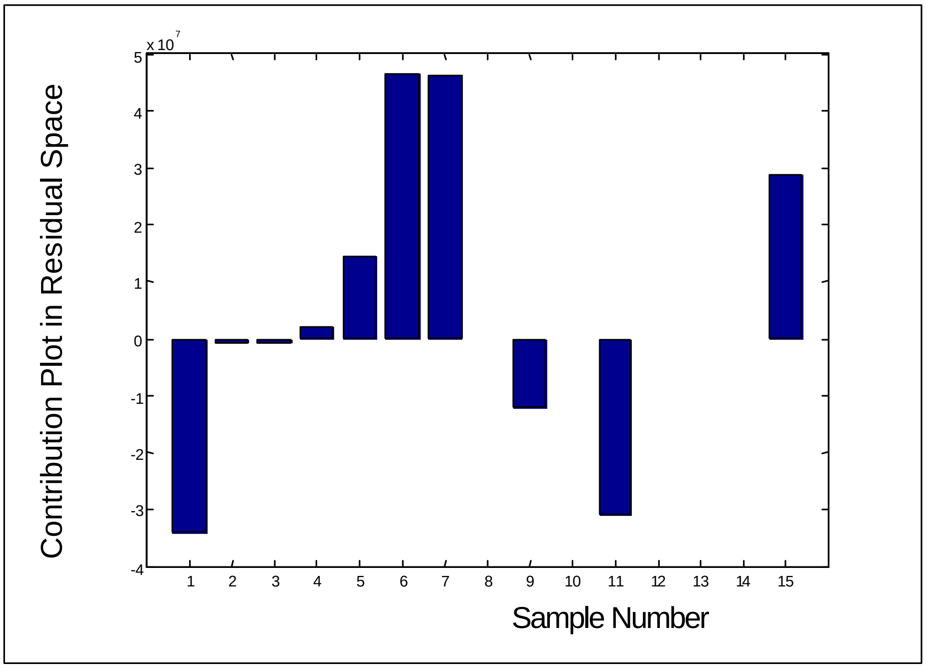


Figure 5.7. Contribution plot in the residual space for feed water flow meter fault

Figure 5.8 shows the fault direction in the model space and in the residual space for SG NR level sensor fault without using SG wide range level sensor. The fault direction is represented by the cosine angle of the fault directions between SG NR level sensor fault and all the 13 reference faults. These reference faults are numbered as follows:

- Feed water flow meter offset fault;
- Steam flow meter offset fault;
- Feed water flow meter offset fault and steam flow meter offset fault;
- Feed water flow meter offset fault and SG level sensor offset fault;
- Steam flow meter offset fault and SG level sensor offset fault;
- SG pressure sensor offset fault;
- Feed water flow meter offset fault and SG pressure sensor offset fault;
- Feed water flow meter offset fault and FCV offset sensor offset fault;
- Steam flow me ter offset fault and FCV position offset fault;
- FCV valve position offset fault;
- SG pressure sensor offset fault and SG level sensor offset fault;
- SG level sensor offset fault;
- Steam flow meter offset fault and SG pressure sensor offset fault.
-

Based on the fault direction in the residual space (green bar in the figure), SG NR level sensor fault can be strongly isolated from feed water flow meter offset fault. Based on the fault direction in the model space (purple bar in the figure), SG NR level sensor fault can be strongly isolated from steam flow meter fault, SG pressure sensor fault and FCV position fault. It proves that the joint fault direction both in the model space and in the residual space is sensitive enough to characterize SG NR level sensor fault.

Figure 5.8 shows the fault direction in the model space and in the residual space for feed water flow meter sensor fault and SG NR level sensor fault without using SG wide range level sensor. As can be seen, the feed water flow meter fault cannot be distinguished from feed water flow meter sensor fault plus SG level sensor fault. This is because the symptoms of a feed water flow meter positive offset fault envelope all those of SG NR level sensor positive offset fault. Therefore, no additional information can be used to uniquely isolate SG NR level sensor fault.

In order to isolate all the selected thirteen faults including dual faults, SG wide range level signal must be used. Figure 5.9 shows the fault direction both in the model space and in the residual space for the feed flow meter fault after the SG wide range level sensor has been used. As can be seen, feed water flow meter fault can then be isolated from feed water flow meter sensor fault plus SG level sensor fault.

Furthermore, an important criterion to judge if the designed FDI scheme is successful or not is to test the stability of the fault signatures in different fault magnitudes and under different initial operation conditions. For this reason, a set of data in fault magnitude of three percent under the initial power level at 80% full power, which are unknown to the fault dictionary, are generated to test the reliability of the designed FDI system.

Figures 5.10-5.21 show the fault direction both in the model space and in the residual space for the defined 13 faults respectively in magnitude of three percent under the initial power level at 80% full power. As can be seen, in most cases, the fault direction in either model space or residual space is not enough to isolate dual faults. For example, the fault direction in the model space (purple bar in the figure) for steam flow meter offset fault is similar to that for steam flow meter offset fault plus SG NR level sensor fault (See Figure 5.10). Nonetheless, the fault direction in the residual space (green bar in the figure) for steam flow meter offset fault is quite different from steam flow meter offset fault plus SG NR level sensor fault. An opposite example is that the fault direction in the model space helps to isolate a fault. The fault direction in the residual space for steam flow meter offset is similar to that for steam flow meter offset fault plus FCV position fault (See also Figure 5.10). Nevertheless, the fault direction in the model space for steam flow meter offset fault is quite different from steam flow meter offset fault plus FCV valve position offset fault. Therefore, when the joint fault direction is used, there is more possibility to isolate faults.

The cosine of the angle between the fault direction of an unknown fault and that of the reference faults can also be used as confidence level when a decision is to be made. Figure 5.21 shows there is no significant margin to isolate a FCV position fault from a steam flow meter offset fault plus FCV position fault. Figure 5.10 shows that no significant margin exists to isolate a SG pressure sensor fault from a SG pressure sensor fault plus a steam flow meter offset fault. Therefore, when decisions are made, the confidence level to isolate these two faults should be taken into account.

5.8.5. Discussion

This chapter has presented the PCA approach to fault detection and isolation and its application to PWR steam generator system. The PCA approach is shown to be in agreement with parity space approach. The linear relationship among measured variables implying analytical redundancy can be consistently represented by the eigenvectors corresponding to the trivial components. The fault directions jointly defined both in the model space and in the residual space is a sensitive fault signature for fault isolation.

PCA approach needs the least information about a system when applied to FDI. It is simple to achieve on-line implementation. It provides an ideal tool to supervise plant status without too much investment. However, PCA approach has many inherent weaknesses. From modeling point of view, linear PCA is only applicable to a linear static system. It is difficult to develop a nonlinear PCA model for a dynamic system accurate enough to reveal the analytical redundancy inherent in a physical system. With regard to fault isolation, the fault characteristics must be defined from fault data for the enumerated faults. This exerts heavy burden on engineering application. In addition, the fault isolation is a process of classification, so the decision has poor interpretability.

Because PCA approach has duality with parity space approach, it is very important to validate the constraint equations extracted from PCA modeling. If process variables are not appropriately chosen, some constraint equations necessary for fault isolation may not be obtained. If the number of principal components is chosen incorrectly, the residual direction cannot be used to characterize a fault.

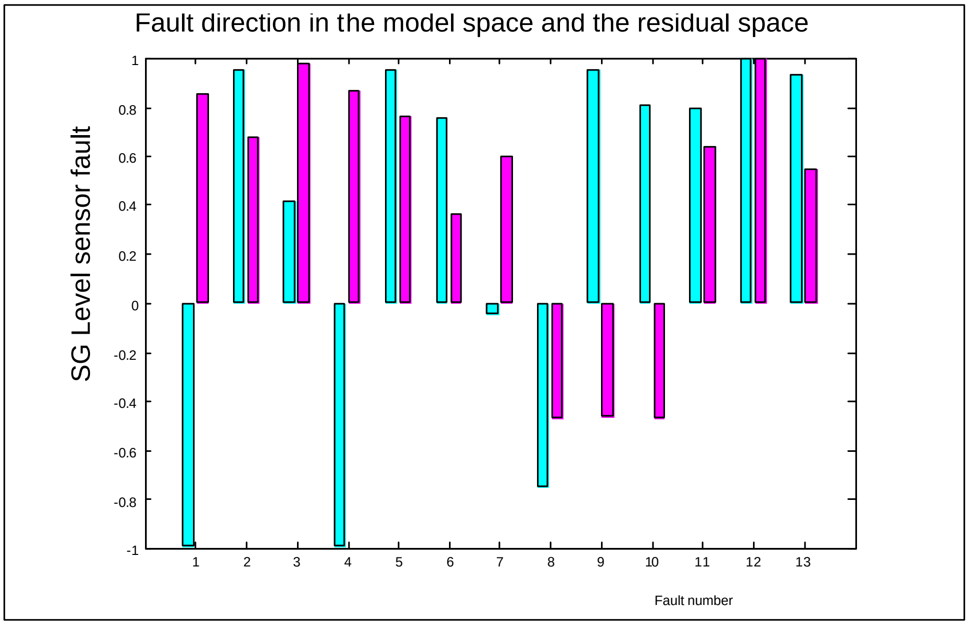


Figure 5.8. Fault direction for SG NR level sensor fault without using SG WR level signal.

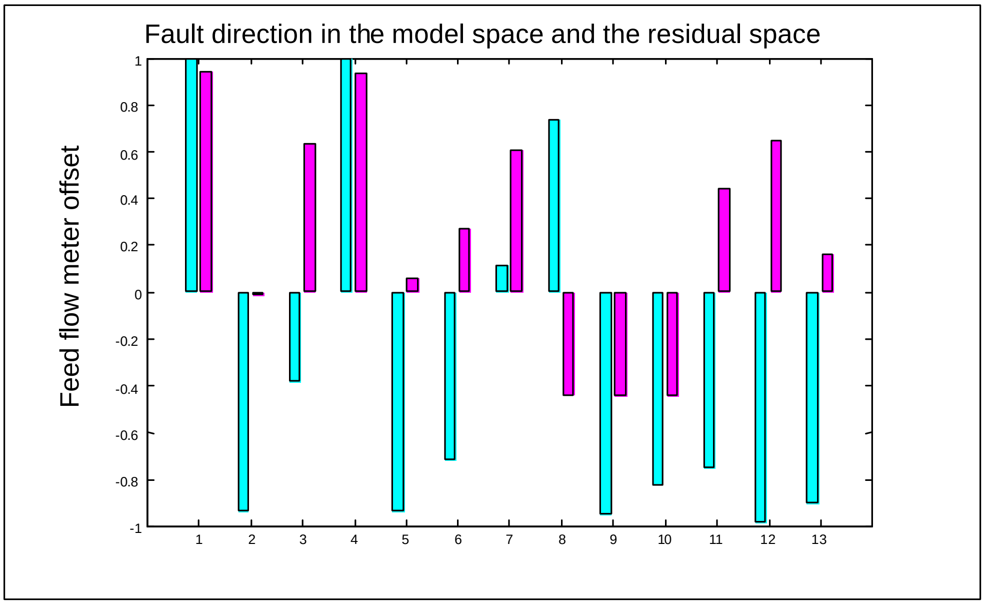


Figure 5.9. Fault direction for feed water flow meter offset fault and SG NR level sensor offset fault without using SG WR level signal

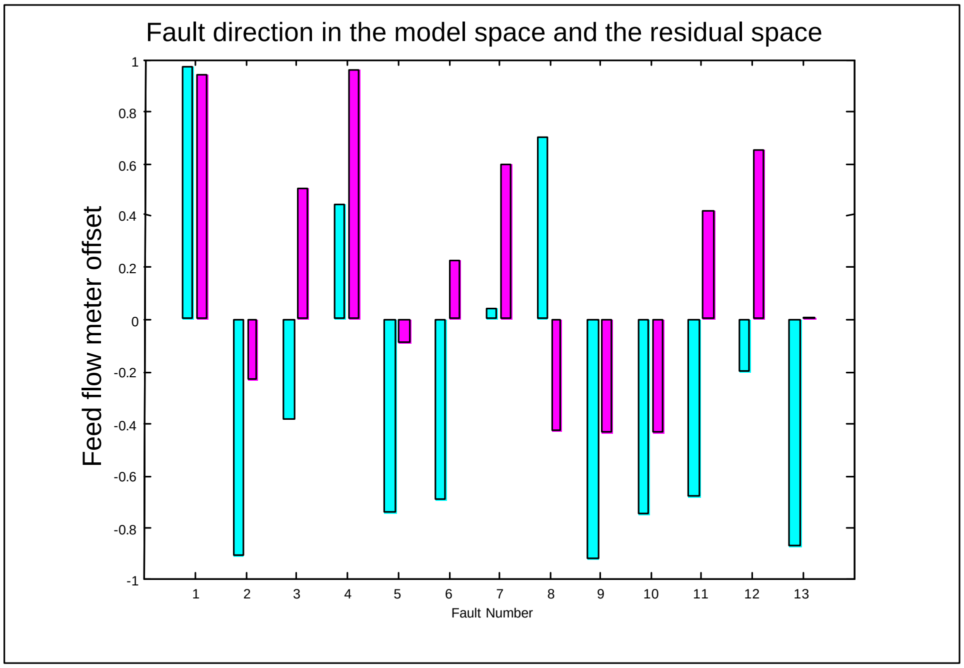


Figure 5.10. Fault direction for feed water flow meter offset fault

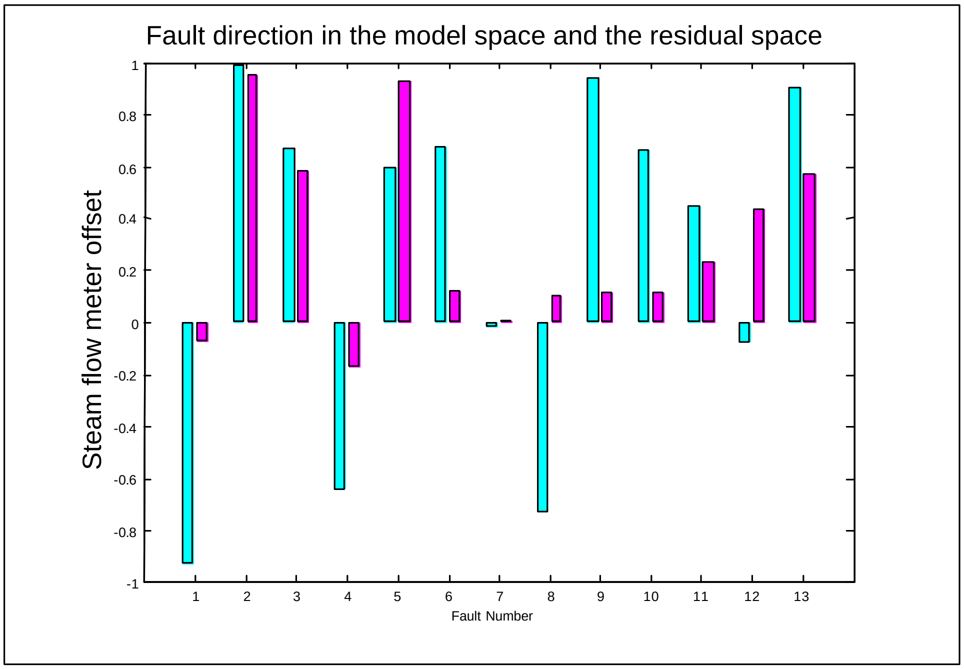


Figure 5.11. Fault direction for steam flow meter offset fault

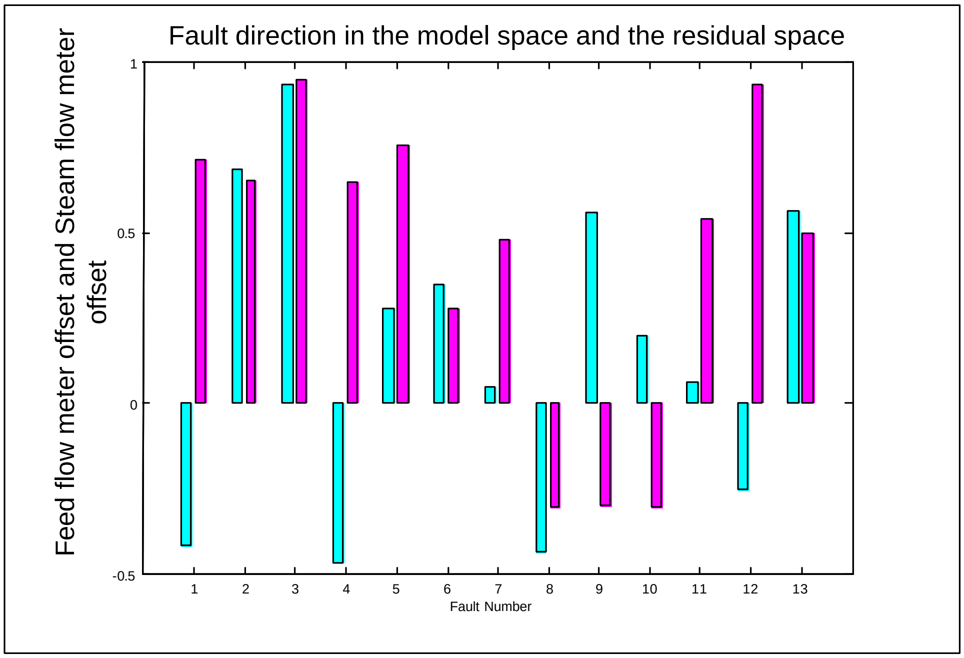


Figure 5.12. Fault direction for feed water flow meter offset fault and steam flow meter offset fault

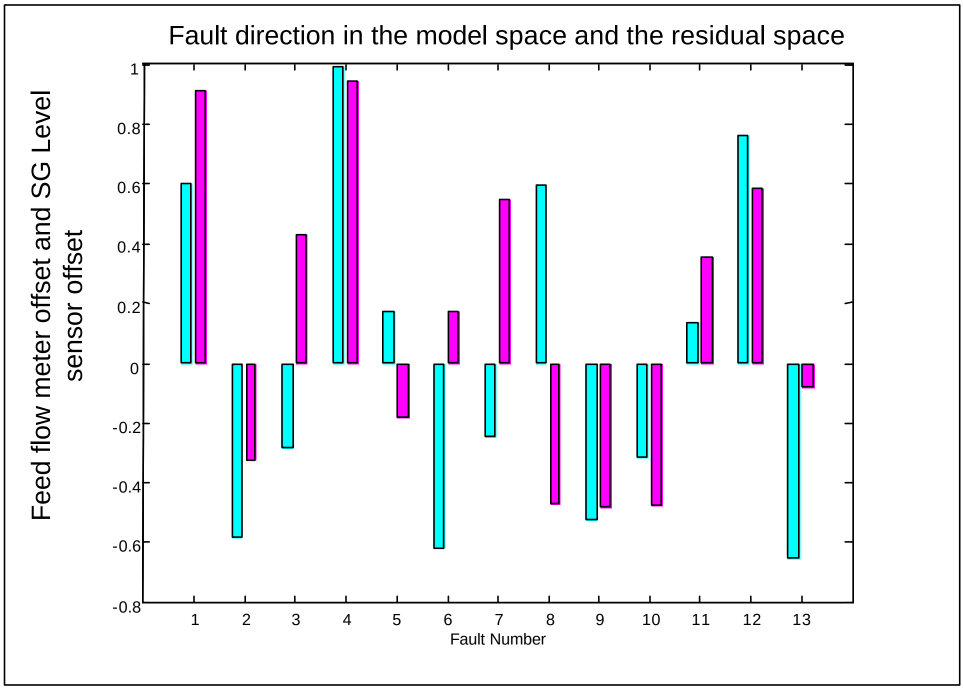


Figure 5.13. Fault direction for feed water flow meter offset fault and SG NR level sensor offset fault

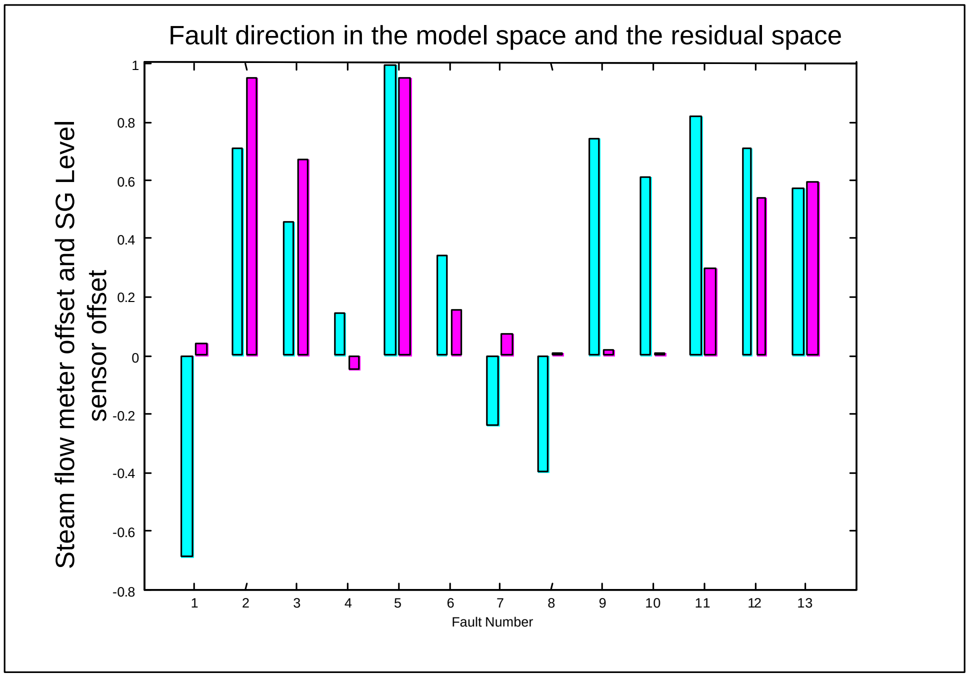


Figure 5.14. Fault direction for steam flow meter offset fault and SG NR level sensor offset fault

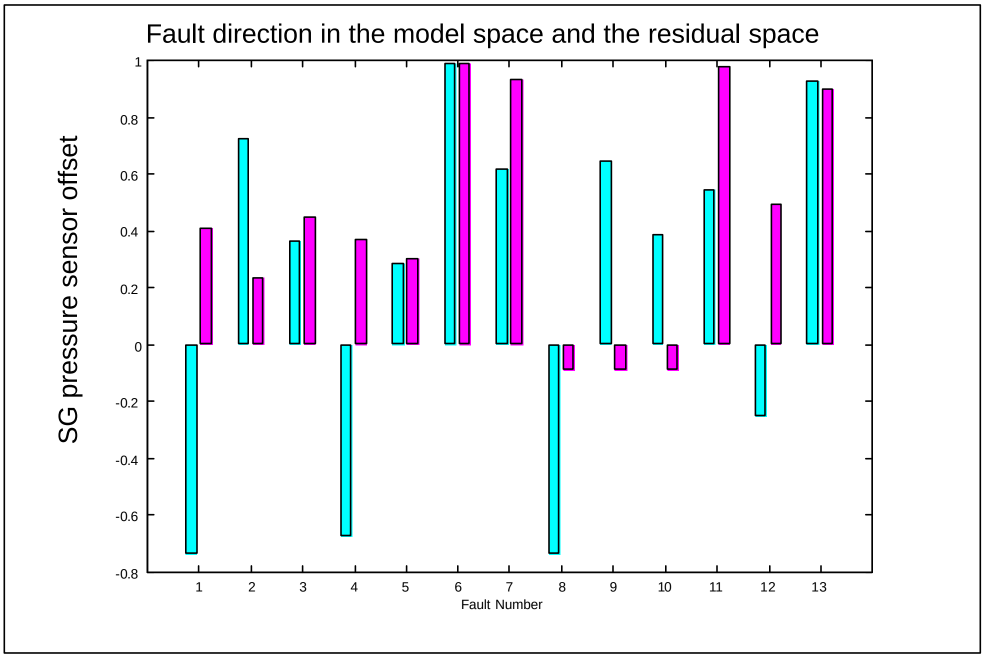


Figure 5.15. Fault direction for SG pressure sensor offset fault.

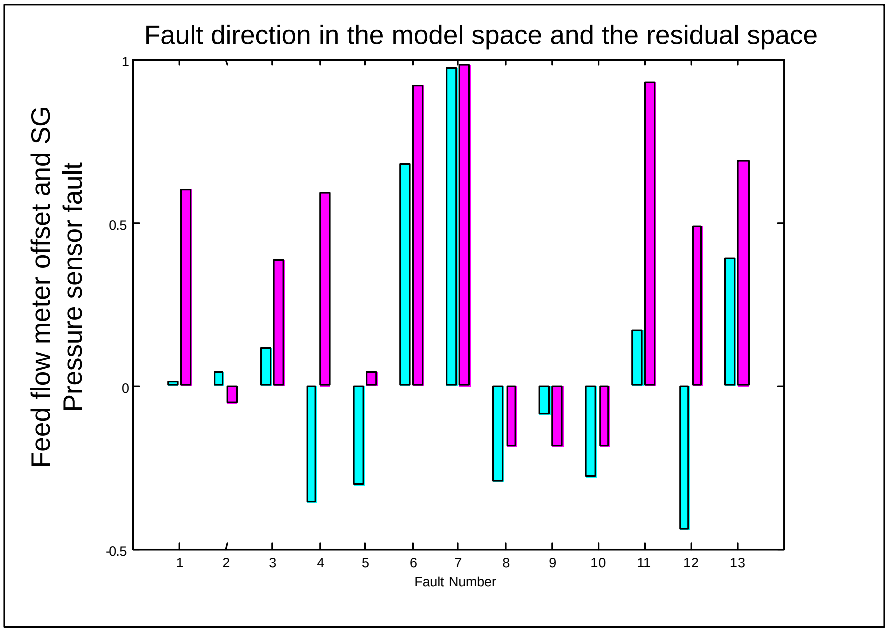


Figure 5.16. Fault direction for feed water flow meter offset fault and SG pressure sensor offset fault.

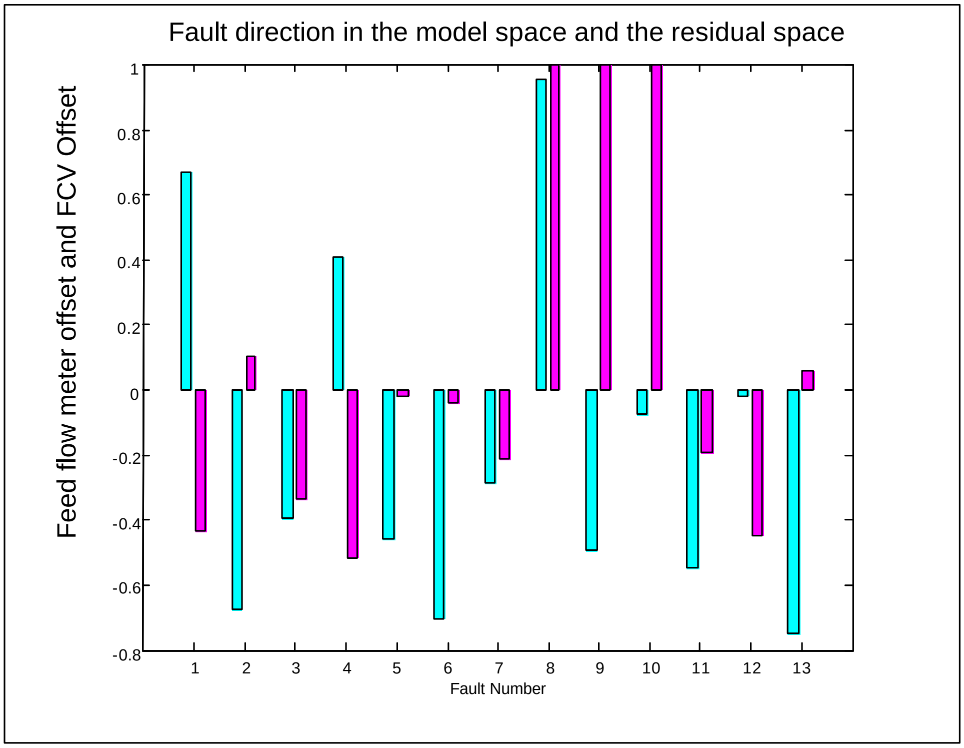


Figure 5.17. Fault direction for feed water flow meter offset fault and FCV position offset fault.

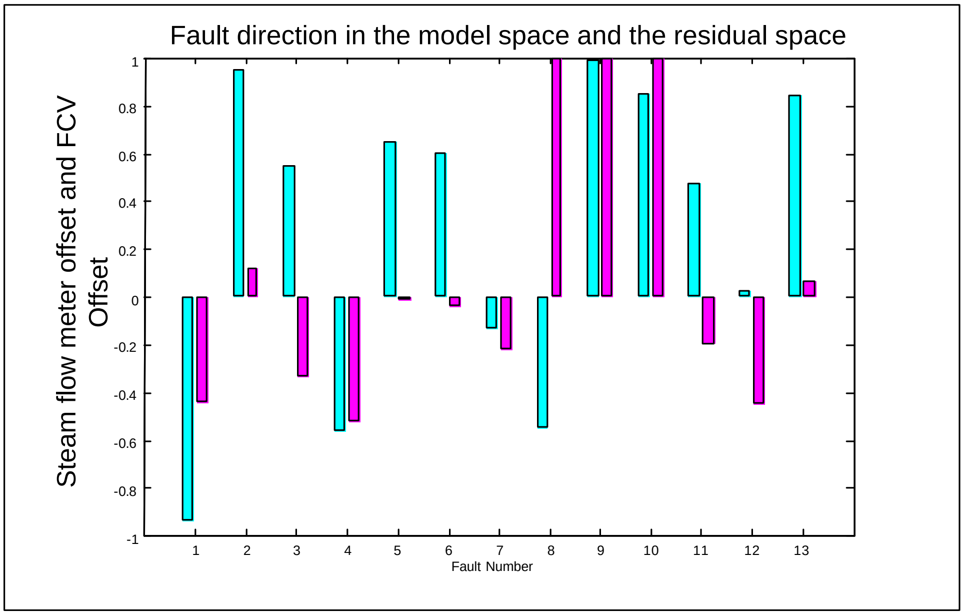


Figure 5.18. Fault direction for steam flow meter offset fault and FCV position offset fault.

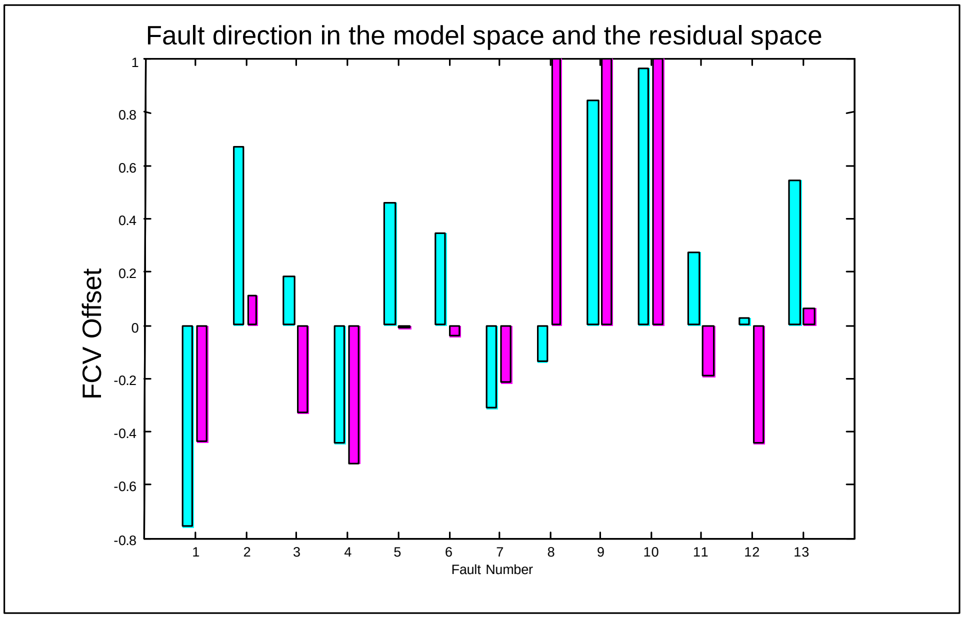


Figure 5.19. Fault direction for FCV position offset fault.

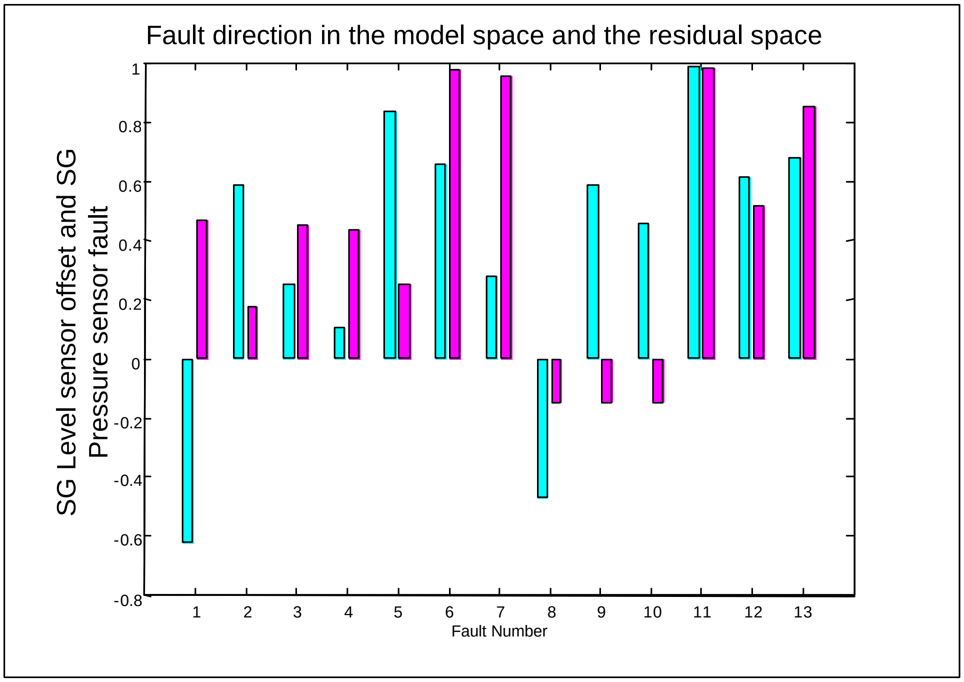


Figure 5.20. Fault direction for SG level sensor offset fault and SG pressure sensor offset fault.

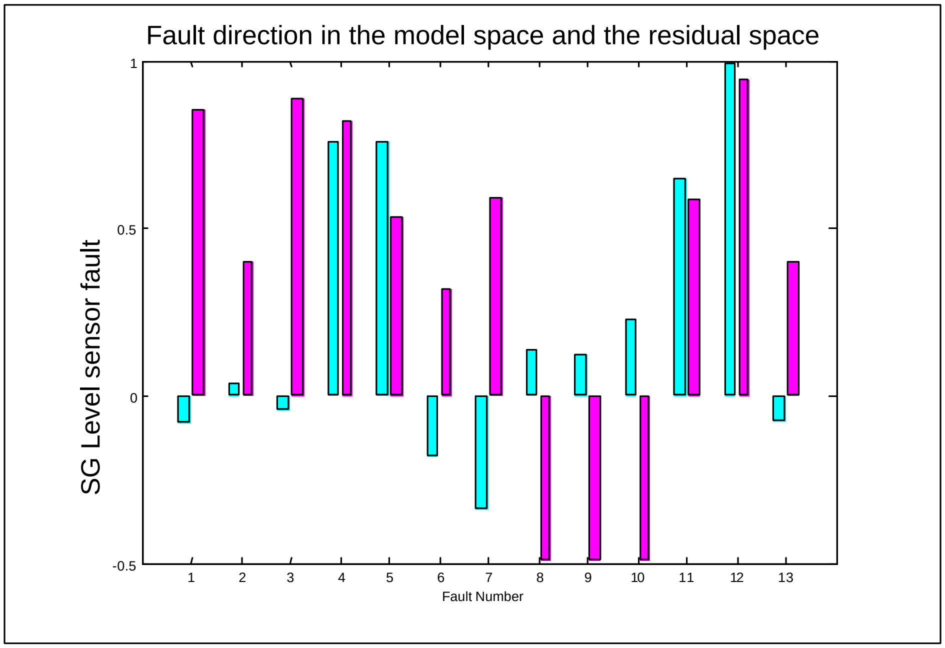


Figure 5.21. Fault direction for SG level sensor offset fault.

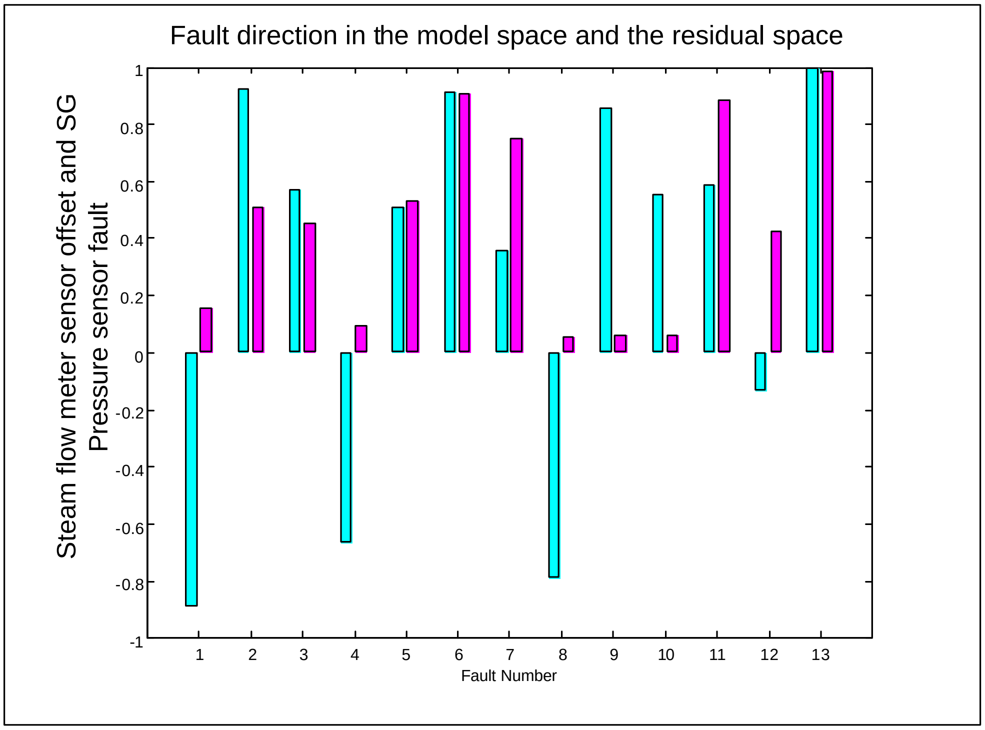


Figure 5.22. Fault direction for steam flow meter offset fault and SG pressure sensor offset fault.

6. ADAPTIVE NETWORK FUZZY INFERENCE SYSTEM (ANFIS) FOR FAULT DETECTION AND ISOLATION (FDI)

(neriFinalReport02-07)

A PCA model cannot take advantage of the available system knowledge. For this reason, sometimes it is very difficult to build an appropriate model with low model uncertainty. Some of these difficulties are as follows:

- When a large number of variables are involved, it is hard to make sure that all the measurements are well excited in order to obtain a model with reliable generalization capability.
- When noisy data are involved, their effects on how the constraint equations can be extracted from the noisy data are unknown to the analyst.
- When nonlinear behavior is involved, it is hard to have a tradeoff between choosing more principal components to have a better approximation and preclude the disturbance of noises.

In order to overcome these problems and keep the features of historical data based FDI approach, Adaptive-Network-based Fuzzy Inference System (ANFIS) is implemented to generate models for FDI. This method can take advantage of the available system knowledge and try to capture the most relevant relationships among measured variables to characterize a fault.

6.1. ANFIS Architecture

ANFIS is a fuzzy inference system implemented in the framework of artificial neural networks (Jang, 1990). It is able to combine the reasoning capability of fuzzy logic and the learning capability of neural network. It is efficient in building a model with only a few inputs and one output. A fuzzy inference system implements inference procedure using fuzzy rules. A fuzzy rule can be expressed linguistically as follows:

$$\text{If x is A then y is B} \tag{6.1}$$

A fuzzy rule is analytically an implication relation R between its antecedent and its consequent, which can be expressed as

$$R(x,y) = \int_{(x,y)} m(x,y)/(x,y) \tag{6.2}$$

where

$m(x,y)$ = the membership function of the antecedent and the consequent pair.

The implication relationship *R(x,y)* can be alternatively explained as the membership function of a fuzzy set defined in a two dimensional universe of discourse *(x,y)*. It can be computed using implication operator ϕ as follows:

$$R(x,y) = f(m_A(x), m_B(y)) \tag{6.3}$$

The most commonly used implication operators are the Larsen *product* and Mamdani *min* operators. If there are several input variables, it is necessary to have several antecedents connected with fuzzy operators. In general, a fuzzy inference system uses a set of fuzzy rules connected with connectives forming fuzzy algorithms.

Fuzzy inference of Generalized Modus Ponens (GMP) is stated as the following problem:

If x is A then y is B

$$x = A' \Rightarrow y = B' \tag{6.4}$$

In the above problem statement, the known part is *R(x,y) and A'*, the unknown part is *B'* associated with *A'*. This inference procedure is a fuzzy composition given by:

$$B' = A' \circ R(x,y) \tag{6.5}$$

The most commonly used fuzzy composition operators are Max-Min if Mamdani Min implication relationship is used and Max-Product if Larsen Product implication relationship is used.

A fuzzy inference system has the following four components (Jang, 1994):

- A rule base containing if-then rules.
- A database defining the membership functions used by the fuzzy rules.
- A decision-making unit performing inference operations on the rules.
- A unit to fuzzify the inputs and a unit to defuzzify the fuzzy outputs.

Five steps need to be taken in a fuzzy inference system as follows:

- Fuzzify the inputs.
- Apply fuzzy operator.
- Apply implication Method.
- Aggregate all the outputs.
- Defuzzify the output.

The first step is to fuzzify the crisp inputs. In this step, the membership values of all the input variables are computed by comparing their values and the membership functions. The second step is to apply fuzzy operators to compute the degree of the fulfillment (DOF) of the whole antecedent for each fuzzy rule by combining the membership values of all the fuzzy inputs. The result is the firing strength of its corresponding rule. In the third step, the membership of the consequent for each rule is computed based on the DOF of the antecedent for the corresponding rule by applying appropriate composition method. The fourth step is to aggregate the membership of the fuzzy outputs for all the rules. The final step is to defuzzify the output using methods such as centroid, maximum criterion, etc.

The simplest fuzzy inference model is of the Sugeno type. It has the following form of fuzzy rules:

If x is A and y is B then

$$z = f(x) \tag{6.6}$$

where

A and B are fuzzy sets.
z = f(x) is a crisp function.

In this model, the consequent of a fuzzy rule is simply a crisp function rather than a fuzzy set. It can significantly simplify fuzzy reasoning. In general, aggregation and defuzzification will involve matrix operation in high dimensional space. However, for a Sugeno fuzzy model, only a simple arithmetic function is involved in computing the output of each rule. The aggregation and defuzzification can be combined into a weighted sum (Hines and Wrest, 1997).

A Sugeno fuzzy model evolves into its first order form if the function f(x) defined in the fuzzy rules is of first order. Given that there are two inputs and one output, two fuzzy rules can be represented by

Rule 1: If x is A1 and y is B1 then z = ax+by+c

Rule 2: If x is A2 and y is B2 then z = px+qy+r

The output *f* can then be obtained as the sum of the two crisp output *f1* and *f2* resulting from the two rules weighted by the firing strength ratio *w1* and *w2*. That is,

$$f = \overline{w}_1 f_1 + \overline{w}_2 f_2 \tag{6.7}$$

When the fuzzy inference system is implemented using an adaptive network, the network system consists of layers of nodes capable of adapting parameters to map the desired input-output relation using fuzzy inference mechanism. For each node, there are several inputs and one output. The processing inside each node is nothing but performing some function computation. There are no weights designated to the connection between two nodes, but there is directional indication.

A classical ANFIS architecture is shown in Figure 6.1. There are two inputs (x and y) and one output (f) in the four network layers. The nodes in the first layer take crisp inputs and compute the degree of fulfillment of fuzzy sets (A_1, A2, B1, B2). These fuzzy sets are parameterized fuzzy set. Their membership functions can be adjusted easily by changing a set of parameters. The two nodes in the second layer correspond to two fuzzy rules. All the nodes in this layer take two inputs to give an output, w_1 or $w_{2,}$ representing the firing strength of each rule based on the product of the two membership values being involved. The third layer is responsible for calculating the relative importance of each rule ($\overline{w}_1$ and $\overline{w}_2$), the ratio of one rule's strength to the sum of the firing strengths of all the rules. Each node in the fourth layer contains a node function to calculate the consequent multiplied by the ratio calculated in the third layer. The output layer gives the final output by summing all its inputs.

6.2. ANFIS Learning Rule

The training process of a first order Sugeno type ANFIS system is to tune the parameterized functions involved in the system such that the desired input-output mapping is achieved. Hybrid learning algorithm is used for training, which combines the gradient descent method and the least square method (Hines and Wrest, 1997). In this training algorithm, all the tuned parameters are classified into two sets. One set of the parameters describes the linear relationship between the inputs and the outputs, which contains the parameters in the crisp function to describe the consequent of each rule. The other set of parameters describes the non-linearity between the inputs and the output, which involves those parameters defining membership functions. In the forward pass, the parameters describing the linear relationships are upgraded by sequential least square training. After the error is computed, the gradient descent training is used, which makes the error propagated from the output layer to the input layer. In this backward pass, the parameters describing the nonlinear relationship are upgraded. The training process does not end until the desired error goal is reached or the designated maximum number of epochs is exceeded.

For the ANFIS structure with two inputs and one output, the learning algorithm is shown in a later section. The system output may be expressed as follows:

$$f = \overline{w}_1(A1, B1) * (p_1 x + q_1 y + r_1) + \overline{w}_2(A2, B2) * (p_2 x + q_2 y + r_2)$$

The parameter space S can be partitioned into two subspaces S_1 and S_2 given by:

$$S = S_1 + S_2$$

where

$$S_1 \supset (A1, B1, A2, B2)$$

$$S_2 \supset (p_1, q_1, r_1, p_2, q_2, r_2)$$

During the forward pass with the fixed set S_1, if off-line learning algorithm is used, the parameters in the subspace S_2 can be determined by least squares estimate as follows:

$$S_2 = (X'X)^{-1} X'Y$$

where

X = input data set.
Y = target output.

If an on-line learning algorithm is used, the following sequential learning algorithm can be used (Jang, 1994).

$$S_{2,i+1} = S_{2,i} + Q_{i+1} X_{i+1} (Y_{i+1} - X^{'}_{i+1} S_{2,i})$$

$$Q_{i+1} = \frac{1}{l}(Q_i - \frac{Q_i X_{i+1} X^{'}_{i+!} Q_i}{1 + X^{'}_{i+!} Q_i X_{i+1}})$$

where

Q =covariance matrix.

X_i =a column vector of the input variables at the ith time.

Y_i = the target output at the ith time.

l =the forgetting factor between 0 and 1.

During the backward pass with the fixed set S_2, the parameters in the subspace S_1 can be determined by gradient descent method. For the output layer, the error rate is defined by

$$\frac{\partial E}{\partial O^5} = -2(T - O^5)$$

where

$$O^5 = \overline{w}_1(A1, B1) * (p_1 x + q_1 y + r_1) + \overline{w}_2(A2, B2) * (p_2 x + q_2 y + r_2)$$

T =target output.

For the two nodes in the fourth layer, the error rate is defined by

$$\frac{\partial E}{\partial O_1^4} = \frac{\partial E}{\partial O^5} \frac{\partial O^5}{\partial O_1^4}$$

$$\frac{\partial E}{\partial O_2^4} = \frac{\partial E}{\partial O^5} \frac{\partial O^5}{\partial O_2^4}$$

where

$O^5 = O_1^4 + O_2^4$ = the system output.

O_i^k = the output of the i-th node of the k-th layer.

For the two nodes in the third layer, the error rate is defined by

$$\frac{\partial E}{\partial O_1^3} = \frac{\partial E}{\partial O^5}(\frac{\partial O^5}{\partial O_1^4} \frac{\partial O_1^4}{O_1^3} + \frac{\partial O^5}{\partial O_2^4} \frac{\partial O_2^4}{O_1^3})$$

$$\frac{\partial E}{\partial O_2^3} = \frac{\partial E}{\partial O^5}(\frac{\partial O^5}{\partial O_1^4} \frac{\partial O_1^4}{O_2^3} + \frac{\partial O^5}{\partial O_2^4} \frac{\partial O_2^4}{O_2^3})$$

where

$$\begin{aligned} O_1^4 &= O_1^3 f_1 \\ O_2^4 &= O_2^3 f_2 \end{aligned}.$$

For the two nodes in the second layer, the error rate is defined by

$$\frac{\partial E}{\partial O_1^2} = \frac{\partial E}{\partial O^5}(\frac{\partial O^5}{\partial O_1^4}\frac{\partial O_1^4}{O_1^3} + \frac{\partial O^5}{\partial O_2^4}\frac{\partial O_2^4}{O_1^3})\frac{\partial O_1^3}{\partial O_1^2} + \frac{\partial E}{\partial O^5}(\frac{\partial O^5}{\partial O_1^4}\frac{\partial O_1^4}{O_2^3} + \frac{\partial O^5}{\partial O_2^4}\frac{\partial O_2^4}{O_2^3})\frac{\partial O_2^3}{\partial O_1^2}$$

$$\frac{\partial E}{\partial O_2^2} = \frac{\partial E}{\partial O^5}(\frac{\partial O^5}{\partial O_1^4}\frac{\partial O_1^4}{O_1^3} + \frac{\partial O^5}{\partial O_2^4}\frac{\partial O_2^4}{O_1^3})\frac{\partial O_1^3}{\partial O_2^2} + \frac{\partial E}{\partial O^5}(\frac{\partial O^5}{\partial O_1^4}\frac{\partial O_1^4}{O_2^3} + \frac{\partial O^5}{\partial O_2^4}\frac{\partial O_2^4}{O_2^3})\frac{\partial O_2^3}{\partial O_2^2}$$

where

$$\begin{aligned} O_1^3 &= \frac{O_1^2}{O_1^2 + O_2^2} \\ O_2^3 &= \frac{O_2^2}{O_1^2 + O_2^2} \end{aligned}.$$

For the j-th node in the first layer, the error rate is defined by

$$\frac{\partial E}{\partial O_j^1} = \frac{\partial E}{\partial O_1^2}\frac{\partial O_1^2}{\partial O_j^1} + \frac{\partial E}{\partial O_1^2}\frac{\partial O_2^2}{\partial O_j^1}$$

where

$$\begin{aligned} O_1^2 &= O_1^1 O_3^1 \\ O_2^2 &= O_2^1 O_4^1 \end{aligned}$$

The derivative of the output error with respect to the parameters used to define the membership functions can be determined by:

$$\frac{\partial E}{\partial S_{1,i}} = \sum_{j=1}^{M} \frac{\partial E}{\partial O_j^1}\frac{\partial O_j^1}{\partial S_{1,i}}$$

where

$S_{1,j}$ = the j-th parameter in the space of S_1.

M = the number of fuzzy sets used to fuzzy rules.

If Gaussian membership function is used as given by

$$O_j^1 = m_j(x) = \exp(-(\frac{x - c_j}{a_j})^2)$$

for the j-th fuzzy set used in the first layer, then

$$\frac{\partial E}{\partial c_j} = \frac{\partial E}{\partial O_j^1}\frac{\partial m_j(x)}{\partial c_j}$$

$$\frac{\partial E}{\partial a_j} = \frac{\partial E}{\partial O_j^1}\frac{\partial m_j(x)}{\partial a_j}$$

The update of the parameters in the space of S_1 is determined by

$$\Delta S_{1,j} = -h\frac{\partial E}{\partial S_{1,j}}$$

where

h = the learning rate.

If online learning is implemented, the update of the parameters is performed after each input-output pair is presented. The hybrid learning algorithm is much faster than gradient descent method only and gradient descent and one pass of least squares method (Jang, 1993). If some membership functions or some rule functions are determined from expert knowledge, the learning algorithms can be easily adapted to develop some hybrid models.

6.3. Structured Residual Design Approach to FDI

If structured residual design approach is used for fault isolation, the residual vector is represented by bit numbers for a set of models. The bit number 1 indicates that the model has a significant residual while the bit number 0 indicates the model has insignificant residual.

Table 6.1 shows the residual structure of four models for four faults. The residual pattern for fault 1 is [1,1,0,0], the residual pattern for fault 2 is [1,0,0,0], the residual pattern for fault 3 is [0,0,1,1], and the residual pattern for fault 4 is [0,0,1,0]. If the bit number of model 3 for fault 2 degenerates, fault 2 will be misdiagnosed as fault 1. If the bit number of model 4 for fault 3 degenerates, fault 4 will be misdiagnosed as fault 1. Therefore, the residual structure can only result in weak isolation between faults.

In order to achieve strong fault isolation, which means a fault will not be misdiagnosed as another fault even if one bit number has degenerated, it is necessary to transform the residual vectors into a structured form shown Table 6.2.

For a linear system, the structured residual can be achieved by a linear transformation on the original residuals. However, for a nonlinear system, it is quite difficult to derive new dependent equations by algebraic combinations of the previous equations in order to obtain the desired residual structure (Garcia, et al, 2000).

6.4. Applications to FDI of Nuclear Steam Generator System

Once the possible faults are enumerated based on engineering judgments structured residual design approach with ANFIS models can be implemented for the fault diagnosis for nuclear SG system. The study shows different residual structures are required to deal with single faults and dual faults.

6.4.1. Residual structure design for single faults

If there are only single faults in the designed FDI system, it is possible to achieve strong fault isolation if the model structures are carefully chosen. The most obvious way to obtain residuals is based on the natural redundancy in a process. From the engineering analysis of PWR SG system, the model structures derived from the physical analysis are as follows:

FCV flow rate (t) = f (FCV valve position (t), SG pressure (t))

FCV valve position (t+1) = f (controller output (t), FCV valve position (t))

SG pressure (t) = f(SG temperature(t))

Steam flow rate(t+1) = f(feed water temperature(t), SG pressure(t), hot leg temperature(t), cold leg temperature(t))

SG level (t+1) = f(SG level(t),Feed water flow rate(t), steam flow rate(t),SG pressure(t))

Table 6.1. Structured residual design for weak fault isolation

Model	Fault 1	Fault 2	Fault 3	Fault 4
Model 1	1	1	0	0
Model 2	1	0	0	0
Model 3	0	0	1	1
Model 4	0	0	1	0

Table 6.2. Structured residual design for strong fault isolation

Model	Fault 1	Fault 2	Fault 3	Fault 4
Model 1	1	1	0	1
Model 2	1	0	0	0
Model 3	0	0	1	1
Model 4	0	1	1	0

Table 6.3 shows the residual patterns for the 13 faults with the above model structure. In the table, the bit number 0 indicates that the model to predict the specific variable will not generate significant residual while the sign 1 indicates that the residual is significant. The threshold to distinguish the significance is determined by the model accuracy and the level of plant disturbance. As can be seen from the table, the residual patterns can be directly used to achieve strong fault isolation for the three single faults (feed water flow meter offset, steam flow meter offset and FCV position offset). However, the SG level sensor fault cannot even be detected. In this case, the residuals refer to the values after the new steady state has been reached. Due to the compensation effects of the SG level controller, the relationship among the feed water flow rate, the steam flow rate, the SG pressure, the FCV position and the SG level are always attempted not to change. Therefore, it is usually very difficult to detect a minor fault of the steam generator water level sensor fault based on a steady state model.

Table 6.3 also shows that some residuals become unstable due to fault competition with different fault magnitudes when dual faults are involved. The unstable residuals of the SG level model, denoted by the sign "?" in Table , correspond to the following combination of faults:

- Steam flow meter offset plus feed flow meter offset.
- SG pressure sensor offset plus SG level sensor offset.
- FCV valve position offset plus feed water flow meter offset.

Although the residual structure can be used directly to isolate the three single faults, it is not sufficient to isolate dual faults. These dual faults are as follows:

- Feed flow meter offset plus SG level sensor offset cannot be separated from feed flow meter offset.
- Steam flow meter offset plus SG level sensor offset cannot be separated from steam flow meter offset.
- SG pressure sensor fault plus Feed flow meter offset cannot be separated from steam SG pressure sensor fault.
- SG pressure sensor fault plus SG level sensor offset and SG pressure sensor fault plus SG steam flow meter offset.

In conclusion, although a set of models derived from physical analysis may result in different residual patterns for fault isolation, they are usually not effective to deal with dual faults. When used for dual fault isolation, some dual faults may result in the same residual pattern as their element faults. In addition, the residuals of some models may become unstable for dual faults with different fault magnitudes.

6.4.2. Dedicated residual design for dual faults

Because a dual fault usually cannot be strongly isolated from its element faults, dedicated residual structure is designed to isolate dual faults. Dedicated residual structure has the following two properties:

- Each residual is only sensitive to one fault and insensitive to all the other faults.
- Different faults will result in different types of residual patterns.

If all possible faults are known, dedicated residual structure can be obtained through appropriately selecting the model structures to generate residuals. The alternative models can be derived based on:

- Natural redundancy

  For instance, for a saturated system, the temperature and the pressure has one to one correspondence. Any model involving either variable can be substituted by the other variable.
- Derived redundancy

  For instance, if the flow rate is determined by the system pressure and the valve position for a system, any model involving the flow rate can be substituted by the system pressure and the valve position.
- Measurement redundancy

  The measurement redundancy is the most primitive one. For the SG system, the SG narrow range level measurement involved in any model can be substituted by the SG wide range level measurements.

In order to isolate the specified 13 faults for the nuclear SG system, the models with dedicated residual structure are defined as follows:

FCV flow rate (t)=f (FCV valve position (t), SG temperature (t))
FCV valve position (t+1)=f (controller output (t), FCV valve position (t))
SG pressure (t)=f(SG temperature(t))
Steam flow rate(t+1)=f(feed water temperature(t), SG temperature(t), hot leg temperature(t), cold leg temperature(t))
SG NR level (t)=f(SG WR level(t),Feed water temperature(t)).

In order to isolate dual faults involving the controlled variable from their element faults in a closed control loop, the measurement redundancy has to be used. For the nuclear SG level system, SG WR level sensor has to be used to isolate the SG NR level sensor fault from SG NR level sensor fault plus another fault in the control loop.

6.4.3. ANFIS modeling for SG system

The ANFIS has been used to construct the five models for the system during normal operation. Before the ANFIS models are constructed, the input variables need to be appropriately scaled. The purpose to scale the inputs is to give equal importance to all the inputs in case that the input variables are in different units.

FCV valve flow rate model is shown as an example to train ANFIS. The network uses two bell-shape membership functions for either input. Two rules have been selected to map the input and output relationship.

Figure 6.2 shows the membership functions for the ANFIS model to predict the FCV valve flow rate before and after training. It can be seen that the ANFIS training has changed the shape of the membership function for the first input (FCV valve position) significantly. In general, this change reflects the degree of nonlinearly contained in the mapping between the input and the output. After three epochs, the ANFIS model has been trained to reach a training error less than 0.5 %.

Table 6.3. Consistency checking using natural redundant relations

Functional Model Faults	FCV flow rate	Steam flow rate	SG pressure	FCV valve position	SG level
Feed flow meter offset	1	0	0	0	1
Steam flow meter offset	0	1	0	0	1
Steam flow meter and feed flow meter offset	1	0	0	0	?
SG NR level sensor offset	0	0	0	0	0
Feed flow meter offset and SG level sensor offset	1	1	0	0	1
Steam flow meter offset and SG level sensor offset	0	1	0	0	1
SG pressure sensor offset	1	1	1	0	1
Feed flow meter offset and SG pressure sensor offset	1	1	1	0	1
Steam flow meter offset and SG pressure sensor offset	1	1	1	0	1
SG level sensor offset and SG pressure sensor offset	1	1	1	0	?
Feed water flow meter offset and FCV position offset	1	0	0	1	?
Steam flow meter offset and FCV position offset	0	1	0	1	1
FCV position offset	0	0	0	1	1

6.4.4. Model testing and validation

The residuals generated by some models can be immediately used for fault detection. If the sum square residuals of all the models are greater than a specified threshold, a fault is assumed to have happened.

In order to reduce the false alarm rates for fault detection, these models must be able to correctly characterize the system behavior under all the fault free conditions. However, if the models are fully static, any changes in the plant status or even plant disturbance will cause false alarms because the dynamic behavior of the system is unknown to the system. For this reason, most FDI systems need to use dynamic models. The dynamic models are able to simulate the normal transients such as a normal power transient.

In order to test the performance in characterizing the dynamic behavior, a power transient from 100% power to 90% power is simulated using the ANFIS models built in the last section. Figures 6.3-6.6 show the comparison between the estimation from the ANFIS models and the actual values obtained by the SimPWR simulation code for the following variables:

- SG narrow range level
- Steam flow rate
- FCV flow rate
- FCV valve position.

From these figures, it can be seen that the ANFIS models can correctly simulate the transient process with low errors. It is also very interesting to examine the sources of the error. One source can be attributed to the modeling error itself. When the models are trained, the training error cannot be absolute zero. The reasons might be:

- There are noises contained in the data.
- Some other contributors to the output are not included in the designed models.
- The training error is specified not to be very low in order to avoid over-training.

Another source is that some individual variables may be outside their training range, which will cause uncertainty in prediction. When a large complex system with strong interaction is involved, it is usually very challenging to build perfect data driven models. For instance, it would be very difficult to build a data driven model for a fast transient due to the fast interaction among systems. For fast transients, it will involve much more complicated model structure and it is harder to collect data to sufficiently excite all the related subsystems.

However, from FDI point of view, slight error of these models will not impose a serious problem. First, different thresholds can be set to the residuals for different models depending on the accuracy of the models. Secondly, fast transient is not of major interest for an incipient fault detection and isolation system. A fast power transient is usually under cautious supervision of operators, so operators can easily switch off the FDI system if the expected transient is any faster than the designed level.

In general, it is expected that these ANFIS models will give correct estimation if the relationship between the input variables and the output variable is not violated. However, if a fault happens, some input variables may be outside their training range. In this situation, the model may perform unreliable extrapolation. Hence, even if the relationship among the variables is not violated, the residual of the model may still exceed the specified threshold. In order to avoid this problem, it is necessary to evaluate whether the models are excited in all fault cases.

Here is an example to show the importance of model testing for SG NR level model. If the data are collected only from 20 % to 100% power ranges, the model will generate unstable residual patterns for different fault magnitudes. Figure 6.7 shows that when the SG steam flow meter and the SG pressure sensor have less than 2% offset faults, the residual is less than 0.5%. However, when the fault magnitude is 3%, the residuals become unstable. To investigate the causes, the training range of the inputs is examined. The minimum values of the feed water temperature and the SG WR SG level are 313.94 *F* and 76.269% respectively. The maximum values of the feed water temperature and the SG WR SG level are 438.4 *F* and 85.599% respectively. However, for the SG steam flow meter and SG pressure sensor offset faults with 3% fault magnitude, the minimum values of the feed water temperature and the SG WR SG level are *440.42 F* and 83.0% respectively and the maximal values are 440.5 *F* and 86.167% respectively. Apparently, the fault data have exceeded the training range, so the ANFIS model is not able to correctly compute the residual of the SG narrow range level.

After more data is collected to cover the range the faults may result in, the residuals exhibit consistent behavior. Figure 6.8 shows that the residuals of SG NR level models are within 1% for the dual faults (the SG steam flow meter and the SG pressure sensor offset fault) when the SG NR level sensors are healthy.

6.4.5. FDI results

Table 6.4 shows the dedicated residual structure based on the variables within the control loop used to isolate the defined 13 single and dual faults. As can be seen, each ANFIS model is dedicated to isolate one fault. For dual faults, the corresponding two models dedicated to the two element faults will generate significant residuals, which provides the full possibility to isolate them.

Figures 6.9-6.15 are plotted to show the residual structures for different fault magnitudes. In these plots, the 13 fault classes correspond to the following faults:

- Fault class 1= Feed water flow meter offset fault
- Fault class 2= Steam flow meter offset fault
- Fault class 3= Feed water flow meter offset fault and steam flow meter offset fault
- Fault class 4= SG level sensor offset fault
- Fault class 5= Feed water flow meter offset fault and SG level sensor offset fault
- Fault class 6= Steam flow meter offset fault and SG level sensor offset fault
- Fault class 7= SG pressure sensor offset fault

- Fault class 8= Feed water flow meter offset fault and SG pressure sensor offset fault
- Fault class 9= Steam flow meter offset fault and SG pressure sensor offset fault
- Fault class 10= SG pressure sensor offset fault and SG level sensor offset fault
- Fault class 11= Feed water flow meter offset fault and FCV offset sensor offset fault
- Fault class 12= Steam flow meter offset fault and FCV position offset fault
- Fault class 13= FCV valve position offset fault.

In each figure, the 13 faults have different residual patterns, so they can be isolated. If the residual patterns are compared for different fault magnitudes, their structures are stable. Moreover, the residuals are approximately equal to the fault magnitudes of the sensor faults such as the feed water flow meter fault and the steam flow meter fault. Theoretically, the residuals should be exactly as same as the fault magnitudes. However, due to the modeling errors of these data driven models, some slight differences still exist and these slight differences in FDI are acceptable. The faults occurring at 80% initial power level other than 100% full power are also tested. Figure 6.15 clearly shows that the performance of the FDI system does not degrade. The residual structures keep the same pattern as those faults at 100% power level.

6.5. Discussion

This chapter has presented ANFIS model based approach to fault detection and isolation and the application to PWR steam generator system. ANFIS model based approach combined with structured residual design is shown to be efficient in fault detection and isolation if the possible faults are enumerable. For single faults, strong isolation scheme can be achieved through appropriate choice of the model structures. For dual faults, it is not possible to achieve strong isolation between the dual faults and one of the element faults. Using natural redundancy and derived redundancy, dedicated residual structure can be achieved to isolate dual faults. In order to detect and isolate a fault related to control variable, sometimes it is necessary to use the information about measurement redundancy itself.

ANFIS model based approach combined with structured residual design does not need to define the fault signatures from fault data if the sensitivity of the developed ANFIS modes to the model inputs is known. Therefore, it is in conformance with the principle of modern fault detection and isolation methods. Since ANFIS is able to learn the relationship between variables from data, it has the power of on-line implementation.

However, ANFIS model based approach still needs to enumerate the possible faults. It still exert heavy burden on engineering application. In addition, for a non-linear complicated system, structured residual design for fault isolation, especially when data driven modeling is used, is essentially a process of trial and error. This exerts additional difficulties in engineering application.

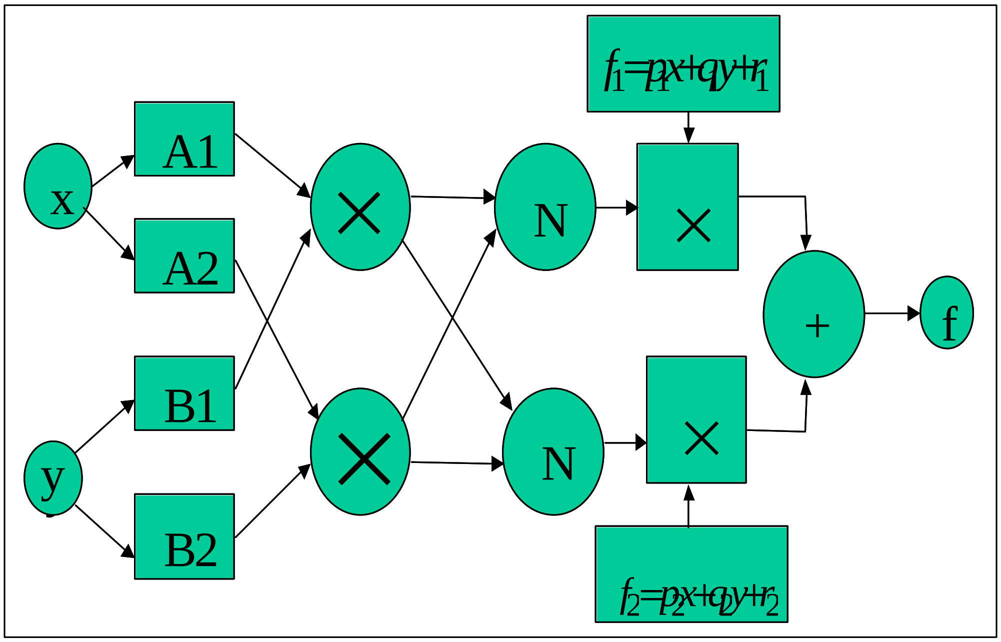


Figure 6.1. ANFIS structure.

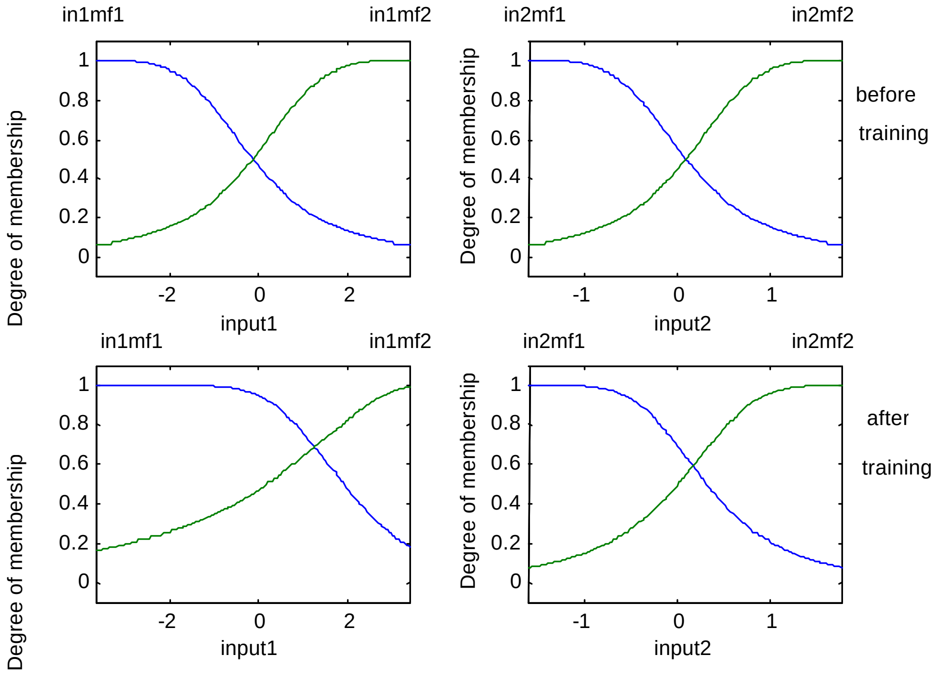


Figure 6.2. Membership for the two inputs to predict SG pressure using ANFIS.

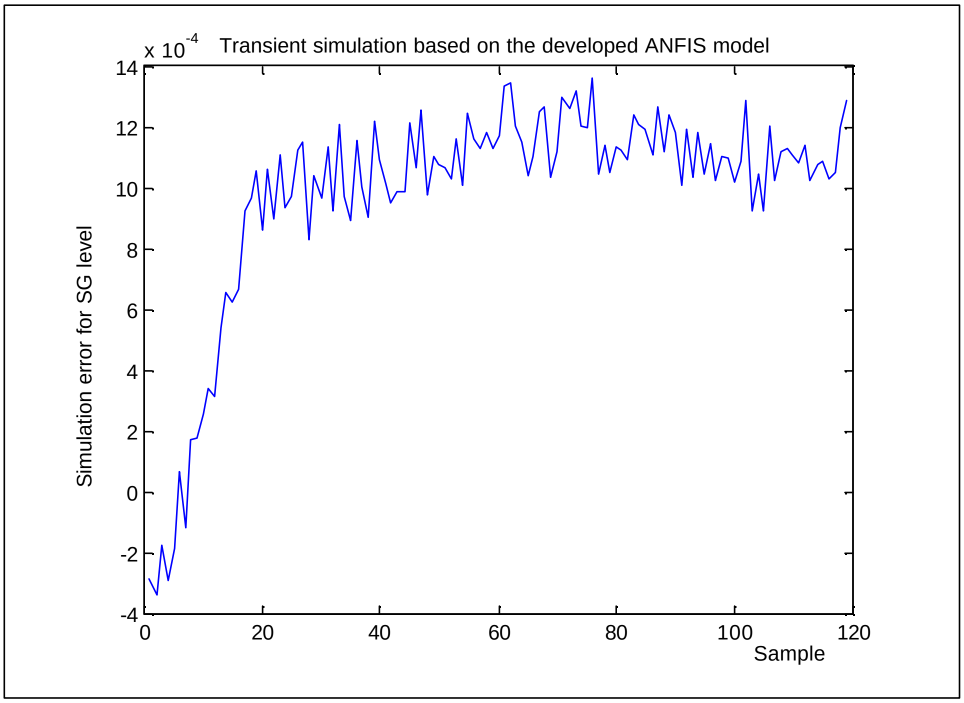


Figure 6.3. Transient simulation of SG NR level using ANFIS model.

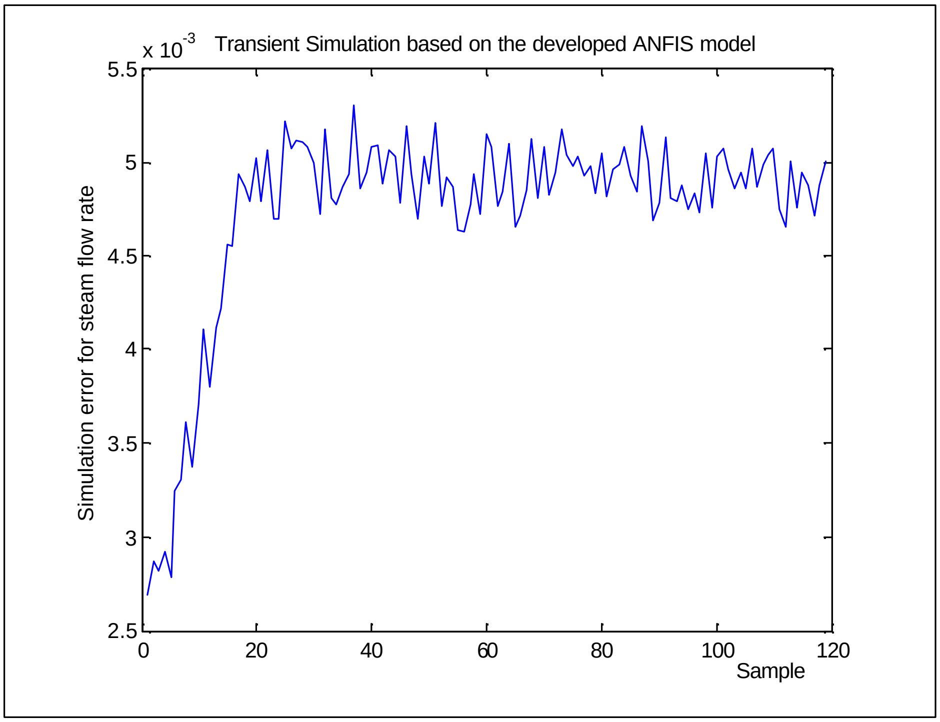


Figure 6.4. Transient simulation of steam flow rate using ANFIS model.

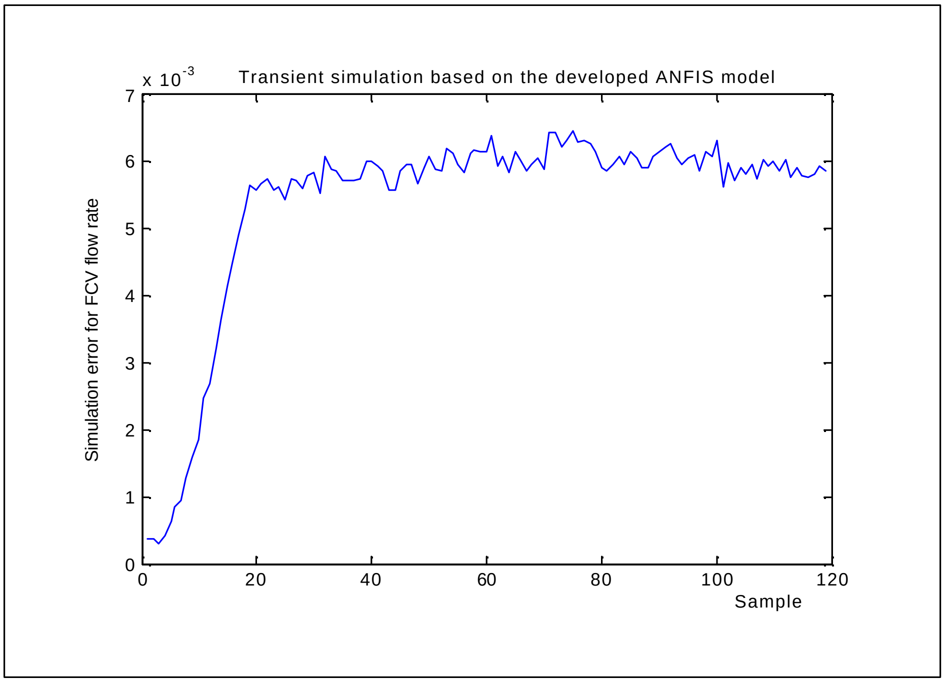


Figure 6.5. Transient simulation of FCV flow rate using ANFIS model.

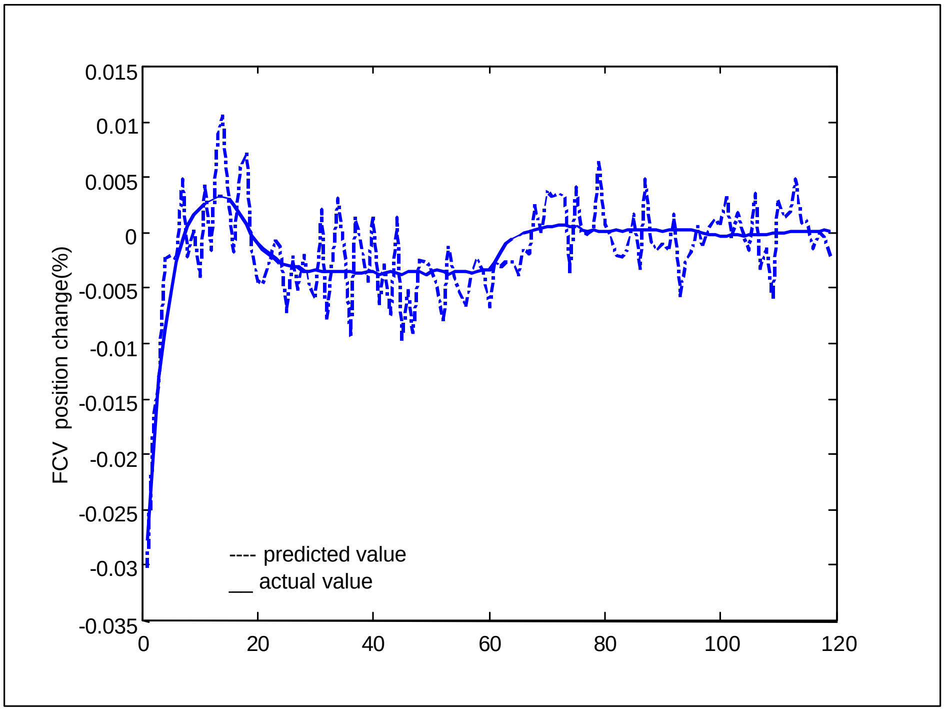


Figure 6.6. Transient simulation of FCV position using ANFIS model.

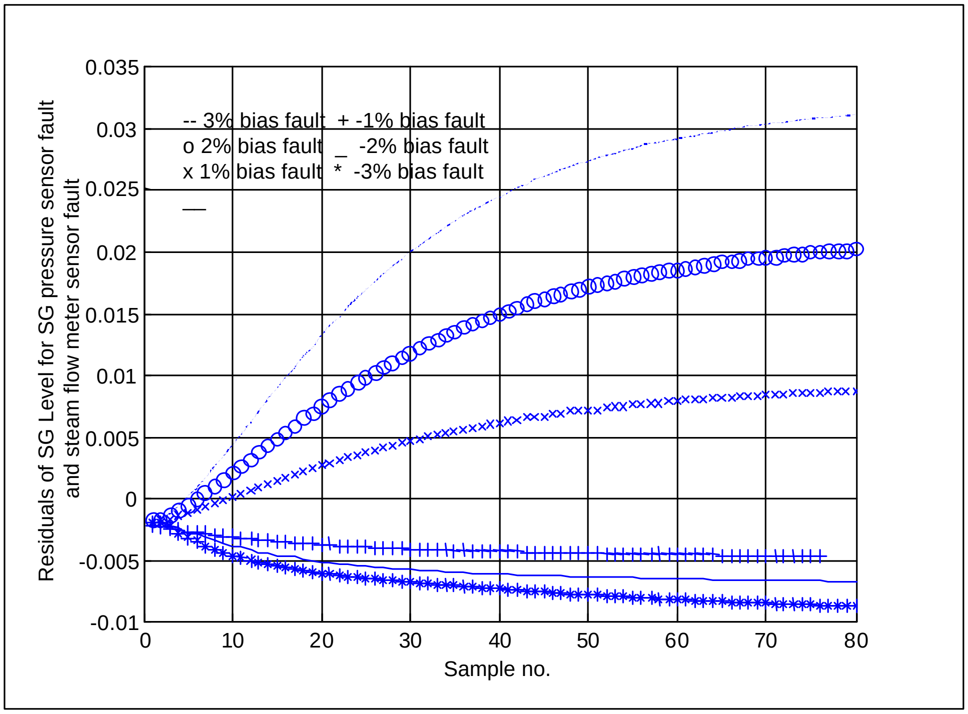


Figure 6.7. A not well-excited model generates unstable residual.

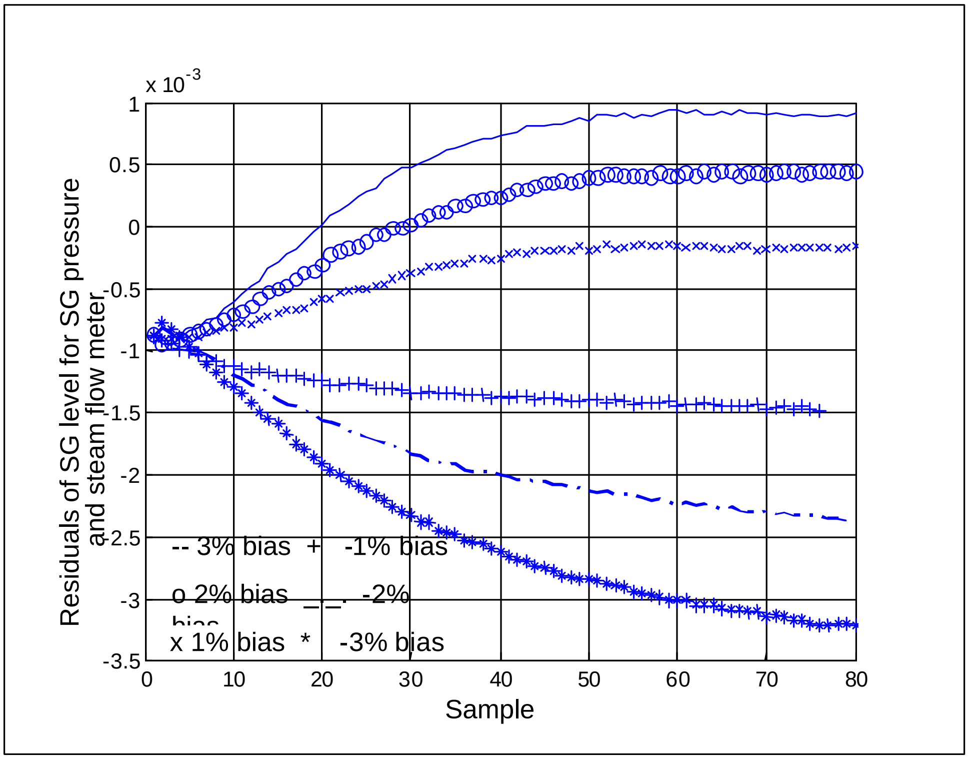


Figure 6.8. Stable residuals are obtained from well-excited model.

Table 6.4. Dedicated residual structure for SG system

Functional Model Faults	Feed water flow rate	Steam flow rate	SG pressure	FCV valve position	SG level
Feed flow meter offset	1	0	0	0	0
Steam flow meter offset	0	1	0	0	0
Steam flow meter and feed flow meter offset	1	1	0	0	0
SG NR level sensor offset	0	0	0	0	1
Feed flow meter offset and SG level sensor offset	1	0	0	0	1
Steam flow meter offset and SG level sensor offset	0	1	0	0	1
SG pressure sensor offset	0	0	1	0	0
Feed flow meter offset and SG pressure sensor offset	1	0	1	0	0
Steam flow meter offset and SG pressure sensor offset	0	1	1	0	0
SG level sensor offset and SG pressure sensor offset	0	0	1	0	1
Feed water flow meter offset and FCV position offset	1	0	0	1	0
Steam flow meter offset and FCV position offset	0	1	0	1	0
FCV position offset	0	0	0	1	0

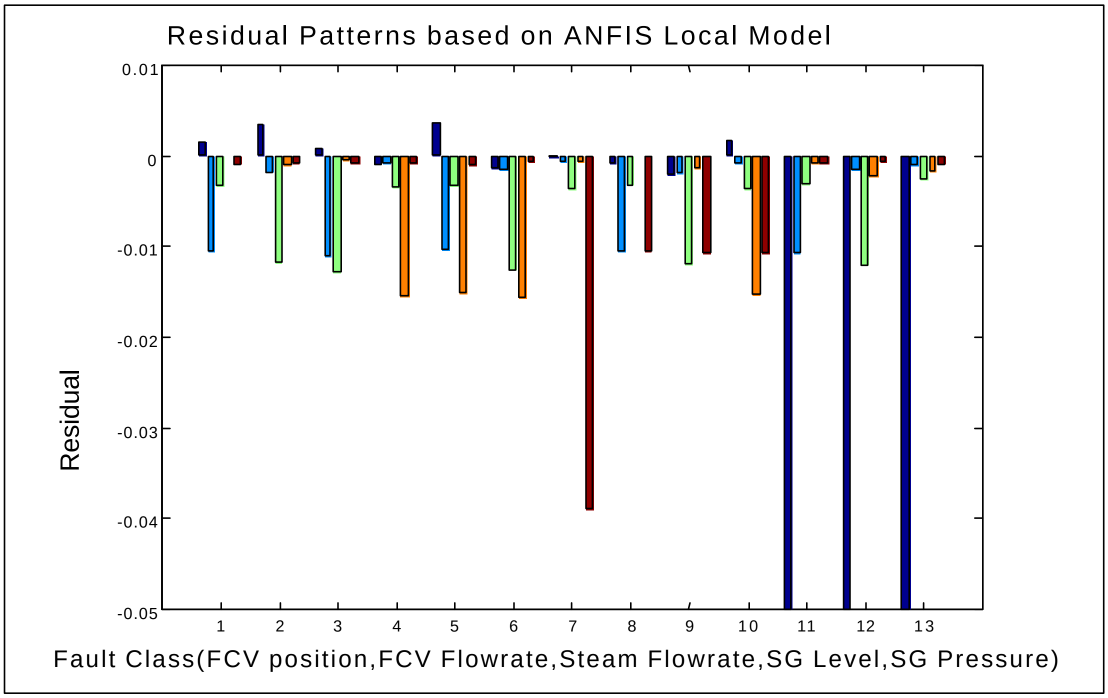


Figure 6.9. Structured residual pattern using ANFIS models

(100% Power, 1% offset fault**).**

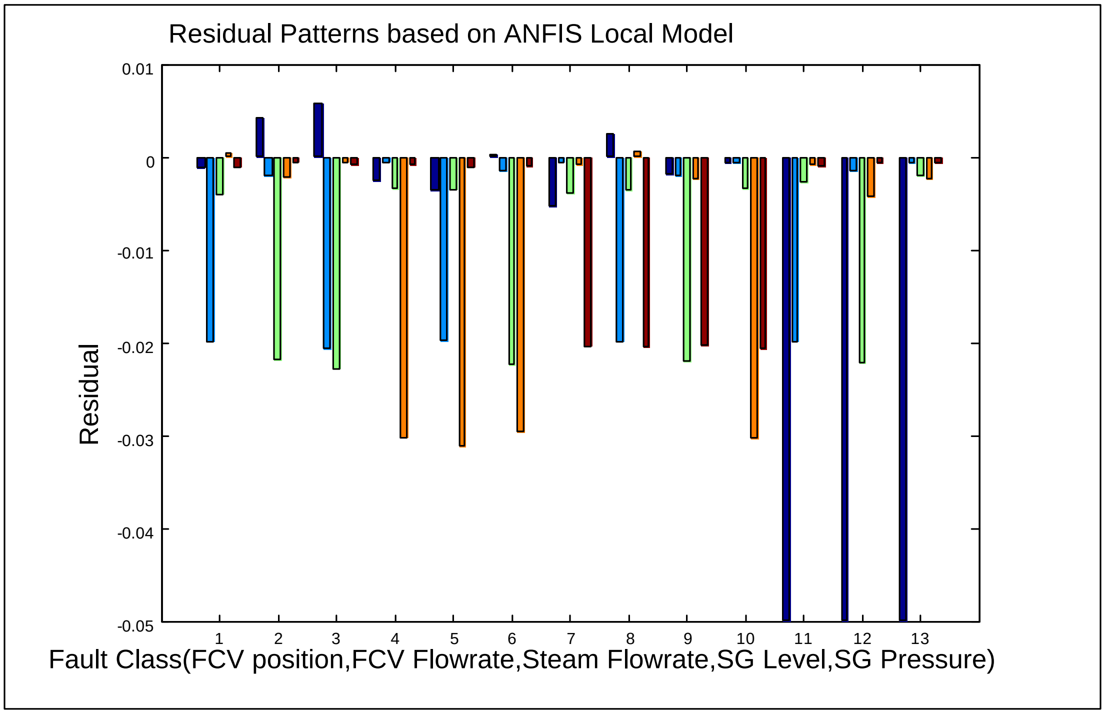


Figure 6.10. Structured residual pattern using ANFIS models

(100% Power, 2% offset fault).

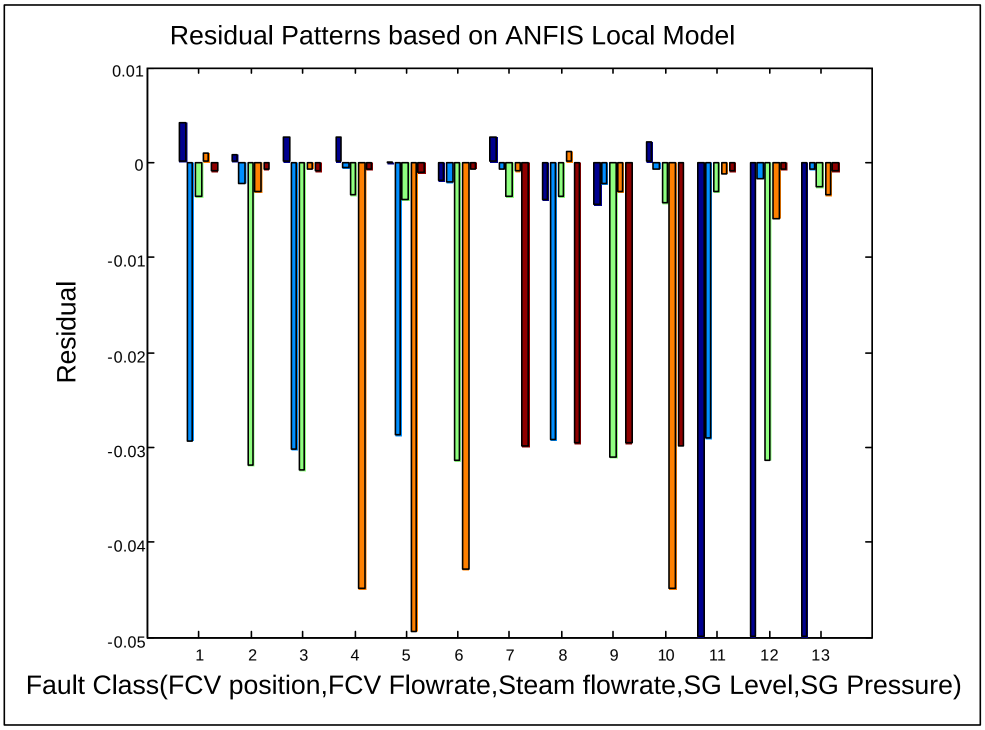


Figure 6.11. Structured residual pattern using ANFIS models (100% Power, 3% offset fault).

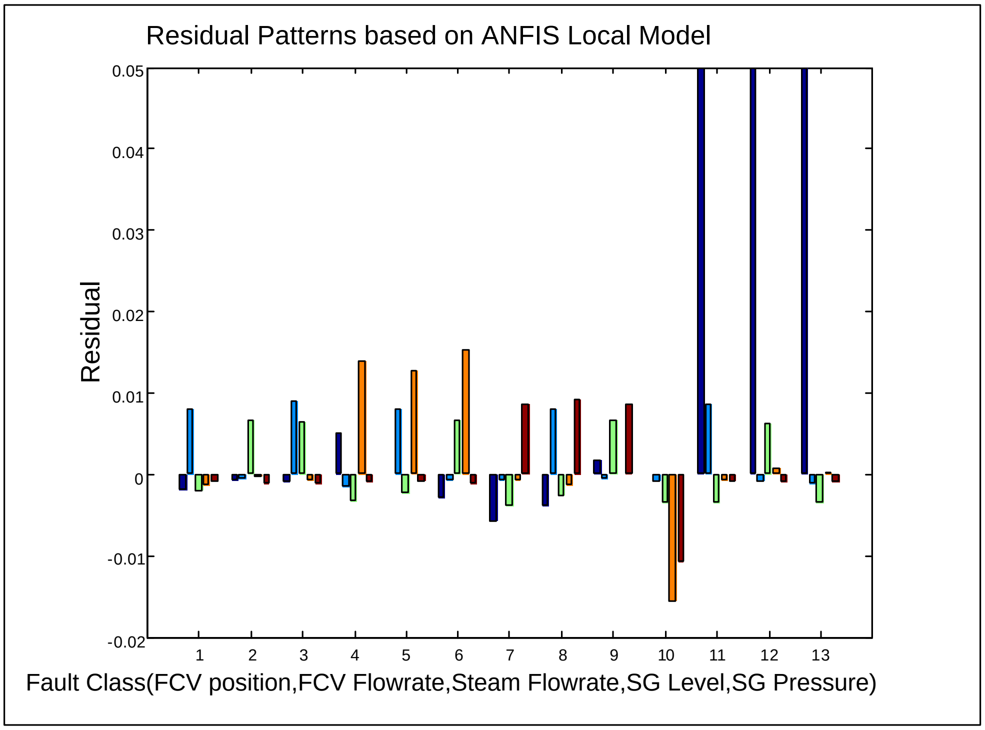


Figure 6.12. Structured residual pattern using ANFIS models. (100% Power, -1% offset fault)

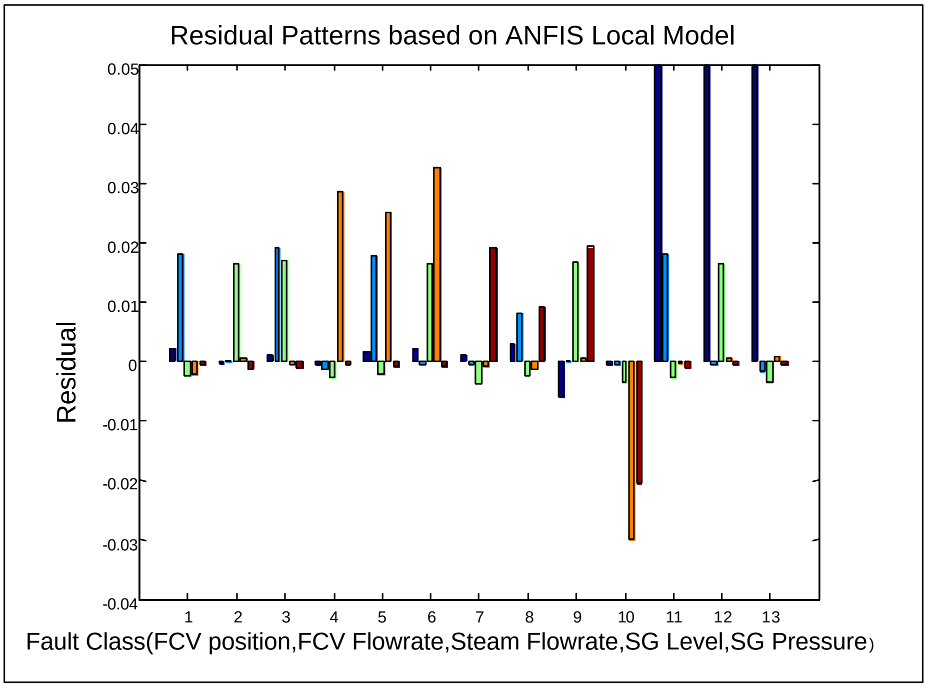


Figure 6.13. Structured residual pattern using ANFIS models (100% Power, -2% offset fault).

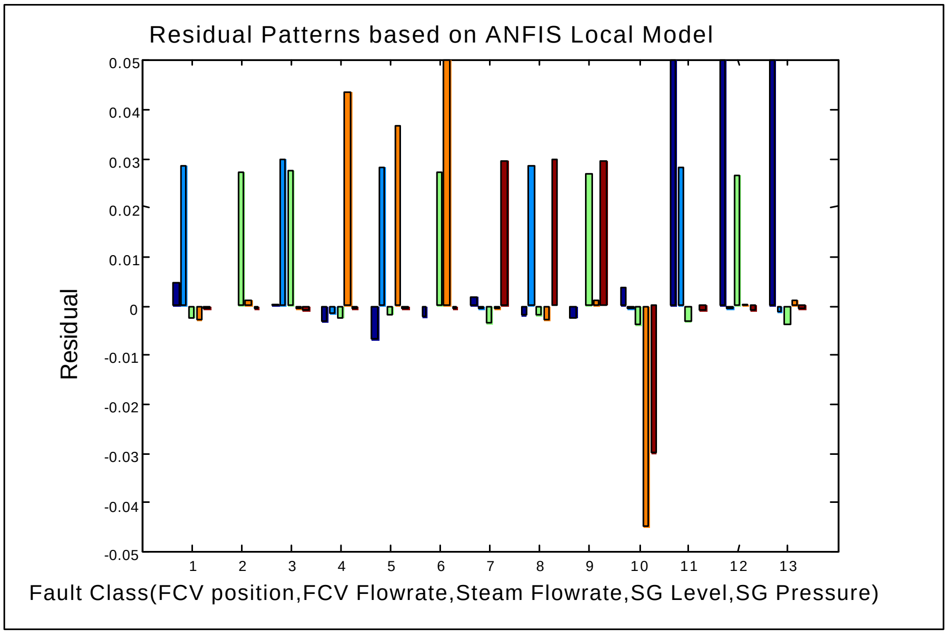


Figure 6.14. Structured residual pattern using ANFIS models (100% Power, -3% offset fault).

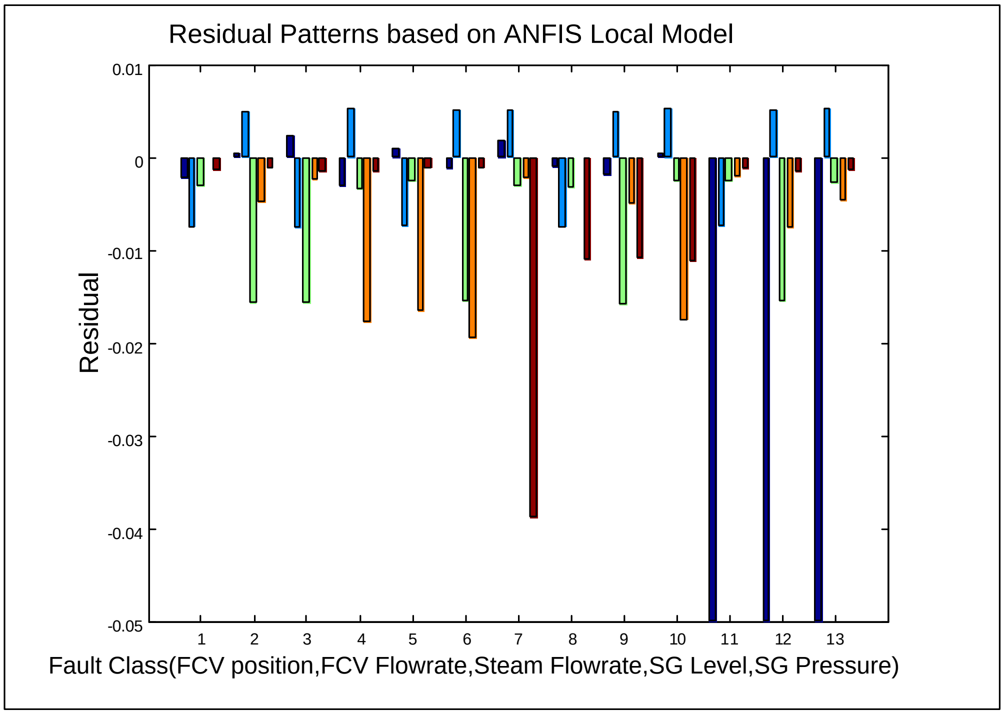


Figure 6.15. Structured residual pattern using ANFIS models
(80% Power, 1% offset fault).

7. DATA DRIVEN MODEL DIRECTIONAL GRAPH (DIGRAPH) APPROACH FOR FAULT DETECTION AND ISOLATION (FDI)

(neriFinalReport02-08)

7.1. Introduction

In Section 6, the dedicated residual structure is achieved based on the assumption that the possible faults are enumerable. With the faults known, the designers may achieve fault isolation through appropriate choice of the models. However, for a large complex system, it will be extremely challenging to enumerate all the possible faults. To avoid the enumeration of possible faults and predefine their associated fault signatures has thus motivated the development of data driven model digraph approach.

Data driven model digraph is proposed as a generic approach to fault diagnosis for fault diagnosis. It is able to combine the reasoning capability of qualitative knowledge based method and the strength in resolution of quantitative knowledge based method. To facilitate on-line implementation, adaptive neural fuzzy inference is used for modeling. Fault detection is fulfilled by monitoring the residual of each model. Fault isolation is achieved by cause effect analysis on the residuals generated from models.

7.2. Cause Effect Reasoning Using Model Digraph

Cause effect reasoning was originally introduced as a reasoning tool to account for the propagation of fault symptoms within a system (Davis, 1983). It has been extended to quantitative model based FDI when mathematical models are available (Montain and Gentil, 2000).

A model digraph consists of individual nodes connected by quantitative models. The individual nodes represent plant parameters, state variables and measurement variables. The quantitative models represent the cause-effect relationship between the nodes. As compared with sign directed graphs using qualitative knowledge only to describe the relationship between variables, a quantitative model is formally introduced to express the cause effect relationships. The model digraph is not a simple network of structural models. It includes the dynamic information about process flow-path, signal flow-path, and control logic so that a fault can be localized based on the cause effect analysis for a process system.

A complex physical system can be represented by the following set of differential equations:

$$\dot{X}_i = g_i(G_i, X_i, U_i) \tag{7.1}$$

where

$\boldsymbol{i = 1,2,.......N}$

$\boldsymbol{X_i}$ = system variables

$\boldsymbol{g_i}$ = a function to estimate $\boldsymbol{X_i}$

$G_i = \{X_j \mid j \neq i\}$

$\boldsymbol{G_i}$ = a set of variables being the inputs of $\boldsymbol{g_i}$. These variables may be measured values or unmeasured values.

U_i =process inputs.

$\boldsymbol{N}$ = number of models to characterize the system.

The moving average form of the above differential equation can be used to arrive at the digraph models, that is:

$$X_i = f_i(G_i, U_i) \tag{7.2}$$

If the model digraph is developed based upon the original process flow and signal flow, the causal relationship between variables will be implicit in it.

The cause effect relationship between the inputs and the outputs of a model has two connotations (Leyal, Gentil, and Stephan, 1994). From physics point of view, the cause-effect relation represents the pathway of the signal propagation. Any changes in the model inputs are always before any changes in the model outputs in the time domain. From computational point of view, the cause-effect relation means that any changes in the model inputs will sufficiently cause some changes in the model outputs and the model outputs will not change without any changes in the model inputs.

Figure 7.1 shows a simple example of a model digraph. In the figure, four models $\boldsymbol{g_1, g_2, g_3, g_4}$ are shown to characterize the system. The four variables X1, X2, X3, X4, X5, and X6 are process variables.

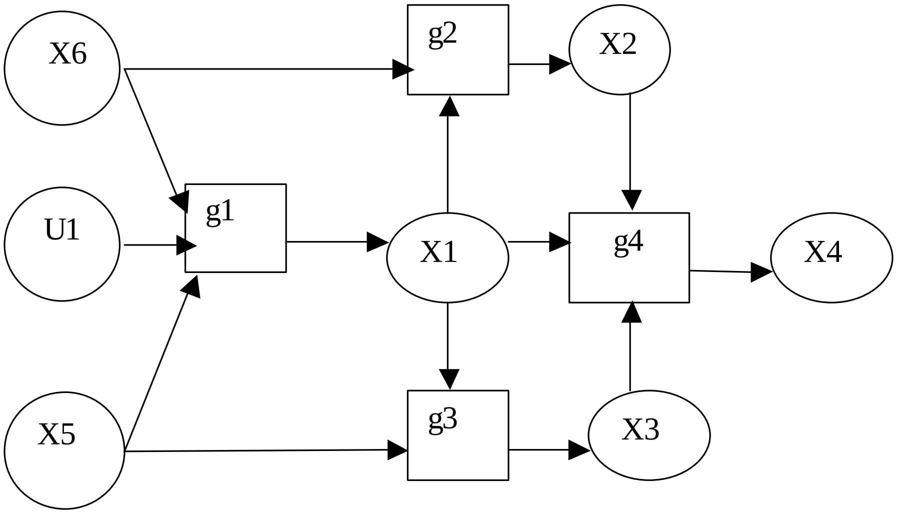


Figure 7.1. A simple example of a dynamic model digraph.

Causal effect reasoning can be easily performed based on analyzing the residuals of the individual models. The original residuals are calculated for each measured variable as follows:

$$R_i = X_i - X_i^* \tag{7.3}$$

where

$\boldsymbol{R_i}$ = the residual of the variable $\boldsymbol{X_i}$

X_i = the measured value of the variable $\boldsymbol{X_i}$

$\boldsymbol{X_i^*}$ = the estimated value of the variable $\boldsymbol{X_i}$ from the model $\boldsymbol{f_i}$ defined previously.

If $\boldsymbol{R_i}$ is significant, it can be determined that a fault has occurred to the system. However, there are still two possibilities that may explain the abnormal residual:

a) A local fault affecting $\boldsymbol{X_i}$

b) A consequence of a fault affecting the inputs of the model $\boldsymbol{g_i}$.

To facilitate fault isolation, a set of reconstructed residuals are calculated as follows:

$$\tilde{R}_i^j = X_i - \tilde{X}_i^j \tag{7.4}$$

where

$\boldsymbol{\tilde{R}_i^j}$ = the residual of $\boldsymbol{X_i}$ after the input $\boldsymbol{X_j}$ of the model $\boldsymbol{g_i}$ has been reconstructed

$\boldsymbol{\tilde{X}_i^j}$ = the estimated value of $\boldsymbol{X_i}$ after the input $\boldsymbol{X_j}$ of the model $\boldsymbol{g_i}$ has been reconstructed.

Fault isolation can then be based on the following decision procedure:

a) If $\boldsymbol{\tilde{R}_i^j} << \boldsymbol{R_i}$, $\boldsymbol{X_j}$ has a local fault.

b) If $\boldsymbol{\tilde{R}_i^j}$ » $\boldsymbol{R_i}$ for all $\boldsymbol{X_j}$ that will affect X_i, $\boldsymbol{X_i}$ has a fault.

c) If $\tilde{R}_i^{j_1} \approx R_i$ and $\tilde{R}_i^{j_2} \approx R_i$ but $\tilde{R}_i^{j_1 j_2} << R_i$, then X_{j_1} and X_{j_2} have faults.

As an example, the above model digraph method is used for a typical feed back control loop as shown in Figure 7.2. Four nodes connected by three models are used to represent the control loop. Set point is an input node of the controller model. The other three nodes are the controller output, the control variable and the regulated variable. The three models are the controller model, the actuator model and the plant model. Since a controller always takes the measured value of the regulated variable as input, the controller model can always be used to isolate a controller fault. For the same reason, the actuator model can be used to isolate an actuator fault. However, fault detection and isolation becomes a challenging task when a fault related to the regulated variable is involved. When a new steady state is reached after the fault, the regulated variable will be brought back to its original level. Hence, the steady state information is not enough to detect such a fault.

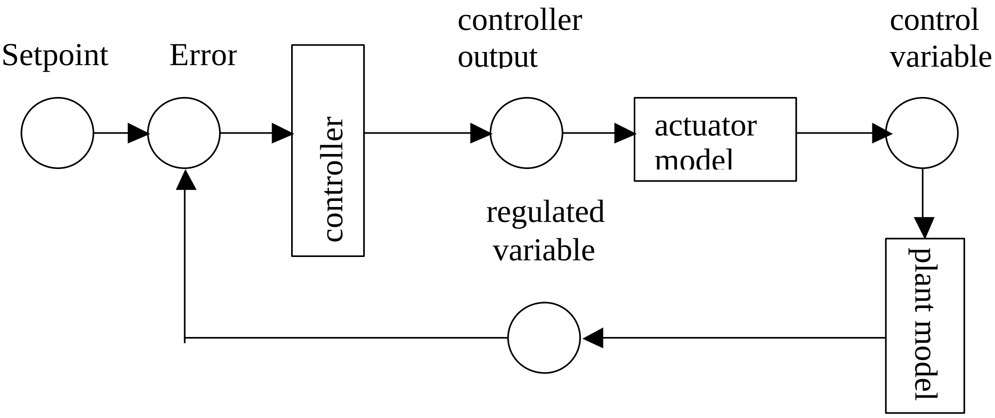


Figure 7.2. Dynamic model digraph representation of a feedback control loop.

The Model digraph method needs to use dynamic models. If a steady state model is used, the developed method can give correct FDI results after a new steady state has been reached. During the fault transient, the steady state models will result in serious false alarms. However, the fault symptoms may become weak after the new steady state due to controller feedback. In addition, dynamic models must be used to isolate a controller fault and some actuator faults such as control valves. By the way, in order to achieve a faster fault detection and isolation for safety concern, dynamic models are also desired.

7.3. Extended Model Digraph

7.3.1. Multi-model digraph

Multiple-model (or multi-observer) digraph is introduced to isolate process input faults. If regular model digraph is used, the cause effect analysis on the model residuals cannot be performed since there are no additional models available to reconstruct these process inputs.

Multi-model approach was proposed for fault isolation (Simani, 2000). The basic idea is to make most use of the knowledge about the process redundancy inherent in a system. For example, in a saturate SG system, there is a one-to-one relation between the SG pressure and the SG temperature. Therefore, any model as a function of SG pressure can always be used to derive a new model as a function of SG temperature. The cause effect analysis on the residuals of these two models can then be performed to isolate the two faults.

Figure 6-3 and figure 6-4 show two types of multiple models designed to isolate output faults and input faults, respectively. In the design scheme shown in figure 6-3, one output and all the inputs drive each model. An output measurement fault affects only the residual of the model driven by this output variable. Therefore, the output faults can then be isolated if there is not input fault. In the design scheme shown in figure 6-4, each model is driven by all but one input and all the outputs, which generates a residual sensitive to all but one input fault. Therefore, the input faults can then be isolated if there is not output fault.

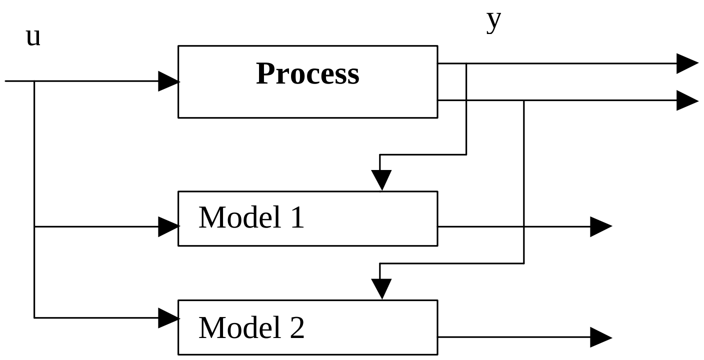


Figure 0-1 Multiple models to isolate output faults

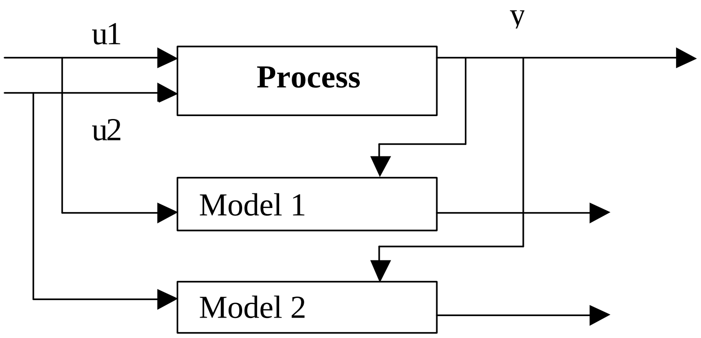


Figure 0-2 Multiple models to isolate output faults

Figure 7.3 shows an example on how to combine multi-model approach and regular model digraph to obtain a multi-model digraph. If X_1 and X_2 cannot be reconstructed from some other models, the multi-model digraph still enables to perform cause effect analysis on the model residuals. After the subsequent models have confirmed there are no faults related to variable Y_1 and Y_2, the following decision logic can be performed:

a) If $R_1 \approx R_2 \approx R \approx 0$, there is no fault with respect to X_1, X_2 and Y_1 and Y_2

b) If both R_1 and R_2 are significant, there is a fault with respect to X_n.

c) If $R_1 \approx 0$ but $R_2 \neq 0$, there is a fault with respect to X_2.

d) If $R_2 \approx 0$ but $R_1 \neq 0$, there is a fault with respect to X_1.

7.3.2. Model digraph with hidden nodes

Model digraph can also be extended to include unmeasured variables. This is even necessary to detect and isolate a process faults. Figure 7.4 shows an example. In the figure, X1, X2, X3, and X4 correspond to four measured variables and H1 corresponds to an unmeasured variable. In this case, the same reasoning logic can be used except that H1 cannot be used as an independent residual generator. It is necessary to prepare an explicit model instead of a data driven model to estimate the value of a hidden node. If a data driven model is to be used, some special learning algorithm must be developed.

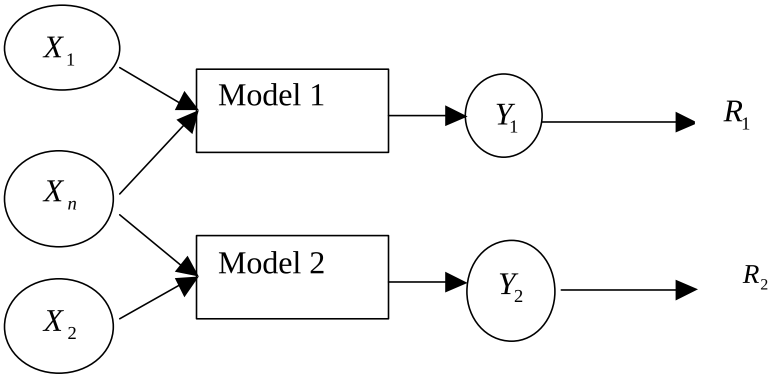


Figure 7.3. An example of multiple-model digraph.

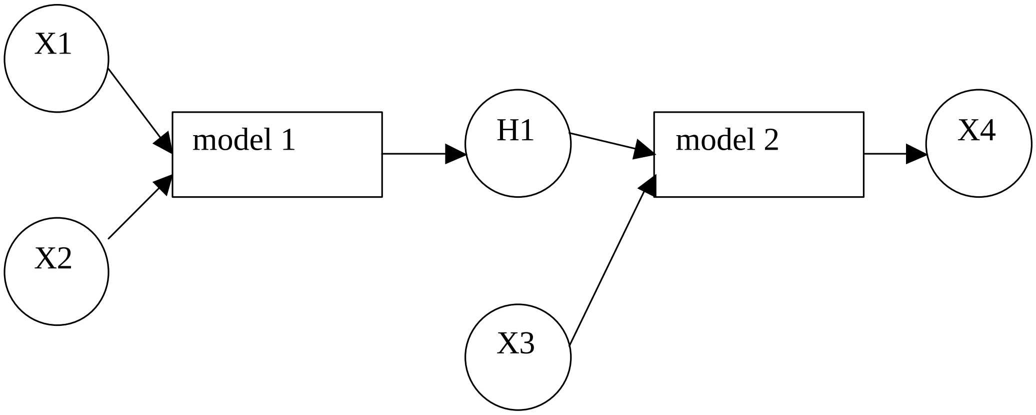


Figure 7.4. Model Digraph with hidden nodes.

7.4. Model Digraph Approach with Fuzzy Inference Modeling

Theoretically, an ANFIS model is able to approximate a system to any desired degree of accuracy. However, in real situation, it is not always able to achieve this accuracy because a too complex model may be required. Even if such a model can be obtained, the desired capability of generalization can still not be guaranteed.

In order to achieve a perfect model, the input variables must always be cautiously selected. On the one hand, the colinearity between the input variables should be avoided since the least square method is used in training ANFIS. On the other hand, the dimensionality of the inputs for the ANFIS models should be as few as possible. In ANFIS, each input variable needs to be fuzzified into problem specific membership functions. When the number of input variables is increased, the number of nodes in the second layer and the third layer of the ANFIS network will be increased exponentially. Correspondingly, the number of rules used in the system will be increased too. This increase will not only have a severe influence on the training speed but also on the stability of the built models because the number of degrees of freedoms is more than necessary. The principle of choosing the number of input variables is that none of the redundant input variables should be retained in the ANFIS inputs.

An efficient ANFIS model with parsimonious number of rules and membership functions can be achieved only through physically correct choice of inputs. In order to characterize the behavior of a dynamic system, physically driving inputs are much more efficient than purely input delay and output delay. If all the input variables driving the dynamic process are included in the model, much fewer delays will be required to perform the input-output mapping.

Model digraph method is in full agreement with the requirement of efficient ANFIS modeling using the prior knowledge of the system. All the physically involved inputs can be exploited through studying the causal-effect relationship. Therefore, the ANFIS model is able to have most appropriate inputs when combined with cause effect analysis. If known, some important nonlinearities can also be directly captured through an unmeasured node in the model graph before it is presented as an input to an ANFIS model. For example, the pressure loss can be assumed as a square function of the flow rate. An unmeasured node can then be designed as the square of the flow rate in the model graph and is used as an input to an ANFIS model to estimate the pressure drop. By explicitly including nonlinear terms in the inputs, fewer membership functions will be needed in the resulting ANFIS models. Model digraph method helps to decompose a complex model into several small models. The model decomposition can significantly enhance the performance of data driven modeling such as ANFIS when used for FDI. In order to achieve an accurate data driven model, the amount of data required is proportional to the number of inputs. When a complex model is decomposed, much fewer inputs are related to each small model, and correspondingly, much fewer data will be required to train the small model than a complex model with a great number of inputs.

In the case that sufficient knowledge is known about a system so that the rules and the member functions of the inputs can be specified, the training algorithm of the ANFIS system can also be adapted for this purpose. Since ANFIS is a fuzzy model in nature, it is easy to integrate expert knowledge in different forms. Knowledge in different confidence can be represented explicitly by appropriate choice of the shape of membership function.

7.5. FDI Procedure for Model Digraph Approach

The following is a summary of the procedure to design a data driven model digraph based FDI.

- Design the model digraph structure.

The structure can be obtained from process flowchart and control system design chart.

- Develop individual models.

In some cases, one model defined in the first step may be decomposed into several models in series. The series of models correspond to the inclusion of some hidden nodes and some multi-model digraph sub-modules.

- Develop fault detection module.

Appropriate thresholds should be specified for all the nodes. A too small threshold may cause false alarms and a too big threshold may cause missing detection.

- Develop causal reasoning algorithm.

If only single faults are involved, the simple residual reasoning algorithm can be directly implemented. If some dual faults are of concern for the FDI system, some extended reasoning schemes may need to be designed.

The implementation of data driven model digraph based FDI can be summarized as following steps:

- Fault detection is fulfilled by monitoring the residual of each model.
- For any abnormal model, the possible root causes are identified by tracking backwards until a model gives insignificant residual.
- All the corrupted signals are reconstructed by tracking forward from the identified fault origin to the input nodes of the detected model.
- Finally, cause effect reasoning is performed on the residuals for fault isolation.

7.6. Applications to Nuclear SG System

Figure 7.5 shows the model digraph of steam generator water level system for a PWR nuclear power plant. The models in series can be summarized as follows:

Controller output (t)=f(steam flow rate(t)-feed water flow rate(t), SG level(t)-SG reference level(t)).
FCV valve position (t+1)=f (controller output (t), FCV valve position (t)).
FCV flow rate (t)=f (FCV valve position (t), SG pressure (t)).
FCV flow rate (t)=f (FCV valve position (t), SG pressure (t)).

Steam flow rate (t+1)=f (feed water flow rate (t), SG pressure (t), hot leg temperature (t), cold leg temperature (t+1)).

Figures 7.6 and 7.7 show a comparison between the estimated controller output and the indicated controller outputs for a controller gain offset fault and a feed water flow meter offset fault. As can be seen, the residual can be used to isolate a controller fault as a local fault. If some other faults related to the controller input signals such as feed water flow rate, steam flow rate, or SG level occur, the residual of the controller output remains close to zero. The reason is that the controller model always uses indicated signals. Even if some faults happen to the input signals of the controller, the controller model itself is still not violated. By the way, the capability of isolating the controller fault as a local fault demonstrates that the ANFIS model is precise enough to capture the dynamic behavior of the controller.

Figures 7.8 and 7.9 show a comparison between the estimated valve position change and the indicated valve position change for a valve position offset fault and a feed water flow meter offset fault. As can be seen, the residual can be used to isolate a valve position fault as a local fault. If feed water flow meter offset fault happens, the residual of the valve position change remains close to zero. The reason is that the valve position change is physically determined by the controller output signal.

Figure 7.10 shows the model digraph to estimate the feed water flow rate. Figure 7.11Figure shows the residual of feed water flow rate before and after SG pressure is reconstructed for a feed water flow meter sensor fault. The figure shows that the residual does not change much before and after all the input signals are reconstructed. Therefore, the detected fault can be isolated as a local fault. Figure 7.12 shows the residual of feed water flow rate before and after SG pressure is reconstructed for SG pressure sensor fault. The reconstruction of SG pressure signal can fully explain the original residual. This indicates that the detected fault is a secondary fault and the fault can be isolated as a SG pressure sensor fault.

Figures 7.13-7.15 show the residual of the steam flow rate for a steam flow meter sensor fault, feed water flow meter sensor fault and SG pressure sensor fault, respectively. Figure 7.13 shows that the residual of steam flow rate does not change much before and after the SG pressure, the feed water flow rate, and the feed water flow rate and the SG pressure is reconstructed. Therefore, the detected fault can be correctly isolated as a steam flow meter sensor fault. Figure 7.14 shows the residual of the steam flow rate for feed water flow meter sensor fault. The reconstruction of feed water flow rate signal can fully explain the original residual. This indicates that the detected fault is a secondary fault and the fault can be isolated as a FCV flow meter sensor fault. Figure 7.15 shows the residual of the steam flow rate for the SG pressure sensor fault, the reconstruction of the SG pressure signal can fully explain the original residual. Therefore, the detected fault is a secondary fault and can be correctly isolated as a SG pressure sensor fault.

In order to detect and isolate the SG narrow range level sensor fault, it is necessary to build a dynamic model to estimate SG level. From the physics point of view, SG level can be determined by the SG mass and the SG thermal parameters. For this reason, an unmeasured node, SG mass, is used to estimate SG level. The SG level is expressed as a function of SG mass, SG pressure, SG temperature as well as feed water temperature, cold leg temperature and hot leg temperature, shown in Figure 7.16. The SG mass can be estimated as a function of feed water flow rate and steam flow rate. In fact, without using the SG mass as an explicit variable, it is extremely difficult to build a data driven model to estimate the SG level. The reason is that the SG mass is the integral effect of the incoming feed water flow rate and the out-flowing steam flow rate. A given value of SG mass may correspond to any value of FCV flow rate and steam flow rate. In the specific case, a model using delay input does not help to track the dynamic behavior either since the SG indicated level would be ultimately brought back to its normal value after a SG level sensor fault due to the controller feedback.

Figure 7.17 shows the residual of the SG level for a SG level sensor fault before and after the input signal is reconstructed. As can be seen, the residual does not change because of the input reconstruction for the detected fault. This indicates that the fault is a local fault. The detected fault can then be successfully isolated.

Figure 7.18 shows the residual of the steam flow rate for simultaneous feed water flow meter sensor fault and SG pressure sensor fault. The original residual can be used to detect the fault. In order to isolate the faults, reconstructed residuals are used. When either SG pressure or feed water flow rate is reconstructed, the residual can be reduced. However, the reconstruction of either signal is not enough to explain 100 percent of the original residual. Only when feed water flow rate and SG pressure are reconstructed can the residual reach minimal. Therefore, from explaining maximal fault signature point of view, the simultaneous dual faults can be correctly isolated.

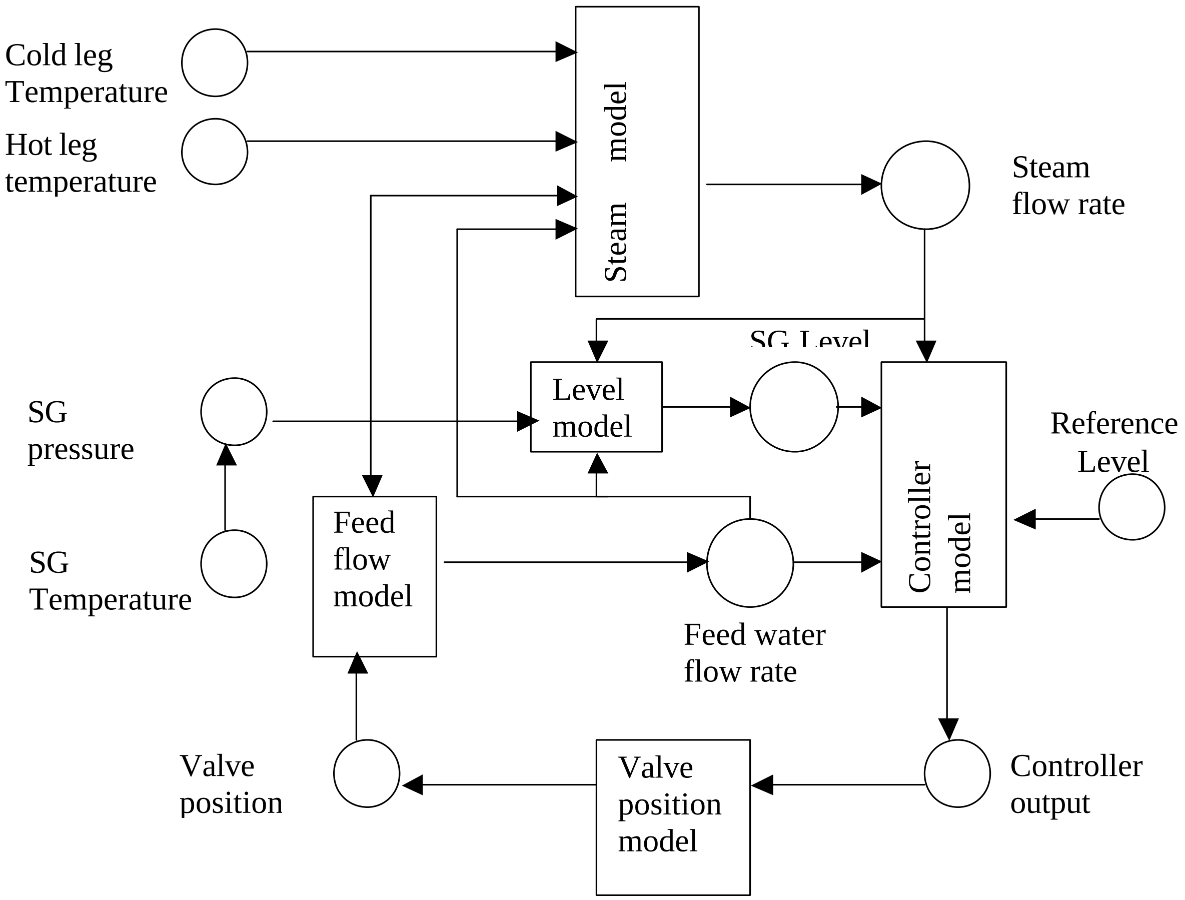


Figure 7.5. Model digraph of nuclear SG system.

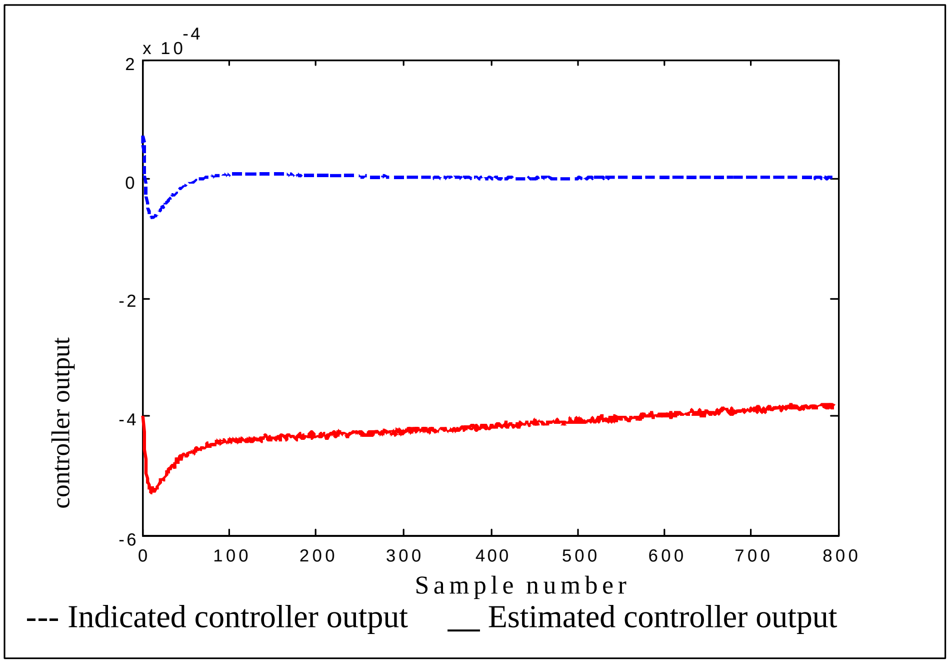


Figure 7.6. Controller output for controller gain offset fault.

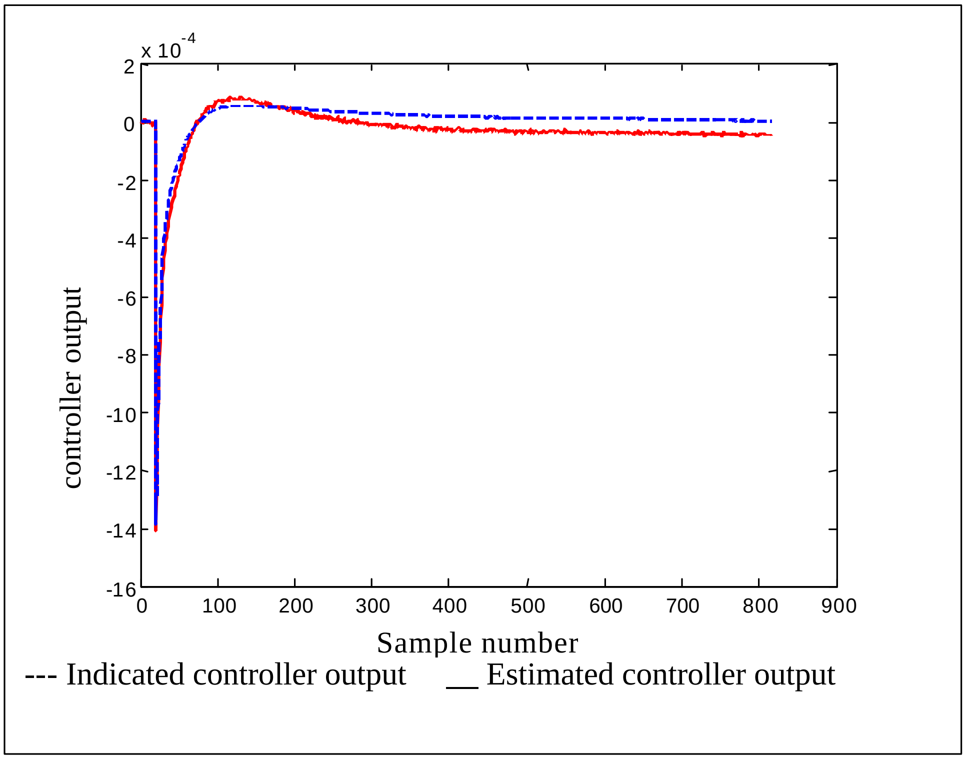


Figure 7.7. Controller output for feed water flow meter sensor fault.

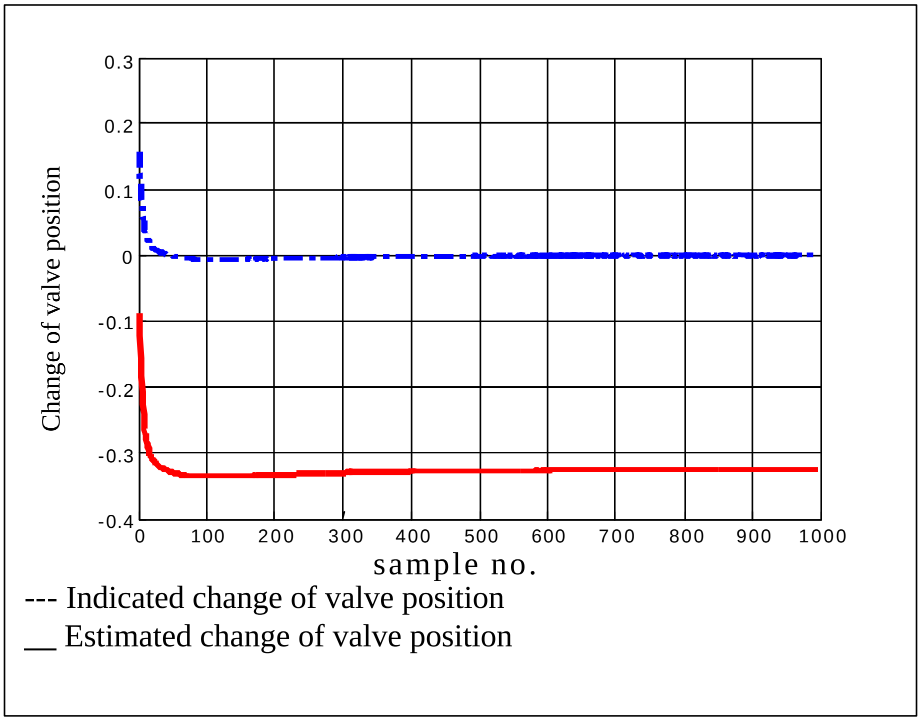


Figure 7.8. Change of valve position for valve position fault.

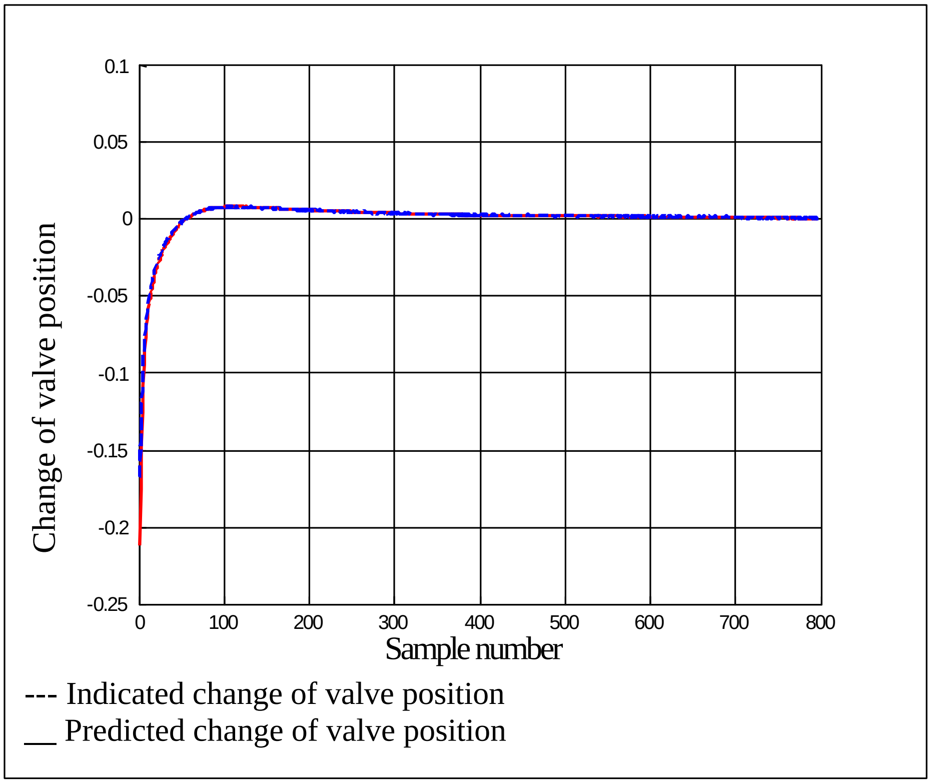


Figure 7.9. Change of valve position for feed water flow meter sensor fault.

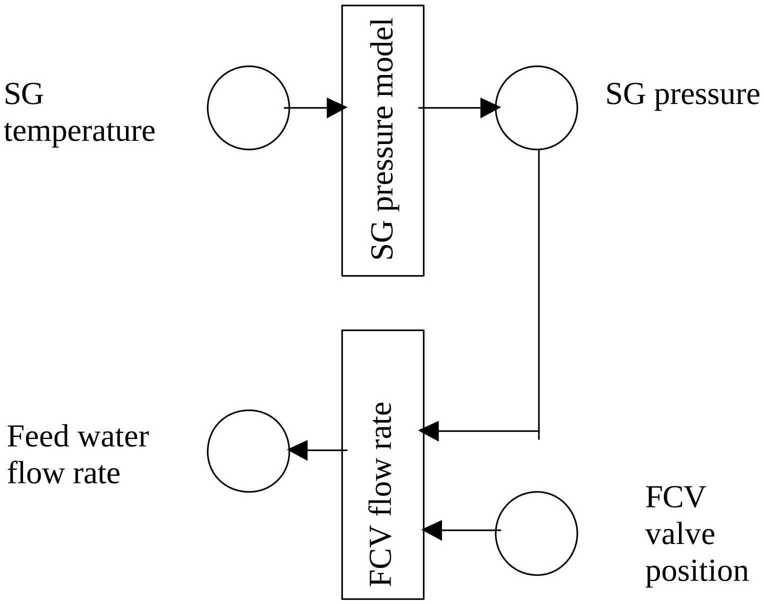


Figure 7.10. Model digraph of feed water flow rate.

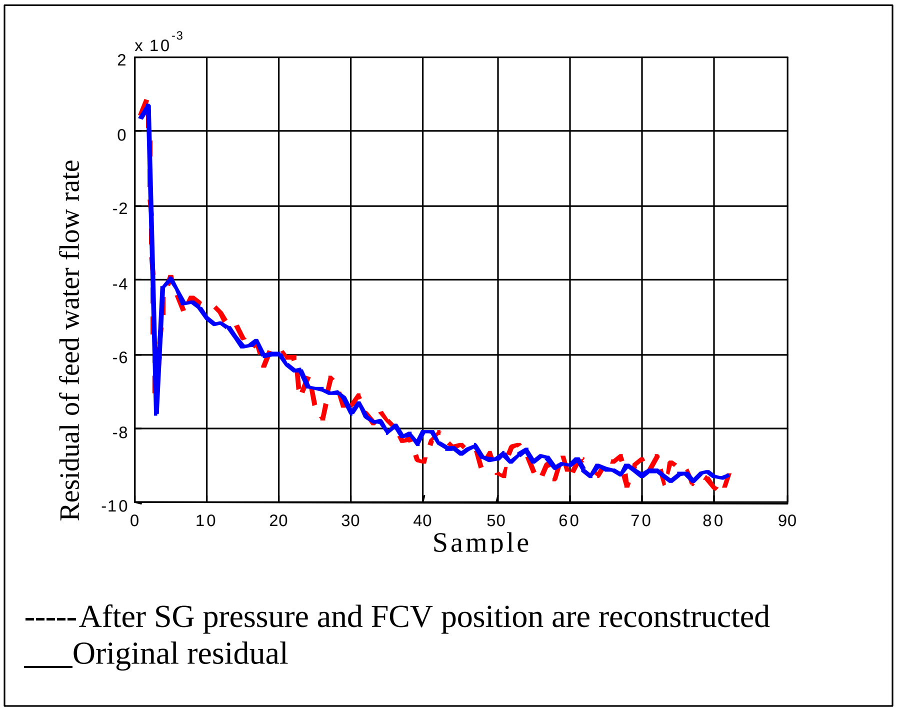


Figure 7.11. Model digraph approach to isolate feed water flow meter sensor fault.

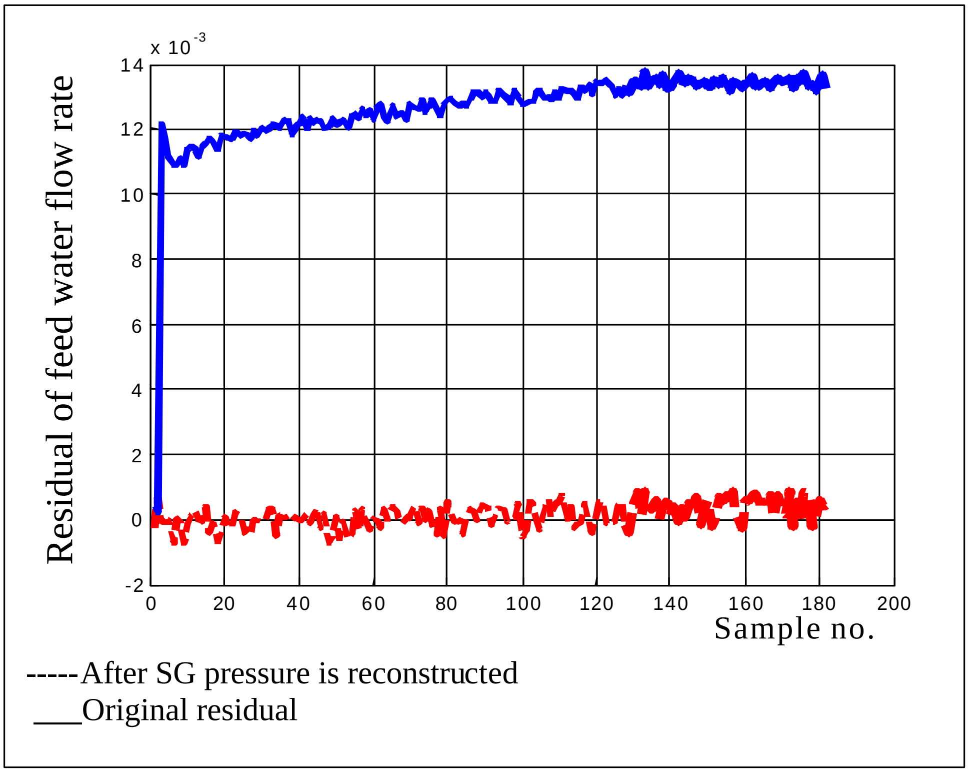


Figure 7.12. Model digraph approach to isolate SG pressure sensor fault using feed water flow rate model.

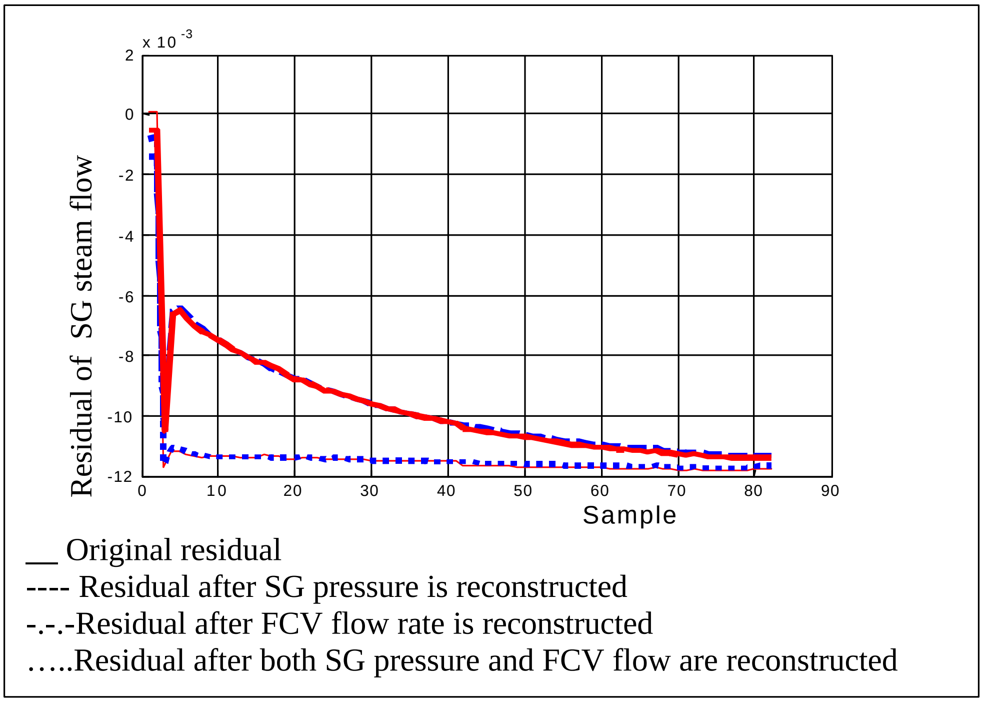


Figure 7.13. Model digraph approach to isolate steam flow meter sensor fault.

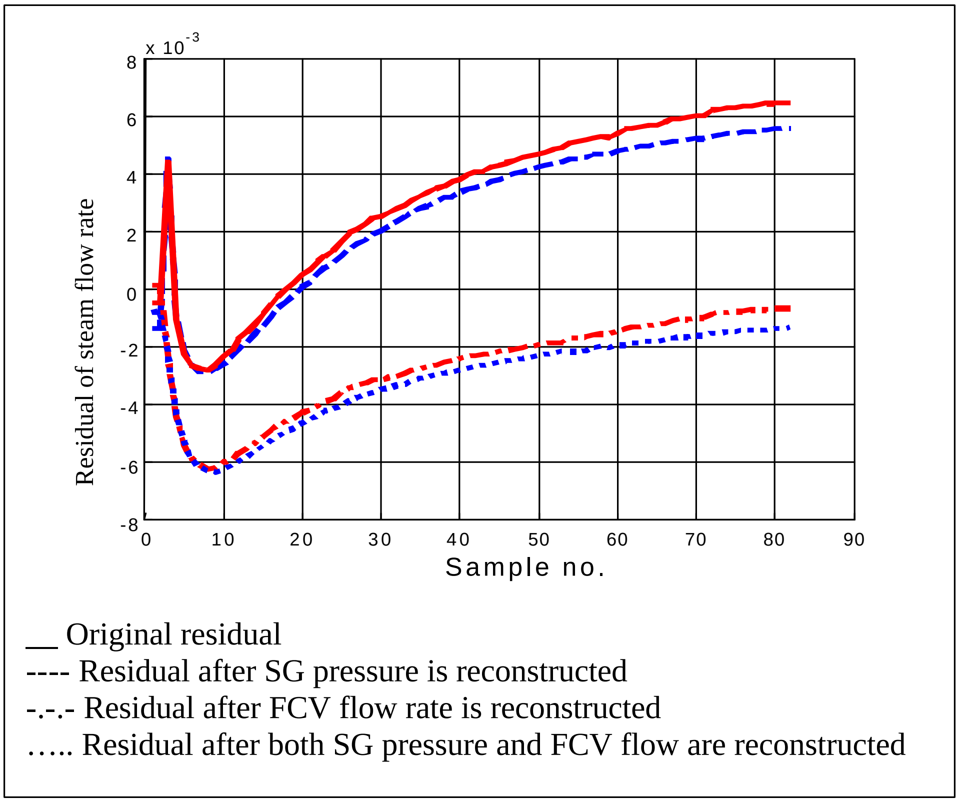


Figure 7.14. Model digraph approach to isolate feed water flow meter sensor fault.

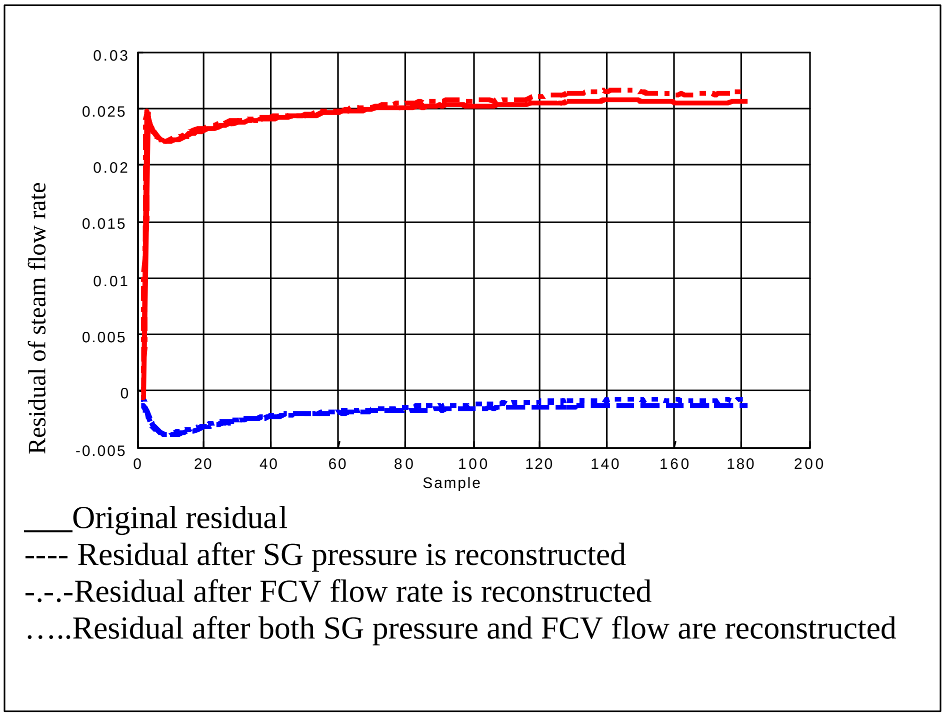


Figure 7.15. Model digraph approach to isolate SG pressure sensor using steam flow rate model.

Figure 7.19 shows the residual of feed water flow rate for simultaneous feed water flow meter sensor fault and SG pressure sensor fault. The original residual can be used to detect the fault. After SG pressure is reconstructed, the residual of FCV flow rate can be reduced by 50%. After both SG pressure and FCV position are reconstructed, the residual of FCV flow rate cannot be further reduced. It can be concluded that SG pressure sensor is faulty and FCV valve position is healthy. However, the remained residual is still about 0.5%. This fraction of the original residual must be explained by the assumption that the feed water flow meter sensor has a fault.

7.7. Comparison with Other Approaches

Although PCA based FDI and ANFIS model based FDI with structured residual design can be used for fault detection and isolation in some applications, model digraph approach has some unique features in FDI system design and application to engineering problem.

Both PCA based FDI and ANFIS model based FDI with structured residual design must be designed for enumerable faults. However, model digraph approach isolates a fault based on cause effect reasoning on model residuals, so it is not necessary to predefine the faults and their fault signatures. To incorporate an automated fault detection and isolation into a large safety critical system such as nuclear power plants, this is a significant step moving forward to engineering application.

Model digraph approach achieves fault isolation by screening out a fault among all the possible fault candidates so that all the abnormal measurements can be explained. Therefore, there is no problem with misdiagnosing one fault as another. However, both PCA based FDI and ANFIS model based FDI with structured residual design do not have a safeguard against the possibility that some unknown faults may have the same fault signatures as defined for a fault in the fault dictionary. Although structured residual design approach is able to avoid misdiagnosing one fault as another for the enumerated faults through manipulating the residual structures, it still cannot fully solve the problem. In addition, structured residual design is not always achievable especially for a non-linear system.

Data driven model digraph approach is able to meet the requirement for automation because only normal operation data are necessary to adaptively upgrade system models. The fault signatures used for fault isolation are extracted from the understanding about the physical system instead of time consuming simulation or additional experiments.

Digraph approach allows accurate data driven modeling. The most parsimonious model structure can be obtained through a model digraph. Therefore, it can improve the accuracy of the developed data driven models significantly. In addition, the model structure is consistent with the system decomposition, so it helps to arrive at a modular FDI system.

Because fault isolation is based on reasoning about model residuals, data driven model diagraph approach is also able to deal with simultaneous faults.

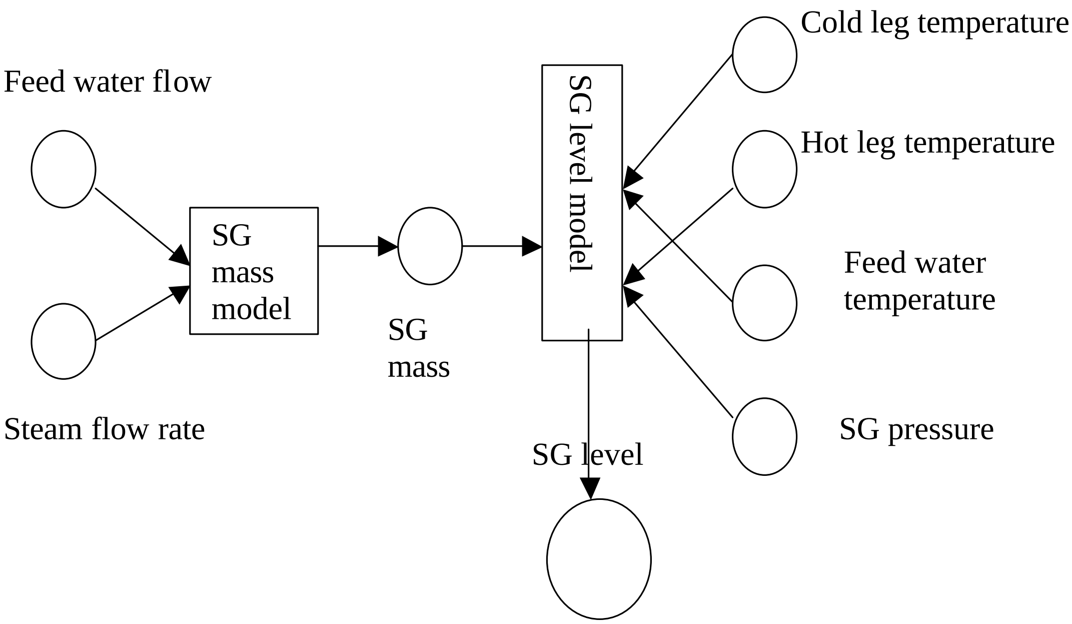


Figure 7.16. Model digraph of SG level measurement.

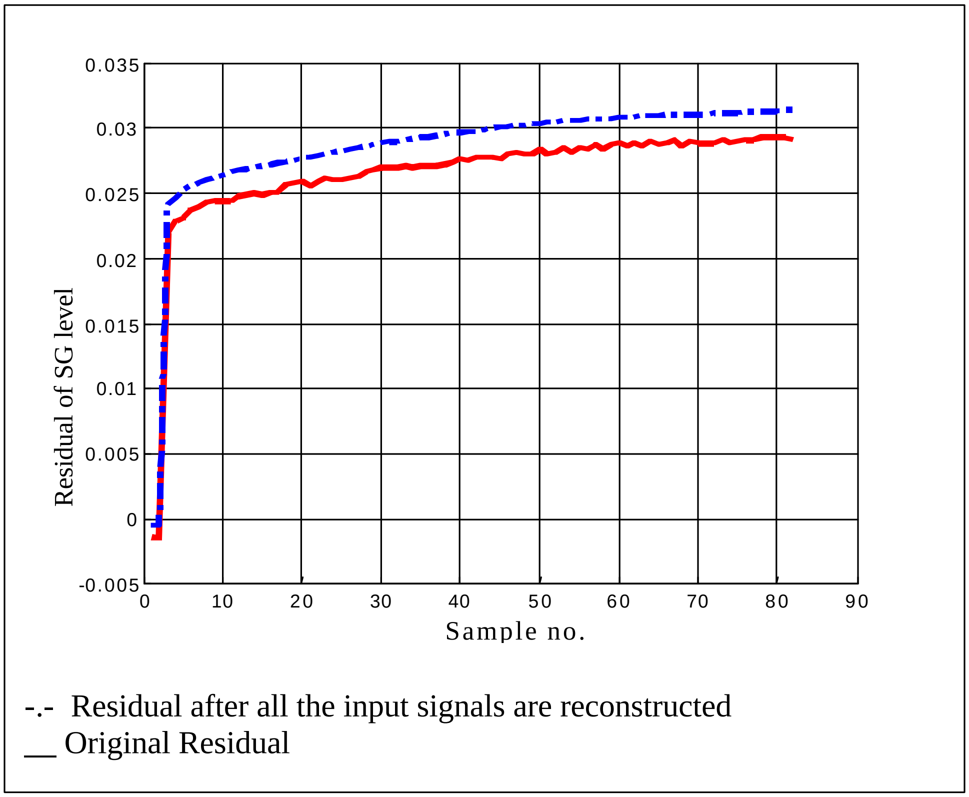


Figure 7.17. Model digraph approach to isolate SG level sensor fault.

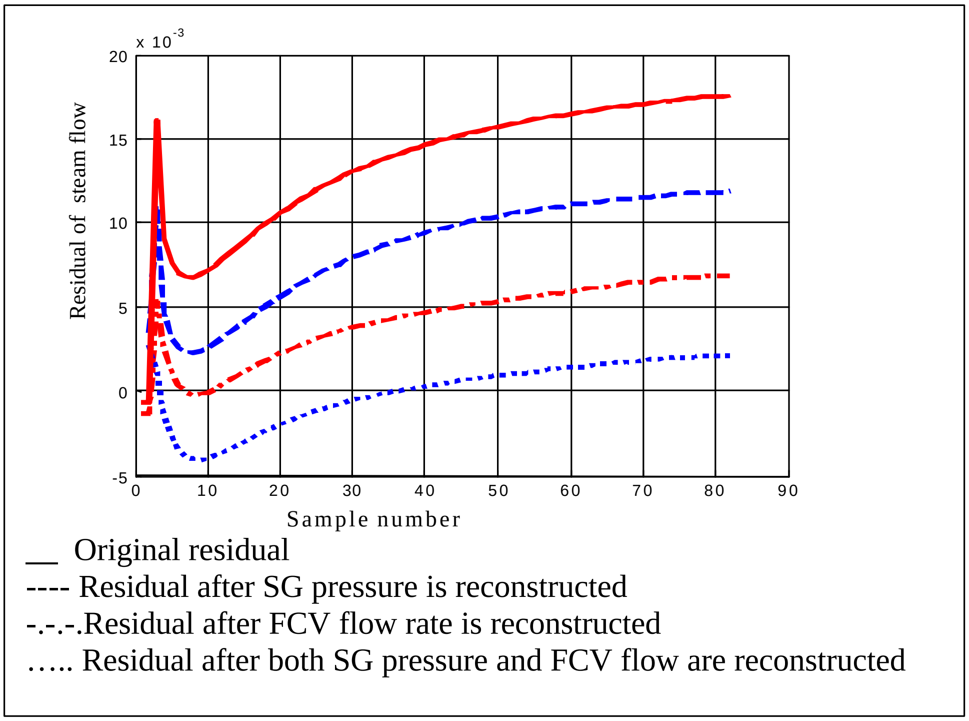


Figure 7.18. Model digraph approach to isolate feed water flow meter sensor fault and SG pressure sensor fault.

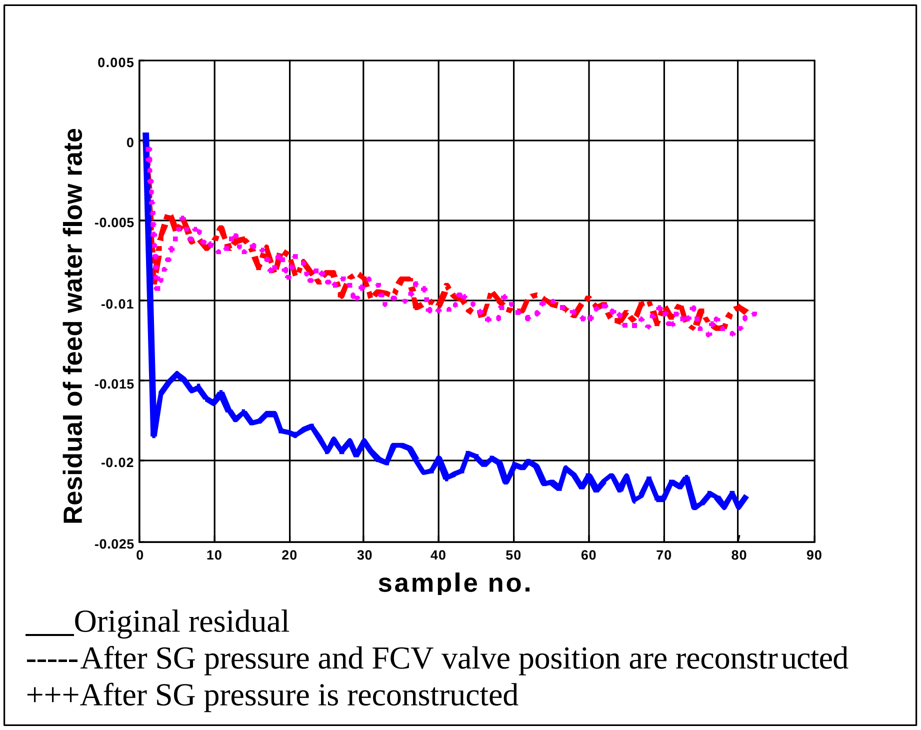


Figure 7.19. Model digraph approach to isolate SG pressure sensor fault and feed water flow meter sensor fault.

8. APPLICATION TO EXTENDED SYSTEM AND THE DEVELOPMENT OF THE FDI DEMONSTRATION MODULE

(neriFinalReport02-09)

8.1. Application of FDI Method to Feed Pump Monitoring

In order to test the feasibility of applying the FDI algorithm to the scope that includes an extension of the SG system boundary, the FDI algorithm is extended to the subsystem beyond the steam generator. In this application an attempt is made in detecting the performance degradation of the feed pump. The deviation of the feed pump from its nominal operating condition is studied using the established FDI modules.

The feed water pump dynamics affects system variables such as the feed flow rate and FCV position. Therefore, the deviation of the feed pump speed from its nominal value (about 1200 RPM) would cause the change of parity relation among related variables. The induced pump fault is an over-speed operation. There is about 25% increase of pump speed and head for the pump fault in NCSU code. The fault signature of the over-speed pump is reflected in the residual space generated from the GMDH models that are established in order to simulate the causal relationship among variables in a SG system. It is found that the fault signature due to the pump over-speed is significantly different from other fault signatures described in Chapter 6. Therefore, it is straightforward to extend the FDI algorithm to equipment beyond the SG system boundary in this case.

Figure 8.1 shows the residual pattern during pump over-speed operation. The residual pattern indicates that this pump anomaly mainly affects the FCV position (the third residual in the figure) change, which is reasonable from the point of keeping the required power output and the SG water level set point. The nine residual directions are the same residuals used in SG residual pattern generation.

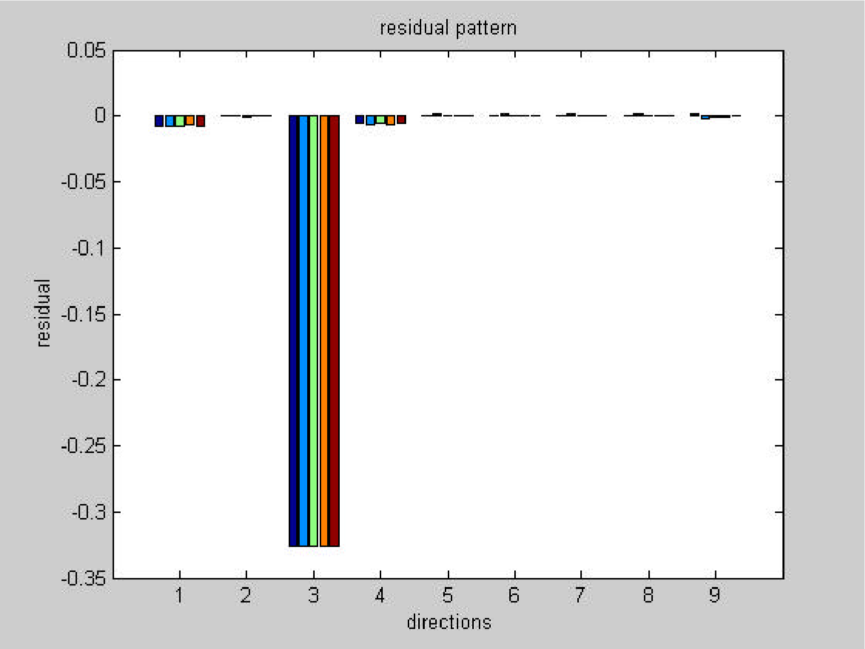
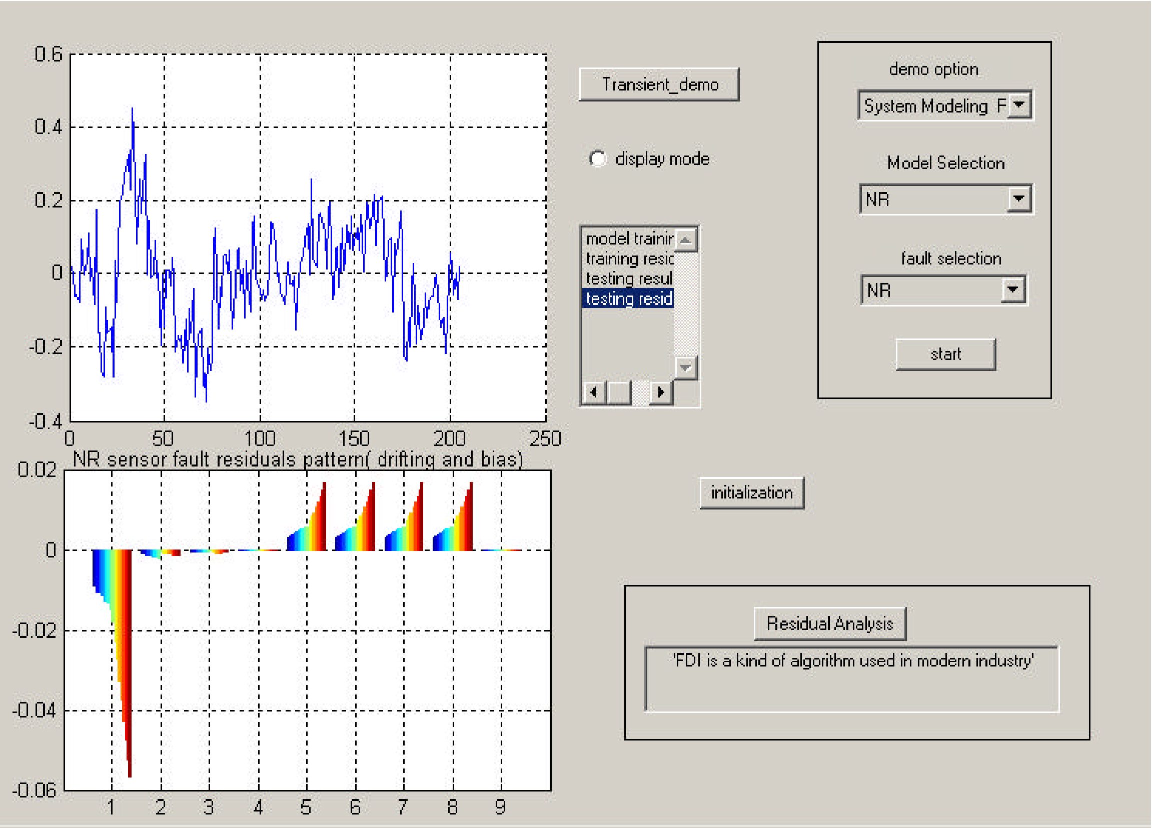


Figure 8.1. Residual Pattern for the case of equipment anomaly caused by feed pump overspeed operation.

8.2. Development of a FDI Demonstration Module

The FDI system developed at The University of Tennessee is part of a control design architecture for nuclear power plants, being developed jointly with ORNL and NCSU, under a DOE-NERI grant. A stand-alone demonstration module has been developed to illustrate the capabilities of the FDI system. As part of this effort a graphical user interface (GUI) are developed using the MATLAB platform. The elements of the GUI are shown in Figure 8.2. The demonstration illustrates the steps involved in establishing an FDI system using the static GMDH models, training phase, fault generation, and classification using structured residual analysis or directional residual analysis. A separate GUI has been developed for the transient FDI analysis.



○ display mode

demo option: System Modeling F

Model Selection: NR

fault selection: NR

'FDI is a kind of algorithm used in modern industry'

Figure 8.2. Graphical user interface for the FDI system.

9. CONCLUDING REMARKS AND RECOMMENDATIONS FOR FUTURE WORK

(neriFinalReport02-10)

9.1. Summary and Concluding Remarks

An integrated method for incipient *fault detection and isolation* (FDI) of sensors and field devices, using data characterization, fault detection, and classification, has been developed and described in this report. The FDI system uses three approaches for characterizing system variables – states and control functions. These are (1) group method of data handling (GMDH) with rational function approximation, (2) principal component analysis (PCA) with nonlinear extension, and (3) adaptive network fuzzy inference system (ANFIS). The isolation of device faults is performed using a rule-based decision-making, a multi-observer digraph logic, and a pattern classification of prediction error vectors in the *fault space*. This integrated approach enhances the fault diagnostics capability and provides a robust method for FDI. The extensions of the basic approaches also include the more difficult situations of detection and isolation of dual faults in the steam generator system of a PWR plant and fault detection during plant transients. A stand-alone demonstration of the FDI system has been developed using the Halden's *PICASSO* GUI platform. A MATLAB-based demonstration of the FDI technique, that illustrates the various computational steps of the software system, is also included as part of the deliverables.

The methods being developed under this task for the DOE-NERI project have been tested and illustrated using process measurements from a laboratory process control loop and a U-tube steam generator in a PWR. All the faults being simulated were detected successfully. The faults were primarily of the bias and drift types. Appendix A provides a summary of the software system. The computer codes and the data used for analysis are available on a CD-R. The outcomes of the projects have been disseminated in scientific meetings and by journal publications.

9.2. Recommendations for Future Work

The following recommendations are presented for extension and implementation of the research and development results of this project:

- Application of the FDI methods to new generation reactors such as the gas-cooled reactor and the IRIS reactor.
- Incorporation of FDI-based techniques for monitoring critical plant equipment for on-line maintenance that would assist in extending the duration of fuel cycles of future power plants.
- Extension of the FDI method by increasing the system boundary to include the whole plant, both primary and balance of plant systems.
- Implementation of fault detection and isolation as an autonomous (without human intervention) monitoring system that would be executed with minimal human interaction.

Appendix A. Summary of Computer Codes Developed for the Fault Detection and Isolation (FDI) System

(neriFinalReport02-11)

A.1. Software System for Single Fault Analysis

A.1.1. Package 1: FDI Modules for the Steady State Case

These MATLAB codes are used for steam generator system modeling based on the GMDH algorithm and residual direction identification using PCA. The training data are generated using a detailed simulation code developed by NCSU.

Implementation Steps:

- Run the code 'FDI_ornl.m' under MATLAB command.
- Follow the instructions that appear on the screen.

Read the file called README.txt for further details.
System requirement: MATLAB 5.3 or higher.

A.1.2. Package 2: FDI Modules for the Transient Case

The GMDH algorithm is adapted for transient process modeling. PCA is used in residual direction identification. The training data are collected during transient processes as the thermal power changes from one level to another.

Implementation Steps:

- Run the codes named 'SG_model_*.m' for system modeling. For example, SG_model_TCV.
- Run the codes in the form of 'fault_det_*.m' for fault detection and directional fault isolation.

Read the file ‘README.txt’ for further details.
System requirement: MATLAB 5.3 or higher versions.

A.1.3. Package 3: GUI for FDI Demonstration

System requirement: MATLAB 6.0 or higher versions.

Demonstration procedure for steady state FDI:

a. Input command ‘guitest’ under MATLAB environment.
b. Select the demo options in the upper right pop-up menu. The second option, that is, residual generation, is required before residual analysis.
c. Choose a component model of interest in the model selection menu. There are totally six GMDH models that can be selected.
d. Select the fault type in the pop-up menu called *fault type*.
e. Press the start button to begin the demo.
f. Choose the display option from the display list box in the middle part of the panel. The results will be illustrated in the upper left plot.

g. Press residual analysis button located in the right bottom corner on the panel. The FDI result will be described in text form under this button. At the same time, the fault direction is plotted in the left bottom corner.

FDI demo for transient condition:

Input the command 'gui_tran' under MATLAB environment. Refer to the static FDI above for menu option information.

Note: at least 256 MB memory is required for transient demonstration.

A.2. Software System for Dual Fault Analysis

A.2.1. Development environment

The codes are developed under MATLAB 6.0 environment. The MATLAB Fuzzy Logic toolbox is necessary to run them. To run the demonstration, Picasso 3.0, MATLAB compiler 2.0, MS FORTRAN 6.0 and MS C++ 6.0 are required.

A.2.2. PCA Based FDI Module

- **MultiplePCA**

  In this code, five single faults (steam flow meter sensor faults, feed water flow meter sensor fault, feed water control valve position fault, steam generator narrow range level sensor fault) can be detected and isolated based on five PCA models. The Q-statistics are calculated for each PCA model. The residual structure defined by the Q-statistics after a fault happens is used for fault isolation. The code has incorporated six cases to show its success in fault detection and isolation.

- **PCADir**

  In this code, five single faults (steam flow meter sensor faults, feed water flow meter sensor fault, feed water control valve position fault, steam generator narrow range level sensor fault) can be isolated based on fault directions. The fault directions are defined by the change of PCA scores and residuals before a fault and after a fault. It has incorporated six cases to show its success in fault isolation.

A.2.3. ANFIS Based FDI Module

- **AnfisFCVFlow, AnfisSGLevel,AnfisSGPrs,AnfisSteamFlow,AnfisValPos**

  These five codes are used to train ANFIS models to predict feed water flow rate, steam generator narrow range level, steam generator pressure, steam flow rate, and feed water control valve position. These models are dynamic models, so they can be used to track the dynamic behavior after a fault happens. Each code also shows the residual responses for the selected 13 faults so that an appropriate rule base can be developed for fault isolation.

- **ANFISvalidation**

  In this code, the four ANFIS models developed to predict steam flow rate, feed water control valve position, feed water flow rate and steam generator level are examined to simulate the power transient from 100% power level to 90% power level with a slow ramping rate. The code gives the prediction error for these four variables.
- **LocalResdGen**

  In this code, the five ANFIS models developed are used to generate the residual patterns for all the considered 13 faults. It can be used to study the stability of residual structures for different fault magnitudes and different initial power level when a fault occurs.

A.2.4. Model Digraph Based FDI Module

- **MAnfisFCVFlow**

  This code computes feed water flow rate based on a) measured feed water control valve position and feed water temperature; b) reconstructed feed control valve position and feed water temperature. It may be used to isolate a feed water flow meter sensor fault.

- **MAnfisSGLevel**

  This code computes steam generator narrow range level based on a) wide range level and steam generator pressure; b) wide range level and reconstructed steam generator pressure. It may be used to isolate a steam generator level sensor fault and a steam generator pressure sensor fault.

- **MAnfisSGPrs**

  This code computes steam generator pressure based on steam generator temperature. It may be used to isolate a steam generator pressure sensor fault.

- **MAnfisSteamFlow**

  This code computes steam flow rate based on a) measured feed water flow rate, measured steam generator pressure, clod leg temperature and hot leg temperature; b) reconstructed feed water flow rate, steam generator pressure, clod leg temperature and hot leg temperature; c) measured feed water flow rate, reconstructed steam generator pressure, clod leg temperature and hot leg temperature. It may be used to isolate a feed water flow meter sensor fault, a steam generator pressure sensor fault and a steam flow meter sensor fault.

- **MAnfisController**

  This code computes feed water controller output based on measured steam generator level, measured feed water flow rate and measured steam flow rate. It may be used to isolate a controller fault.

- **ManfisValPos**

  This code computes feed water control valve position based on controller output. It may be used to isolate a feed water control valve position fault.

A.2.5. FDI Demonstration Using PICASSO Platform

This demonstration is able to show the effectiveness of the developed FDI approach for nuclear power plants under the environment of Picasso-3, a user interface management system. The software has the following functions a) Create a fault by changing the fault characteristic parameters; b) Display essential parameters on the flowchart of the reactor system; c) Display the residual patterns specific to a fault; and d) Trend the process variables relevant to the fault, and echo the FDI results.

BIBLIOGRAPHY

1. P.B. Ferreira and B.R. Upadhyaya, *"Incipient Fault Detection and Isolation of Sensors and Field Devices,"* Research Report, Nuclear Engineering Department, The University of Tennessee, Knoxville, UTNE/BRU/99-02, December 1999.

2. A.S. Erbay and B.R. Upadhyaya, "*A Personal Computer-Based On-Line Signal Validation System for Nuclear Power Plants*," Nuclear Technology, Vol. 119, No. 1, pp. 63-75, July 1997.

3. B.R. Upadhyaya and W. Yan, "*Hybrid Digital Signal processing and Neural Networks for Automated Diagnostics Using NDE Methods*," Final Report prepared for the U.S. Nuclear Regulatory Commission, November 1993.

4. X. Xu, J.W. Hines and R.E. Uhrig, "*Sensor Validation and Fault Detection Using Neural Networks*," Proc. MARCON 99, Vol. **2**, pp. 58.01-58.09, May 1999.

5. S. J. Qin, and W. Li, "*Detection, Identification, and Reconstruction of Faulty Sensors with Maximized Sensitivity*," *AIChE Journal*, Vol. 45, p. 1963, 1999.

6. R. Dunia, S. J. Qin, T. F. Edgar, and T. J. Mcavoy, "*Sensor Fault Identification and Reconstruction Using Principal Component Analysis,"* Proceedings of the 13th Triennial World Congress, pp. 259-264, San Francisco, 1996.

7. J. Gertler, W. Li, Y. Huang, and T. J. McAvoy, "*Isolation Enhanced Principal Component Analysis*," *AIChE Journal*, Vol. 45, p 323, 1999.

8. R. S. Mangoubi, *"Robust Estimation and Failure Detection,"* Springer, 1998.

9. A. S. Erbay, *"A PC Based Signal Validation System for Nuclear Power Plants,"* M.S. Thesis, The University of Tennessee, Knoxville, December 1994.

10. C. Chen, *"Statistical Pattern Recognition,"* Spartan Books, New Jersey, 1973.

11. B. R. Upadhyaya, N. Kaistha, I. Goncalves, and B Lu, "*Fault Detection and Isolation of Nuclear Plant System Sensors and Field Devices,"* Annual report prepared for DOE-NERI Project, August 2000.

12. J.C. Bezdek, "*Pattern Recognition with Fuzzy Objective Function Algorithms,"* Plenum Press, New York, 1981.

13. J. Fox, *"Applied Regression Analysis, Linear Models, And Related Methods,"* Sage Publications, Thousand Oaks, CA, 1997.

14. S. K. Pal and P.P. Wang, "*Genetic Algorithms for Pattern Recognition,"* CRC Press, Boca Raton, 1996.

15. K. Fukunaga, "*Introduction to Statistical Pattern Recognition,"* Academic Press, New York, 1972.

16. S. Watanabe, *"Methodologies of Pattern Recognition,"* Academic Press, New York, 1969.

17. Y.J. Lee, H.J. Lee, and K.Y. Kim, *"The level control system design of the nuclear steam generator for robustness and performance,*" Journal of the Korean Nuclear Society, Vol. 32, No.2, pp. 157-168, April 2000.

18. M. Naghedolfeizi and B.R. Upadhyaya, *"Dynamic Modeling of a Pressurized Water Reactor Plant for Diagnostics and Control,"* Research Report, University of Tennessee, DOE/NE/88ER12824-02, June 1991.

19. C. Antoniades and P.D. Christofides, "*Non-linear feedback control of parabolic partial differential difference equation systems,*" Journal of International Control, Vol. 73, No.17, pp. 1572-1591, 2000.

20. T.H. Guo and J. Musgrave, *"Neural Network Based Sensor Validation for Reusable Rocket Engines,"* Proceedings of the American Control Conference, Seattle, Vol. 2, pp. 1367-1672, 1995.

21. W.C.Merrill, J.L. Musgrave, and T.H. Guo, "*Integrated Health Monitoring and Controls for the Rocket Engines,"* SAE Technical Paper Series 1992 SAE Aerospace Atlantic Conference, pp. 1-5, Dayton, OH, 1992.

22. C.R. Hild, "*Development of The Group Method of Data Handling With Information-based Model Evaluation Criteria: A New Approach to Statistical Modeling,"* Ph.D. Dissertation, The University of Tennessee, Knoxville, August 1998.

23. D.T. Pham, *"Neural Networks for Identification, Prediction and Control,"* Springer, London, 1998.

24. N. Kaistha and B. R. Upadhyaya, *"Incipient Fault Detection and Isolation of Field*

Devices in Nuclear Power Systems Using Principal Component Analysis," Nuclear Technology, Vol. 136, No. 2, pp. 221-230, November 2001.

25. G. Ivakhnenko, Home page at http://www.stormloader.com/gmdh/GMDH_his.htm, 2002.

26. MATLAB Statistics Toolbox, User's Guide, MathWorks, Inc., 2000.

27. R.O. Duda, P.E. Hart, and D.G. Stork, *"Pattern Classification,"* Second Edition, John Wiley, New York, 2001.

28. S. Yoon and J.F. MacGregor, *"Statistical and Causal Model-Based Approaches to Fault Detection and Isolation," AIChE Journal*, Vol. 46, pp. 1813-1824, September 2000.

29. L.C. Shen, S.K. Chang, and P.L. Hsu, *"Robust Fault Detection and Isolation with Unstructured Uncertainty Eigenstructure Assignment,"* Journal of Guidance, Control and Dynamics, Vol. 21, No.1, pp. 50-57, January-February 1998.

30. C.D. Persis and A. Isidori, *" A Geometric Approach to Nonlinear Fault Detection and Isolation," IEEE Transactions on Automatic Control*, Vol. 46, No. 6, pp. 853-865, June 2001.

31. A. Ray and M. Desai, *"A Redundancy Management Procedure for Fault Detection and Isolation,"* Transaction of the AMSE, Vol. 108, pp. 248-254, September 1986.

32. S.J. Farlow, *"Self-Organizing Methods in Modeling: GMDH Type Algorithms,"* Marcel Dekker, New York, 1984.

33. J. W. Hines, D. W. Miller, and B. K. Hajek, *"Fault Detection and Isolation: A Hybrid Approach,"* Proceedings of the 1995 American Nuclear Society Annual Meeting and Embedded Topical Meeting on Computer-Based Human Support Systems: Technology, Methods and Future, Philadelphia, 1995.

34. J. W. Hines, D. W. Miller, and B. K. Hajek, *"A Hybrid Approach for Detecting and Isolating Faults in Nuclear Power Plant Interacting Systems,"* Nuclear Technology, Vol. 115, September 1996.

35. B. Rasmussen, J.W. Hines and R.E. Uhrig, "*Nonlinear Partial Least Squares Modeling for Instrument Surveillance and Calibration Verification,*" Proc. MARCON 99, Vol. **2**, pp. 36.01-58.09, May 2000.

36. X. Xu and J.W. Hines, “*On-Line Sensor Calibration Monitoring and Fault Detection for Chemical Processes*,” Proc. MARCON 98, Vol., pp. 01-, May 2000.

37. C. Black, J. W. Hines and R. E. Uhrig, “*Inferential Neural Networks for Nuclear Power Plant Sensor Channel Drift Monitoring,”* Proc., The 1996 American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control and Human Machine Interface Technologies, University Park, PA, May 6-9, 1996.

38. D. Wrest, J. W. Hines, and R. E. Uhrig, *“Instrument Surveillance and Calibration Verification Through Plant Wide Monitoring Using Autoassociative Neural Networks,”* Proc., American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control and Human Machine Interface Technologies, University Park, PA, May 6-9, 1996.

39. E. Russell, L.H. Chiang and R.D. Braatz, “Data-driven Methods for Fault Detection and Diagnosis in Chemical Process,” Springer, London, 2000.

40. J. Amsterdam, “Automated Qualitative Modeling of Dynamic Physical Systems,” Ph.D. Dissertation, MIT, 1992.

41. J.S. Albuquerque and L.T. Biegler, “Data Reconciliation and Gross Error Detection for dynamic systems,” AIChE Journal, Vol. 42, pp. 2841-2856, 1996.

42. J. Chen and R.J. Patton, Robust Model Based Fault Diagnosis for Dynamic Systems, Kluwer Academic Publisher, 1999.

43. M. Doster, “A Code for PWR System Thermal Hydraulic Analysis,” North Carolina State University, 2000.

44. R. Davis, “Diagnosis via causal reasoning: paths of interaction and the locality principle,” American Association of Artificial Intelligence, pp. 88-94, 1983.

45. S. Dash and V. Venkatasubramanian, “Challenges in the Industrial Applications of Fault Diagnostic Systems,” Comput. & Chem. Engng, Vo. 24, pp. 785-791, 2000.

46. Electric Power Research Institute, EPRI/Utility Program Demonstrates Advanced Nuclear Control System, June 1994.
(http://www.nuc.berkeley.edu/thyd/ne161/rtse/softcon.html)

47. P.M. Frank, "Fault Diagnosis in Dynamic Systems Using Analytical and Knowledge-Based Redundancy -- a Survey and some new Results, Automatica, Vol. 26, pp. 459-474, 1990.

48. F. Garcia, J.V. Izquierdo, L.J. Miguel and J.R. Peran, "Fault Diagnostic System Using Analytical Fuzzy Redundancy," Engineering Applications of Artificial Intelligence, Vol. 13, pp. 441-450, 2000.

49. B. Flury, Common Principal Components and Related Multivariate Models," John Wiley, New York, 1989.

50. J. Gertler and D. Singer, "Augmented Models for Statistical Fault Isolation in Complex Dynamic System," Proceedings of the American Control Conference, pp. 317-322, 1985.

51. J. Gertler and W. Li, "Isolation Enhanced Principal Component Analysis," AIChE Journal, Vol. 45, pp. 323-333, 1999.

52. O. Glockler and M.V. Tublett, "Application of Reactor Noise Analysis in the Candu Reactors of Ontario Hydro," Progress in Nuclear Energy, Vol.29, pp. 171-191, 1995.

53. D.M. Himmelblau, "Fault Detection and Diagnosis in Chemical and Petrochemical Processes," Elsevier, New York, 1978.

54. J.W. Hines, D.J. Wrest, and R.E. Uhrig, "Signal Validation Using an Adaptive Neural Fuzzy Inference System," University of Tennessee, 1997.

55. H.L. Jones, "Failure Detection in Linear Systems," PhD Dissertation, Dept. of Aeronautics, MIT, Cambridge, 1971.

56. J.E. Jackson and G.S. Mudholkar, "Control Procedures for Residuals Associated With Principal Component Analysis," Technometrics, Vol. 21, pp. 341-349, 1979.

57. J.R. Jang and C. Sun, "ANFIS: Adaptive-Network-based Fuzzy Inference Systems," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 23, 1993.

58. O. Jakobsen, The PICASSO-3 User Interface Management System, 1994.

59. A.B. Kjell, "Picasso: a User Interface Management System for Real Time Applications," OECD Halden Reactor Project, 1992.

60. A.B. Kjell, “Developing Graphics Applications in an Interactive Environment,” 1994.

61. A.B. Kjell, “Implementation of Graphical User Interfaces in Nuclear Applications,” 1994.

62. M.A. Kramer and B.L. Palowitch, Jr., “A Rule Based Approach to Fault Diagnosis Using the Signed Directed Graph,” AIChE Journal, Vol. 33, pp. 1067-1078, 1987.

63. G. Lee, “Multiple Fault Diagnosis Under Uncertain Conditions by Quantification of Qualitative Relations,” American Chemical Society, 1999.

64. L. Leyval, S. Gentil and F. Stephan, “Model Based Causal Reasoning for Process Supervision,” Automatica, Vol. 30, pp. 1295-1306, 1994.

65. P.J. Montmain and S. Gentil, Dynamic Causal Model Diagnostic Reasoning for Online Technical Process Supervision,” Automatica, Vol. 36, pp.1137-1152, 2000.

66. P.J. Mosterman and G. Biswas, “Monitoring, Prediction and Fault Isolation in Dynamic Physical Systems,” Proceedings of AAAI, pp. 100-105, 1997.

67. A. Raich and A. Cinar, “Statistical Process Monitoring and Disturbance Diagnosis in Multivariable Continuous Processes, AIChE Journal,Vol. 42, pp. 995-1009, 1996.

68. E.L. Russell and L.H. Chiang, “Data-Driven Methods for Fault Detection and Diagnosis in Chemical Processes,” Springer-Verlag, New York, 2000.

69. S. Simani, “Model Based Fault Diagnosis in Dynamic Systems Using Identification Technique,” University of Hull, UK, 2000.

70. B.R. Upadhyaya et al., “Incipient Fault Detection and Isolation of Sensors and Field Devices. Research Report, Nuclear Engineering Department, The University Of Tennessee, UTNE/BRU/99-02, 2000.

71. B.R. Upadhyaay, “Introduction to Digital Signal Processing and Power Plant Monitoring, University of Tennessee, 2001.

72. H. Vedam and V. Venkatasubramanian, PCA-SDG based Process Monitoring and Fault Diagnosis,” Applications of Artificial Intelligence, Vol. 8, pp. 689-701, 1995.

73. J.D. White, “Comparative Assessments of Nuclear Instrumentation and Controls in the U.S., Canada, West Europe, Japan and former Soviet Union," Oak Ridge National Laboratory, (http://itri.loyola.edu/ar93_94/canic.htm), 1994.

74. S. Wold, “Cross Validation Estimation of the Number of Components in Factor and Principal Component Analysis,” Technometrics, Vol. 20, pp. 397-406, 1978.

75. S. Yoon and J.F. MacGregor, “Fault Diagnosis with Multivariate Statistical Models Part I: Using Steady State Fault Signatures,” Journal of Process Control, Vol. 11, pp. 387-400, 2001.

76. A. Zakarian and A. Kusiak, “Analysis of Process Models,” IEEE Transactions on Electronics Packaging and Manufacturing, Vol. 23, 2000.