EPRI

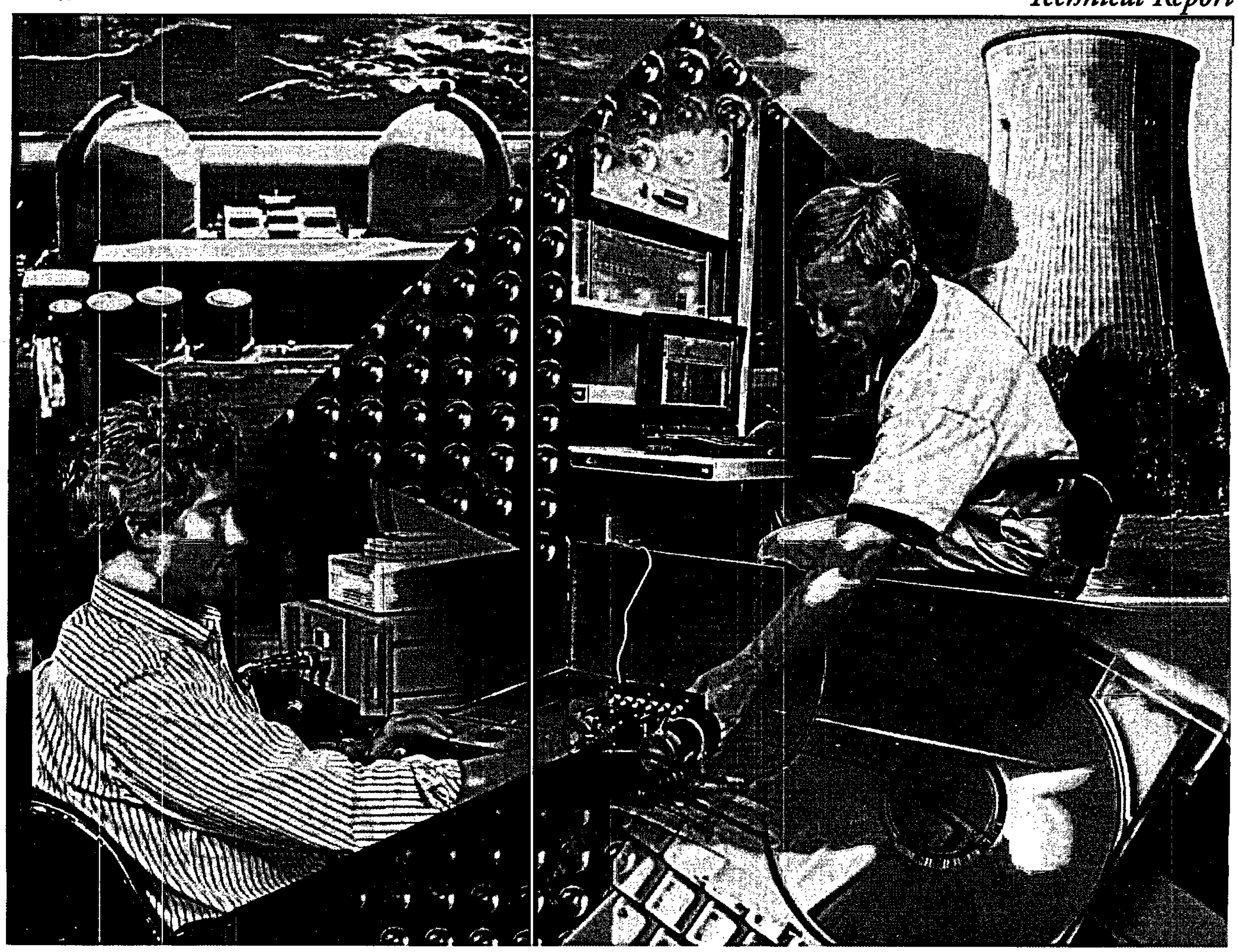
On-Line Monitoring of Instrument Channel Performance

Volume 1: Guidelines for Model Development and Implementation

LICENSED MATERIAL

WARNING:
Please read the License Agreement on the back cover before removing the wrapping material.

Technical Report



Instrumentation & Control
I&C

On-Line Monitoring of Instrument Channel Performance

Volume 1: Guidelines for Model Development and Implementation

1003361

Final Report, December 2004

EPRI Project Manager
R. Shankar

EPRI • 3412 Hillview Avenue, Palo Alto, California 94304 • PO Box 10412, Palo Alto, California 94303 • USA
800.313.3774 • 650.855.2121 • askepri@epri.com • www.epri.com.

DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITIES

ORGANIZATION(S) THAT PREPARED THIS DOCUMENT

EPRI

Edan Engineering Corp.

ORDERING INFORMATION

Requests for copies of this report should be directed to EPRI Orders and Conferences, 1355 Willow Way, Suite 278, Concord, CA 94520, (800) 313-3774, press 2 or internally x5379, (925) 609-9169, (925) 609-1310 (fax).

CITATIONS

This report was prepared by

Edan Engineering Corporation
900 Washington St., Suite 830
Vancouver, WA 98660

Principal Investigator
E. Davis

EPRI I&C Center
Kingston Fossil Plant
714 Swan Pond Road
Harriman, TN 37748

Principal Investigator
B. Rasmussen

This report describes research sponsored by EPRI.

The report is a corporate document that should be cited in the literature in the following manner:

On-Line Monitoring of Instrument Channel Performance, Volume 1: Guidelines for Model Development and Implementation, EPRI, Palo Alto, CA: 2004. 1003361.

REPORT SUMMARY

Background

The On-Line Monitoring (OLM) Group operates under the Instrumentation and Control (I&C) Nuclear Program. There are at present two programs under this group (separately funded under individual subscription) that address the needs of power plants with regard to instrument monitoring, instrument calibration reduction/extension, and sensor validation. The two groups are the Instrument Monitoring and Calibration (IMC) Users Group, formed in 2000, and the On-Line Monitoring Implementation Users Group, formed in 2001.

EPRI's strategic role in on-line monitoring is to facilitate its implementation and cost-effective use in numerous applications at power plants. EPRI has sponsored an on-line monitoring implementation project at multiple nuclear plants specifically intended to install and use on-line monitoring technology. The goal is to apply on-line monitoring to all types of power plant applications and to document all aspects of the implementation process in a series of EPRI deliverables. These deliverables will cover installation, modeling, optimization, and proven cost-benefit.

EPRI will continue to foster the development of on-line monitoring technology and its application via the IMC Users Group. Through this group, on-line monitoring as a key technology will continue to be supported technically as its use grows throughout the industry. The EPRI IMC Users Group will also continue to support generic technical issues (such as providing implementation guidance for calibration reduction of safety-related instrumentation) associated with on-line monitoring.

This report is the first in a three-volume set. *On-Line Monitoring of Instrument Channel Performance, Volume 2: Model Examples, Algorithm Details, and Reference Information* contains more detailed descriptions of the empirical modeling algorithms, specific examples and results of developed models, and further evaluations of the software used in this project. *On-Line Monitoring of Instrument Channel Performance, Volume 3: Applications to Nuclear Power Plant Technical Specification Instrumentation* provides an overview of how to extend calibration intervals by the use of on-line monitoring, describes the technical specification changes that are recommended to extend calibration intervals, addresses measurement and estimation uncertainty, provides guidance regarding on-line monitoring acceptance criteria, and addresses software verification and validation criteria for on-line monitoring applied to technical specification-related instruments.

Objectives

- To provide technical information regarding on-line monitoring as a calibration extension and performance-monitoring tool and to address data acquisition, data quantity, and data quality issues related to modeling
- To provide guidance regarding evaluating and responding to identified failures
- To provide an overview of the Multivariate State Estimation Technique (MSET) and the SureSense monitoring software
- To provide technical information describing software installation and setup for on-line monitoring and to address data management and interface issues
- To explain the steps and actions necessary to implement an on-line monitoring system and to list the steps and actions necessary to declare that a model is ready for use

Approach

This report provides detailed information regarding the application of on-line monitoring to nuclear plant instrument systems. The MSET is described because it has been the basis for the EPRI OLM implementation project from 2000–2003. Recent modifications to the SureSense software (supplied by Expert Microsystems, Inc. for use in this project) have introduced an alternative technique, the Expert State Estimation Engine (ESEE) with at least equivalent capabilities. The ESEE model was used for all model development and implementation work in 2004. Issues related to the implementation and use of on-line monitoring systems are presented in this report to enable users to assess plant-specific needs and the limitations of the techniques.

Results

Industry and EPRI experience at several plants has shown on-line monitoring to be very effective in identifying out-of-calibration instrument channels or potential equipment degradation problems. The results have been very encouraging. Substantial progress has been made over the course of this multiyear project.

EPRI Perspective

EPRI's strategic role in on-line monitoring is to facilitate its implementation and use in numerous applications at power plants. On-line monitoring of instrument channels provides increased information about the condition of monitored channels through accurate, more frequent evaluation of each channel's performance over time. This type of performance monitoring is a methodology that offers an alternative approach to traditional time-directed calibration. EPRI is committed to the development and implementation of on-line monitoring as a tool for extending calibration intervals and evaluating instrument performance.

Keywords

Calibration
Condition monitoring
Instrumentation and control
Maintenance
Nuclear plant operations and maintenance
Signal validation

ACKNOWLEDGEMENTS

EPRI, the EPRI I&C Center, and Edan Engineering recognize the following individuals for their contributions to this project. Their time and attention in support of this project are greatly appreciated.

Randy Bickford	Expert Microsystems, Inc.
David Carroll	South Carolina Electric and Gas
Pat Colgan	Exelon Corporation
Steve Dixon	Exelon Corporation
William Drendall	Amergen Energy
Dave Hooten	Carolina Power and Light
Jerry Humphreys	CANUS Corporation
Aaron Hussey	EPRI
Robert Kennedy	Exelon Corporation
Calvin C. King Jr.	Public Service Electric and Gas Co.
Vo Lee	Expert Microsystems, Inc.
Hubert Ley	Argonne National Laboratory
David Lillis	British Energy
Edwina Liu	Expert Microsystems, Inc.
Connie Love	Tennessee Valley Authority
Adrian Miron	Argonne National Laboratory
Karl Nesmith	Tennessee Valley Authority

Mike Norman	Tennessee Valley Authority
Ken Olenginski	Exelon Corporation
Steve Orme	British Energy
Keith Pierce	Public Service Electric and Gas Co.
Jeff Richardson	British Energy
Richard Rusaw	South Carolina Electric and Gas
Larry Straub	Exelon Corporation
Bill Turkett	South Carolina Electric and Gas
Tom Wei	Argonne National Laboratory
Bill Winters	Exelon Corporation
John Yacyshyn	Exelon Corporation
Chenggang Yu	Argonne National Laboratory
Nela Zavaljevski	Argonne National Laboratory
Jack Ziegler	Exelon Corporation

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1
INTRODUCTION

1.1 Report Purpose

This report discusses the actual implementation of on-line monitoring (OLM) to nuclear plant instrument systems. A considerable amount of application and development has been completed by the EPRI On-Line Monitoring Implementation Project. The experience gained has been documented here for the benefit of EPRI members.

This report, the first volume of a three-volume set, provides an overview of the EPRI on-line monitoring project activities and definitions for the majority of the terminology used to describe on-line monitoring and its implementations. The report discusses data management issues related to implementations and describes various modes of operation possible for on-line monitoring. Overviews of on-line monitoring and the software product used throughout this project are presented. This report is devoted mainly to presenting the various tasks that must be completed to prepare models for and implement an on-line monitoring system including data preparation, signal selection, model training and evaluation, model deployment, and model retraining. Data quality, data quantity, fault-detection techniques, and alarm response mechanisms are related issues that are also discussed. An extensive glossary of on-line monitoring terms is provided in Appendix A.

The second volume of this three-volume set, *On-Line Monitoring of Instrument Channel Performance, Volume 2: Model Examples, Algorithm Details, and Reference Information* [1], serves mainly as a reference to this first volume. It contains detailed descriptions of the Multivariate State Estimation Technique (MSET) and the instrument calibration and monitoring program for redundant channels. These two algorithms are discussed in detail because they were the primary tools used under the EPRI on-line monitoring projects. Numerous examples are presented for models that were developed for the participants of this project. Model maintenance (or retraining) is also demonstrated to illustrate the process of updating models when they require modifications to their training datasets. Finally, a recent software product developed specifically for cleaning data files and removing bad data prior to developing on-line monitoring models is reviewed and demonstrated.

The third volume, *On-Line Monitoring of Instrument Channel Performance, Volume 3: Applications to Nuclear Power Plant Technical Specification Instrumentation* [2], builds on the groundwork presented in the first two volumes and discusses on-line monitoring applications specifically for safety-related technical specification instrumentation at nuclear power plants. The report presents recommendations for the safety-related channels that are suitable for model deployment along with the related issue of single-point monitoring. The U.S. Nuclear Regulatory Commission's (NRC's) safety evaluation report [4], which reviews on-line monitoring for

nuclear power applications, is provided for reference. The results from a detailed uncertainty analysis performed on the MSET and an additional summary of previous results obtained for the instrument calibration and monitoring program for redundant sensors are also provided. Verification and validation studies of both MSET and the SureSense[1] on-line monitoring software are discussed along with a software acceptance test procedure for the MSET. Additional discussions are provided regarding redundant vs. nonredundant empirical modeling techniques as applied to safety-related instrumentation.

The objectives of this first report include the following:

- To provide technical information regarding on-line monitoring as a calibration extension and performance-monitoring tool and to address data acquisition, data quantity, and data quality issues related to modeling
- To provide guidance regarding evaluating and responding to identified failures
- To provide an overview of MSET and the SureSense monitoring software
- To provide technical information describing software installation and setup for on-line monitoring and to address data management and interface issues
- To explain the steps and actions necessary to implement an on-line monitoring system and to list the steps and actions necessary to declare that a model is ready for use

1.2 Report Applicability

This report addresses specific issues and considerations applicable to any on-line monitoring system. The EPRI on-line monitoring implementation project applied MSET to nuclear plant instrument systems. The modeling guidelines directly describe an MSET approach. Argonne National Laboratory (ANL) originally developed the MSET approach for nuclear power applications. The SureSense software contains an optional MSET toolkit that was selected for use by the EPRI On-Line Monitoring Implementation Project members. In some cases, SureSense software features are used to illustrate certain aspects related to modeling. In addition, the latest version of the SureSense software (Version 2.0) incorporates a proprietary empirical model of the Expert State Estimation Engine (ESEE). Version 2.0 of SureSense has been used for all applications and implementations in the last year. The guidelines presented here apply in general to both empirical models—MSET and the newer proprietary algorithm. The software can be utilized to easily convert models of one specific model type to the other.

This report is not intended to serve as a software user's guide. Instead, it addresses modeling issues at a higher level and rarely discusses specific model settings. The EPRI On-Line Monitoring Implementation Project has produced a separate report [3] that should be reviewed for SureSense-specific considerations. For other software systems, refer to the specific supplier's user's guide.

In summary, the modeling guidelines provided in this topical report apply directly to MSET and the examples have been illustrated with the SureSense software. However, many of the principles of modeling described here can be applied to other on-line monitoring methods. In

[1] SureSense is a trademark of Expert Microsystems, Inc.

particular, issues associated with data quality, model training, failure detection, and retraining apply to most empirical model-based on-line monitoring methods.

1.3 Report Audience

This report is primarily intended for instrumentation and control engineers and technicians at nuclear plants. The examples provided in this report apply directly to nuclear plant systems, but the concepts of on-line monitoring can be applied to virtually any application involving signal analysis and validation.

Readers of this report are assumed to have the following skills:

- A basic knowledge of nuclear plant protection and control instrumentation
- A general understanding of statistics and statistical analysis methods (although this report has been prepared in a manner that minimizes the in-depth discussion of underlying statistical methods)

The readers of this report are not expected to have detailed knowledge of on-line monitoring theory. Accordingly, this report attempts to maintain a balance between providing too much versus too little information or technical detail. Much of the information provided has not been previously assembled or published in a comprehensive and instructive format.

1.4 Considerations Before Starting Model Development

On-line monitoring is conceptually simple—send plant computer data to the software, and it will identify any drifting channels. The reality is that data management, software model configuration, and subsequent failure detection require some effort and knowledge. Before starting model development with an on-line monitoring software package, the following steps are recommended:

- Use a very good personal computer for model development. Managing and evaluating the amount of data recommended in this effort require substantially better computing ability than other tasks typically performed on the computers of most engineers. Beginning with an older computer with inadequate memory will be a frustrating experience. Everything possible should be done to ensure that the computer does not hinder the modeling and analysis effort.
- Test the data acquisition method that will be used by obtaining large data files containing 1-minute sample rate data. Historical data will be required to develop and test the model; these data are typically obtained from a data archive. Models cannot be developed or tested unless historical data are readily retrievable. Some nuclear plants might have outdated local area networks (LANs). Accessing data across a LAN can become the limiting part of data acquisition. Ensure that personnel from the computer services department are involved and understand the quantity of data that will be managed.

- Determine the intended purpose of on-line monitoring at your facility. On-line monitoring for calibration optimization might require a different set of signals than on-line monitoring for performance monitoring or equipment condition monitoring. Fault evaluation techniques also might differ depending on the purpose.
- Decide what models to develop. Review *On-Line Monitoring for Instrument Channel Preformance, Volume 2: Model Examples, Algorithm Details, and Reference Information* [1] for examples of the typical models developed by the EPRI On-Line Monitoring Implementation Project. For each model selected, determine which signals have accessible data archived by a computer system.
- Understand the importance of training a model for proper performance. Determine response methods to identify failures. The body of this report addresses these topics in detail. Initial training of the project team will be helpful.

The previously mentioned steps assume that the desired on-line monitoring software has been identified and obtained. Table 1-1 summarizes the recommended steps to take before starting an on-line monitoring program.

Table 1-1
On-Line Monitoring Implementation Preliminary Checklist

Item	Recommendation	Ready?
Computer Equipment		
1.	Obtain good computers for model development personnel.	
2.	Test the data acquisition method with sample data.	
3.	Verify that the data file storage location is acceptable.	
Determine Project Goals and Identify Models		
4.	Confirm the intended purpose and users of on-line monitoring.	
5.	Identify the models to be developed.	
6.	Set up a project plan with achievable milestones.	
Training		
7.	Train project personnel.	

1.5 On-Line Monitoring Overview

On-line monitoring is an automated method of monitoring instrument performance and assessing instrument calibration without disturbing the monitored channels while the plant is operating. In the simplest implementation, redundant channels are monitored by comparing each individual channel's indicated measurement to a calculated best estimate of the actual process value—referred to as the *parameter estimate* or *estimate*. By monitoring each channel's deviation from

the parameter estimate, an assessment of each channel's calibration status can be made. An on-line monitoring system can also be referred to as a *signal validation system* or *data validation system.*

Several different implementations of on-line monitoring for nuclear plant systems currently exist. Examples include the EPRI Instrument Calibration and Monitoring Program (ICMP), the ANL Multivariate State Estimation Technique (MSET), and the Organization for Economic Cooperation and Development (OECD) Halden Reactor Project PEANO (Process Evaluation and Analysis by Neural Operators). Some plants currently implement on-line monitoring in addition to their traditional calibration programs to provide additional performance assessment, troubleshooting, and maintenance planning capabilities.

Electricité de France (EDF) plants have received approval from the France Safety Authority to use on-line monitoring as a basis for extending calibration intervals. Additionally, the NRC has issued a safety evaluation report authorizing the application of on-line monitoring as a calibration extension tool [4].

Due to the ease with which data acquisition and analysis of instrument channel data can be performed, on-line monitoring of instrument channels is possible and practical. In essence, on-line monitoring provides a proactive and beneficial approach to performing periodic instrument surveillances. It accomplishes the surveillance or monitoring aspect of calibration by comparison between redundant or correlated instrument channels and with independent estimates of the plant parameter of interest. It does not replace the practice of instrument adjustments; instead, it provides a performance-based approach for determining when instrument adjustment is necessary as compared to a traditional time-directed calibration approach.

1.6 EPRI's Role in On-Line Monitoring

EPRI's strategic role in on-line monitoring is to facilitate its implementation and cost-effective use in numerous applications at power plants. To this end, EPRI has sponsored at multiple nuclear plants an on-line monitoring implementation project specifically intended to install and use on-line monitoring technology. The EPRI on-line monitoring implementation project serves two purposes:

- To apply on-line monitoring to all types of power plant applications
- To document all aspects of the implementation process in a series of EPRI deliverable reports

These reports cover installation, modeling, optimization, and proven cost-benefits. The following EPRI reports resulted from this project:

- *On-Line Monitoring of Instrument Channel Performance, Volume 1: Guidelines for Model Development and Implementation* (this report) addresses all aspects of modeling for on-line monitoring applications and their implementation. This report describes model development, data quality issues, training requirements, retraining criteria, responding to failure alarms, and declaring that a model is ready for use.
- *On-Line Monitoring of Instrument Channel Performance, Volume 2: Model Examples, Algorithm Details, and Reference Information* [1] presents detailed model examples, empirical algorithm details, and further evaluations of the software utilized during this project.
- *On-Line Monitoring of Instrument Channel Performance, Volume 3: Applications to Nuclear Power Plant Technical Specification Instrumentation* [2] addresses on-line monitoring for safety-related applications and the NRC's safety evaluation report [4] for on-line monitoring. Topics include technical specifications, uncertainty analysis, procedures and surveillances, MSET application considerations, and miscellaneous technical considerations. Nuclear Energy Plant Optimization (NEPO) projects related to software verification and validation and uncertainty analysis provide input to this report.
- *SureSense Diagnostic Monitoring Studio User's Guide, Version 2.0* [3] provides detailed guidance in the application of SureSense for nuclear plant systems. This report is updated periodically as a result of user feedback or software revisions.
- *On-Line Monitoring Cost-Benefit Guide* [6] discusses the expected costs and benefits of on-line monitoring. Direct, indirect, and potential benefits are covered. The project participants' experiences with on-line monitoring are included.

EPRI fosters development of on-line monitoring technology and its application via the Instrument Monitoring and Calibration (IMC) Users Group. Through this group, on-line monitoring will continue to be supported as a key technology as its use grows throughout the industry. Finally, the EPRI IMC Users Group will continue to support generic technical issues associated with on-line monitoring such as providing implementation guidance for calibration reduction of safety-related instrumentation.

1.7 Terminology Used in This Report

Appendix A provides a glossary of terms used in this report. Some terms require additional clarification in support of using this report. The following sections explain key terms.

1.7.1 Channel, Sensor, and Signal

The terms *channel*, *sensor*, and *signal* are often used almost interchangeably in this report, but there is an important distinction between the three terms. The sensor is the device that measures the process value. The sensor and associated signal conditioning equipment are referred to as the

instrument channel or channel. The electrical output from the channel is the signal. Figure 1-1 shows the relationship between the three terms for a safety-related channel. A non-safety-related channel might not have the isolator or bistable as shown.

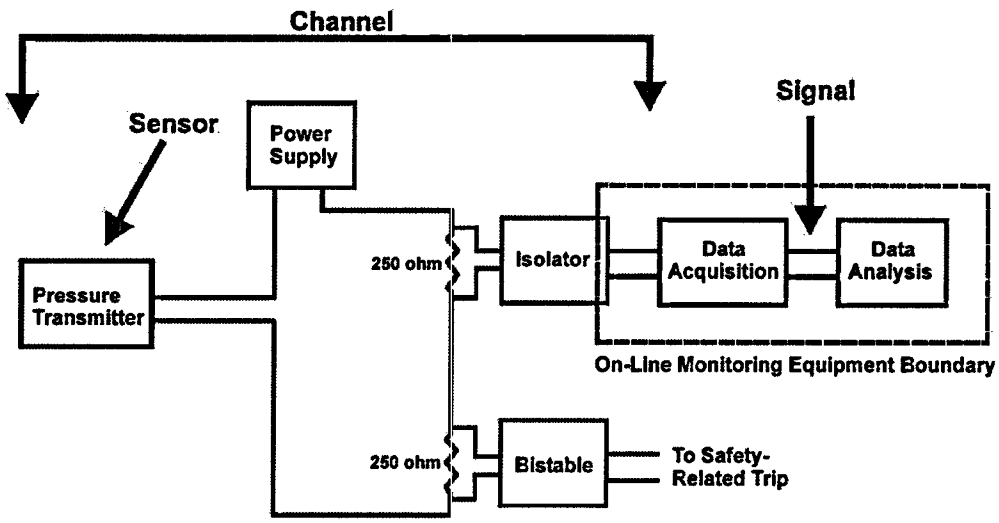


Figure 1-1
Instrument Channel in Terms of On-Line Monitoring

For a non-safety-related channel, there might be little in the way of signal conditioning because the sensor is the only real monitored device. More complex measurements might contain several signal conditioning modules. This report will usually refer to the channel rather than the sensor in terms of what is monitored. Although other industry documents and published papers often discuss on-line monitoring using the term *sensor*, it is the channel (or some portion of the channel) that is actually monitored. The discussion provided in the following sections will frequently refer to sensor drift because the sensor is usually the most common source of drift, but any portion of the channel might actually be the cause of drift.

The on-line monitoring system does not know the layout of the channel—it receives only a digitized signal from the plant computer or from a historical file. Although the instrument channel is typically producing a milliampere or voltage output, the signal acquired by the on-line monitoring system is often scaled into the expected process units such as pressure, temperature, or percent. When this report refers to signals, it means the scaled or unscaled digitized output signals from the monitored channels.

1.7.2 Modeling Terms

The term *model* is used to describe the group of signals that have been collected together for the purpose of signal validation and analysis. Depending on the context, model might refer to only the selected group of signals, or it might also include the various settings defined by the on-line monitoring system that are necessary to optimize the performance of the signal validation

procedure. In the context of on-line monitoring, *model* does not refer to some functional relationship between model elements defined by a set of equations.

The term *vector* is used to describe the observed values for all of the signals in the model at a particular instant in time. For example, if the signal data is contained in a spreadsheet, a single row of data for a particular time would be a vector.

The term *domain* is used to describe the operating states that form the basis for training a model. The domain contains a range for each signal in the model, and it also defines different operating states within that range. For example, a domain for a pressure sensor might cover a range of 800–1200 psig (5516–8274 kPa). Within this range, there might be several distinct operating states associated with different equipment lineups or plant power levels.

The term *estimate* is used to describe the best estimate or approximation of the actual process or sensor value calculated by the on-line monitoring system. The term *residual* refers to the mathematical difference between an observed value and the corresponding estimate for that observation. The residual is important because fault detection is often based on the behavior of the residual.

2
DATA MANAGEMENT

On-line monitoring is a data intensive process, and the amount of data to be managed should be understood in advance. Section 2 discusses various aspects of data-file management. This information applies to either a batch mode or an on-line mode of operation.

2.1 On-Line Monitoring System Architecture

On-line monitoring can be applied in various modes ranging from batch mode using data files to a real-time mode receiving a continuous data stream. The term *on-line* has not been well defined in the industry. Originally, the term *on-line* was used to indicate that signal validation was performed while the plant was operating at power, without regard to how the data were acquired. In most cases, the on-line monitoring method has actually been off-line in terms of data acquisition, meaning that data are accumulated in batch files for processing.

The analysis procedures within the monitoring system treat all signals as if they originate from an on-line data collection system. The on-line monitoring system is not concerned with the layout of the channel; it simply receives a digitized time series of signals from the plant computer, a historical file, or another data acquisition system. Although the instrument channel is typically producing a milliampere or voltage output, the signals acquired by the on-line monitoring system are often scaled into the expected process units such as pressure, temperature, or percent.

The EPRI publication *On-Line Monitoring of Instrument Channel Performance* [5], defines the following possible options for an on-line monitoring system:

- An automated system that performs data acquisition and analysis essentially continuously in real time at a specified sample rate
- An automated system that performs data acquisition and analysis at discrete specified intervals
- An automated system that is normally off and is manually activated to perform data acquisition and analysis at a set interval (at least quarterly)
- A manual system in which data is acquired manually on at least a quarterly interval and entered manually into a computer program for the purpose of analysis

The differences between each of the these options most often involve the degree of automation in the data acquisition step, including the method of data collection and the frequency of the data analysis. Most of these options actually operate in a batch mode in which data are accumulated in stored files. The differences in these options relate to the locations of the data files and to how

frequently data files are generated and evaluated. For typical nuclear plant applications, either option 2 or option 3 will usually be used.

The typical on-line monitoring system consists of the following building blocks:

- Separate off-line computer hardware on which the system resides.
- Communications hardware and software to obtain data from the plant process computer, plant data historian, or other source if the data are automatically acquired. Manual data acquisition is also possible.
- The on-line monitoring software that analyzes, displays, and archives the data and presents results interactively in graphs and reports.

2.1.1 Off-Line Batch Mode Using Historical Data

The term *batch mode* means that data files are stored in some location and are accessed by the on-line monitoring system. The term *off-line batch mode* applies if the data files are manually extracted from an archive or must be manually specified by the user. Depending on the frequency of data collection, batch mode might be used to evaluate the previous quarter, month, week, or day. Notice that a system operating in batch mode is evaluating specified data files that cover some period of time—batch mode is not receiving a real-time data stream.

Even if the intent is to apply on-line monitoring in a true on-line mode, certain modeling and analysis functions are generally best performed in a batch operation, including the following:

- **Training** – The data used for training require careful review and cleanup. Training data must be error-free and must properly characterize the normal operating states that will be monitored. The data used for training must be kept available for retraining if certain model settings are modified.
- **Model development** – Historical data files are used to evaluate model performance as part of the model development process. Some of the historical data is used for training, and some data are typically used for verification and performance assessment.
- **Look back** – After model development is complete and the model has been placed in service, it can be useful to review the historical performance of a signal or group of signals when evaluating current signal performance. It is often easier to maintain historical data in pre-configured files for ready access and comparison. Extracting data for an extended time period from a plant data historian or other archive can be time-consuming.

The effort to complete these functions for a model is about the same, regardless of the selected on-line monitoring approach—all models require historical data for training, retraining, model development, and historical performance analysis. The differences in the on-line monitoring mode of operation show up after the model has been developed, tested, and placed in service and relate primarily to how subsequent data are acquired and tested. In the case of an off-line batch model, the user must periodically extract data for each model, typically at monthly intervals,

involving a recurring maintenance cost. In its simplest implementation (in which no changes are made to the existing data archive software), extracting data typically involves the following steps:

- Extract the previous month of data into a separate file for each model.
- If the data are extracted in text format, optionally convert each extracted file into a binary format for improved performance during analysis. Required formats are specific to the on-line monitoring software application used.
- Link the data file to the model and run the model.

If there are many models, this can be a time-consuming process. Therefore, some level of automation is preferable. The next section describes an automated approach to data acquisition and file management.

2.1.2 On-Line Batch Mode Using Current Data

All users initially start with an off-line batch mode of operation as they develop models and learn how to use the on-line monitoring system software. As the models are placed in service, the transition from off-line to on-line mode of operation should be considered. Figure 2-1 shows a typical system architecture for a true on-line monitoring system, which includes a more efficient approach to data management.

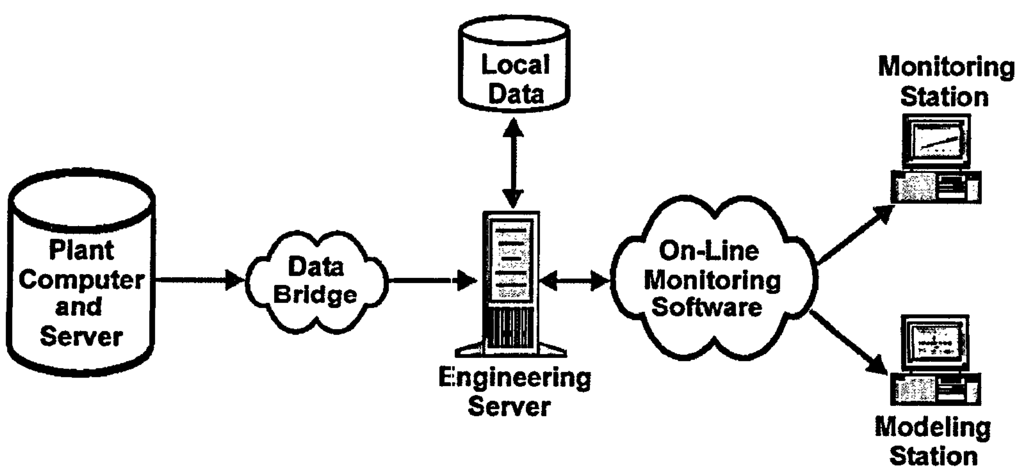


Figure 2-1
On-Line Monitoring System

The system shown in Figure 2-1 operates as follows:

- No changes are made to the plant computer software and its associated data historian. Plant process data are acquired from the plant computer and archived just as they were prior to installing an on-line monitoring system.
- Data are acquired from the data historian using an extraction routine (referred to in Figure 2-1 as a data bridge) that might be unique to the plant-specific computer configuration. At periodic intervals, data are downloaded from the data historian and stored in the appropriate format for each model. For example, an on-line monitoring system might acquire the previous day's data for each model and store it in a designated location.
- The on-line monitoring software combines the new data with previously acquired data, automatically runs each model with the new data included, and stores the run results. The latest run results are available on demand by plant personnel.

In terms of power plant requirements, this description represents a true on-line monitoring system. It should be noted that the models are still technically operating in a batch mode because the models run on historical files that include up to the previous day's data. This is not real time, but it is close enough for most purposes (especially for calibration reduction and performance monitoring). The key differences between this periodic on-line approach and the off-line batch mode described in Section 2.1.1 occur in the following automated steps:

- Data acquisition – An engineer or technician does not manually extract data files.
- Data file formatting – File conversions are handled automatically.
- Data file linking – The model automatically links to the latest dataset.
- Model run – The model runs automatically and stores the run results in a specified location for ready retrieval by any user.

Making the transition from off-line batch mode to on-line mode usually requires some unique programming that can be provided by either the software supplier or by plant personnel. The benefits of on-line mode are considered important enough that it is the recommended approach for most users.

2.1.3 Real-Time Mode

Real-time on-line monitoring is unnecessary for most, if not all, nuclear plant applications, especially considering the dedicated computing overhead required to operate numerous models simultaneously in a real-time environment. The NASA space shuttle is one example of a true real-time on-line monitoring application where the software used in this project has been employed. In this case, a continuous data stream is sampled and processed through the signal validation software throughout the period from just before launch until the completion of launch. For the few minutes that it takes to reach orbit, a large quantity of data is evaluated as it is received to ensure that the mission critical sensors are providing valid data to the flight computers.

Nuclear plant requirements are quite different from the space shuttle in terms of the requirements for data acquisition. Rather than operating for only a few minutes during launch, a power plant might operate at near 100-percent power for an entire operating cycle of 18–24 months. During this extended period of operation, many sensors will experience only small process changes, essentially monitoring the process about a single point. Setting up on-line monitoring software to operate in a real-time mode can be done, but it involves an additional level of complexity that is not needed for typical plant applications because the software's output will not be used in real time.

2.2 Data File Naming

When on-line monitoring is applied to its fullest potential, there will be many data files stored in the computer system and linked to the various models. When many data files are stored in a common location, the file-naming convention becomes important. The following naming convention works well and is recommended:

Unit Number (if needed) – Model Name – Year – Month – Portion of Month

Unit number refers to the power plant designation (or designation for some monitored asset). Model name refers to the specific diagnostic software model used to monitor some asset or system. The dates refer to the period of operation during which the data were acquired. The following three examples illustrate typical naming conventions:

- Unit 1 Vessel Level-2002-01
- RPS-2001-10
- OTSG-A-2001-12

If this naming convention is used, all files for a single model will be located as a group within a directory, sorted by year and month. Missing months or time periods are easily recognized. If data cover portions of a month (such as the first 10 days in one file, the next 10 days in a second file, and the last 10 days in a third file), it is recommended that the three files be combined into a single file for the month.

Notice that this naming convention implies that the data for all signals in a given model will be acquired collectively as a group. This is the preferred approach. Some commercial software packages such as SureSense readily support separating signal data. However, separating signal data for a single model for a specified period of time into multiple files complicates data cleanup and review. This can limit the applicability of the data for use with other software applications. Also, the removal of bad data within separate files must be carefully managed to prevent files with time-stamp simultaneity inconsistencies.

It should also be noted that when using this naming convention, that data for several models are not contained in a single file. For example, it might be easier during data retrieval to extract the data for five models at once. It is strongly recommended that a single data file contain the data for only a single model. While combined data files are readily supported in some commercial software packages such as SureSense, combining the data can actually complicate other aspects

of data management. For example, suppose the data for six models, each containing 25 signals, are simultaneously extracted and stored in a single file. If data are acquired at a 1-minute rate, this file will contain up to 44,600 rows of data for 150 columns of signals—a total of 6,690,000 data points. Files will likely be in a spreadsheet format to allow review and editing as necessary. The size of each file will be about 60 megabytes, which is time consuming to manage and difficult to review. These files are also treated as text files by spreadsheet programs. Text files take longer to read because each data access has to read through the entire file to read the data of interest. In addition, when sorting through historical data files, there are often cases where certain signal data are not valid. When reviewing a large data set for all models, simultaneously, removing periods of data due to specific invalid signal values will result in the simultaneous removal of these data from all other signals regardless of their quality. Based on the experience to date, each file should be kept as small and simple as possible.

2.3 Data Storage Format

On-line monitoring is a data-intensive activity. The overall objective is to acquire plant data and to process the acquired data to determine whether it is indicative of a healthy, normal state of the system or, alternatively, of some degraded or unhealthy state of the system. Power plants are characterized by large numbers of subsystems and signals, many of which are important targets of on-line monitoring. Data acquisition rates for these large plants are reasonably high. As a result, very large data sets will be managed by either the on-line monitoring system or its operators. In most cases, a high level of on-line monitoring system automation will be preferred.

When implementing an on-line monitoring system, plant-specific data issues must be addressed and resolved. One such issue is the format of the acquired data. Computers preferentially process data in binary format, while human operators prefer text or graphical formats.

Human operators often process measurement data in the form of text. As a result, many plants have established data extraction utilities to extract operating data in a text-based format, often to ASCII text files. The signal data are typically acquired by the plant computer, stored in a data archive, and extracted on demand by a user as shown in Figure 2-2. These existing procedures will typically provide the initial data extraction capability for on-line monitoring. What is often required is a final automation step to perform the extraction from the data archive. Some on-line monitoring systems such as SureSense will provide a ready means to implement this data archive connection either on-demand by the user or via a regularly scheduled data extraction service program (also known as a data bridge in the case of the SureSense software).

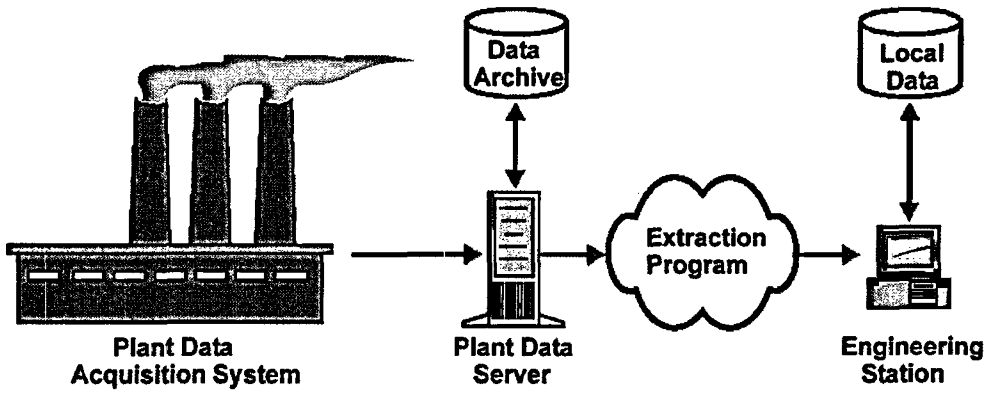


Figure 2-2
Typical Plant Data Archiving and Retrieval System

Although text files are readable by humans, they are extremely inefficient for use with a computer program. The program must read each character individually and interpret the series of characters in order to determine the number represented. Further, the space required for each value is unknown until the value has been read. This reduces file access speed when attempting to read data in the middle of the file. In contrast, computers can read binary format files with greater efficiency and speed and can readily access any data value in the file immediately and directly. Therefore, it is desirable to store plant data in a binary format for efficient use by the on-line monitoring system.

The SureSense software used by the EPRI On-Line Monitoring Implementation Project provides a general-purpose data interface that can be readily adapted to any input data format. It can be used to convert a set of data in a plant-specific format into one of several standard formats that can be readily shared between the project participants without further translation or processing.

It is recommended that initial implementation work be performed using one of these standard formats. After developing an initial set of models, a plant-specific data acquisition interface can be implemented if supported by the capabilities of the user's on-line monitoring software. The SureSense software uses a plug-in data reader capability to enable a highly adaptable and low-cost interface to any plant data system. This interface is discussed in Section 9.3 of this report, which describes the data bridge implementation.

Most users will already have text format data available from their plant data system. One common format for text-based data is the Microsoft Excel comma delimited file (CSV) format. An advantage to this format is that Microsoft Excel can be used to examine and characterize the data. The SureSense software includes a plug-in data reader for CSV files. SureSense users should consult the user's manual for further details.

Standards for binary format files are typically specific to the industry and the software. The EPRI On-Line Monitoring Implementation Project has selected a simple nonproprietary format known as the signal data file (SDF) format. SDF files provide fast and efficient data access. The SureSense software comes with a plug-in data reader for the SDF file format. SureSense users should consult the user's manual for further details.

2.4 Data File Configuration Management

Data configuration management is an important element of any on-line monitoring implementation program. The following items require configuration management for an effective on-line monitoring implementation in a nuclear power plant environment:

- On-line monitoring model definition files, including any supporting phase-determiner plug-ins and time-formatter plug-ins
- On-line monitoring model training data files, including any supporting data-reader plug-ins.
- On-line monitoring software acceptance test files and acceptance test results

On-line monitoring configuration management should follow procedures currently in place for document and data management at the plant.

3 OVERVIEW OF THE MULTIVARIATE STATE ESTIMATION TECHNIQUE, SURESENSE ON-LINE MONITORING SOFTWARE, AND MODEL DEVELOPMENT

The EPRI on-line monitoring implementation project selected MSET as the base algorithm to use for the on-line monitoring project. Argonne National Laboratory (ANL) developed MSET for nuclear power plant applications. This section provides a brief overview of MSET and a description of the software used during this project that contains a commercial implementation of MSET. Volume 2 of this series of reports [1] contains more detailed information describing the mathematical basis of MSET. Volume 3 [2] provides additional information regarding the verification and validation of both MSET and the SureSense software. MSET is an empirical modeling program that can be used for on-line monitoring. Because it is not a user-friendly product by itself, it requires a commercial implementation that provides all of the necessary user interfaces, graphic capabilities, and data management aspects. During the course of this project, Expert Microsystems, Inc., provided a commercial implementation with the SureSense software product.

It is important to note that there are alternative methods that can be used for on-line monitoring; however, to review all of these methods is beyond the scope of this report. The focus of this project was implementation. With that focus in mind, a method was selected and initially reviewed. The greater effort then followed to understand on-line monitoring as a whole and the implementation of systems at participating plant sites.

3.1 Overview

MSET is a software-based tool for on-line monitoring that was specifically developed by ANL for nuclear power applications. MSET is general in scope and can also improve performance, prevent downtime, and reduce operating costs for fossil power plants and many other industrial applications. MSET is a statistical modeling technique that learns a high-fidelity model of a process or apparatus from a sample of its normal operating data. Once built, the model produces an accurate estimate for each observed signal, given a new data observation from the process or equipment. Each estimated signal is compared to its actual signal counterpart using a highly sensitive fault-detection procedure to statistically determine whether the actual signal agrees with the learned model. MSET produces estimates for all signals included in the model.

To utilize MSET, the user starts by collecting sensor-generated data from the process under consideration that bound all normally expected operational states. These data are used by the

MSET system to establish the domain of normal process operation (trains MSET is trained to recognize normal behavior) and are used in the monitoring phase to identify abnormal behavior. During monitoring, sensor data are read by MSET. An estimate of the current state of the process is determined by comparing the measured sensor data with that obtained during training. The difference between this estimate and the measurement is calculated. This difference or residual error is then analyzed by a statistically based hypothesis test that determines if the process is operating normally or abnormally. If an abnormal condition is detected, the initial diagnostic step identifies the cause as either a degraded sensor or an operational change in the process. When a degraded sensor is identified, MSET uses the estimated value of the signal from this sensor to provide a highly accurate virtual sensor that can be used to replace the function of the faulted sensor in the MSET estimation process.

3.2 MSET Software Functions

MSET consists of 1) a pattern recognition system that provides empirically estimated values of all monitored signals and 2) a statistically based hypothesis test that compares the estimated signal values with the measured values to detect the development of incipient faults. MSET consists of three essential modules and a number of supporting modules. The essential modules are the following:

- A training algorithm for selecting and characterizing a subset of representative data from sensors during normal operation of the system. The training module is used to produce a training matrix of operating data that ideally encompasses all expected normal operating states of the system.
- A system state estimation module for parameter estimation. This module is used to calculate estimates for all signals in the model.
- A statistically based fault-detection algorithm. This module is used to detect abnormal disturbances in the monitored signals by examination of the difference between the estimated and the measured signal values.

MSET provides a high-fidelity estimate of the expected response of an asset's data signals by using advanced pattern recognition techniques to measure the similarity between the signals within a learned domain of operation. The learned patterns or relationships among the signals (that is, the training data) are used to identify the operating state that most closely corresponds with the current measured set of signals. By quantifying the relationship between the current and the learned states, MSET estimates the current expected response of the signals.

The difference between a signal's estimated value and its measured value is used as the indicator for sensor and equipment faults. The sequential probability ratio test (SPRT) technique provides ANL's basis for detecting statistical changes in the sensor signals at the earliest possible time, including usable information regarding the type and location of a disturbance. The SPRT technique provides a superior surveillance tool because it is sensitive not only to disturbances in signal mean, but also to very subtle changes in the statistical quality (variance, skewness, and bias) of the monitored signals. Instead of threshold or control limits, the SPRT technique utilizes user-specified false-alarm and missed-alarm probabilities, allowing the user to control the likelihood of missed or false alarms. For sudden gross failures of sensors or system components,

the SPRT can annunciate the disturbance as fast as a conventional threshold limit check. However, for slow degradation that evolves over a long time period (such as gradual decalibration in a sensor, wear-out or buildup of a radial rub in rotating machinery, loss-of-time constant degradation in a pressure transmitter, or change-of-gain failure without a change in signal mean), the SPRT can indicate the onset of a disturbance long before it would be apparent by a visual inspection of the strip chart or CRT signal traces and well before conventional threshold limit checks would be tripped.

3.3 Functional Overview and General Capabilities of SureSense

The SureSense software provides the option of a commercial implementation of the MSET software licensed from ANL for use in the power industry. SureSense provides numerous additional capabilities that build upon the ANL algorithms to significantly enhance the usability and diagnostic performance of the MSET procedures. The EPRI on-line monitoring implementation project is using SureSense for its on-line monitoring efforts.

The SureSense Diagnostic Monitoring Studio automates the production of application-specific software modules that reliably detect signal data faults and equipment malfunctions. These real-time capable on-line diagnostic modules enable improved safety, reduced operations and maintenance costs, and optimal performance for a wide range of systems and processes. SureSense diagnostic monitoring is applicable to any process monitoring system where time-critical functions depend on sensor input or where unexpected process interruptions due to sensor and equipment failures or false alarms are unsafe or uneconomical.

A signal fault is defined as any failure in the data path that corrupts the data signal, thereby providing erroneous information to the process monitoring system. A signal validation module will also detect abnormal operating conditions for the monitored process or equipment. SureSense uses several advanced predictive modeling techniques that calibrate a high-fidelity model of a process or apparatus from a sample of its normal operating data. Once built, the models provide an accurate estimate for each observed signal, given a new data observation from the process or equipment. Each estimated or virtual signal is compared to its actual signal counterpart using a highly sensitive fault-detection procedure to determine statistically whether the actual signal agrees with the calibrated model. Inconsistencies, if any, between the observed signals and their corresponding estimates are used to detect a wide variety of equipment problems (as well as to verify that such problems are not present).

The SureSense software automates the production of application-specific signal validation and equipment condition monitoring modules. The software user's guide [3] describes the operation and use of the development environment. The development environment is used to design a diagnostic monitoring model, to perform automated model training from historical operating data, and to verify and evaluate the resulting model. The development environment can then be used directly for on-line process surveillance or the software can be configured for inclusion in another process monitoring application.

In a process monitoring environment, the validation algorithm samples on-line signal data values and uses these observed values as the input to a parameter estimation module. The parameter estimation module produces an estimate of the signal value for each observed signal value. The

difference between the observed and estimated signal values provides a residual error value. The fault-detection procedure uses an advanced statistical technique to determine whether the residual error value is uncharacteristic of the process and, thereby, indicative of a signal or process fault. Finally, a fault decision is made using a conditional probability analysis of a series of fault-detection results in order to reduce the potential for single observation false alarms. Figure 3-1 illustrates the SureSense estimation and fault-detection procedure.

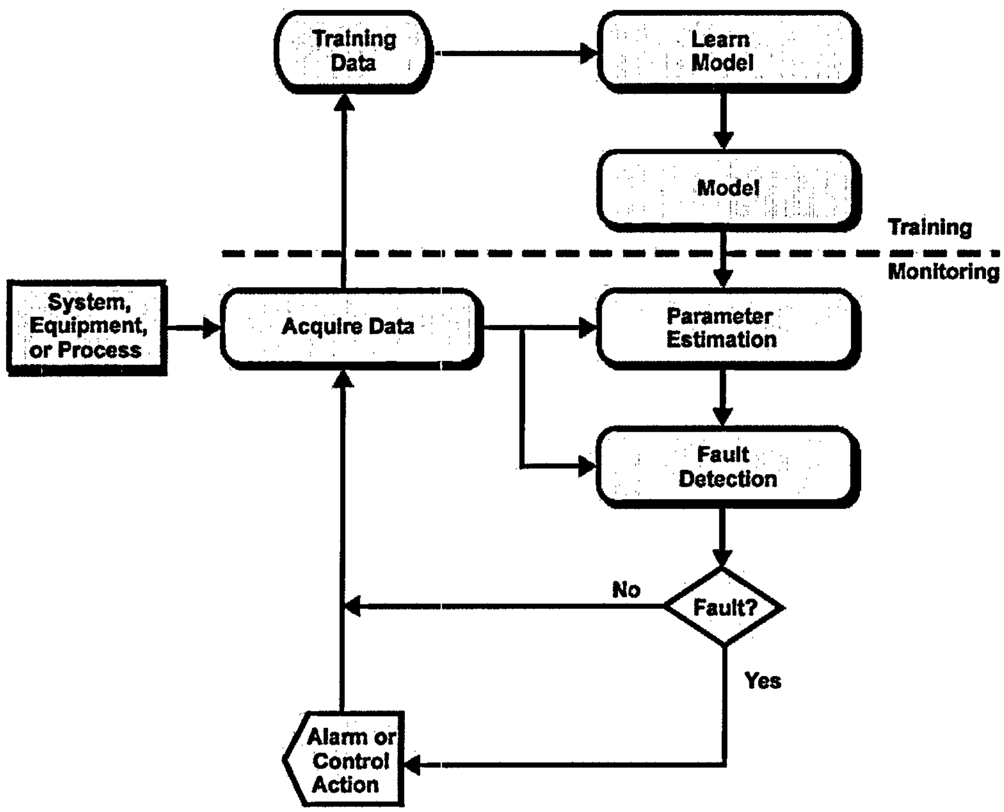


Figure 3-1
SureSense Operation

3.4 Model Development Overview

Model is a general term used to describe a group of signals that have been collected for the purpose of signal validation and analysis. Typically, the model will contain the information necessary to estimate the expected state of the monitored asset or system, given an observation of its signals. The model includes the specifications for acquiring the incoming data as well as the various settings necessary to optimize the performance of the signal validation and predictive condition monitoring procedures. The model includes control settings determining how the model is trained to estimate the expected behavior and the individual signal settings for the purpose of identifying abnormal signal behavior.

Figure 3-2 illustrates one view of model development, which includes the following basic steps:

- Select the parameters to evaluate as a group, and confirm adequate correlation between the selected parameters.
- Acquire training and verification data. Ensure that training data are free of abnormal signals and abnormal operating states. Remove bad data as necessary.
- Select estimation and fault-detection settings for initial analysis.
- Evaluate the model using the training and verification data. Adjust the estimation and fault-detection settings as needed to optimize performance. Acquire additional training data, if needed, to bound the operating space.
- Evaluate and test fault-detection settings in detail. Evaluate model sensitivity to false alarms and missed failure detection.
- Continue testing new data as the data are acquired. Evaluate the calibration status of each validated signal.

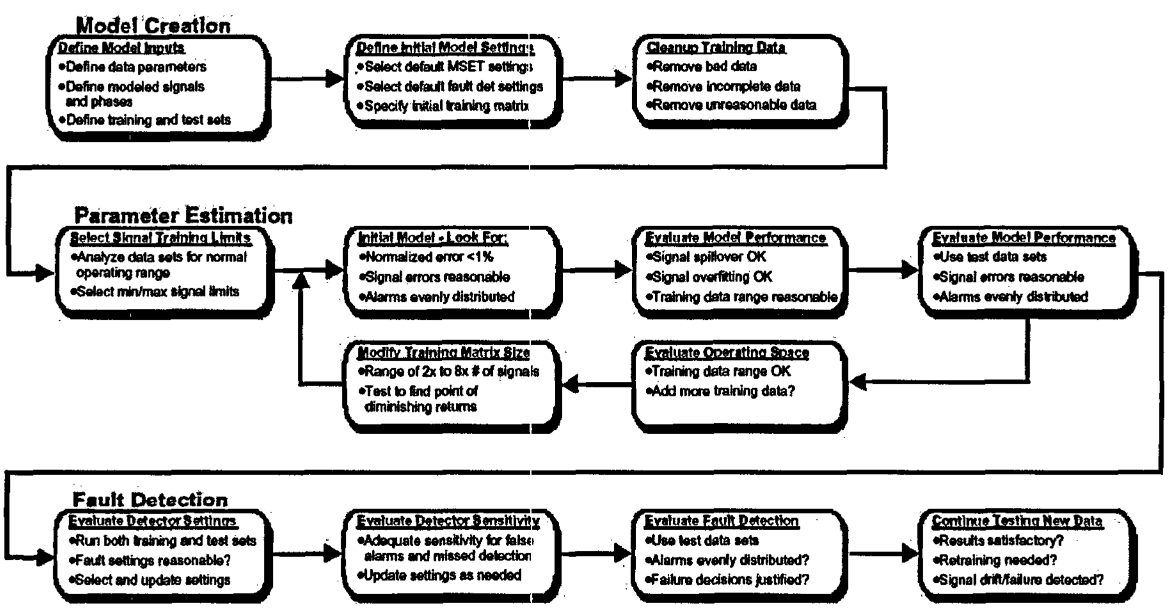


Figure 3-2
Model Development Overview

4 ON-LINE MONITORING PROCEDURES

Section 4 provides general on-line monitoring procedures to assist with the plant-specific on-line monitoring implementation process. These procedures should be used as guidance for items to consider during plant-specific procedure development.

4.1 Software Use

4.1.1 Procedure

On-line monitoring software tends to be quite complex. The underlying algorithms can be especially difficult to understand. For this reason, the software user's manual should provide adequate background information to facilitate a basic understanding of the principles of operation as well as detailed user's instructions. The *SureSense Diagnostic Monitoring Studio User's Guide* [3] was developed in support of the EPRI On-Line Monitoring Implementation project and has been periodically updated as required to reflect software revisions or users' requests for the duration of this project. This user's manual has been designed to serve as the software use procedure at a nuclear plant.

4.1.2 Personnel Training

The software used for on-line monitoring typically implements a complex set of algorithms. The principles behind modeling and software use are not intuitive to most users. Furthermore, the application of on-line monitoring ultimately affects the on-site instrument calibration program, which also requires some training and preparation of plant personnel. The plant should conduct various training classes as part of the implementation process. Typical training classes include:

- Detailed software use training for those personnel who will develop and maintain models
- General software use training for those personnel who might review on-line monitoring system results
- I&C training to explain how on-line monitoring affects the instrument calibration program, including how to assess the need for calibration and how to evaluate alarms as they occur

4.2 Model Development and Evaluation Procedures

The following report sections provide guidance regarding how to develop, maintain, and evaluate the in-service models.

4.2.1 Model Documentation

In the context used here, the term *model* refers to the following items:

- The selected signals that have been grouped for the purpose of signal validation and analysis
- The various settings defined by the on-line monitoring method that are necessary to optimize the performance of the signal validation procedure
- The data used for training, including any filtering of the data

The term *model documentation* refers to the configuration management of the completed model. For each model, the following items should be documented:

- The file name and file date of the completed model as approved for use, including any revision information.
- A list of the instrument channels covered by the model, including which channels have been designated as calibrate on demand, based on the model run results by the instrument calibration program.
- Model settings that form the basis for model operation. These model settings include the definition of signal, data set, phase determiner, and estimation settings.
- The file name and date of any plug-ins used by the model.
- Training files and file dates for those files that form the training basis of the model.

4.2.2 Periodic Model Evaluation

After a model has been placed in service, plant personnel will evaluate it periodically. This evaluation is intended to accomplish the following objectives:

- Confirm that the instrument channels included in the model are performing properly
- Identify any channels that appear to have drifted beyond acceptable limits
- Identify any equipment condition monitoring concerns, depending on whether the model was designed to function as a condition monitoring tool
- Determine whether the model appears to be adequately trained for the most recent data
- Confirm that model settings continue to be adequate for the model and plant operating state

4.2.2.1 Example Model Evaluation Procedure

The following provides a typical procedure for periodic model evaluation. This procedure is based on a single model and the instrument channels evaluated by that model. The model used to illustrate the procedure (HP FW HEATERS) contains various instrument channels in the high-pressure feedwater heater system.

1.0 Purpose

1.1 To confirm that instrument channels included within the scope of the on-line monitoring program are operating within acceptable limits

1.2 To verify that the on-line monitoring system for the evaluated model is operating normally and does not require modification to the model settings

The procedure applies to the HP FW HEATERS model, which contains the instrument channels in the high-pressure feedwater heater system as shown in Table 4-1.

Table 4-1
Instrument Channel Description

Instrument Channel Description	Tag Number	Computer Point
MFW Pump A Pressure	PT 3-66	P2214A
MFW Pump B Pressure	PT 3-80	P2215A
MFW Pump A Outlet Temperature	TE 3-68	T2362A
MFW Pump B Outlet Temperature	TE 3-82	T2363A
MFW Pump Outlet Header Temperature	TE 3-2	T2364A
MFW Pump A Outlet Flow	FT 3-70	F2250A
MFW Pump B Outlet Flow	FT 3-84	F2251A
FW Heater A1 Inlet Temperature	TE 3-6	T2240A
FW Heater B1 Inlet Temperature	TE 3-8	T2260A
FW Heater C1 Inlet Temperature	TE 3-16	T2241A
FW Heater A1 Outlet Temperature	TE 3-18	T2261A
FW Heater B1 Outlet Temperature	TE 3-26	T2242A
FW Heater C1 Outlet Temperature	TE 3-28	T2262A
FW Heater 1 Outlet Pressure	PT 3-34	P2273A
SG 1A Inlet Flow	FT 3-35A	F0403A
SG 1B Inlet Flow	FT 3-35B	F0404A
SG 2A Inlet Flow	FT 3-48A	F0423A
SG 2B Inlet Flow	FT 3-48B	F0424A
SG 3A Inlet Flow	FT 3-90A	F0443A
SG 3B Inlet Flow	FT 3-90B	F0444A
SG 4A Inlet Flow	FT 3-103A	F0463A
SG 4B Inlet Flow	FT 3-103B	F0464A
SG 1 Inlet Temperature	TE 3-36	T0418A
SG 2 Inlet Temperature	TE 3-49	T0438A
SG 3 Inlet Temperature	TE 3-91	T0458A
SG 4 Inlet Temperature	TE 3-104	T0478A
SG 1 Inlet Pressure	PT 3-37	P0403A
SG 2 Inlet Pressure	PT 3-50	P0423A
SG 3 Inlet Pressure	PT 3-92	P0443A
SG 4 Inlet Pressure	PT 3-105	P0463A

2.0 Plant Status

2.1 Normally operating

Note that the phase determiner used to partition the model into submodels is often defined based on reactor power level. If the plant is shut down or operating at a low power level, signal validation might not be performed while in a state excluded by a phase determiner.

3.0 Prerequisites

3.1 The on-line monitoring system software is operational.

3.2 The test data for the evaluated model are available. The test data typically contain plant operating data for the most recent time period.

Note that the period of time evaluated by this procedure is plant specific. Some plants might choose to operate in an on-line mode in which data are available on a near real-time basis. Other plants might choose to operate in a batch mode and evaluate the last quarter, last month, or some other period of data.

3.3 The performer of this procedure has been trained to use the on-line monitoring software and evaluate the test results.

4.0 Procedure

4.1 Start the on-line monitoring software and open the HP FW HEATERS model.

4.2 Ensure that the model used for evaluation is the approved model. Verify the following initial conditions:

- The model file name, date, size, and revision number are correct.
- The model is trained for use and has been trained on the file(s) specified by the model documentation.
- The model settings are correct in accordance with the model documentation.
- Acceptance limits have been specified for the instrument channels in the model.

4.3 Locate the test file containing the most recent data. If necessary, link this file to the model, and run the model using this latest test data.

Note that, depending on the plant method of on-line monitoring implementation, the file might be a manually acquired batch file, an automatically acquired batch file, or an automatically acquired and run file.

4.4 Upon completion of the monitoring run, review the run results. Identify any channels that were identified as failed during the run.

4.5 Review the observation estimate plot and the residual plot for each channel identified as failed during the run. Classify the identified failure into one of the following categories:

- The channel has drifted beyond acceptable limits. A channel recalibration will be necessary.
- The channel shows evidence of some drift, but a review of the residual plot confirms that the drift is not significant. A channel recalibration should not be necessary.
- Alarms were generated because of a plant or system operating transient for which the model was inadequately trained. If channel performance is acceptable before and after the transient, recalibration should not be necessary.
- The model is not adequately trained for the plant operating state. Model settings or model retraining with additional data might be necessary.

Note that Section 8 provides for additional guidance regarding alarm assessment. It is assumed in this procedure that the performer has been trained to recognize when model settings require adjustment.

4.6 Initiate a work order to recalibrate any instrument channels that have drifted beyond acceptable limits.

4.7 Initiate a request to update the model if alarms were generated because of inadequate training for the plant or system-operating state.

5
SIGNAL SELECTION

Section 5 describes the starting point for model development—signal selection. Technical issues associated with signal selection are discussed, including the criteria for deciding how large a model should be. *On-Line Monitoring of Instrument Channel Performance, Volume 2* [1] provides numerous examples of models that have been developed using the recommended criteria.

5.1 Signal Selection - Where to Start

5.1.1 Signals as Part of a Model

The term *model* refers to the procedural settings in combination with the selected signals that are collectively evaluated and validated as a group. Figure 5-1 shows a typical on-line monitoring (OLM) group model of a pressurized water reactor (PWR) steam system as displayed by the SureSense software.

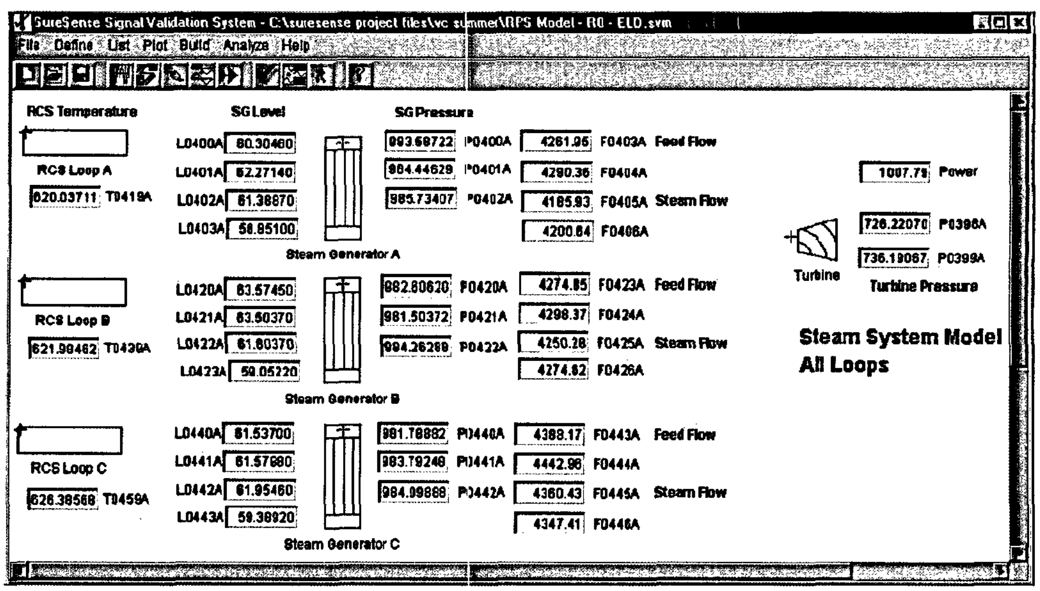


Figure 5-1
Typical SureSense Model

The SureSense model shown in Figure 5-1 includes the following signals:

- Reactor power (used to partition the model into submodels based on power level)
- RCS hot-leg temperature
- Steam generator level
- Steam pressure
- Steam flow and feedwater flow
- Turbine impulse pressure

Figure 5-2 shows the location of these typical signals in the steam system.

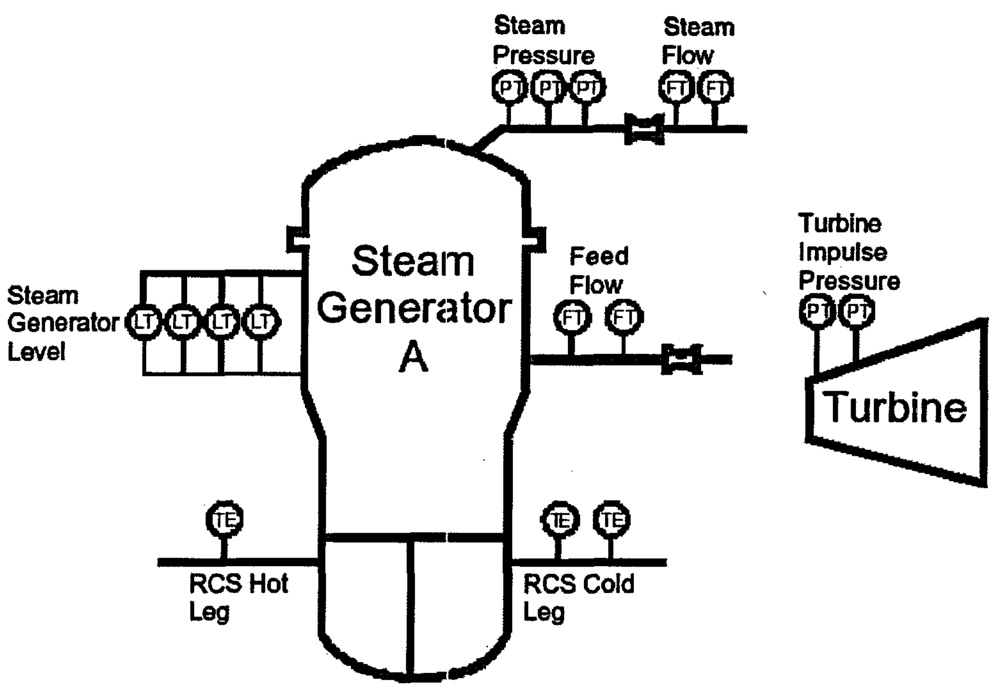


Figure 5-2
Steam System Signals Modeled for One Steam Generator

5.1.2 Where to Start in Signal Selection for a Model

The MSET estimation procedure presumes a moderate to high level of correlation between signals included in the model. This is intuitive when one considers that the MSET estimates are based on the information provided by the signals in the model. Therefore, information from related signals is required. Nonrelated signals will have independent or mildly dependent variations and cannot be monitored accurately with MSET. The MSET training step includes a procedure to build a training matrix from the historical operating data that characterizes the

operating space. In the MSET monitoring step, the training matrix is used to generate estimates based on a weighted combination of the reference patterns most closely matching each new observation.

In cases such as the typical steam system model shown in Figure 5-1, most signals have a strong correlation. Some signals (such as steam generator level) are strongly correlated as a group but might have little or no correlation to the other signals depending on the plant design. As a model is developed, the concept of correlation should be specifically considered. If all signals in a model have little or no correlation to one another, the selected training vectors might model only noise in the signals rather than some actual correlated process relationship or pattern. This will be the limiting consideration when the model does not contain redundant signals.

The first models developed should be simple ones such as the typical steam system model illustrated here. This recommendation is provided for the following reasons:

- The first models will test the data acquisition method and the quality of the retrieved data. Based on the experience to date, this rarely goes smoothly the first time.
- When developing the first model, the user is simultaneously learning several complex parts of model development including how the software works, the problems with data acquisition, how to review large amounts of data for acceptable quality, how process parameters vary over time, how to train and retrain the model, and how to evaluate identified failures. The first models should be simple; otherwise, the model's complexity might further confuse the overall progress.
- Steam system models tend to contain redundant signals for most, if not all, parameters. Data quality issues are more readily identifiable if redundant signals are available for direct comparison.

Refer to the reference list (Section 11) for publications listing examples of numerous models that have been developed. These examples best illustrate the approach to signal selection. Locate comparable instruments for a given model on the plant's piping and instrument drawings, and confirm that the selected instruments have accessible computer points. If there are additional sensors that might be beneficial to include in the model, expand the model to include these signals. It is easier to remove signals later than it is to add signals to an existing model.

5.2 Model Size Considerations

Determine the model size as part of the signal selection. Although there are no strict limits, consider the following:

- Signal validation processing time varies with the square of the model size. Very large data files are also time consuming to manage. For these reasons, models containing hundreds of signals are discouraged simply because of the limitations of signal processing and data handling. Even if a very large correlated group of signals can be identified, improved performance might be achieved by separating the signals into smaller groups for the purpose of signal validation. Typical models might be as few as 3 signals to as many as 80 signals; models containing less than 30 signals will be most common. Optimal model size is generally between 5 and 25 signals.
- Numerical modeling using large data files requires more computer memory than typical business software applications. For very large models or data files, available computer memory can become the limiting factor.
- Model retraining and maintenance requirements favor smaller models. If a sensor is recalibrated or replaced or if the process-operating characteristics change over time, it will likely be necessary to retrain the model. As the model becomes larger, it might require retraining more frequently simply because it contains more signals and, therefore, represents a more complex operating state space.

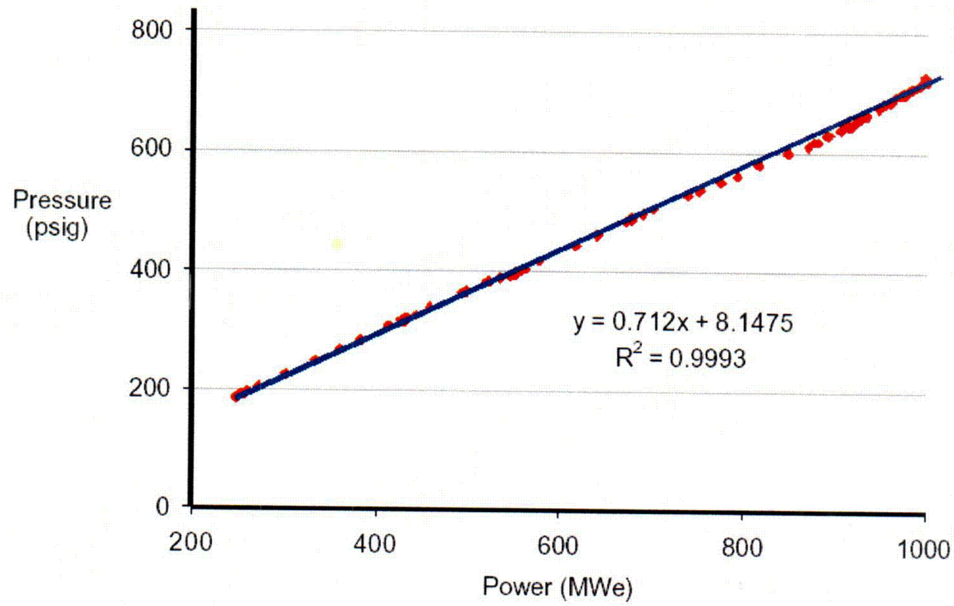
5.3 The Importance of Correlation

This section discusses the importance of correlation in developing models for on-line monitoring. SureSense has the ability to compute correlations and a series of other basic statistics that aid in assessing the validity of the current model.

5.3.1 Why Correlation Matters

Correlation is important in a model because the estimates are based on the learned behavior or correlation patterns among a group of signals. Implicit in this learning method is an assumption that drift or failure in one signal can be detected, based on the normal behavior of the remaining signals. This is an important point because the estimate has to distinguish between valid process changes and sensor drift or failure. A process change will be reflected in several correlated parameters, whereas a sensor drift will occur independently within a set of correlated parameters. For this reason, the signals combined in a model should be checked for correlation as part of model development.

Figure 5-3 shows a real example of highly correlated data. It can be seen that there is a clear relationship between turbine first-stage pressure and reactor power. As power increases, turbine first-stage pressure directly increases in a virtually linear manner. The square of the correlation coefficient is 0.9993, indicative of a high linear correlation. Drift in one pressure signal can be detected by the continued normal behavior of the other pressure signals as well as by reactor power and other similarly correlated signals.



1 psi = 6.894757 kPa

Figure 5-3
Example of Highly Correlated Data - Turbine Pressure and Reactor Power

Figure 5-4 shows a real example of steam generator level data, which has a poor correlation to reactor power in this case. As can be seen, steam generator level remains virtually constant at about 61 percent regardless of power level. Notice that the square of the correlation coefficient is 0.00003, indicative of no linear correlation between level and power. Drift in one steam generator level signal will not be predicted with changes in power. The learned behavior is that the steam generator level is about 61 percent regardless of the state of most other signals, with only the other redundant level signals providing any useful information. This does not necessarily mean that steam generator level is an unlikely candidate for on-line monitoring, but it does mean that its fault-detection capability will rely more on the joint response of the redundant channels than of other uncorrelated signals.

C01

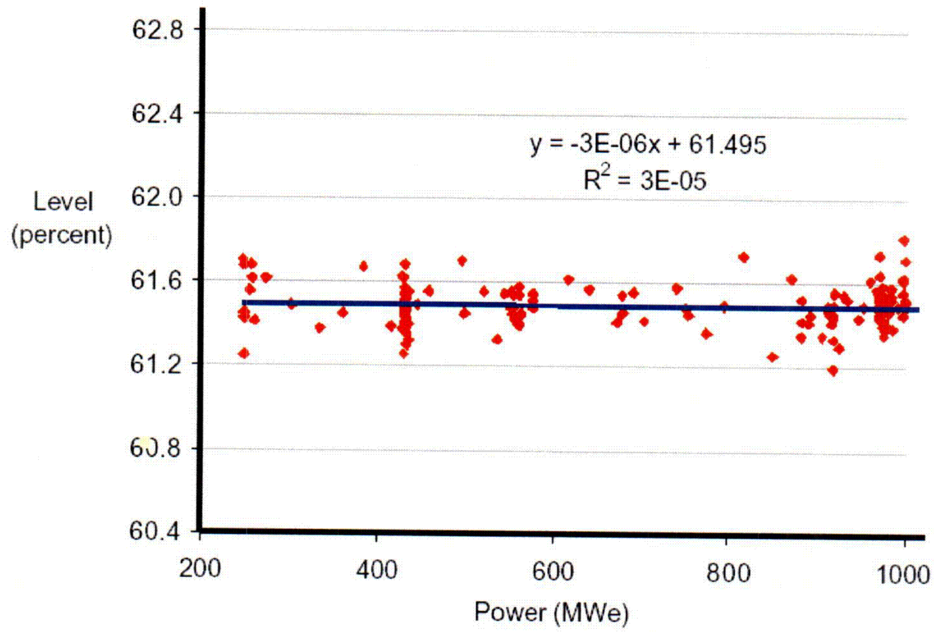


Figure 5-4
Example of Poorly Correlated Data - Steam Generator Level and Reactor Power

Correlation at some level is presumed in the MSET training procedure. The MSET MinMax training procedure (Section 7.1.1 describes the MinMax algorithm) establishes the boundaries of the presumed operating state space by selecting the training data vectors containing the smallest and largest value for each signal channel. The MSET vector-ordering training procedure adds to the MinMax vectors by selecting vectors within the operating state space that are approximately evenly distributed across this space. When MSET computes an estimate, it assigns greater weight to the training matrix vectors closest to the new observation vector. The assumption of correlation is inherent in this method. In other words, it is presumed that an estimate can be determined based on the training vectors closest to the observation vector. If all signals in a model were completely uncorrelated, the model would effectively be trained on noise, meaning that the training vectors closest to an observation might actually have little or no relevance to the actual observed values.

5.3.2 Why Correlated Parameters Might Not Appear to Correlate

The data used to check for correlation should be reviewed. Even if there is a known physical relationship between parameters, the data might not always show a correlation. In cases such as this, additional data might be needed. The following sections provide examples of such potential data issues.

C02

5.3.2.1 Small State Space

Figure 5-5 shows an example of highly correlated data—reactor power and turbine first-stage pressure vary in a virtually linear manner over the entire power range. The square of the correlation coefficient is 0.996, indicative of the high correlation.

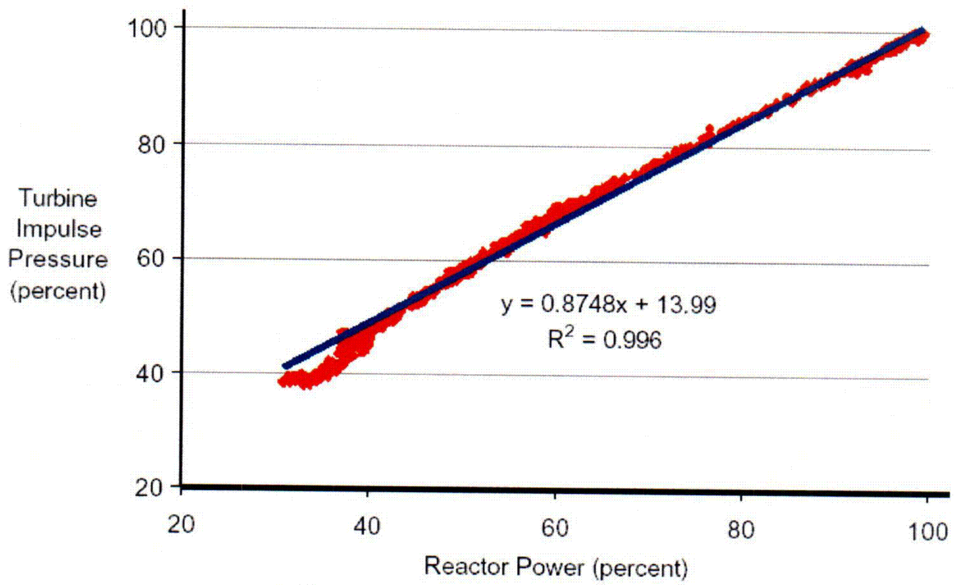


Figure 5-5
Typical Correlation of Reactor Power to Turbine Pressure - Entire Range

The method by which correlation measures are calculated will not produce the same result if only a portion of the range is evaluated. As an example, the graph provided in Figure 5-5 contains 2645 points. If the highest 645 points alone are plotted, the result is shown in Figure 5-6. Notice that these data points are in a very small range, all close to 100 percent power, and the square of the correlation coefficient is only 0.0995—indicative of a negligible correlation. This example illustrates the value of engineering knowledge in the modeling procedures. Signals in a particular model should be initially selected based on their known physical relationships. Next, data that adequately characterize the known relationship should be gathered and included in the model training procedures. As this example illustrates, it is preferable if the training data include normal operating points having correlated variations in these signals rather than data for a single operating point.

C03

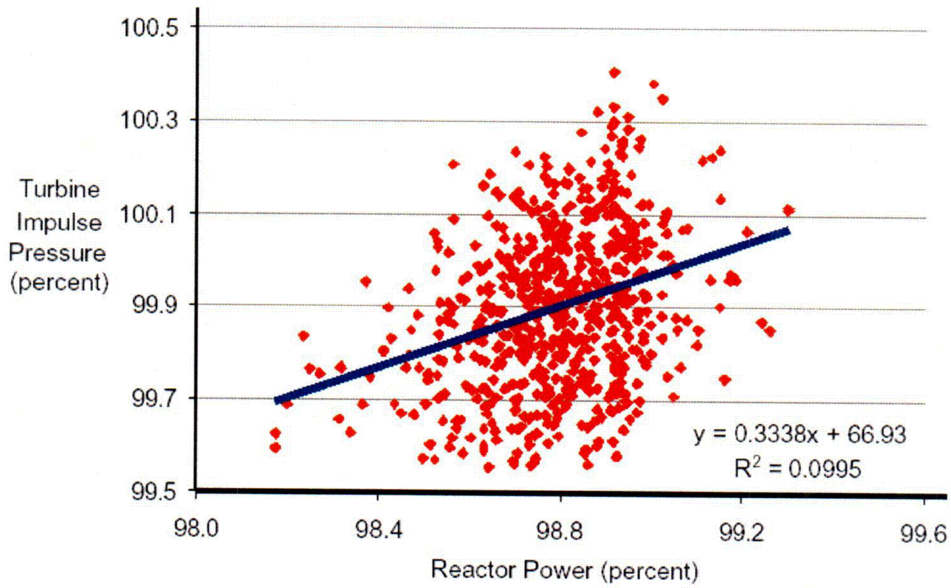


Figure 5-6
Typical Correlation of Reactor Power to Turbine Pressure - 100 Percent Power

If the data deviation from the regression line is small with respect to the range of the evaluated data, the correlation coefficient can be quite high as shown in Figure 5-5. If the data deviation from the regression line is large with respect to the range of the evaluated data, the correlation coefficient will be low as shown in Figure 5-6. This is a common problem with nuclear plant data. Most plants operate at or near 100 percent power for an extended period. Data extracted during this period will show little variation over a range, and any correlation analysis is effectively evaluating the correlation in the noise of the data. In these cases, it will be necessary to acquire additional data that cover a wider range if possible.

5.3.2.2 Data Archive Historian

Some data historian systems apply data compression techniques to minimize archive file size. If the signal value does not vary outside a specified range—referred to as the data compression limit, factor, or tolerance (depending on the computer system), the value is assumed to be unchanged. When the value eventually exceeds the data compression limit, the signal value is updated in storage. When the data are later extracted from archive, a linear interpolation routine might be used to derive intermediate points between the recorded values. By this approach, a significant reduction in file storage size can be realized for archived data. Unfortunately, the data historian creates false data between the stored points, which can adversely affect any signal correlation analysis. Section 6.3 discusses this problem in detail.

C04

5.3.3 Correlation Equation and Meaning

The discussion of correlation is usually based on an assumed linear correlation between signals. The linear correlation coefficient is given by:

$$r = \frac{\sigma_{xy}}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n\sigma_x \sigma_y}$$

where

R	=	Correlation coefficient
σ_{xy}	=	Covariance of x and y
σ_x , σ_y	=	Standard deviation of x and y
x_i, y_i	=	Pairs of x and y
$\bar{x}$, $\bar{y}$	=	Sample mean of x and y

The calculated value of r is an indicator of how well the points (x_i , y_i) fit a straight line, and r can range from -1 to +1. If the absolute value of r is close to 1, the points lie close to a straight line; if the absolute value of r is close to 0, the points are not correlated. The following interpretation of r is often used:

- If $0 \le |r| \le 0.3$, there is little or no linear relationship between x and y.
- If $0.3 < |r| \le 0.7$, there might be a weak linear relationship between x and y.
- If $0.7 < |r| \le 1.0$, there is a basis to claim some type of linear relationship between x and y.

Figure 5-7 shows how the value of r will vary with the relationship between x and y.

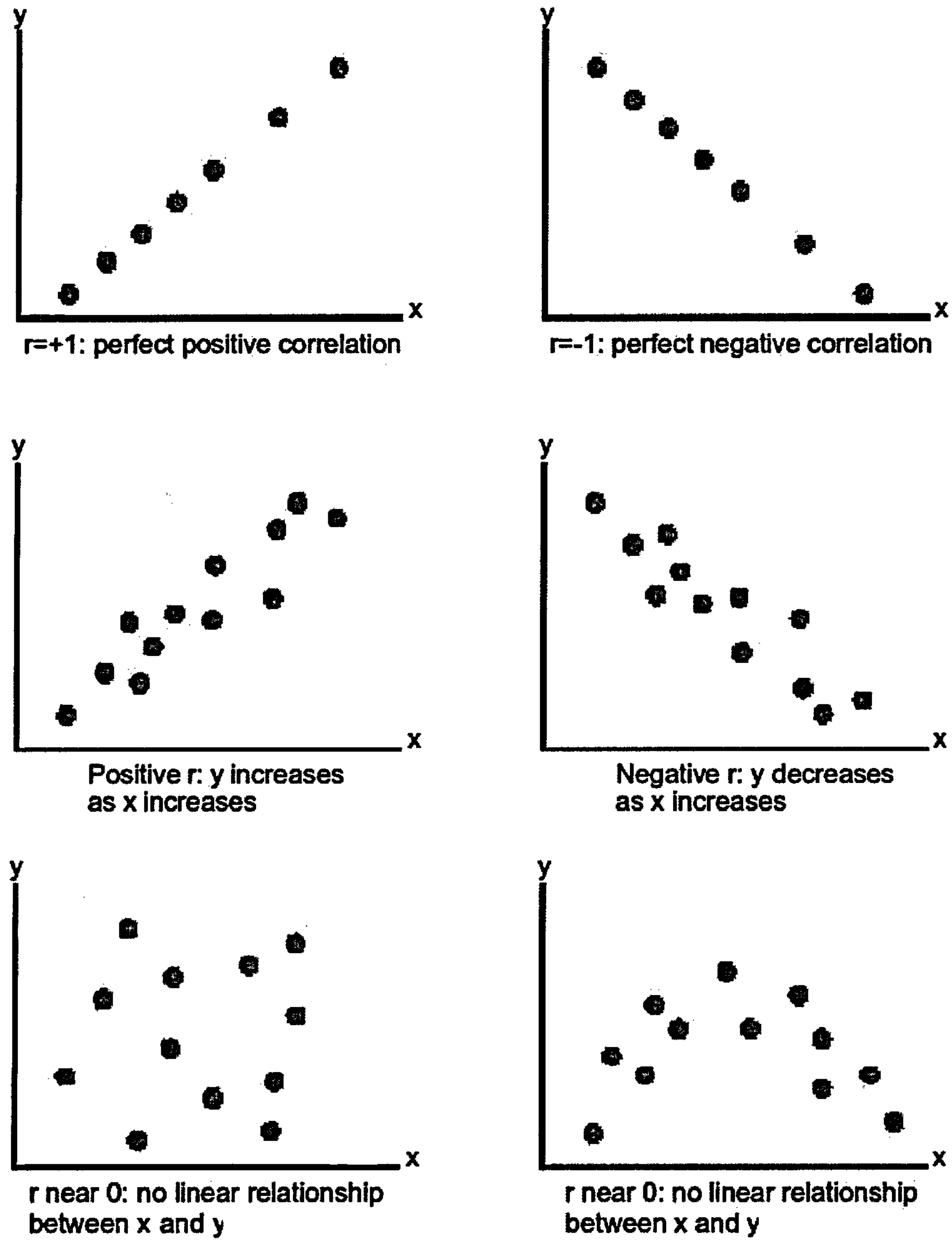


Figure 5-7
Correlation Values

When applied to samples, the correlation coefficient is often expressed as *Pearson's sample correlation coefficient* in which the sample standard deviations are used and the denominator is adjusted by n-1 rather than n. Statistics textbooks can provide additional information regarding the interpretation of the correlation coefficient.

Notice that an assumed linear relationship is often used for evaluation purposes because the associated statistic is relatively easy to calculate and understand. If the actual relationship between signals is correlated in some nonlinear relationship, as is often the case, this correlation test will indicate a lower level of correlation than actually exists. Also, the correlation coefficient does not by itself imply any sort of functional relationship between the two parameters; it only indicates the degree to which the two parameters are linearly correlated. For these reasons, engineering knowledge of the physical relationship between a group of signals should also be considered.

In addition, it is important that there is more than one signal in a model that has a strong correlation with any other signal (such that a given signal's estimate is not based solely on a correlation with only one other signal). This is a cautionary issue when developing models with four or five signals, where the average correlation might be significantly influenced by a single high correlation.

5.3.4 Example of an Uncorrelated Model

Figure 5-8 shows a small model for circulating water pump discharge pressures. When this model was first developed, it was not initially recognized that the discharge pressure signals for each of the four circulating water pumps did not sense a common pressure. In this context, the discharge pressure for each circulating water pump is entirely uncorrelated to the other pump discharge pressures. Figure 5-8 also shows two signals for each pump. Actually, these are two different computer points for the same physical sensor—they are not independent measurements even though they appear slightly different in value. Thus, there is no correlation between pumps because the two signals for each pump are actually from the same sensor. The correlation analysis readily shows this. This is an example of a model that looked promising at first. However, it was concluded upon further review that it was not a suitable candidate for on-line monitoring.

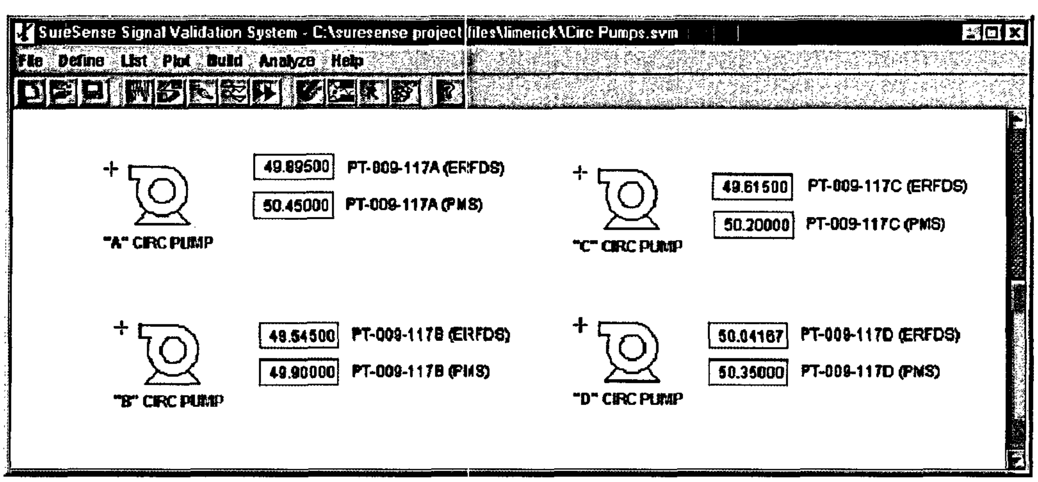


Figure 5-8
Circulating Water Pump Discharge Pressure Model

5.4 Planning for Phases (Submodels)

5.4.1 What Is a Submodel?

Some software applications such as SureSense will readily allow a single physical model to be divided into multiple submodels based on the various operating states (also referred to as phases) of the monitored asset. Experience has shown that dividing a model into submodels for certain distinct operating phases can significantly improve the model's diagnostic performance. Submodels are defined regions of operation that are separately trained in a model; signal validation can be performed on each phase independent of the signal behavior in other phases. This is an important feature to consider when process values can change considerably over the entire domain (range) of system operation. Phases in this context are very simply defined as regions where a given signal's value is between an upper and a lower limit.

For a given domain, typical phases that might be defined include:

- Power level – Submodels might be established for 100 percent power, 90 percent power, or other power levels.
- Ambient temperature – Process variations might be a function of ambient temperature so that a change in ambient conditions will shift the signal validation model into a different phase.
- System pressure – The correlated group of instruments in the model might behave differently at low pressure rather than at high pressure.
- Seasonal – As a variation on ambient temperature, it might be preferable to create a summer model and a winter model.
- Equipment lineups – There might be operating states that vary depending on which pumps are running. This will often be a consideration during low-power operation.

5.4.2 Power Level as a Phase Determiner

Many parameters in a power plant are either strongly or weakly directly correlated to power. Virtually all parameters in a power plant are affected by power level even if they are not correlated to power. For this reason, power level is a natural signal to include in a model for the purpose of defining and detecting the plant's operating phases.

The application of phases divides the model into separate submodels. Each submodel is separately trained and evaluated. For example, consider the use of several submodels based on reactor electric power in which the measured signal has units of percent. Suppose that the

principal operating modes (or phases) are near 100 percent power, 90 percent power, or about 70 percent power. For this operating behavior, the phases might be defined as follows:

- OPERATING_100 >98 percent power
- OPERATING_90 Between 90 and 98 percent power
- OPERATING_70 Between 60 and 80 percent power
- NONOPERATING <10 percent power
- OPERATING_OTHER Any other power level

For the purpose of signal validation, only the 100 percent, 90 percent, and 70 percent power levels might be validated. The model would then be trained with three submodels based on the power level. As input signals are processed, the power level would first be checked to determine which submodel applied. The input data would then be evaluated using the appropriate model training matrix and settings. Furthermore, the sensitivity and other settings can be individually tailored to the optimized performance of each submodel. During the period that the power level is outside the trained regions, the phase determiner would identify the phase state as either NONOPERATING or OPERATING_OTHER. If these two regions were not trained for signal validation, no fault-detection processing would be performed.

The use of phases is particularly useful because they enable the model to ignore data outside the regions of diagnostic interest or regions with insufficient training data. By this approach, failure alarms caused by data outside of the training range can be minimized. Regions with insufficient training data can be quite common for a power plant model. For example, a typical nuclear plant might operate above 90 percent power for more than 99 percent of the time that it is at power (an assertion that is based on data provided by the participants in the EPRI On-Line Monitoring Implementation Project). A typical plant might operate at better than 99 percent power for up to 98 percent of the time that it is at power. This means that there is an enormous amount of data near 100 percent power operation, but typically very little data for low-power and intermediate states. This is further complicated by equipment lineups that can vary at low power levels for which it might take years to observe suitable training data for each possible operating state. There is also the possibility of physical changes to this equipment, necessitating that new data be collected. The application of phases allows signal validation to be performed for virtually all of the period at power while excluding low-power states that have little or no available training data.

5.4.3 Choosing a Power Signal

Different types of power signals are readily available at each plant:

- Nuclear instrumentation signals
- Calculated reactor power (expressed either in megawatts or as a percent)
- Electric power output (expressed either in megawatts or as a percent)

The preferred power signal to use might depend on the signals in the model. The calculated reactor power signal might be preferred for models involving primary system signals. The electric power output signal might be preferred for models involving secondary system signals, mainly because the generated electric power will vary with some secondary system parameters.

5.4.4 Phases and Signal Selection

The application of phases affects signal selection in that the data for the phase determiner must also be acquired and verified for training. There will be very few models that should not initially include a power signal, even if there is not an obvious correlation between power and the signals in the model. As part of signal selection, it is important to think through the concept of phases and to choose signals that might have potential value as a phase determiner. It is always easier to remove signals from a model than it is to add signals later.

6 DATA QUANTITY AND QUALITY

The model's learned definitions of both the normal operating states and the normal signal behavior are established during training. The model's learned definition of its own parameter estimation performance is also established during training. Therefore, bad training data will be learned as normal and can significantly degrade fault-detection sensitivity. Actual operating states not included in the training data will often cause many or all signals to annunciate faults after the signals exceed the boundaries of the trained state space. For these reasons, the quantity and quality of training data warrant special care in the model development process.

The data used for training and testing should be of the best available quality. In this context, the term *data quality* refers to several attributes including:

- Training data should be representative of the signals' expected behavior under expected monitoring conditions.
- Bad data should be removed from training data sets so that the model does not learn incorrect operating states or bound a larger state space than is reasonable. In this instance, bad data refers to erroneous data recorded by the data acquisition system that is beyond the range of what is physically possible.
- Abnormal process variations should be evaluated for removal from the training data set. Even if the process variations really did occur, the associated data might not be desirable for training. Any data contained in the training data sets will be learned as normal behavior. Hence, an abnormal process included in the training data will result in the assumption that the given abnormal process is an expected phenomenon.
- If operating in batch mode, bad data should be removed from monitored data sets to minimize verification efforts due to fault-detection events that might occur. It should be recognized that bad data can produce false alarms or fault-detection events when operating in on-line mode.

The following sections address various aspects of data quality that should be considered when developing a model.

6.1 Data Quantity

6.1.1 Quantity of Data - Sample Frequency

It is important to ensure that the training data set contains adequate data for each operating phase to be validated and that the data are representative of the evaluated operating conditions. A phase

is a defined region of operation. Phases are used to separate the model into submodels (refer to Section 5.4 for more information). If a particular phase does not appear to have sufficient data for training, monitoring for that phase should not be enabled so that detection events are not annunciated during these phases.

Data files should generally contain data sampled at a frequency consistent with normal process variations—for large systems such as power plants, sampling once every minute is recommended. If data are obtained at a frequency that is too low (such as once every hour for a power plant), some data filtering and analysis options will not be meaningful or appropriate. Although monitoring data might not need to be acquired at the same sampling frequency as training data, it is recommended that all data should be acquired at the same frequency for the following reasons:

- As a model is initially developed, it is not always known how much training data will be needed. In some models, the first month of data might be adequate for training. In other models, it might take several months to acquire data that covers the expected operating space. Subtle process changes might occur over time that prompt retraining with additional data.
- If equipment or sensors are repaired, recalibrated, or replaced, it might be necessary to retrain the model with either new data or additional data.
- Data acquired at a 1-minute rate are typically frequent enough to allow data filtering by an averaging or median-select approach (refer to Section 6.4). The dynamics of typical power plant systems do not usually cause large fluctuations in signals over a period of a few minutes while at steady-state conditions. However, wider fluctuations are possible as the sampling interval is extended, which can eliminate potential data filtering options.
- The method by which signals are identified as faulted depends on the number of alarms received over a set of sequential measurements. As a signal drifts outside the expected range and the initial alarms are generated, it generally requires additional time points before the signal is declared faulted. If signals are acquired at longer frequencies (such as every hour), it might take one day or more for signal failures to be identified. Higher sampling frequencies increase the likelihood of early fault detection.
- When a problem is identified with a monitored channel, the user will typically want to evaluate the monitored data in more detail. By having monitored data available at a relatively high data acquisition frequency, the user can more easily evaluate the potential problem. For example, the user might initially monitor available data sampled at a 15-minute rate. However, if a problem is identified, the user might instead look more closely at the results by reprocessing the data at a 1-minute rate.
- The method by which data are acquired should be consistent. Attempting to acquire monitoring data at a rate different than that used for the training data increases the likelihood of data processing inconsistencies. Note that even if monitoring data are acquired at a higher rate (such as every minute), it need not be validated at that frequency.

6.1.2 Quantity of Data - How Much Historical Data to Acquire

Model development includes testing a model for adequate performance using actual plant data. Even if a plant operates at about 100 percent power for an entire operating cycle, process values

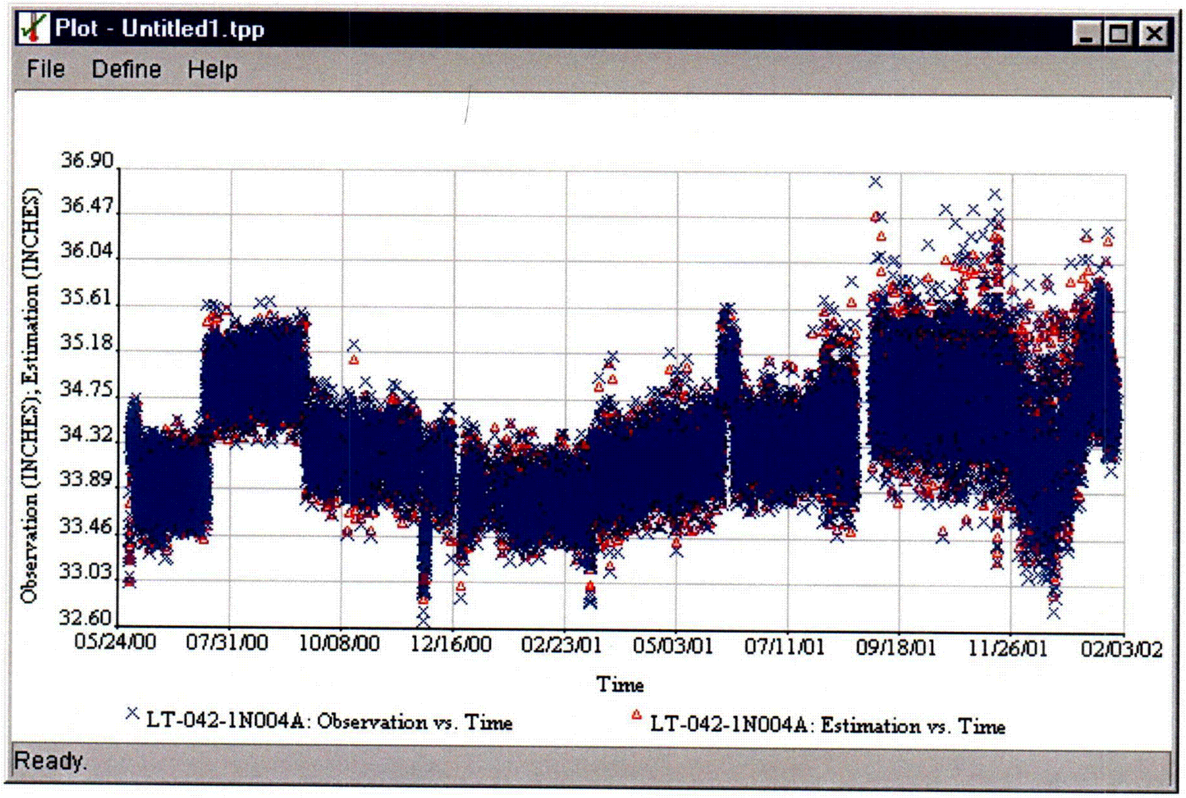
can change over time. For this reason, it is important to test the model with as much historical data as possible so that the model response to these process changes can be anticipated. The use of data from the previous operating cycle up to the present is recommended to evaluate the model's performance. In some cases, this might mean acquiring up to 24 months of data. Although this is a large amount of data, it allows the model to be thoroughly tested and evaluated during development. The response to fault alarms that might occur in the future associated with different equipment lineups, periodic transients, or minor process changes (temporary or permanent) can be determined before deploying the model for in-service use. Acquiring this amount of data will also identify any data acquisition issues or problems.

6.2 Dealing with Bad Data

Bad data is a general term that covers several types of data problems and erroneous data that will likely be encountered. In this context, the presence of bad data is usually caused by failures in the data acquisition system or by programming limitations in the manner by which data are archived. Whenever possible, bad data should be removed from files to minimize evaluation and fault-detection errors that might occur. Bad data absolutely must be removed from training data files so that the model is not trained to treat the bad data as normal system behavior. In the following subsections, the most common types of bad data are presented. The second volume of this three-volume report series [1] presents an evaluation of a stand-alone software product that was designed to aid in data preprocessing efforts related to OLM model development.

6.2.1 The Effect of Bad Data on the Estimation Process

The model's learned definitions of both normal operating states and normal signal behavior are established during training. The model's learned definition of its own parameter estimation performance is also established during training. Bad data that are allowed to remain in the data provided for training will affect the calculation of the estimate for each corresponding observation. Figure 6-1 shows an example of a model that has been trained with acceptable data. Although this particular signal has a relatively high noise content, the estimates (shown as red triangles) track the observations (shown as blue crosses) reasonably well.

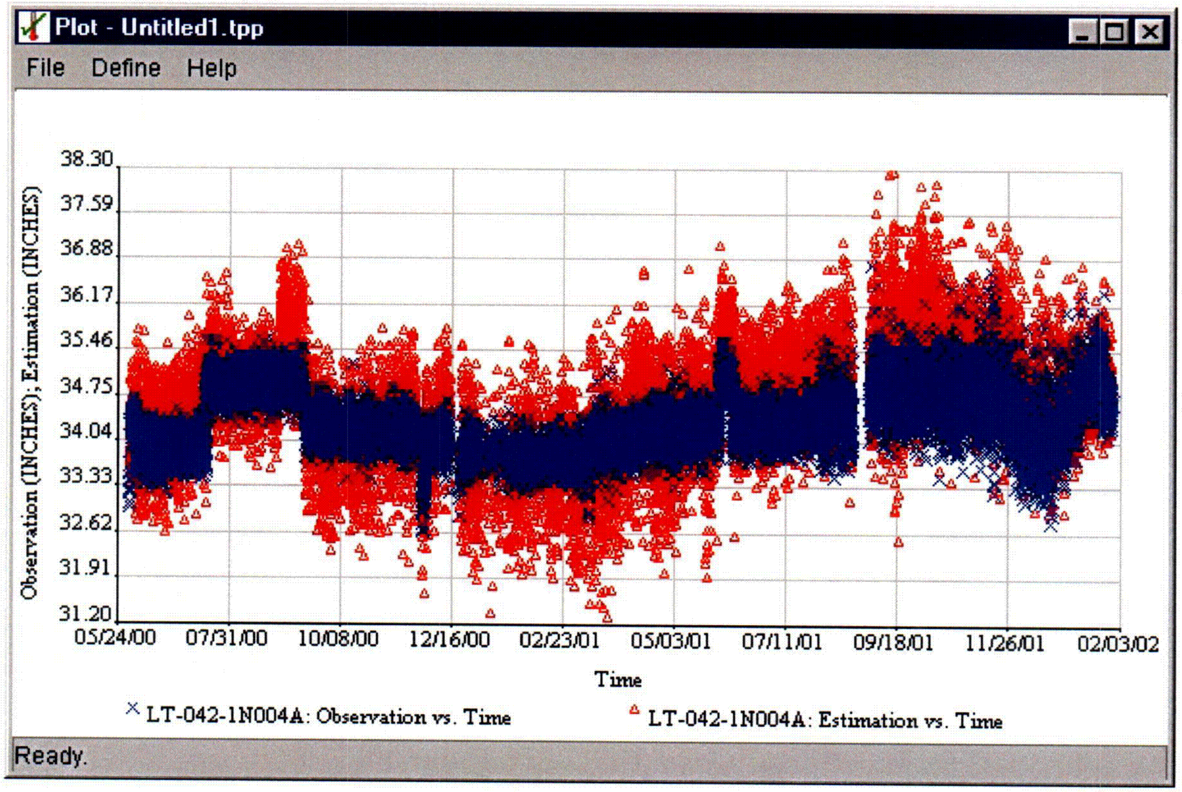


1 in. = 25.4 mm

Figure 6-1
Observations and Corresponding Estimates With Acceptable Training Data

As shown in Figure 6-1, this level signal ranges from about 32–37 inches (812.8–939.8 mm) over a two-year period. To demonstrate how bad data can affect the quality of the training process, two vectors were added to this model—each contained bad data for this signal. One bad data point had a value of 5 inches (127 mm), and the second bad data point had a value of 55 inches (1,397 mm) (which ensures that these two vectors will be selected by the MinMax training method). Figure 6-2 shows the result. It should be noted that the estimates no longer track the observations well. The two outlying bad data points have influenced the estimation calculations for this signal, resulting in poor model performance for the signal.

C05



1 in. = 25.4 mm

Figure 6-2
Observations and Corresponding Estimates With Two Bad Data Points

Bad data must be removed from training data sets. Otherwise, incorrect data values will adversely affect both estimation and fault detection in the model.

6.2.2 Examples of Data Problems

Figure 6-3 shows an example of bad data acquisition. The values are clearly in error and are well outside of any expected signal range. If the data are left in the training data set, the MSET training vectors will select these values as part of the expected operating state space boundaries. This type of data problem can identify failures with the data acquisition modules.

C06

Microsoft Excel - Examples of Data Problems.xls

D5 = 33.57833633

	A	B	C	D	E	F	G	
1	Time	LT-042-1N004A	LT-042-1N004B	LT-042-1N004C	LT-042-115A	LT-042-115B	LT-042-1N017	LT-04
2	10/11/00 11:32 AM	34.50999948	34.26499997	34.12999842	30.08833382	29.79333178	37.51166673	37.
3	10/11/00 11:33 AM	34.08666697	33.70500207	33.52666417	29.91333327	28.37666613	36.66500055	3
4	10/11/00 11:33 AM	34.36166488	33.66166742	34.00333247	29.59833352	29.51833277	37.20166548	37.
5	10/11/00 11:34 AM	34.12500048	34.52333193	33.57833633	28.89166675	30.10666667	35.76333403	36.
6	10/11/00 11:35 AM	34.4550014	34.68166707	34.19333313	29.70500028	27.84999982	38.15333167	36.
7	10/11/00 11:36 AM	34.03333475	33.88000218	33.73333403	29.905	31.32333233	37.44666602	3
8	10/11/00 11:37 AM	34.4233332	34.1016663	33.75833155	30.69666565	28.91999948	37.5900005	
9	10/11/00 11:38 AM	34.00333278	32.76833382	33.64500052	29.58499997	28.74333185	37.44833292	36.
10	10/11/00 11:39 AM	33.98999972	33.5616654	33.70500137	30.17333287	29.96333368	37.7166651	3
11	10/11/00 11:41 AM	28.87499873	28.52666705	28.79999919	25.95666593	25.43666737	31.25333321	3
12	10/11/00 11:42 AM	-6.10948E-07	3.04542E-07	7.00355E-07	7.7486E-07	4.76837E-07	9.53674E-07	-5.3
13	10/11/00 11:43 AM	-6.10948E-07	3.04542E-07	7.00355E-07	7.7486E-07	4.76837E-07	9.53674E-07	-5.3
14	10/11/00 11:44 AM	-6.10948E-07	3.04542E-07	7.00355E-07	7.7486E-07	4.76837E-07	9.53674E-07	-5.3
15	10/11/00 11:45 AM	-6.10948E-07	3.04542E-07	7.00355E-07	7.7486E-07	4.76837E-07	9.53674E-07	-5.3
16	10/11/00 11:46 AM	-6.10948E-07	3.04542E-07	7.00355E-07	7.7486E-07	4.76837E-07	9.53674E-07	-5.3
17	10/11/00 11:47 AM	-6.10948E-07	3.04542E-07	7.00355E-07	7.7486E-07	4.76837E-07	9.53674E-07	-5.3
18	10/11/00 11:48 AM	11.53333359	11.40000054	11.3666668	10.00000085	9.866667318	12.06666797	12.
19	10/11/00 11:49 AM	34.63666527	33.9700017	34.22333197	30.5599998	29.4733337	36.8633318	37.
20	10/11/00 11:50 AM	34.08666813	33.40666593	33.50666707	29.26000107	27.3333328	37.06666453	36.
21	10/11/00 11:51 AM	34.82666427	33.8533334	34.0866656	30.03333333	28.09333387	37.5933332	3
22	10/11/00 11:51 AM	34.2800016	33.02666593	33.62000107	29.94000053	29.65333333	37.6466664	37.
23	10/11/00 11:52 AM	34.6033335	34.1666657	34.02333293	30.0533328	29.85333227	37.79999983	36.
24	10/11/00 11:53 AM	34.2133342	33.9599996	33.8599992	31.2133336	27.55333307	37.76666587	3
25	10/11/00 11:54 AM	34.5600003	34.15333303	33.84333433	30.47333377	30.09999917	37.40666703	36.
26	10/11/00 11:55 AM	34.18666797	33.61333603	33.96333443	29.0200008	29.31666667	38.46333063	37.
27	10/11/00 11:56 AM	34.34999973	33.90999937	34.103335	30.139999	29.90999947	37.4299983	
28								

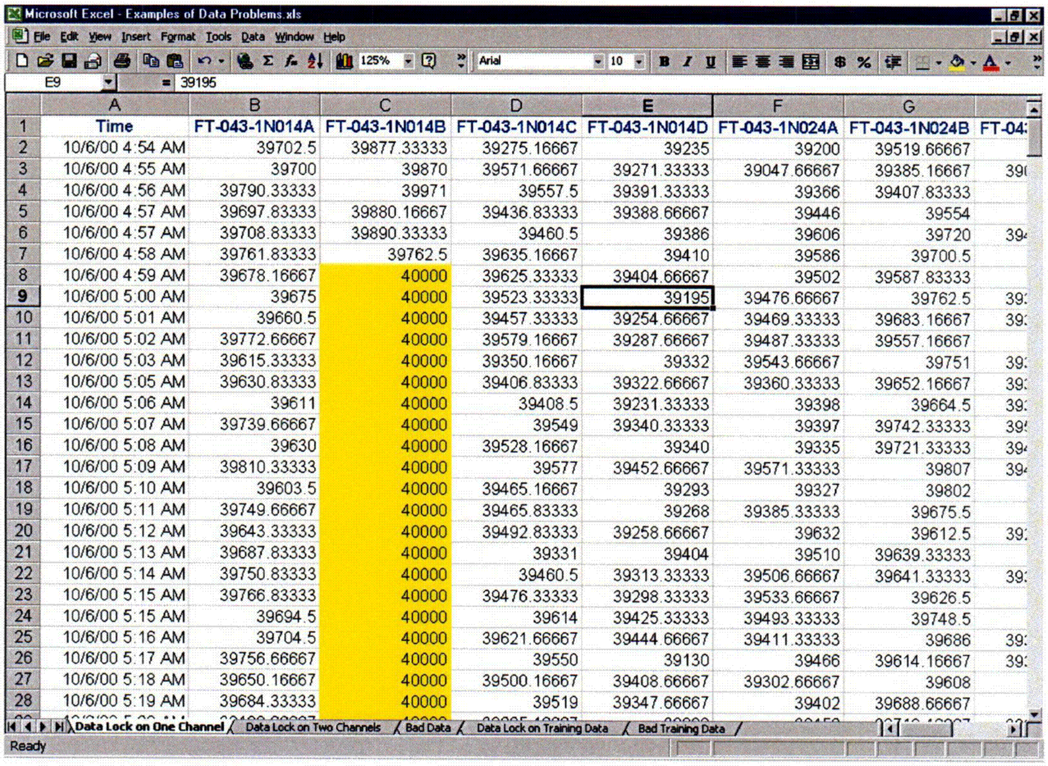
Data Lock on an Entire Group / Data Lock on One Channel / Data Lock on Two Channels / **Bad Data** / Data Lock on Training Data / Bad Training

Ready

Figure 6-3
Bad Data Acquisition

Figure 6-4 shows an example of data lockup for one channel. Because of a data acquisition problem, one signal is frozen at a value of 40,000, and its actual value is unknown. This data should not be used for training because it can degrade the estimation results.

C07



	A	B	C	D	E	F	G
1	Time	FT-043-1N014A	FT-043-1N014B	FT-043-1N014C	FT-043-1N014D	FT-043-1N024A	FT-043-1N024B
2	10/6/00 4:54 AM	39702.5	39877.33333	39275.16667	39235	39200	39519.66667
3	10/6/00 4:55 AM	39700	39870	39571.66667	39271.33333	39047.66667	39385.16667
4	10/6/00 4:56 AM	39790.33333	39971	39557.5	39391.33333	39366	39407.83333
5	10/6/00 4:57 AM	39697.83333	39880.16667	39436.83333	39388.66667	39446	39554
6	10/6/00 4:57 AM	39708.83333	39890.33333	39460.5	39386	39606	39720
7	10/6/00 4:58 AM	39761.83333	39762.5	39635.16667	39410	39586	39700.5
8	10/6/00 4:59 AM	39678.16667	40000	39625.33333	39404.66667	39502	39587.83333
9	10/6/00 5:00 AM	39675	40000	39523.33333	39195	39476.66667	39762.5
10	10/6/00 5:01 AM	39660.5	40000	39457.33333	39254.66667	39469.33333	39683.16667
11	10/6/00 5:02 AM	39772.66667	40000	39579.16667	39287.66667	39487.33333	39557.16667
12	10/6/00 5:03 AM	39615.33333	40000	39350.16667	39332	39543.66667	39751
13	10/6/00 5:05 AM	39630.83333	40000	39406.83333	39322.66667	39360.33333	39652.16667
14	10/6/00 5:06 AM	39611	40000	39408.5	39231.33333	39398	39664.5
15	10/6/00 5:07 AM	39739.66667	40000	39549	39340.33333	39397	39742.33333
16	10/6/00 5:08 AM	39630	40000	39528.16667	39340	39335	39721.33333
17	10/6/00 5:09 AM	39810.33333	40000	39577	39452.66667	39571.33333	39807
18	10/6/00 5:10 AM	39603.5	40000	39465.16667	39293	39327	39802
19	10/6/00 5:11 AM	39749.66667	40000	39465.83333	39268	39385.33333	39675.5
20	10/6/00 5:12 AM	39643.33333	40000	39492.83333	39258.66667	39632	39612.5
21	10/6/00 5:13 AM	39687.83333	40000	39331	39404	39510	39639.33333
22	10/6/00 5:14 AM	39750.83333	40000	39460.5	39313.33333	39506.66667	39641.33333
23	10/6/00 5:15 AM	39766.83333	40000	39476.33333	39298.33333	39533.66667	39626.5
24	10/6/00 5:15 AM	39694.5	40000	39614	39425.33333	39493.33333	39748.5
25	10/6/00 5:16 AM	39704.5	40000	39621.66667	39444.66667	39411.33333	39686
26	10/6/00 5:17 AM	39756.66667	40000	39550	39130	39466	39614.16667
27	10/6/00 5:18 AM	39650.16667	40000	39500.16667	39408.66667	39302.66667	39608
28	10/6/00 5:19 AM	39684.33333	40000	39519	39347.66667	39402	39688.66667

Figure 6-4
Data Lockup for One Channel

Data lockup creates constant-valued signals that should not be included in any training data. The MSET model training algorithms are intolerant of signals that are constant valued, whether these arise from data lockup or from signals that are truly unchanging over time. Constant-valued signals cannot be effectively correlated to any other signal's behavior because such a signal is clearly independent of changes in other signals. Constant-valued signals in the training data also introduce numerical instabilities in the MSET training procedures.

If a signal is stuck for an extended period, it will be necessary to remove it from the model until satisfactory data are available (in one instance, two signals were stuck for several months because dummy signals had been inadvertently inserted in place of the actual signals). Figure 6-5 shows an example of a stuck channel during an on-line monitoring analysis. It should be noted that the estimate (shown as red triangles) continues to predict where the value should be even though the observations are stuck for an extended period. This is the usual result when a stuck data value occurs during monitoring. This behavior can be identified by an OLM system through the use of the SPRT variance test (see Section 8). While it is essential that stuck data should not be included in the training data, stuck data occurring during monitoring will be readily identified, and the corresponding instrument channel can be investigated to determine the cause of the error.

C08

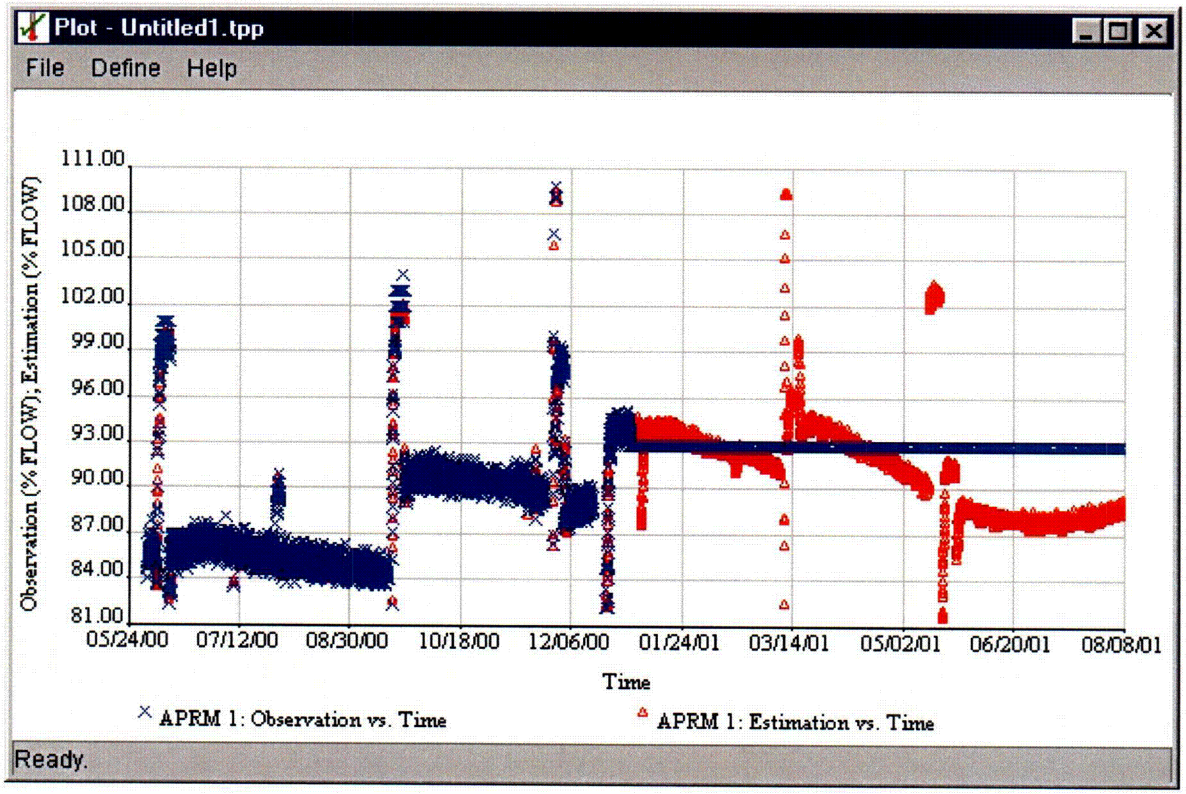


Figure 6-5
Example of Data Lockup in an Analysis

Figure 6-6 shows an example of data lockup for all channels. Because of a data acquisition problem, all channels are frozen, and the actual signal values are unknown. These data should not be used at all. The data immediately before and after the frozen data are often also suspect.

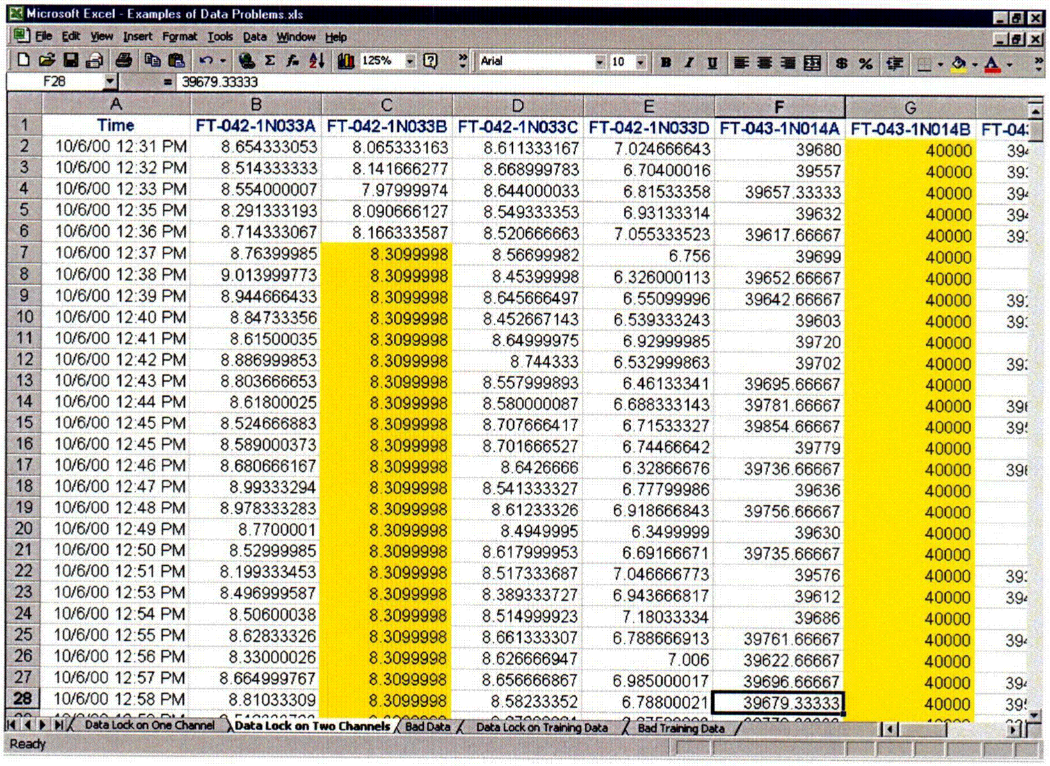
C09

	Time	LT-042-1N004A	LT-042-1N004B	LT-042-1N004C	LT-042-115A	LT-042-115B	LT-042-1N017	LT-04
2	6/16/00 6:05 PM	33.74499985	34.14666687	33.87333173	32.75333353	31.5400007	38.52333187	37.
3	6/16/00 6:06 PM	34.17999893	33.67666725	33.79499943	31.29333498	29.20166663	39.45500007	37.
4	6/16/00 6:07 PM	33.8716666	32.66666702	33.53666702	30.2983334	30.0833324	38.12000107	3
5	6/16/00 6:08 PM	33.7350007	34.04166413	33.78166662	31.38500018	32.31333092	38.39666518	36.
6	6/16/00 6:09 PM	33.97833227	34.10166635	34.13999895	31.64333333	31.11000175	38.02666927	
7	6/16/00 6:09 PM	33.63833282	33.91166788	33.42000132	30.62	30.88999957	37.01666763	37.
8	6/16/00 6:10 PM	33.85166815	33.1900011	34.09999848	30.89666618	30.35499955	37.37999948	37.
9	6/16/00 6:11 PM	33.99499975	33.20833353	33.71333382	30.59666662	32.10666692	37.42500088	37.
10	6/16/00 6:12 PM	34.17999962	33.53999797	33.95666478	31.24166667	32.02166682	38.06000038	36.
11	6/16/00 6:13 PM	33.92666458	33.72999817	33.67500077	30.81666745	31.23333343	36.90333633	37.
12	6/16/00 6:14 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
13	6/16/00 6:15 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
14	6/16/00 6:17 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
15	6/16/00 6:18 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
16	6/16/00 6:19 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
17	6/16/00 6:20 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
18	6/16/00 6:21 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
19	6/16/00 6:22 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
20	6/16/00 6:23 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
21	6/16/00 6:24 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
22	6/16/00 6:25 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
23	6/16/00 6:26 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
24	6/16/00 6:27 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
25	6/16/00 6:27 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
26	6/16/00 6:28 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
27	6/16/00 6:29 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	
28	6/16/00 6:30 PM	33.899997	33.599998	33.600001	30.700001	30.8	36.500004	

Figure 6-6
Data Lockup for All Channels

Figure 6-7 shows an example of data lockup for two channels. Because of a data acquisition problem, two signals are frozen, and their actual value is unknown. These data should not be used for training because they can degrade the estimation results.

c10



	A	B	C	D	E	F	G	
1	Time	FT-042-1N033A	FT-042-1N033B	FT-042-1N033C	FT-042-1N033D	FT-043-1N014A	FT-043-1N014B	FT-04
2	10/6/00 12:31 PM	8.654333053	8.065333163	8.611333167	7.024666643	39680	40000	39
3	10/6/00 12:32 PM	8.514333333	8.141666277	8.668999783	6.70400016	39557	40000	39
4	10/6/00 12:33 PM	8.554000007	7.97999974	8.644000033	6.81533358	39657.33333	40000	39
5	10/6/00 12:35 PM	8.291333193	8.090666127	8.549333353	6.93133314	39632	40000	39
6	10/6/00 12:36 PM	8.714333067	8.166333587	8.520666663	7.055333523	39617.66667	40000	39
7	10/6/00 12:37 PM	8.76399985	8.3099998	8.56699982	6.756	39699	40000	
8	10/6/00 12:38 PM	9.013999773	8.3099998	8.45399998	6.326000113	39652.66667	40000	
9	10/6/00 12:39 PM	8.944666433	8.3099998	8.645666497	6.55099996	39642.66667	40000	39
10	10/6/00 12:40 PM	8.84733356	8.3099998	8.452667143	6.539333243	39603	40000	39
11	10/6/00 12:41 PM	8.61500035	8.3099998	8.64999975	6.92999985	39720	40000	
12	10/6/00 12:42 PM	8.886999853	8.3099998	8.744333	6.532999863	39702	40000	39
13	10/6/00 12:43 PM	8.803666653	8.3099998	8.557999893	6.46133341	39695.66667	40000	
14	10/6/00 12:44 PM	8.61800025	8.3099998	8.580000087	6.688333143	39781.66667	40000	39
15	10/6/00 12:45 PM	8.524666883	8.3099998	8.707666417	6.71533327	39854.66667	40000	39
16	10/6/00 12:45 PM	8.589000373	8.3099998	8.701666527	6.74466642	39779	40000	
17	10/6/00 12:46 PM	8.680666167	8.3099998	8.6426666	6.32866676	39736.66667	40000	39
18	10/6/00 12:47 PM	8.99333294	8.3099998	8.541333327	6.77799986	39636	40000	
19	10/6/00 12:48 PM	8.978333283	8.3099998	8.61233326	6.918666843	39756.66667	40000	
20	10/6/00 12:49 PM	8.7700001	8.3099998	8.4949995	6.3499999	39630	40000	
21	10/6/00 12:50 PM	8.52999985	8.3099998	8.617999953	6.69166671	39735.66667	40000	
22	10/6/00 12:51 PM	8.199333453	8.3099998	8.517333687	7.046666773	39576	40000	39
23	10/6/00 12:53 PM	8.496999587	8.3099998	8.389333727	6.943666817	39612	40000	39
24	10/6/00 12:54 PM	8.50600038	8.3099998	8.514999923	7.18033334	39686	40000	
25	10/6/00 12:55 PM	8.62833326	8.3099998	8.661333307	6.788666913	39761.66667	40000	39
26	10/6/00 12:56 PM	8.33000026	8.3099998	8.626666947	7.006	39622.66667	40000	
27	10/6/00 12:57 PM	8.664999767	8.3099998	8.656666867	6.985000017	39696.66667	40000	39
28	10/6/00 12:58 PM	8.81033309	8.3099998	8.58233352	6.78800021	39679.33333	40000	39

Figure 6-7
Data Lockup for Two Channels

Missing data will cause fault alarm problems with the model in operation. Removing these occurrences from the data sets before use will reduce the occurrence of alarms. Figure 6-8 shows an example of this problem type. Note that if these data were used during monitoring, the missing data would result in annunciated alarms. While thorough data cleanup is required for training, it is not a requirement for monitoring. Bad data that are included in the monitoring data will result in alarms from the OLM system that can then be investigated to determine the source of the alarm. Although it is not a requirement, data cleanup of monitoring data sets will ensure that the number of alarms is reduced. This discussion is intended to address data cleanup concerns for a real-time implementation. It is not the case that every bad data point needs to be removed prior to monitoring; however, it is the case that these bad data points will produce alarms from the OLM system. Note that removing bad data from the training data is still required.

C11

	A	B	C	D	E	F
1	Time	FT-043-1N024A	FT-043-1N024B	FT-043-1N024C	FT-043-1N024D	AVG APRM PWR
2	11/20/00 2:55 PM	38995.1	39103.96667	38731.26667	38456.36667	99.01666792
3	11/20/00 2:56 PM	39041.75	39105.5	38912.5	38619.5	99.1
4	11/20/00 2:57 PM	38903.9	39089.16667	38879.9	38591.2	99.0016672
5	11/20/00 2:59 PM	39014.88333	39167.8	38714.66667	38422.9	99.0533328
6	11/20/00 3:00 PM	39033.18333	39151.83333	38830.16667	38537.9	99.0533328
7	11/20/00 3:01 PM	38984.93333	39123.8	38819.6	38535.53333	99.12166597
8	11/20/00 3:02 PM	38844.1	38988.43333		38543.1	99.100006
9	11/20/00 3:03 PM	39026.56667	38959.5		38457	99.10000445
10	11/20/00 3:04 PM	39043.4	39265.36667		38613.16667	99.153331
11	11/20/00 3:05 PM	39040.68333	39097.6		38395.4	99.06500158
12	11/20/00 3:06 PM	38882.36667	39186.03333		38477.6	99.08666312
13	11/20/00 3:07 PM	38848.71667	39107		38545.73333	99.14166678
14	11/20/00 3:08 PM	39006.75	39225.73333		38498.4	99.04833437
15	11/20/00 3:09 PM	38995.38333	39198.63333		38445.33333	99.1966662
16	11/20/00 3:09 PM	38908.53333	39070.5		38414.7	99.06666842
17	11/20/00 3:10 PM	39025.58333	39221.83333		38547.83333	99.0849964
18	11/20/00 3:11 PM	39046.86667	39130.2		38575.2	99.1249993
19	11/20/00 3:12 PM	38992.43333	38998.96667		38403.86667	99.056668
20	11/20/00 3:13 PM	38991.63333	39057.16667		38433.93333	99.1116677
21	11/20/00 3:14 PM	38859.06667	39230.5		38598.5	99.15166702
22	11/20/00 3:15 PM	38882.21667	39161.06667		38435.8	99.10166972
23	11/20/00 3:17 PM	38930.91667	39166.33333		38561.56667	99.04666605
24	11/20/00 3:18 PM	38865.45	39143.26667		38523.63333	99.099994
25	11/20/00 3:19 PM	38891.63333	39191.36667		38471.13333	99.099994
26	11/20/00 3:20 PM	39025.15	39197.13333		38549.93333	99.1149959
27	11/20/00 3:21 PM	38906.2	39107.63333		38539.93333	99.14166792
28	11/20/00 3:22 PM	39014.78333	39249		38497	99.10000293

Figure 6-8
Missing Data

Sometimes the data values are incorrect but might appear to be within a reasonable range. Figure 6-9 shows four steam-flow signals as well as the total steam-flow signal. The total steam flow should be the sum of the four individual signals. Throughout the period shown, reactor power was constant at about 100 percent; other correlated signals behaved normally. As can be seen, three of the four steam flow signals are fluctuating because of a problem with the data acquisition system. The total steam-flow signal is nearly constant during this period and is at the expected value. These erroneous individual steam-flow signals must be removed from any training data sets to prevent the model from learning this type of behavior as normal or expected.

C12

Time	Steam Flow A	Steam Flow B	Steam Flow C	Steam Flow D	Total Steam Flow
6/1/00 0:00	3.74	3.61	3.70	3.66	14.70
6/1/00 0:01	3.74	3.62	3.70	3.66	14.70
6/1/00 0:02	3.74	3.43	3.50	3.47	14.70
6/1/00 0:03	3.73	1.88	1.91	1.89	14.70
6/1/00 0:04	3.73	3.60	3.71	3.66	14.75
6/1/00 0:05	3.73	3.61	3.69	3.66	14.80
6/1/00 0:06	3.74	3.60	3.70	3.66	14.71
6/1/00 0:07	3.73	3.60	3.70	3.65	14.75
6/1/00 0:08	3.74	3.60	3.69	3.65	14.70
6/1/00 0:09	3.73	3.19	3.26	3.23	14.75
6/1/00 0:09	3.73	2.11	2.15	2.13	14.70
6/1/00 0:10	3.72	3.60	3.70	3.64	14.70
6/1/00 0:11	3.73	3.60	3.70	3.66	14.70
6/1/00 0:12	3.73	3.61	3.69	3.66	14.70
6/1/00 0:13	3.74	0.54	0.55	0.55	14.73
6/1/00 0:14	3.73	1.92	1.97	1.95	14.72
6/1/00 0:15	3.73	2.59	2.66	2.62	14.70
6/1/00 0:17	3.73	3.62	3.71	3.65	14.70
6/1/00 0:18	3.73	3.43	3.51	3.47	14.70
6/1/00 0:19	3.72	0.00	0.00	0.00	14.70
6/1/00 0:20	3.73	1.63	1.67	1.64	14.70
6/1/00 0:21	3.73	1.69	1.72	1.70	14.75
6/1/00 0:22	3.73	2.46	2.53	2.49	14.70
6/1/00 0:23	3.74	2.83	2.90	2.85	14.70
6/1/00 0:24	3.73	3.61	3.71	3.65	14.70
6/1/00 0:25	3.73	3.18	3.26	3.22	14.70
6/1/00 0:26	3.73	2.10	2.15	2.12	14.70

Figure 6-9
Incorrect Data Values - Several Signals

Sometimes, a single signal fluctuates wildly because of a data acquisition problem. Figure 6-10 shows an example in which a steam pressure signal, (normally about 970 psig/6688 kPa), varies from -280 psig to over 2000 psig (-1931 kPa to over 13790 kPa) in a 30-minute period during which reactor power was constant. This is not a sensor problem—it is a data acquisition problem. Notice also that another signal (FW-FT-7) is stuck during this time period.

C13

	A	B	C	D	E	F
1						
2	Time	MWE	FW-FT-7	EX-PT-29	EX-PT-30	FW-PT-64
3	Sec	MWE	MLB/H	PSIG	PSIG	PSIG
4	Time of Data	A5004	A0021	A0024	A0025	A0032
5	11/10/99 19:13	860.996	5.48438	478.516	269.238	969.238
6	11/10/99 19:14	860.996	5.48438	478.516	269.336	969.238
7	11/10/99 19:15	860.933	5.48438	478.516	269.336	530.273
8	11/10/99 19:17	860.933	5.48438	478.516	268.262	397.949
9	11/10/99 19:18	855.666	5.48438	479.883	269.238	1575.68
10	11/10/99 19:19	862.71	5.48438	478.125	269.727	1575.68
11	11/10/99 19:20	862.71	5.48438	478.906	269.043	2008.3
12	11/10/99 19:22	856.3	5.48438	478.906	269.531	2008.3
13	11/10/99 19:23	856.3	5.48438	478.711	269.629	2004.88
14	11/10/99 19:24	861.06	5.48438	479.492	269.434	2004.88
15	11/10/99 19:25	860.806	5.48438	478.711	269.531	1990.15
16	11/10/99 19:26	860.806	5.48438	478.32	269.336	1990.15
17	11/10/99 19:27	860.806	5.48438	478.516	269.336	1986.7
18	11/10/99 19:28	860.933	5.48438	478.516	269.141	1997.04
19	11/10/99 19:30	856.871	5.48438	478.32	269.141	1997.04
20	11/10/99 19:31	856.871	5.48438	478.711	268.848	2004.41
21	11/10/99 19:32	856.871	5.48438	478.711	268.945	1997.04
22	11/10/99 19:33	856.871	5.48438	478.125	269.141	997.273
23	11/10/99 19:34	856.871	5.48438	478.32	269.043	997.273
24	11/10/99 19:35	861.377	5.48438	477.734	268.555	-1.88184
25	11/10/99 19:36	861.377	5.48438	478.32	269.141	-1.88184
26	11/10/99 19:37	856.617	5.48438	476.367	268.164	8.00464
27	11/10/99 19:38	856.491	5.48438	477.734	268.75	1004.97
28	11/10/99 19:39	856.491	5.48438	478.32	269.141	1004.97
29	11/10/99 19:40	856.491	5.48438	478.32	269.238	1793.67
30	11/10/99 19:41	861.123	5.48438	478.906	269.238	1093.59
31	11/10/99 19:42	856.237	5.48438	478.516	269.238	762.108
32	11/10/99 19:43	856.808	5.48438	478.516	269.043	270.332
33	11/10/99 19:45	856.808	5.48438	478.125	269.043	-1.41162
34	11/10/99 19:46	856.808	5.48438	478.32	269.043	8.00464
35	11/10/99 19:47	862.075	5.48438	477.734	268.652	-281.324
36	11/10/99 19:49	855.919	5.48438	479.102	268.457	-281.324
37	11/10/99 19:50	862.392	5.48438	477.93	269.531	992.946
38	11/10/99 19:51	862.392	5.48438	478.125	268.75	992.946

Figure 6-10
Incorrect Data Values - One Signal

Data files will occasionally contain a single bad data point for no apparent reason. Figure 6-11 shows an example in which steam pressure is about 985 psig (6791 kPa), and the value for one measurement is instead stored as -4.99 psig (-34.4 kPa). This erroneous value cannot remain in training data sets because it will be selected in the training data as representative of normal behavior. Some on-line monitoring applications such as SureSense provide user-configurable training data quality filters that can detect and remove some of these problem types from the training data automatically.

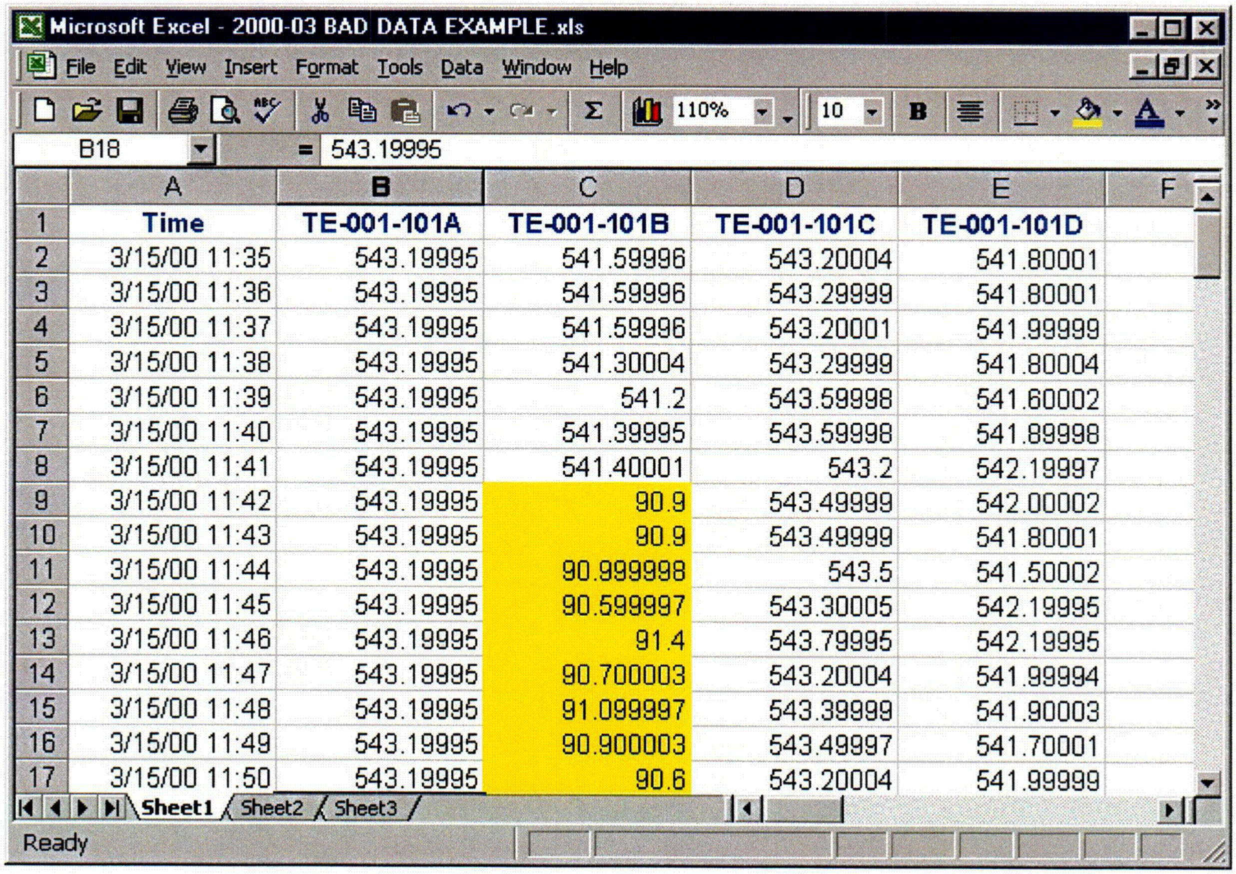
C14

	A	B	C	D
1				
2				
3	Time or	Transmitter	Transmitter	Transmitter
4	Sequence	P0420A	P0421A	P0422A
3334	3330	984.43	986.73	986.72
3335	3331	984.92	987.06	987.86
3336	3332	985.08	987.23	987.21
3337	3333	984.92	987.04	987.04
3338	3334	985.08	987.23	987.70
3339	3335	984.92	986.90	987.04
3340	3336	984.90	987.23	987.86
3341	3337	984.40	986.72	987.21
3342	3338	984.26	986.88	987.04
3343	3339	984.26	-4.99	987.04
3344	3340	984.28	987.04	987.37
3345	3341	983.94	986.57	987.04
3346	3342	984.26	987.04	987.04
3347	3343	984.24	987.04	987.53
3348	3344	984.26	986.90	987.37
3349	3345	984.10	987.04	987.04
3350	3346	984.12	987.06	987.53

Figure 6-11
Incorrect Data Value - One Point

Data quality problems are not always apparent at first review. Figure 6-12 shows an example in which one redundant channel experiences a step-change drop from 541°F to 91°F (283°C to 33°C). The other redundant channels show no significant change. The new values for TE-001-101B are incorrect and are probably caused by a data archive or data extraction error. Notice that another signal is stuck during this period. Figure 6-13 shows a later result when the frozen channel clears from a slightly different value—it takes a step change from 542°F to more than 10,000 °F (283°C to more than 5538°C). This is truly terrible data that requires careful screening before use. Some on-line monitoring applications such as SureSense provide user-configurable training data quality filters that can detect and remove some of these problem types from the training data automatically.

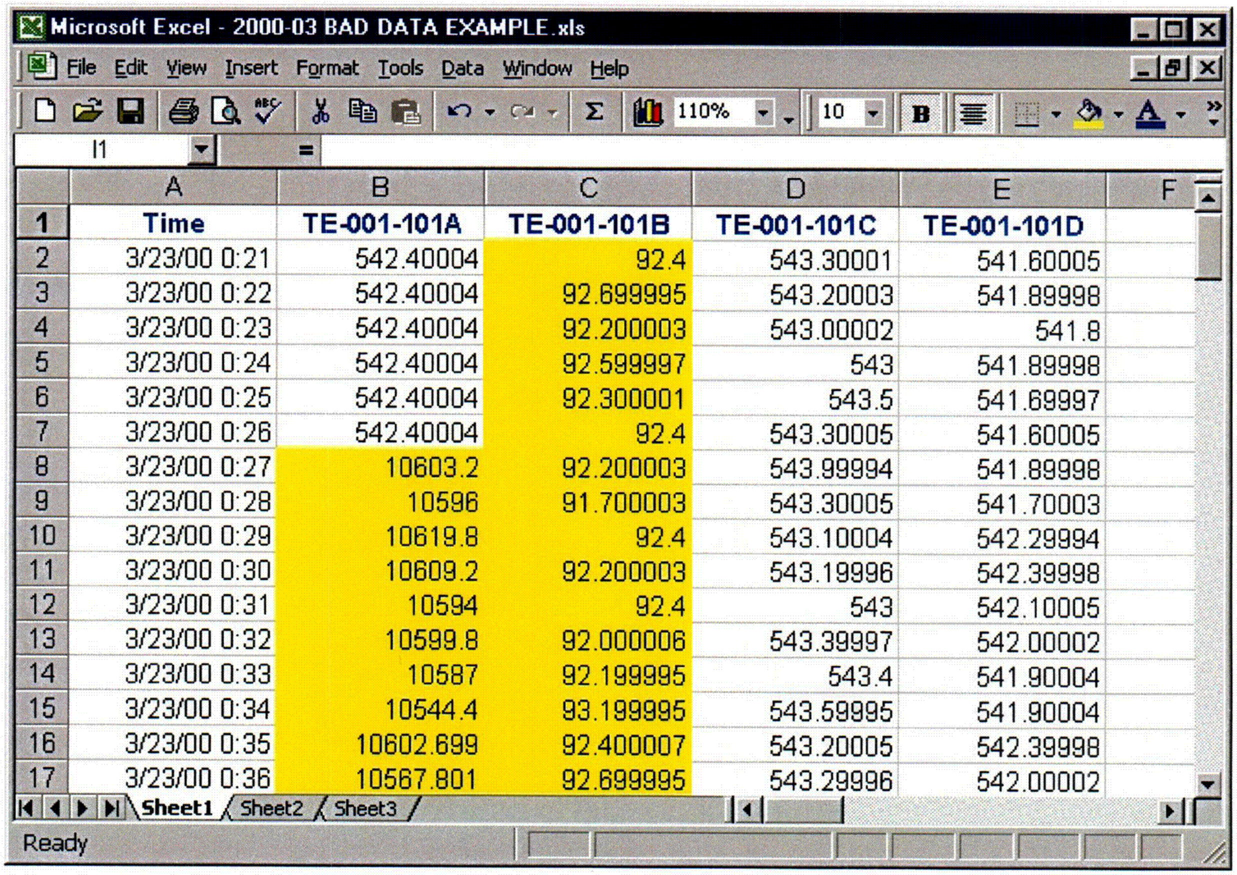
C15



	A	B	C	D	E
1	Time	TE-001-101A	TE-001-101B	TE-001-101C	TE-001-101D
2	3/15/00 11:35	543.19995	541.59996	543.20004	541.80001
3	3/15/00 11:36	543.19995	541.59996	543.29999	541.80001
4	3/15/00 11:37	543.19995	541.59996	543.20001	541.99999
5	3/15/00 11:38	543.19995	541.30004	543.29999	541.80004
6	3/15/00 11:39	543.19995	541.2	543.59998	541.60002
7	3/15/00 11:40	543.19995	541.39995	543.59998	541.89998
8	3/15/00 11:41	543.19995	541.40001	543.2	542.19997
9	3/15/00 11:42	543.19995	90.9	543.49999	542.00002
10	3/15/00 11:43	543.19995	90.9	543.49999	541.80001
11	3/15/00 11:44	543.19995	90.999998	543.5	541.50002
12	3/15/00 11:45	543.19995	90.599997	543.30005	542.19995
13	3/15/00 11:46	543.19995	91.4	543.79995	542.19995
14	3/15/00 11:47	543.19995	90.700003	543.20004	541.99994
15	3/15/00 11:48	543.19995	91.099997	543.39999	541.90003
16	3/15/00 11:49	543.19995	90.900003	543.49997	541.70001
17	3/15/00 11:50	543.19995	90.6	543.20004	541.99999

Figure 6-12
Incorrect Data Values - Unreasonable Change for One Redundant Channel

C16

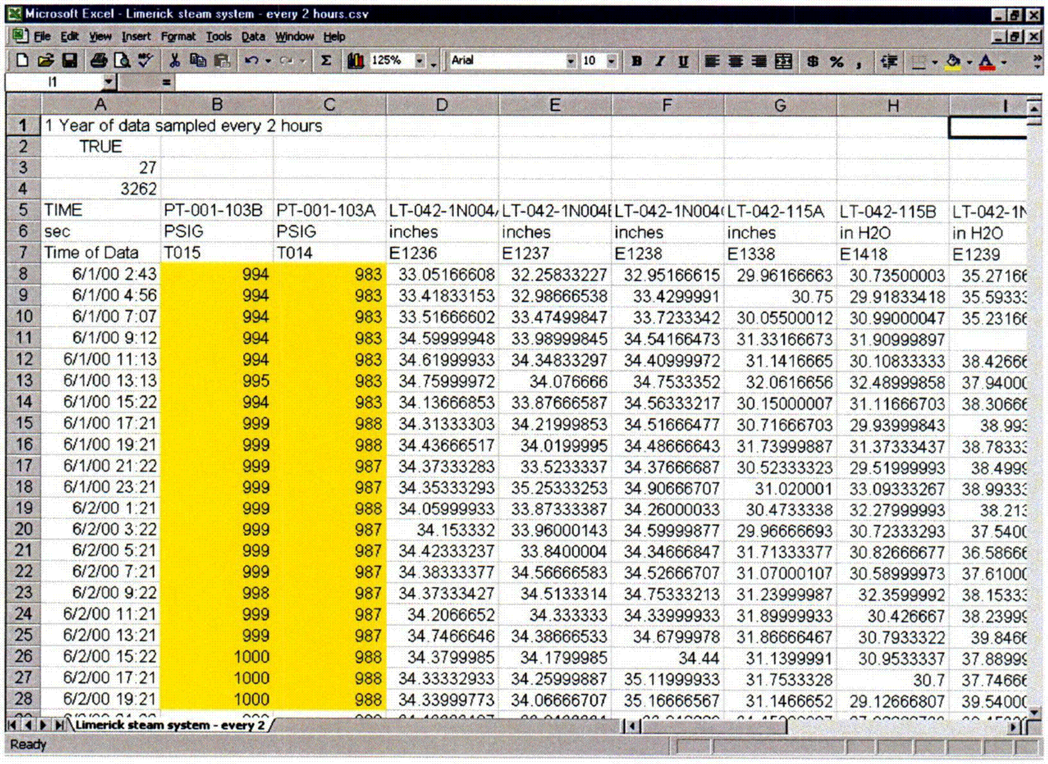


	A	B	C	D	E
1	Time	TE-001-101A	TE-001-101B	TE-001-101C	TE-001-101D
2	3/23/00 0:21	542.40004	92.4	543.30001	541.60005
3	3/23/00 0:22	542.40004	92.699995	543.20003	541.89998
4	3/23/00 0:23	542.40004	92.200003	543.00002	541.8
5	3/23/00 0:24	542.40004	92.599997	543	541.89998
6	3/23/00 0:25	542.40004	92.300001	543.5	541.69997
7	3/23/00 0:26	542.40004	92.4	543.30005	541.60005
8	3/23/00 0:27	10603.2	92.200003	543.99994	541.89998
9	3/23/00 0:28	10596	91.700003	543.30005	541.70003
10	3/23/00 0:29	10619.8	92.4	543.10004	542.29994
11	3/23/00 0:30	10609.2	92.200003	543.19996	542.39998
12	3/23/00 0:31	10594	92.4	543	542.10005
13	3/23/00 0:32	10599.8	92.000006	543.39997	542.00002
14	3/23/00 0:33	10587	92.199995	543.4	541.90004
15	3/23/00 0:34	10544.4	93.199995	543.59995	541.90004
16	3/23/00 0:35	10602.699	92.400007	543.20005	542.39998
17	3/23/00 0:36	10567.801	92.699995	543.29996	542.00002

Figure 6-13
Incorrect Data Values - Unreasonable Change for a Second Redundant Channel

Signal data should have a resolution of several decimal places. The method by which data are stored in archives or obtained from archives sometimes results in truncating the signal data, resulting in integer data as shown in Figure 6-14. Figure 6-15 shows the effect of this truncation on the fault-detection analysis. MSET continues to produce estimates (shown as red triangles) between the integer data values (shown as blue crosses), which results in spurious fault-detection alarms. In general, the estimation results cannot be more accurate than the data used for training.

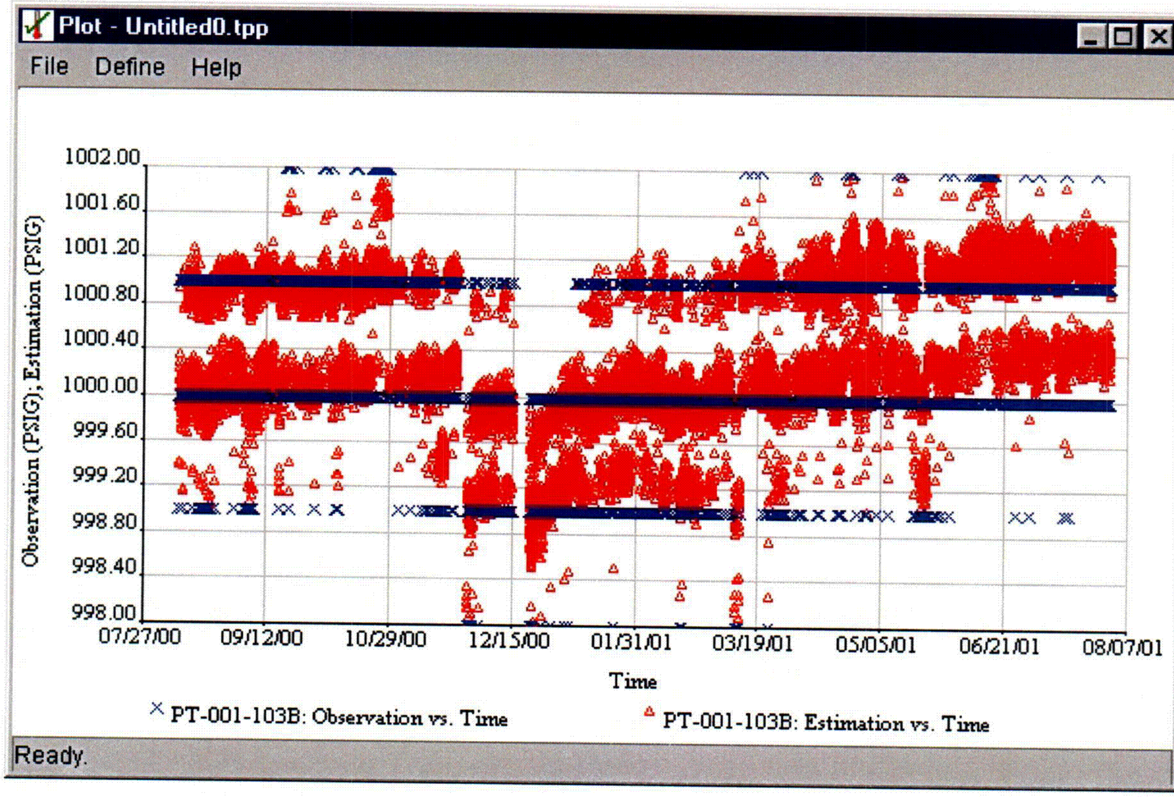
C17



	A	B	C	D	E	F	G	H	I
1	1 Year of data sampled every 2 hours								
2	TRUE								
3	27								
4	3262								
5	TIME	PT-001-103B	PT-001-103A	LT-042-1N004	LT-042-1N004	LT-042-1N004	LT-042-115A	LT-042-115B	LT-042-1N
6	sec	PSIG	PSIG	inches	inches	inches	inches	in H2O	in H2O
7	Time of Data	T015	T014	E1236	E1237	E1238	E1338	E1418	E1239
8	6/1/00 2:43	994	983	33.05166608	32.25833227	32.95166615	29.96166663	30.73500003	35.27166
9	6/1/00 4:56	994	983	33.41833153	32.98666538	33.4299991	30.75	29.91833418	35.59333
10	6/1/00 7:07	994	983	33.51666602	33.47499847	33.7233342	30.05500012	30.99000047	35.23166
11	6/1/00 9:12	994	983	34.59999948	33.98999845	34.54166473	31.33166673	31.90999897	
12	6/1/00 11:13	994	983	34.61999933	34.34833297	34.40999972	31.1416665	30.10833333	38.42666
13	6/1/00 13:13	995	983	34.75999972	34.076666	34.7533352	32.0616656	32.48999858	37.94000
14	6/1/00 15:22	994	983	34.13666853	33.87666587	34.56333217	30.15000007	31.11666703	38.30666
15	6/1/00 17:21	999	988	34.31333303	34.21999853	34.51666477	30.71666703	29.93999843	38.993
16	6/1/00 19:21	999	988	34.43666517	34.0199995	34.48666643	31.73999887	31.37333437	38.78333
17	6/1/00 21:22	999	987	34.37333283	33.5233337	34.37666687	30.52333323	29.51999993	38.4999
18	6/1/00 23:21	999	987	34.35333293	35.25333253	34.90666707	31.020001	33.09333267	38.99333
19	6/2/00 1:21	999	988	34.05999933	33.87333387	34.26000033	30.4733338	32.27999993	38.213
20	6/2/00 3:22	999	987	34.153332	33.96000143	34.59999877	29.96666693	30.72333293	37.5400
21	6/2/00 5:21	999	987	34.42333237	33.8400004	34.34666847	31.71333377	30.82666677	36.58666
22	6/2/00 7:21	999	987	34.38333377	34.56666583	34.52666707	31.07000107	30.58999973	37.61000
23	6/2/00 9:22	998	987	34.37333427	34.5133314	34.75333213	31.23999987	32.3599992	38.15333
24	6/2/00 11:21	999	987	34.2066652	34.333333	34.33999933	31.89999933	30.426667	38.23999
25	6/2/00 13:21	999	987	34.7466646	34.38666533	34.6799978	31.86666467	30.7933322	39.84666
26	6/2/00 15:22	1000	988	34.3799985	34.1799985	34.44	31.1399991	30.9533337	37.88999
27	6/2/00 17:21	1000	988	34.33332933	34.25999887	35.11999933	31.7533328	30.7	37.74666
28	6/2/00 19:21	1000	988	34.33999773	34.06666707	35.16666567	31.1466652	29.12666807	39.54000

Figure 6-14
Loss of Significant Digits

C18



1 psi = 6.894757 kPa

Figure 6-15
Effect of Loss of Significant Digits

6.2.3 Removing Bad Data From Data Sets

The previous section provides several examples of bad data in actual data sets. Bad data must be removed from any files used for training; it is beneficial to remove bad data from historical data sets used for verification testing. Data representing actual sensor problems or component failures must also be removed from the training data. Our intention is not to train the system to treat bad data or signal failures as normal. Regardless of whether the data are bad or represent an actual sensor or equipment problem, the data do not belong in the training data set.

C19

It is recommended that the following points be considered and followed for the identification and removal of bad data:

- Many users will generate their data sets in a form readily accessible by Microsoft Excel or similar spreadsheet programs. The Microsoft Excel conditional format tool can be used to identify missing data or data that are outside of an expected range. It is necessary to highlight the data to be checked and to specify the conditional format criteria. This approach is valuable for data sets containing blanks or bad data that are well outside of the expected operating range. Figure 6-16 shows an example of the conditional format feature. In this example, any cells with a value of less than 1.0 are highlighted in yellow, and any blank cells are highlighted in red.

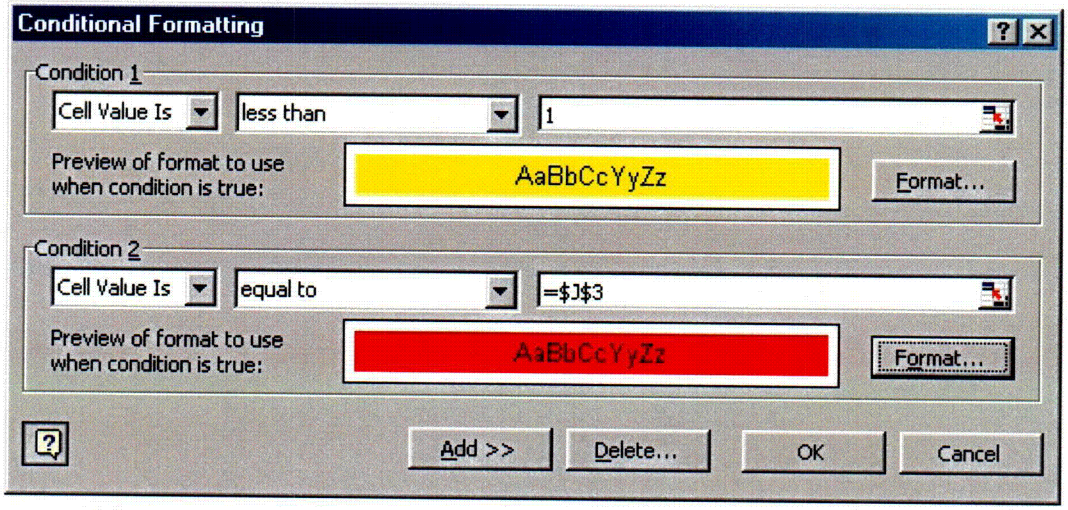


Figure 6-16
Example of Conditional Format Feature

- The data set should be reviewed for data lockup or frozen data. This check is usually performed visually (although it could be checked using a macro that calculates a running standard deviation). Frozen data left in the training set might be selected as part of the training vectors. The estimation technique would then treat this frozen data as normal behavior. Any rows containing frozen data should be removed. The rows just before and just after the data that were removed should be reviewed because these rows might also have problems. The value of this check will vary from plant to plant, depending on how the computer system acquires and stores data. In some computer systems, data lockup is readily identifiable because all signals randomly vary in value each minute, and the computer stores the measured value at the sampling frequency. In other computer systems, data values are updated in storage only as values vary by more than some specified amount from the last reading. In these systems, most or all signals tend to remain unchanged for some period of time and then experience a step change to the next value.
- After removing obviously bad data, basic statistics for the remaining data should be reviewed to identify any other potential problems. Additional rows of data might be removed by this screening.

C20

- It might sometimes be difficult to distinguish between bad data and actual problems with sensors or other signal conditioning components. Bad data will usually "heal" itself at some point, whereas failed sensors rarely recover.
- Data limits should be applied in the SureSense signal definition settings to keep data that are outside of the possible range of sensor operation from being considered in the training set.
- The training data should be tested after the model has been trained to review the data quality. If necessary, additional outliers in the training data set should be removed. Even if outlying measurements appear to be real data rather than bad data caused by data acquisition errors, the following question should be asked: Should the model be trained on this data, thereby allowing the model to treat this data as normal behavior?
- The basic data statistics should be reviewed to check for data that are outside the expected range. Note that this check is distinctly different than using the data limits described previously. Simply stated, this review is intended to identify the presence of data that physically do not make sense. For example, a reactor coolant system temperature of 200°F (93°C) is not possible if the reactor is operating at 100 percent power. Steam system temperature cannot be 10,000°F (5538°C). Similarly, an emergency water storage tank (such as a refueling water storage tank), probably will not have a low level when the plant is operating at power. Turbine first-stage pressure will not be negative when the reactor is at 100 percent power. These and other examples have occasionally been found in the data obtained from the plant computer archive.
- The data should be reviewed for unreasonable signal value step changes. For example, tank level cannot instantaneously change from full to empty, and reactor power cannot step change from 25–100 percent power. This check works best with data acquired at a relatively high frequency such as every minute. If the data are acquired at longer intervals, larger step changes in the data might well be possible.

Bad data *must* be removed from training data sets. A question that is frequently asked is whether bad data also need to be removed from historical data sets used for testing. Bad data in testing sets degrade the model performance and hinder the evaluation of the model. Each time the model is used, the location of the bad data must be remembered. Questions regarding the bad data will continue to come up with each review. It is recommended that all historical data files be cleaned up before use regardless of whether they are used for training or testing.

6.2.4 Data Limit Filters

SureSense can also be used to automatically identify signal data that are outside of a reasonable minimum or maximum range, that exhibit regions of abnormally high or low standard deviation, or that exhibit unreasonably large positive or negative changes in value between successive observations. These tasks are collectively handled through the use of data limit filters.

Data limit filters for each signal provide an automated method for screening out bad data. The following four types of data limit filters are provided in SureSense:

- Range - Minimum and maximum reasonable observed signal value
- Delta - Maximum positive and negative change in sequential observed signal values
- Noise - Minimum and maximum standard deviation of observed signal values
- Not Present - Not present or invalid signal as marked by the data source

Each filter type is specific to both a signal and a phase (each of which is optionally specified by the user during signal definition). For each observation used in training or on-line monitoring, the limit-filter procedure is performed on its specified signal. The limit-filter procedures return the results for each signal indicating the results of each type of limit test. A *FAIL* result is returned if the signal fails the test.

Limit filters can be selectively enabled or disabled for both training and on-line monitoring. The net effect of these filters is to exclude any data outside the specified limits from training or to declare a signal fault during on-line monitoring if enabled. The following information describes each of the limit filters:

- *MIN_RANGE* – A lower limit on the value of the signal. User-definable constants are the minimum reasonable value (*Minimum Value*) expected for the signal and the number (*Filter Interval*) of consecutive single-cycle alarms required to generate a *FAIL* indicator. To return a *FAIL* indicator for each and every observation within which the lower reasonableness limit is violated, the interval must be set to 1. To filter spurious limit violations and return *FAIL* only when the signal stays out of range for multiple observations, set the interval to an integer greater than 1.
- *MAX_RANGE* – An upper limit on the value of the signal. User-definable constants are the maximum reasonable value (*Maximum Value*) expected for the signal and the number (*Filter Interval*) of consecutive single-cycle alarms required to generate a *FAIL* indicator. To return *FAIL* for each and every observation within which the upper reasonableness limit is violated, the interval must be set to 1. To filter spurious limit violations and return *FAIL* only when the signal stays out of range for multiple observations, the interval must be set to an integer greater than 1.
- *MAX_POS_DELTA* – An upper limit on the positive change in value for a signal between two consecutive observations. The user-definable constant is the limit value (*Positive Limit*). After the first observation is acquired in a new applicable phase, the difference between the current observation and the previous observation is computed. Missing values are ignored, and the most recent previous value is used to compute the change in value for the current observation. The delta value (current - previous) is compared to the absolute value of the filter's positive change limit value. If the delta value is greater than the absolute value of the limit, a *FAIL* condition is returned.

- *MAX_NEG_DELTA* – An upper limit on the negative change in value for a signal between two consecutive observations. The user-definable constant is the limit value (*Negative Limit*). After the first observation is acquired in a new applicable phase, the difference between the current observation and the previous observation is computed. Missing data values are ignored, and the most recent previous value in the current phase is used to compute the change in value for the current observation. The delta value (current - previous) is compared to the negative of the absolute value of the filter's negative limit value. If the delta value is less than the maximum negative limit, a *FAIL* condition is returned.
- *MAX_NOISE* – An upper limit on the standard deviation of a time series of signal values. User-definable constants are the standard deviation limit value (*Maximum Standard Deviation Limit*), the number of consecutive observations used to compute the standard deviation (*Standard Deviation Window Size*), the delay on entering a new phase (*New Phase Delay Interval*) before filter application is initiated, and an option control flag (*Remove Trend Line*). The algorithm computes the standard deviation of a time series of observations for the signal after adjusting for an optional line fit to the data. If the current standard deviation value exceeds the *Maximum Standard Deviation Limit*, a *FAIL* condition is reported.
- *MIN_NOISE* – A lower limit on the standard deviation of a time series of signal values. User-definable constants are the standard deviation limit value (*Minimum Standard Deviation Limit*), the number of consecutive observations used to compute the standard deviation (*Standard Deviation Window Size*), the delay on entering a new phase (*New Phase Delay Interval*) before filter application is initiated, and an option control flag (*Remove Trend Line*). The algorithm computes the standard deviation of a time series of observations for the signal after adjusting for an optional line fit to the data. If the current standard deviation value is less than the *Minimum Standard Deviation Limit*, a FAIL condition is reported. The operation of the settings is the same as for the *MAX_NOISE* filter. If *Remove Trend Line* is checked, the standard deviation is calculated with respect to a line fit to the data in the window. Otherwise, it is calculated with respect to the mean of the data in the window.
- *NOT_PRESENT* – Checks whether the signal value matches the *Not Present* value exactly. If a match is found, the value is treated as a flag that the signal is not present or has been marked as invalid by the data source. If the signal is not present, the observation containing the *Not Present* value receives special handling during processing. The entire observation will be ignored during training. During monitoring or the various analysis routines, those procedures that depend on the *Not Present* value will be modified or skipped.

Limits provide an additional method of screening data to ensure that the best possible data quality is allowed. If a signal value is outside its specified range, that vector of data (the observation) is excluded from training if enabled. During on-line monitoring, a signal-specific fault is declared when a limit filter *fail* condition is detected.

If data limits are set too close, a significant amount of data can be excluded from training. The data limits are intended to exclude obviously bad data and should be established on the basis of the full range of reasonably expected data values for each of the defined phases. Similarly, if data limits are set too close, false alarms might occur during on-line monitoring.

6.3 Data Archive Historian and Its Effect on Data Quality

6.3.1 Problem Statement

Some data acquisition systems apply a data compression technique, included as part of a data historian, to minimize archive file size. If the signal value does not vary outside a specified range (depending on the computer system, referred to as the *data compression limit, factor*, or *tolerance*), the value is assumed to be unchanged. When the value eventually exceeds the data compression limit, the signal value is updated in storage. When the data are later extracted from archive, a linear interpolation routine is used to derive intermediate values between the recorded values. By this approach, a significant reduction in file storage size can be realized for archived data.

This data compression technique has an adverse effect on data quality with respect to on-line monitoring. The following list highlights the main problems:

- The linear interpolation routine generates artificial data between recorded data points, which affects the apparent correlation between signals. In some cases, signals that are known to have a high physical correlation can have data with effectively no apparent correlation. Section 5.3 explains why correlation is important in an on-line monitoring system.
- Training data can consist of a combination of real data and artificial data with a degraded correlation between signals. This degrades the overall training quality.
- Analysis of training and test data can be no more sensitive than the applied compression tolerance. In some cases, the compression limit has been set at several percent. This can cause a substantial reduction in the on-line monitoring system sensitivity.

6.3.2 The Effect of a Data Archive Historian on Signal Correlation

A data historian reduces the size of the archive data file by the following process:

1. A tolerance limit is specified for each signal.
2. Data are acquired by the computer system at some frequency.
3. The current data point is compared to the previous data point for that signal. If the difference between the data points exceeds the specified tolerance, the latest data point is stored. If the difference between the data points is less than the specified tolerance, the latest data point is not stored.
4. Data are added to the archive by the this process only if the change in a data point exceeds the tolerance. When data are later retrieved, data for the missing sample times are generated by a linear interpolation between whatever data were stored. The generated data are artificial because they are, at best, estimates of the real value based on a simple linear assumption about the data behavior.

Figure 6-17 shows an example of data archive historian data. Notice that random fluctuations have been filtered out of the data and the signal varies linearly between data points. Referring to Figure 6-17, notice the change in power level from April 6 to April 7. It appears that power was carefully increased from 99.9–100.2 percent in a linear manner over this one-day period. Actually, power was so constant at near 100 percent during this period that only two data points were stored; the rest of the data were artificially generated.

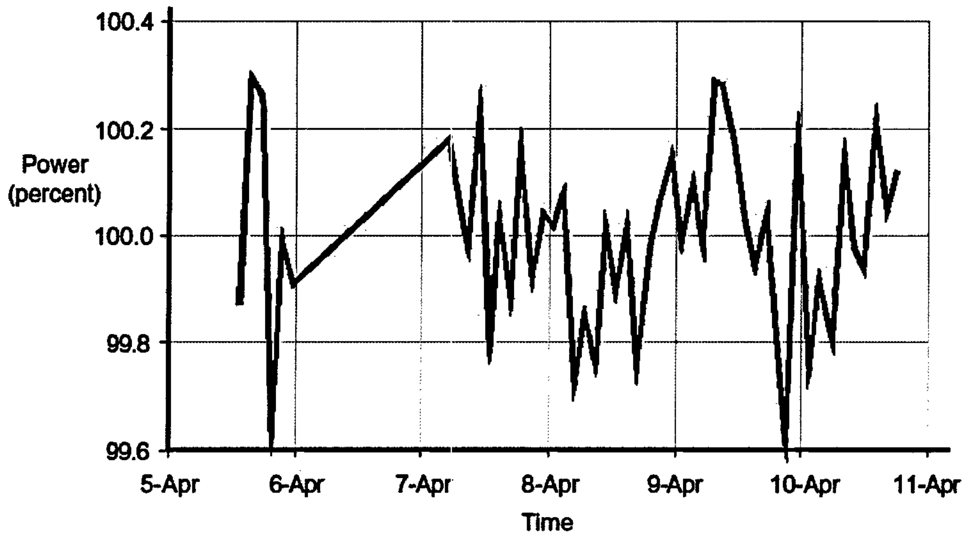


Figure 6-17
Typical Example of Compressed Historian Data

Figure 6-18 shows another example of historian archive data. Notice that the signal varies linearly between data points, clearly indicative of a linear interpolation routine in a data historian. Notice also that the compression limit is several percent, which adversely affects the possible model accuracy.

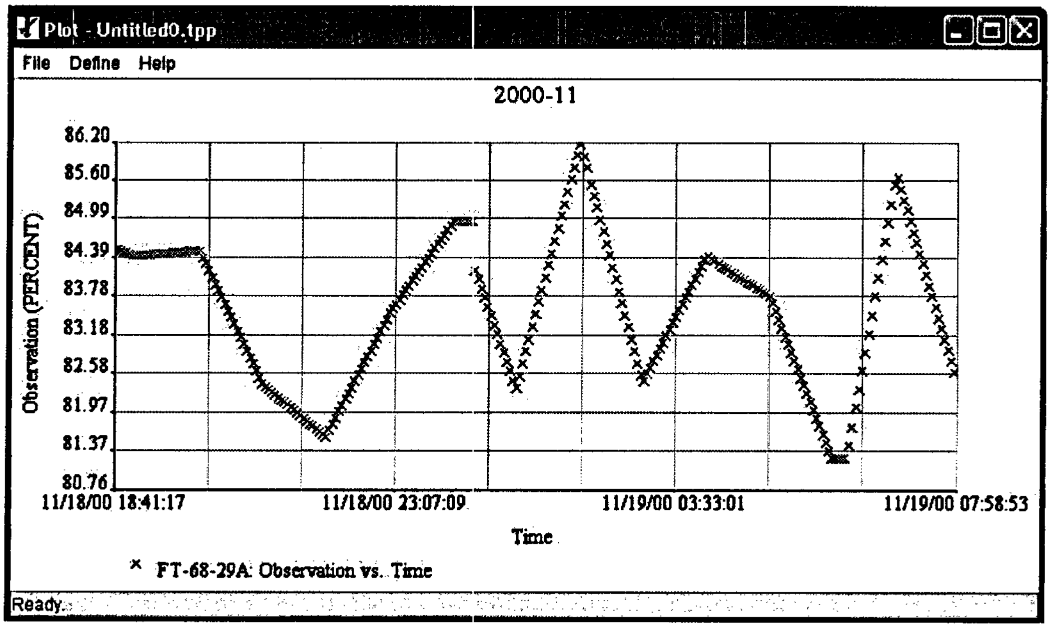


Figure 6-18
Historian Data Example - Flow Signal

An on-line monitoring method such as MSET depends on the behavioral pattern observed between signals. Included in this pattern recognition is an assumption that a model can be developed based on the learned behavior of a group of correlated signals. For example, if a model consists of four signals that have a positive linear correlation, the model would be trained (hopefully) on data that show how the signals tend to perform as a group. MSET develops an estimate for one signal based on the observed values of the other signals.

The historian data compression technique has the effect of degrading the correlation between signals, even if the raw data are highly correlated. The degree to which the correlation is degraded depends on the specified compression limit. As the limit is made larger, fewer data are stored in the archive with the result that most of the data retrieved from the archive are artificially generated by the linear interpolation routine.

As an example, Figure 6-19 shows three signals that have a high linear correlation. Data for the three signals were created so that the signals are correlated with some noise content by the following equations:

$$Signal\ 1 = sin(t) + random\ number$$
$$Signal\ 2 = sin(t) + random\ number$$
$$Signal\ 3 = sin(t) + random\ number$$

Figure 6-19 shows the result in which the three signals vary in a sinusoidal manner together with noise added by the random number. With the noise content, the correlation coefficient for these signals is about 0.7.

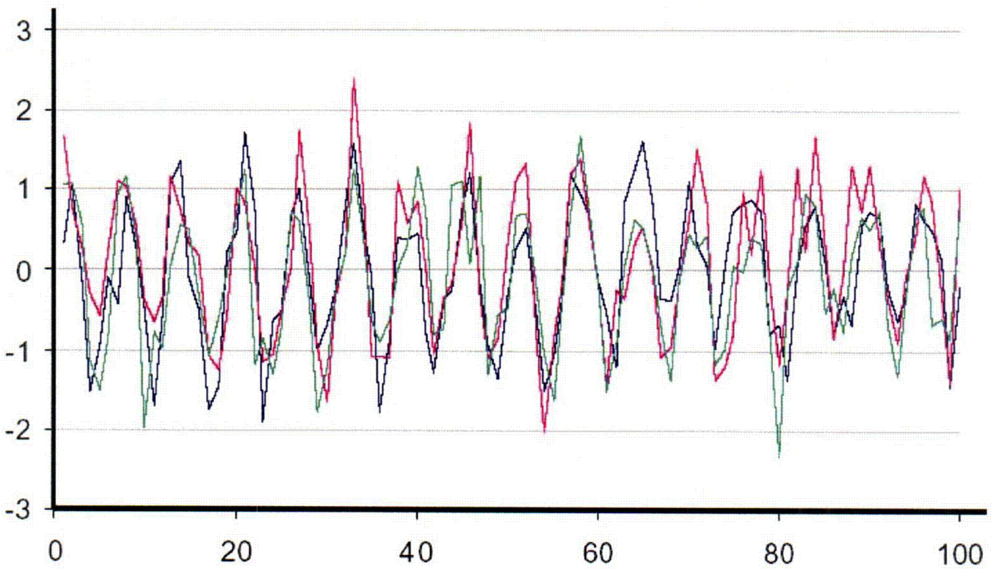


Figure 6-19
Sample Data With Random Variation Included

The data historian compressed the data in Figure 6-19 by specifying a compression limit of 1.2. With this tolerance, the original 300 data points were reduced by about two-thirds, leaving just over 100 data points. A linear interpolation routine was applied to the gaps in the data to retrieve the stored data. The results are shown in Figure 6-20. By generating artificial data, the correlation coefficient for these signals is reduced from about 0.7 to 0.3. With actual plant operating data acquired from such a data historian, it is common to find data correlations between physically correlated signals to be near zero.

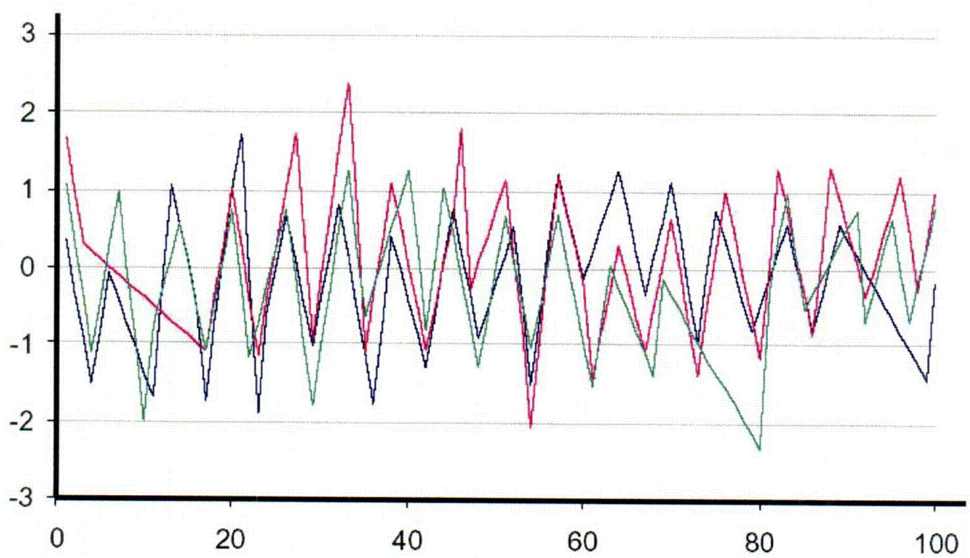


Figure 6-20
Sample Data After Data Compression and Subsequent Archive Retrieval

C21

6.3.3 The Effect of Data Archive Historian With Bad Data

Anomalies can occur in the data acquisition system that result in bad or missing data. A common problem occurs with data dropouts in which the signal value falls to zero for some period of time. Data dropouts can also occur in which the value falls to some low level, but still greater than zero. Data spikes can also occur.

Figure 6-21 shows an example of a data dropout. Three values are recorded at different times—a normal value, an anomalous zero value, and the normal value again. The period in which the signal was unavailable is almost two days. Notice that the historian linear interpolation routine generates artificial data down to zero followed by artificial data up again to the normal value. This type of data error is readily observable in plots of the data by the characteristic slanted "*V*" shape.

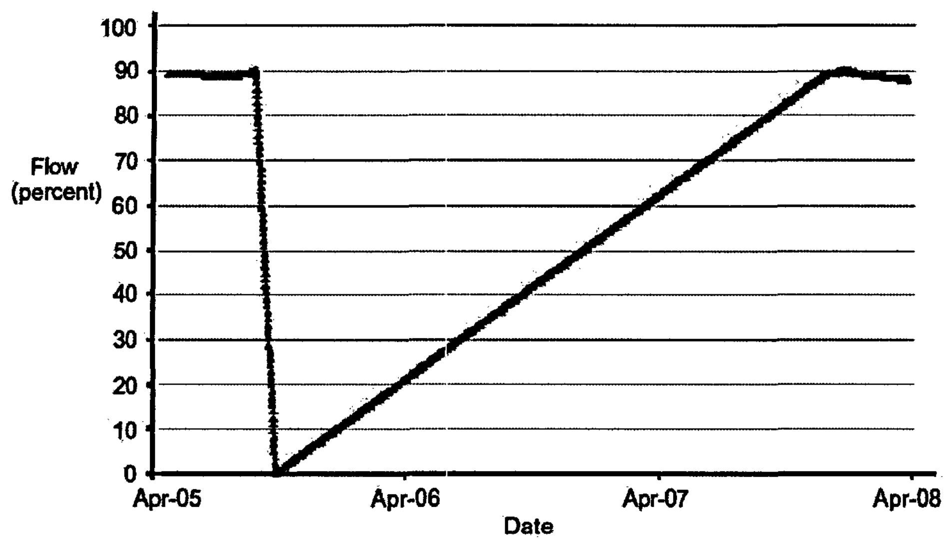


Figure 6-21
Data Interpolation Errors

Data dropouts in a historian are particularly detrimental to on-line monitoring. If the type of data shown in Figure 6-21 is left in training data sets, the on-line monitoring system will train the model to recognize this incorrect behavior as normal.

6.3.4 Dealing With a Data Archive Historian

As shown in the previous sections, a data archive historian can degrade the natural correlation between signals by generating artificial data between stored data points. The following steps are recommended for systems that utilize this data compression feature:

- Disable the historian, and allow the system to acquire data at some specified frequency.
- If the historian cannot be disabled, lower the compression limit to the smallest possible setting for each signal. Evaluate the data after reducing all tolerance settings, and determine if the correlation between signals is acceptable.

Models are generally created in batch mode using historical data. Recognize that archived data will be subject to the problems just described. Hopefully, the models will perform well enough with this historical data to allow model settings to be defined adequately.

7
INITIAL TRAINING AND ESTIMATION

The MSET method is based on developing a training data matrix from which process value estimates are calculated for comparison with corresponding data observations. Although they are treated separately, initial training, fault detection, and retraining are directly related. The quality of the initial training data affects how faults are identified and includes the number of false alarms. Fault identification during model development subsequently leads to retraining in which expected operating states not covered by the initial training are captured. In other words, the evaluation of failure alarms often identifies operating states not adequately described by the data used for initial training.

Section 7 describes initial training (including how estimates are developed). Section 8 continues this discussion by describing fault detection, the evaluation of identified faults, and retraining.

7.1 Training and Estimation Methods - Technical Overview

7.1.1 Training

After a comprehensive and error-free set of training data has been assembled, training algorithms are used to build a diagnostic model of the selected signals. Training is a two-step procedure that begins with the development of an MSET parameter estimation model followed by calibration of the fault detectors.

Training characterizes the expected behavior of signals in a model using historical operating data. Specified data files are used for training; MSET selects those observation vectors from the files that best define the expected operating space. An observation vector is a complete set of signal data values for a given point in time. For example, if the data are contained in a typical spreadsheet, an observation vector is one row of data.

Two types of training methods are currently available in MSET:

- MinMax
- Vector ordering

The following sections describe each of these methods.

7.1.1.1 MinMax Training

The MSET training procedure evaluates the training data and selects a subset of the observations that are determined to best characterize expected operation for the selected signals. The MinMax procedure is used to identify and select the observations containing the minimum and maximum observed values for each included signal. The minimum and maximum observed values define the boundaries of the valid operating range of the MSET model. The selected observations are placed in the MSET training matrix, which is also known as the D-matrix (from its mathematical description).

MinMax builds the smallest possible trained model and is alone only suitable for a model having a very limited operating state space (in fact, no models described in this report use MinMax alone for training). The MinMax option bounds the operating state space represented in the training data by selecting the extreme value vectors for each included signal. Thus, two vectors are selected for each signal—the vectors containing the smallest value and the largest value. In many instances, the selected vectors for other signals will be the same; when one signal is at its extreme value, other signals might also be at their extreme values. Therefore, MinMax selects between 2 and 2*n* vectors, where *n* is the number of included signals. Typically, the MinMax option chooses a number of vectors that is greater than the number of signals, but less than twice the number of signals.

The method by which MinMax selects training vectors and subsequently affects estimation is important to understand. Suppose the training data used in a model contain 2998 vectors for four signals. If MinMax is used for training, eight vectors might be selected for training as shown in Table 7-1. (From the possible 2998 vectors, the minimum and maximum vector for each signal is highlighted in yellow and green, respectively.)

Table 7-1
Level Example Vectors Selected by MinMax

Signal	Vector #1	Vector #2	Vector #3	Vector #4	Vector #5	Vector #6	Vector #7	Vector #8
LT-01	62.950	62.963	63.174	63.439	63.680	63.590	63.590	63.351
LT-02	62.611	62.598	62.722	62.976	63.200	63.212	63.164	62.888
LT-03	61.728	61.716	61.653	61.806	62.080	62.030	62.118	61.816
LT-04	58.738	58.725	58.798	58.486	58.735	58.848	58.926	59.065

The following observations are important to consider here:

- Out of 2998 vectors available for training, MinMax selected only the eight vectors that bounded the signals in the data set. By this method, the boundary of the operating state space is mapped, but the interior states are not characterized at all.
- Regardless of whether the operating space covers a small or a large region, MinMax defines only the extreme boundaries of the region. If the operating space is large, the estimates produced from the MinMax training vectors will likely include significant uncertainty.

C22

MinMax selects between 2 and $2n$ vectors, where n is the number of included signals. For highly correlated signals, one would expect that one signal is at its maximum (or minimum) value at the same instant that the other signals are also at their maximum (or minimum) values. This line of thought would lead one to expect that the number of vectors selected would routinely be close to the minimum number of two. This turns out not to be the case for typical power plant data. For the models presented in the appendices with n signals, the number of vectors selected by MinMax tends to vary anywhere from n to $1.5n$ vectors. There are enough random variations in the data that multiple minimum and maximum vectors are selected. Oddly enough, this seems to be relatively insensitive to the size of the training file. For example, a boiling water reactor (BWR) steam system model contains 39,566 vectors of data in the training file for the 19 signals; the MinMax algorithm selected 33 vectors. If the data in the training set are sampled at every 150 points, the number of available vectors is reduced to only 263 vectors of data, yet MinMax still selected 32 vectors.

MinMax is not recommended as the sole training method. In the following section, a training method is described that uses MinMax to define the operating state space boundaries, then fills in the training matrix with additional vectors from the interior operating states.

7.1.1.2 Vector Ordering Training

After MinMax vector selection is complete, the training procedure selects a number of additional points that best characterize the model's operating states between the minimum and maximum limits. The method used to fill in the additional states is a statistical procedure known as the *vector ordering* technique. The procedure begins by first ordering the training data observations based on the weighted value of each vector (described in the following paragraph). The procedure then selects observations at equal intervals using a spacing criterion (excluding those vectors previously selected by the MinMax procedure) to fill the training matrix with the user-specified number of unique observation vectors.

The vector ordering technique operates as described here.

First, the algorithm calculates an ordered vector $\vec{\mathrm{E}} = \left|E_1, E_2, \ldots, E_p\right|$ such that the elements of the vector are sorted in ascending magnitude for each of P observations in the training data. The elements of $\vec{\mathrm{E}}$ are each the weighted value of the individual observation vectors in the training data. The weighted values are computed as the square root of the sum of the squares of the signal data values contained in an observation vector, or

$$\mathrm{E}_j = \sqrt{\sum_{i=0}^{N} x_i^2}$$

where X_i is the value of signal i within the observation vector and $\mathrm{E_j}$ is the calculated weighted value for vector j.

Note: One consideration for the vector ordering procedure is the average value of each signal in the model. If the input data have been normalized, each signal will have approximately equal weight in the vector ordering method. If the input data have not been normalized, the larger-

valued signals will have greater influence in the vector ordering method. The feature is software specific and should be discussed with the software provider. As a rule of thumb, data should be normalized prior to performing vector ordering to eliminate these effects.

The number of vectors chosen depends upon a user-specified spacing parameter, *F*, which ranges between 0–1. The selection procedure begins by selecting the column vector that corresponds to element E_1. It then loops through each element *j* in the vector $\vec{E}$. The algorithm finds the next vector element E_j that satisfies the equation:

$$E_j - E_{prev} > F\left(E_P - E_1\right)$$

where E_{prev} is the element of the vector $\vec{E}$ that was previously selected (initially $E_{prev} = E_1$).

The value of *F* determines the number of vectors that are selected. SureSense automatically computes the value of *F* that will result in a training matrix of the user-specified size. All vectors selected are compared to those vectors previously selected by the MinMax procedure. Only those vectors that were not already included by the MinMax procedure are added to the training matrix.

The final step in the MSET model training procedure is a matrix inversion operation that computes a similarity matrix (also known as the G-inv matrix) from the training matrix. The similarity matrix is used during monitoring to compute a measure of similarity or overlap between a new observation and the observation vectors stored in the training matrix. The matrix inversion operation is a computationally intensive procedure that prepares the MSET model for on-line monitoring use. The matrix inversion operation is required only once during the training step.

In summary, vector ordering begins with MinMax and fills in the operating space with a user-defined number of additional training vectors. For example, the previous section provided an example for four signals in which MinMax selected eight vectors. If 20 vectors are specified for the vector ordering option, the eight MinMax vectors will be selected as well as 12 additional vectors. The additional training vectors are intended to be as evenly spaced as possible across the operating state space so that the estimation technique is likely to have training vectors available for pattern matching that are reasonably close to observations to be validated.

For steady-state operation, the model might not need a large number of vectors in the training matrix. However, for a larger operating space, a correspondingly larger number of vectors might be needed to cover the operating space adequately.

The training matrix size (number of training vectors selected) is related to the estimation uncertainty or modeling error. Using too few vectors reduces the estimation accuracy because of inadequate coverage of the operating state space. Using too many vectors “overfits” the model to include noise from the training data. Figure 7-1 shows an example of the estimation error as a function of the number of user-specified training vectors. There is usually a point of diminishing returns beyond which the model starts including noise in the training data. In this example, the point of diminishing returns occurs at approximately five times the MinMax size. The observation processing time varies with the square of the number of vectors, which provides additional motivation for not specifying too many vectors.

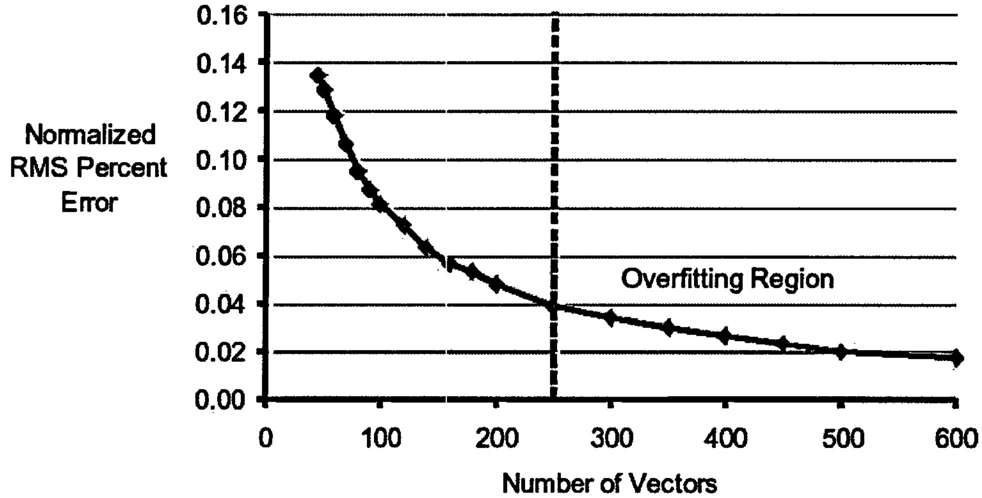


Figure 7-1
Estimation Error as a Function of Vector Specification

The training matrix size also affects the susceptibility for overfitting and spillover as described here.

Overfitting is a tendency for the estimate to follow a signal disturbance. Models exhibiting overfitting typically contain groups of poorly correlated signals or a limited state space with a large number of training vectors. Overfitting delays fault detection (reduces sensitivity) because the estimate follows the observations for some period of time. Overfitting can be minimized by improving the level of correlation among the signals in the model or by reducing the number of training vectors so as to minimize the modeling of noise. Another common approach to preventing overfitting is regularization. While some empirical models incorporate differing levels of regularization, the standard MSET algorithm does not.

Spillover is a tendency for the estimate of one signal to follow a disturbance in a second highly correlated signal. Models exhibiting spillover typically contain groups of highly correlated signals or very few signals. Spillover is the most common cause of false alarm problems and can be minimized by desensitizing the model for the affected signals. Methods of minimizing spillover effects include adjusting model settings, adding more signals to the model, or possibly increasing the number of training vectors.

Typically, more vectors should be specified for transient data than for steady-state data. Consider the following guidelines for the training matrix size (the software user's guide provides additional information):

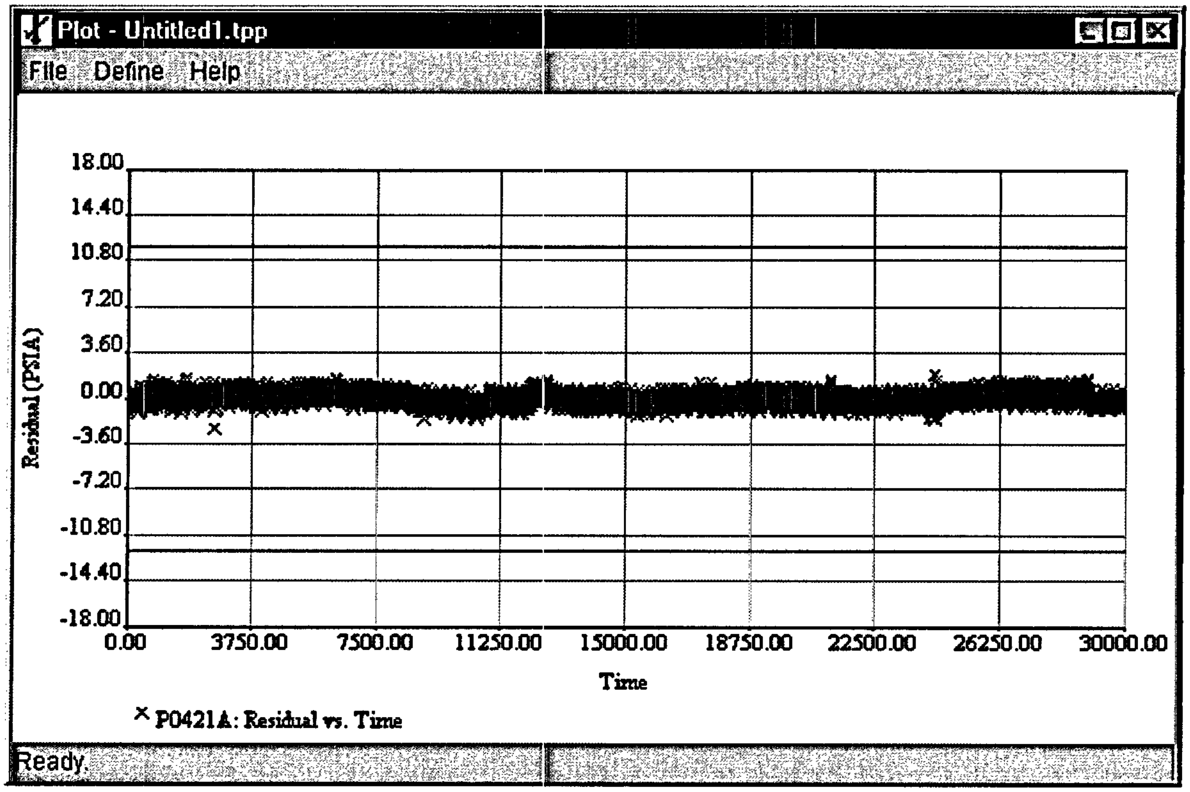
- 4–10 times the size of the MinMax size for transient data with a large operating state space
- 3–6 times the size of the MinMax size for steady-state data with a large operating state space
- 2–4 times the size of the MinMax size for steady-state data with a relatively small operating state space

7.1.1.3 Fault Detector Calibration as the Final Step in Training

Training results are stored in an MSET parameter estimation model that is used during system monitoring to estimate the expected values of the signals. However, before the fault detectors can be used, they must be calibrated for the expected predictive performance of the model. This is accomplished by processing the training vectors that were not included in the training matrix through the model to produce estimates for the signals. The difference between the signals and the estimates are used to compute the residual error values—also referred to as the *residuals*—over the range of operation. The residuals for the unselected training data are used to calibrate the fault detectors to accommodate the expected modeling error.

The fault detector calibration procedure operates the model as if in monitoring mode with the unselected training data vectors as input. Each vector is sequentially processed, and the residuals are computed and stored. For large training data sets, this can be a memory-intensive procedure. The fault detector calibration procedures are applied to calculate the expected statistical properties of the training data residuals for each signal included in the model. These statistical properties implicitly capture the various uncertainties in the model. Uncertainty or noise in the underlying signal, modeling errors, and noise in the measured process will all be represented in the training data residuals. These uncertainties can affect the resulting sensitivity of fault detection.

Figure 7-2 shows an example in which the residuals vary only about 0.15 percent of the sensor's calibrated span (the inner set of horizontal lines represent ±1 percent of span). In this example, fault detection can be quite sensitive. Figure 7-3 shows a different example in which the residuals vary by up to ±1 percent of span, which can result in less-sensitive fault detection. These examples illustrate that fault detection sensitivity is directly related to the underlying uncertainty in both the measured and predicted signal data values. Thus, it is not realistic to expect that on-line data errors in a signal with expected operating noise of 1 percent could be detected at a signal error magnitude of 1 percent or less.



1 psi = 6.894757 kPa

Figure 7-2
Small Residual Results - Sensitive Fault Detection

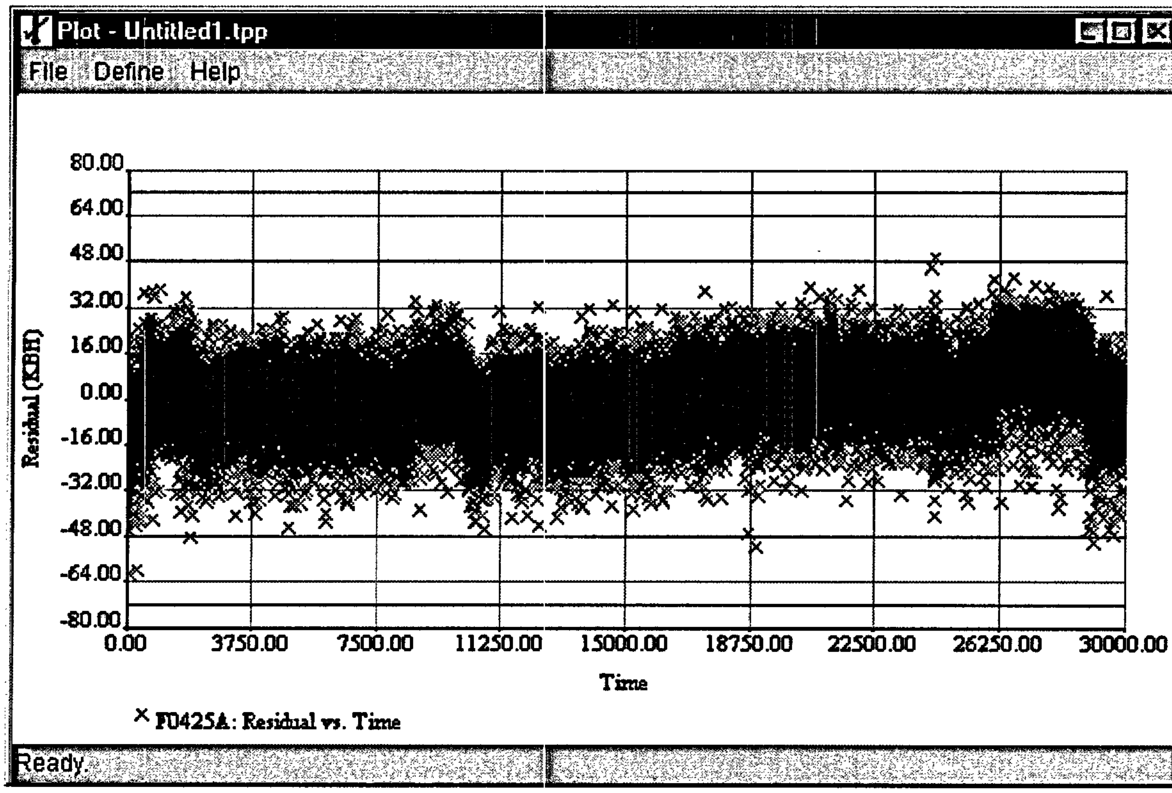


Figure 7-3
Larger Residual Results - Less Sensitive Fault Detection

As discussed in Section 8, the fault-detection technique is used to distinguish between a signal that is operating normally and a signal that is operating abnormally. This is accomplished by statistically evaluating a series of new residual data values to determine whether the series is characteristic or uncharacteristic of the residual data distribution determined in the training step. As long as the signal characteristics remain the same (including the uncertainties), the fault detectors will judge the signal to be operating normally.

7.1.2 Estimation

During monitoring, a new observation of the model's signals is acquired and is compared to the previously trained MSET model to estimate the expected values of the signals. The estimation procedure is accomplished by comparing the new observation to the previously learned examples stored in the training matrix. Similarity between the current observation and the learned examples is computed using multivariable pattern matching techniques. The weighted combination of the most similar learned examples is used to compute the estimated signal values for the current set of observations. Those examples most similar to the current observation are heavily weighted; those that are dissimilar are negligibly weighted.

There are several nonlinear similarity operators and methods that can be used to perform the multivariable pattern matching and parameter estimation calculations. For example, depending on the toolkit selections included in the user's version of the software, SureSense provides a

number of these alternatives. For general use in the government and power markets, three similarity operators developed by ANL are provided in a single toolkit: the Bounded Angle Ratio Test (BART) operator, the Vector Pattern Recognizer (VPR) operator, and the Vector Similarity Evaluation Technique (VSET) operator. U.S. government users can also request the ANL-developed System State Analyzer (SSA) operator. The Universal Process Modeling (UPM) operator and toolkit is a proprietary alternative to the ANL toolkit. The UPM toolkit is available in all fields of use and can be provided separately or in combination with the ANL toolkit. The latest release of SureSense (Version 2.0) includes additional options for empirical estimators—the ESEE and the Parity Space Averaging (PSA) toolkit.

For power plant applications, the following methods are the more commonly used:

- BART – For most applications, the BART operator is used as the principal estimation method. The algorithm defines the similarity between two signals as a function of the angle formed by drawing lines from each measurement to a reference point. The domain of measurements collected during normal operation of the sensor defines a number line for the signal data. The reference point conceptually represents a point that lies above the number line in such a way that a line drawn through the median and the reference point will be perpendicular to the number line. The height of the reference point is such that the angle between the line drawn between the reference point and the minimum value and the line drawn between the reference point and the maximum value is 90 degrees. The user-defined domain extension is a value that extends the range of the number line by the domain extension amount (max-min) on either end. The number line is extended past the minimum value of the domain extension (max-min) units and past the maximum value of the domain extension (max-min) units. This has the effect of increasing the height of the reference point.
- VSET – The VSET algorithm defines the similarity of two vectors as a function of the ratio between the Euclidean distance between the vectors and the sum of the root sum squared (RSS) values of the vectors.
- VPR – The VPR algorithm compares two vectors and defines their similarity as a function of the inverse of the Euclidean distance between the vectors.

Regardless of which similarity operator is chosen, the similarity calculations basically follow a similar procedure. The procedure begins by computing a measure of the similarity of each new observation relative to each of the training observations stored in the training matrix. While the means by which similarity is computed will differ, the end result is the identification of the most similar vectors contained in the training matrix. The derived measure of similarity is then used to construct an estimate of the signal values based on a weighted combination of the observations in the training matrix. The specific approach taken to construct an estimate based on a weighted combination of the observations in the training matrix will again depend on the method chosen.

The k-nearest neighbors approach provides a simple example of how a pattern matching and weighting method are used to compute an estimate. The weighted value of each new observation vector is computed as described in the previous section for vectoring ordering. The k vectors in the training matrix having the most similar weighted value are selected as the k-nearest neighbors for the new observation. This completes the pattern-matching step.

A simple weighting scheme combines the values for a specific signal in each of the k-nearest neighbor vectors to produce the estimate. The estimated signal values are derived by computing the weighted average of the k-stored values using a linear combination with inverse similarity weights, or:

$$X_{est} = \frac{\sum_{i=0}^{k}\left(\frac{X_i}{|E_{obs} - E_i|}\right)}{\sum_{i=0}^{k}\left(\frac{1}{|E_{obs} - E_i|}\right)}$$

where

X_{est} = The estimated signal value

E_{obs} = The weighted average of the current observation vector

X_i = The training signal value

E_i = The weighted average of the selected training observation vector

Notice that the individual observation values are not used in the estimation method illustrated here. The weighted value (square root of the sum of the squares) of all observations, or E_{obs}, is used as the link between the vector of observations and the data that are stored in the training matrix.

7.2 How to Train a Model

7.2.1 Selecting the Initial Training Data

Initial training represents a milestone in model development. The signals for a model have been selected, the preliminary model settings have been specified, and the data have been acquired for training and verification testing. The model is trained with some portion of this data for the purpose of evaluating the model's settings and its response to various conditions. This is referred to as *initial training*. Early in model development, the various system operating states might not be known or fully understood. As a system's normal operating variations become better understood, it is common to add more training data to the model to characterize these additional operating states. This addition of training data to the initial training data is one form of retraining.

Initial training data are expected to bound most but not all of the possible operating state space. There are likely to be valid operating states for which the model was not initially trained. Even if a large amount of data are made available for training, this does not guarantee that all possible operating states have been covered. For example, Section 6.1 discusses data quantity in terms of sample frequency. Generally, a 1-minute sample rate is recommended, which can result in very large data files. Although a 1-minute sample rate can create as much as 44,640 vectors of data in a single month, what is important for training is not the quantity of data as much as the operating states described by the data. Suppose that data for two months were selected for training. This

amount of data might contain over 80,000 vectors of data. However, if process values do not vary during this period, only a single operating state might be defined. Figure 7-4 shows an example in which power is virtually constant at about 100 percent power for two months. If these data are used for training, the 100 percent power level is defined quite well, but no other power levels are included. It is important to assess the data selected for training and to understand the limitations.

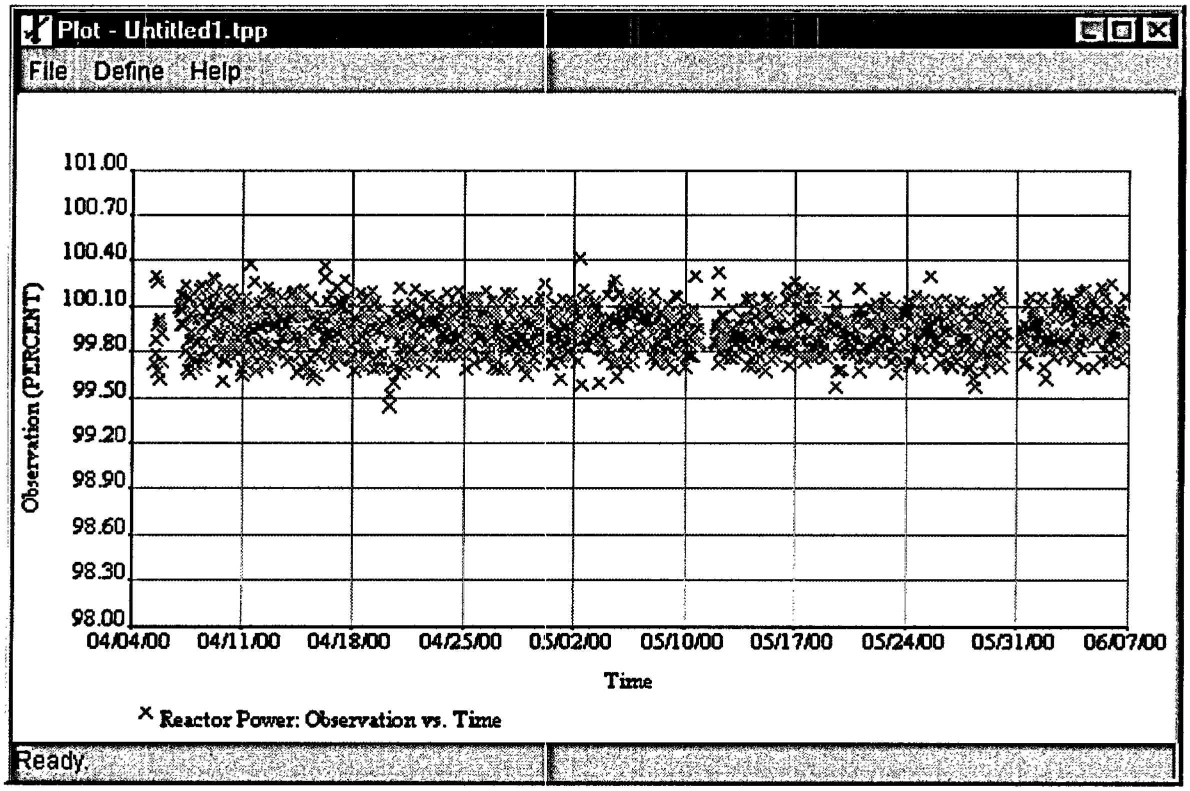


Figure 7-4
Reactor Power - Constant for an Extended Period

7.2.2 Evaluating the Initial Training Data Adequacy

The data used for training must be error-free and should represent the normal expected behavior and variation of the signals contained in the model. With regard to data quality, the following four attributes are important for the historical operating data used to train the model:

- The data should contain observations from all modes and ranges of operation that are to be considered the normal domain of operation. MSET will produce a high-fidelity model of this domain of operation and will determine any other modes and ranges of operation to be abnormal.
- The data should contain one or more signals that are reasonably well-correlated to each signal included in the model. Correlated behavior between the included signals is essential for effective modeling using the MSET procedures.

- The data should not contain any operating anomalies, sensor failures, or equipment failures that would be considered abnormal operation. If included, the modeling procedures will learn these conditions as normal behavior.
- The data should not contain any signals that are constant valued over the range of operation. These provide no diagnostic information and will introduce singularities (resulting in numerical instabilities) into some of the mathematical procedures.

These criteria are prerequisites to characterize normal operation properly and completely. Recommended procedures for acquiring a comprehensive and error-free set of training data are provided in Section 6.

In addition to verifying acceptable data quality, the data set should be reviewed to confirm that it adequately represents the desired calibrated state of the evaluated sensors. For example, if two redundant sensors with identical calibrated spans are monitoring the same process, one should expect that the measurements from the sensors should be about the same. Figure 7-5 shows an example in which identical flow transmitters each monitor reactor coolant system flow, yet the two sensors display a difference of approximately 9 percent in measured values; something is probably not right with the data being used for training.

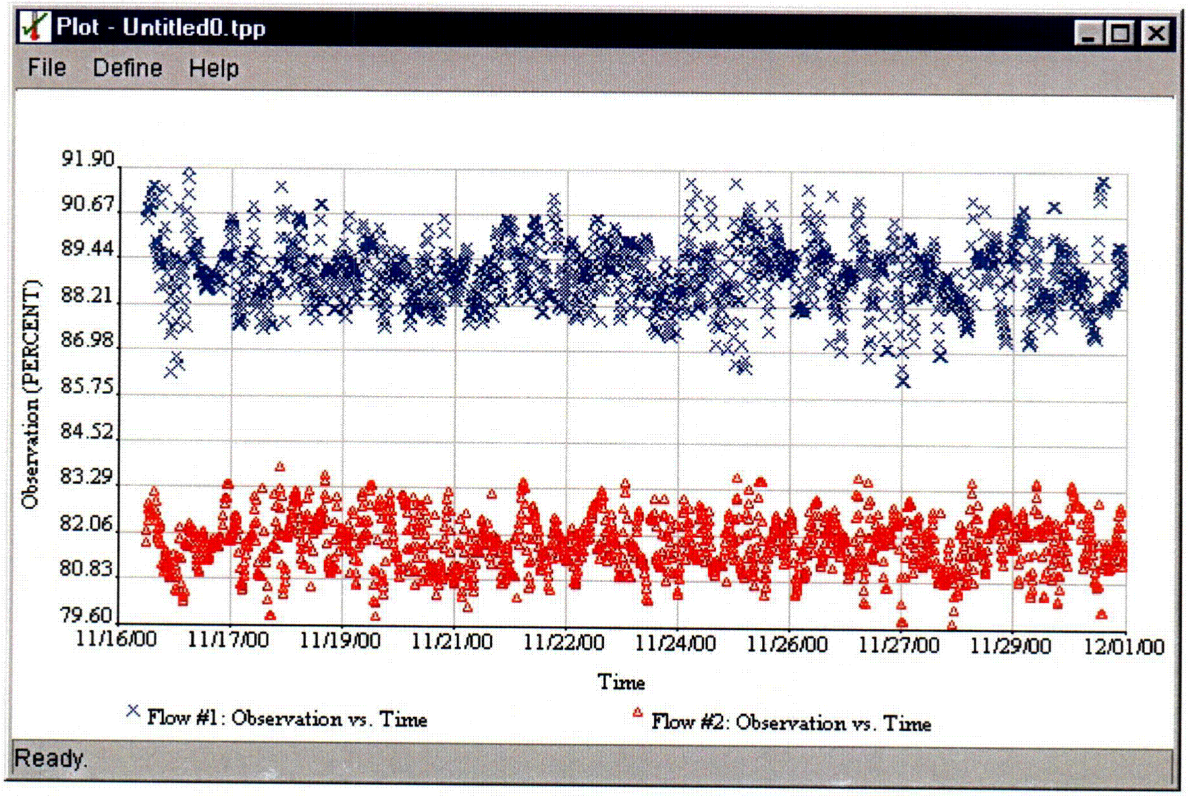


Figure 7-5
Large Difference in Redundant Flow Measurements

C23

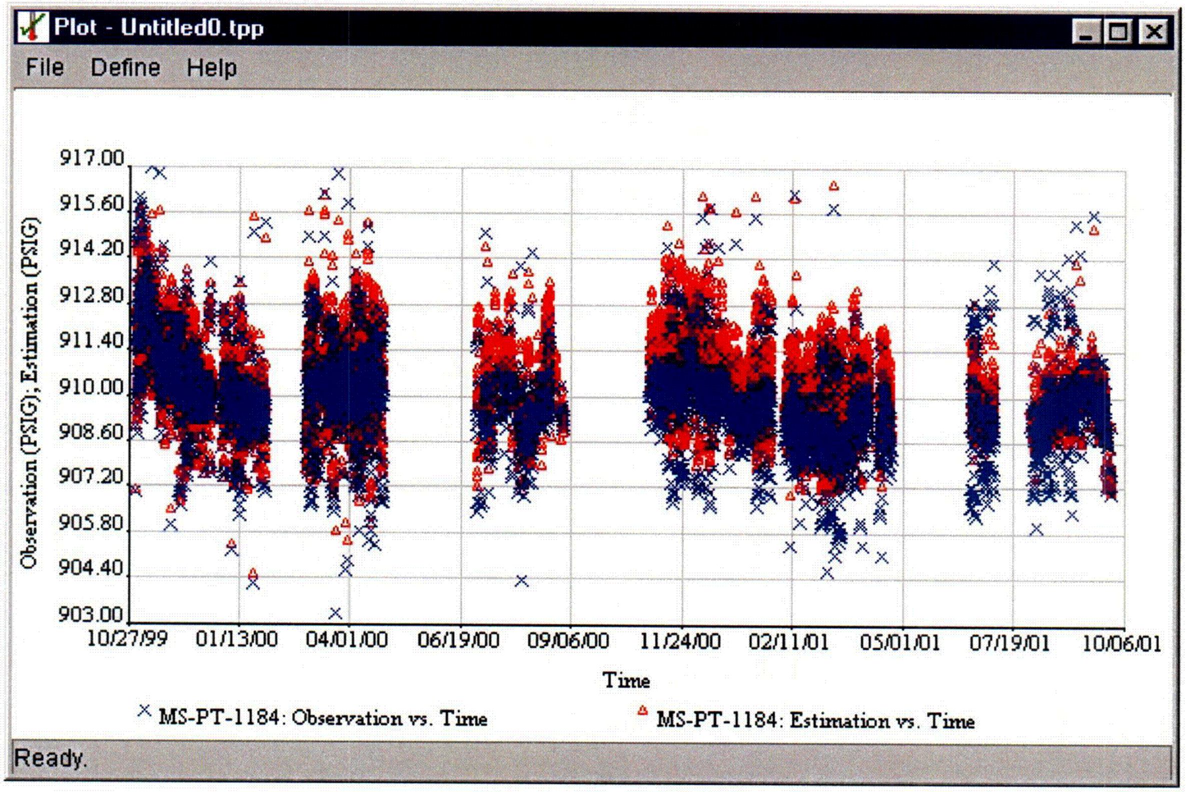
Verifying adequate data quality is fairly straightforward, although it can be tedious. Verifying that the data used for training represent an in-calibration condition is more difficult and requires a careful review of the data. The presence of redundant signals helps by providing a means of direct comparison for some signals. The algorithm used by EPRI's ICMP is easy to set up and run in a spreadsheet program. It provides one method of independently assessing the calibrated state of the sensor data used for training. *On-Line Monitoring of Instrument Channel Performance, Volume 2* [1] provides an overview of the ICMP algorithm.

7.2.3 Evaluating Training Adequacy

Most nuclear plants are on an 18-month or 24-month fuel cycle. For a given model, it is unlikely that the signals remain constant over the entire fuel cycle. Signal values can change for a variety of reasons such as the following:

- System or process changes that occur over time
- Seasonal variations
- Operational changes such as operating at less than 100 percent power
- Equipment lineup changes
- Instrument drift

The data used for training should be compared to historical data to evaluate how well a model can be trained and allow subsequent signals to remain within the training space. The best method of evaluating how well the initial training will cover all possible operating states is to obtain historical verification testing data for an extended period of operation and to compare it to the training data. Figure 7-6 shows an example of the desired result; the data appear to remain essentially constant valued over a two-year period. Notice that main steam pressure varied from about 905–915 psig (6240–6309 kPa), which covers a range of about 0.83 percent of span for this sensor. With respect to this signal, the model has an excellent chance of performing well without requiring retraining.



1 psi = 6.894757 kPa

Figure 7-6
Process Values Relatively Unchanged Over the Operating Cycle

Figure 7-7 shows a different example in which a steam generator level transmitter signal tended to remain almost constant until near the end of the operating cycle when extended low-power operation resulted in a wide variation in the level signal. If the model was trained using data from only the beginning of the operating cycle, the model training will be ineffective during this new operating state. Figure 7-8 shows the final two months of power operation in more detail. It should be noted that this particular model was trained for this condition because the observations and the estimates are close together.

C24

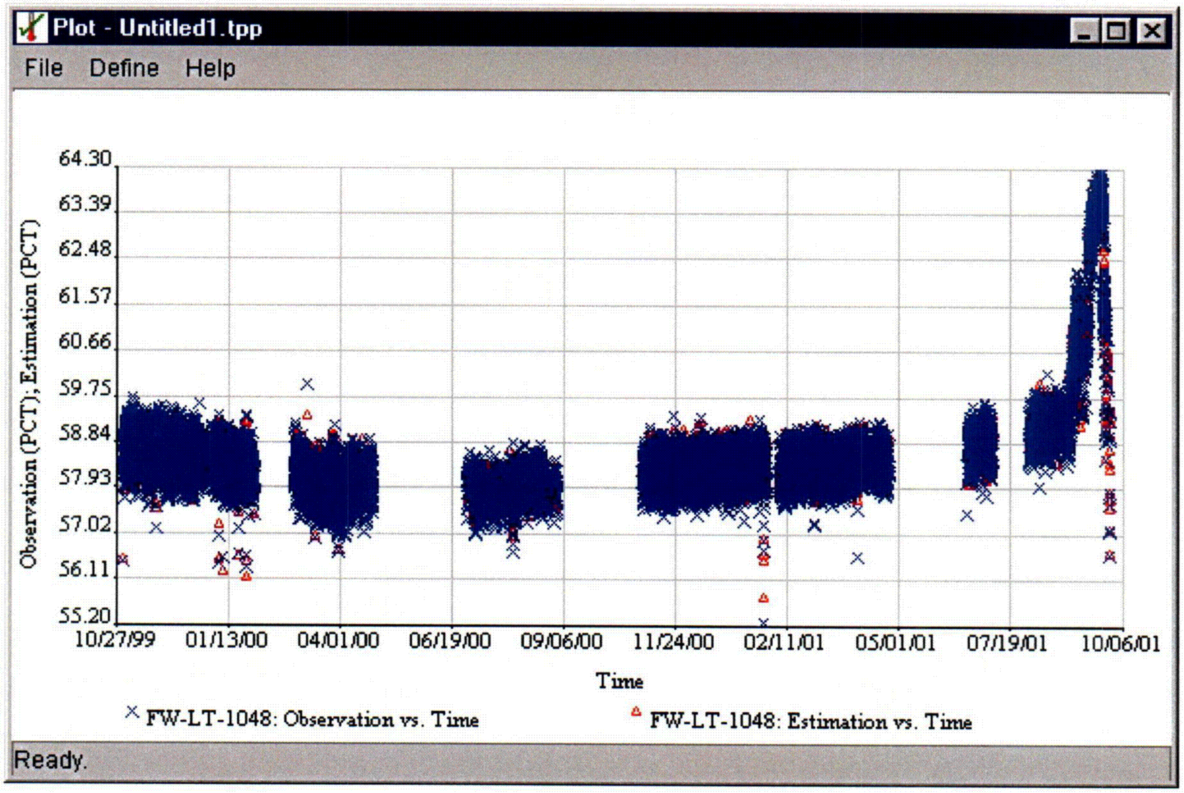


Figure 7-7
Process Value Change During End-of-Cycle Low-Power Operation

C25

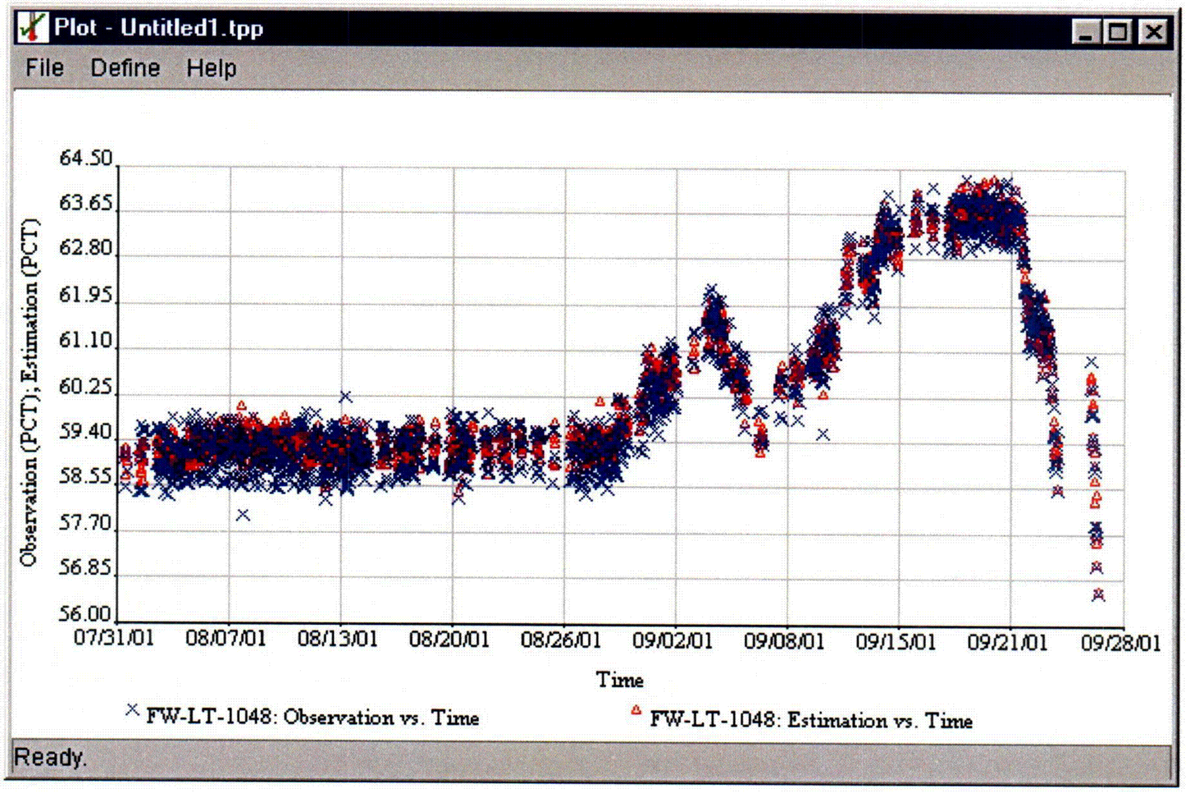


Figure 7-8
Steam Generator Level During Two Months of Low-Power Operation

Figure 7-9 shows an example of recirculation pump flow in a BWR. There are routine changes in flow that appear as transients even though power is constant. Historical data are particularly important for evaluating a model such as this one. Obtaining training data that include all operating states can be challenging.

C26

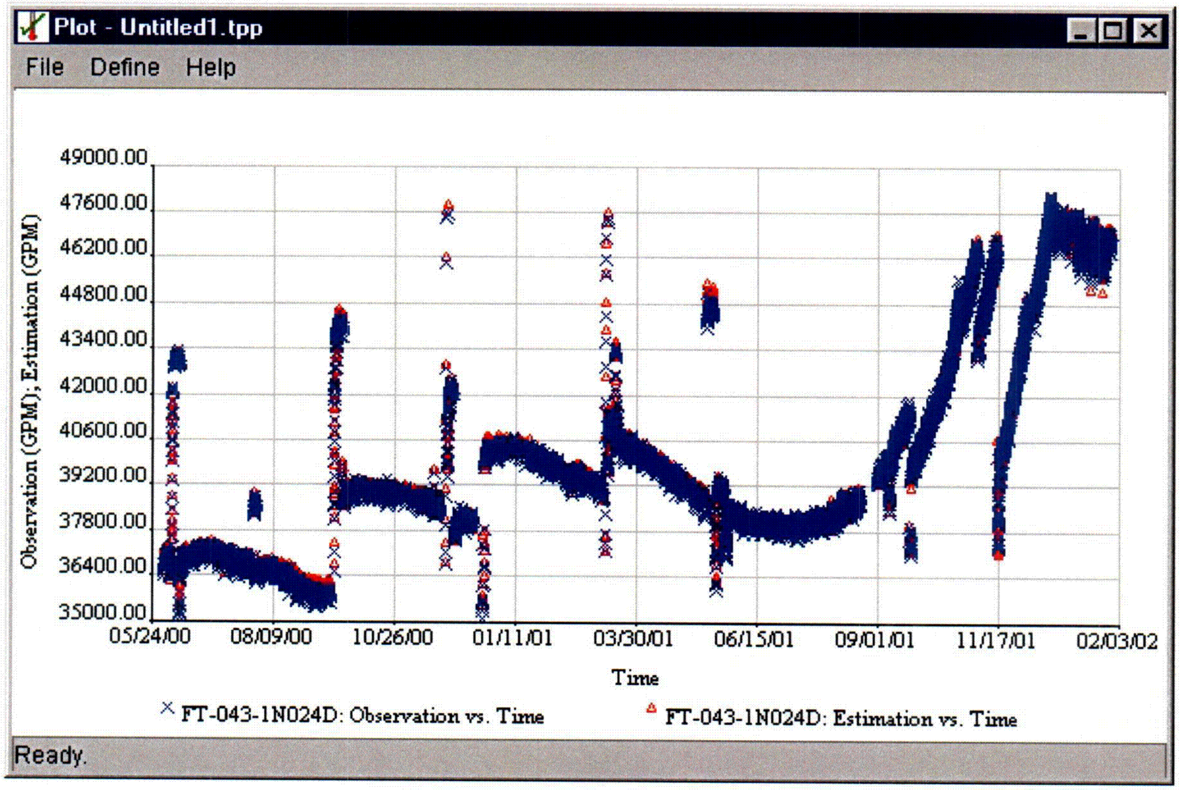


Figure 7-9
Routine Changes in Process Values

Historical data have an important role in model development because they readily show the typical operating states that might occur. The plots shown in the previous figures illustrate the types of variations that can often be encountered.

7.2.4 Retraining the Model

Training, fault detection, and retraining are closely related. When a model is initially trained, it often performs well for some period of time until alarms are generated as the operating space starts deviating outside of the initial training boundaries. The first indication that a model is operating in an untrained space is usually a significantly increased number of signal failure alarms. When this occurs, it is time to consider the need for retraining the model.

C27

7.2.4.1 Retraining Terminology

The selection of the training matrix represents an essential step in the training of a model. When the training matrix is changed, the model has been retrained. This might be necessary for the following reasons:

- Certain model settings are modified. Changing estimator settings, changing the number of signals, adjusting data screening limit filters, or modifying phase determiner submodel partitioning definitions will require retraining. This is referred to as *retraining for settings,* and it optimizes model performance for a given set of training data.
- The data used for training are modified. If the pool of historical data used for training are modified, the vector selection for the training matrix will likely change even if the model settings are unchanged. This is referred to as *retraining for operating space.*

This section discusses issues associated with retraining for operating space. Instances where retraining for settings is undertaken are usually to further optimize model performance through fine tuning of fault-detection settings or phase-determiner settings.

7.2.4.2 Retraining Philosophy and Limitations

At some point during model development or a model's life, a need for retraining will be identified. Any of the following data-related conditions can warrant retraining:

- The data files used for initial training contained bad data that had to be removed. This is not considered retraining in the context of this discussion. Instead, the model was inadequately trained initially. With cleaned-up data files, the model is again initially trained.
- Signals are added to or removed from the model, which requires a change in the training matrix to accommodate these signals. Again, this is not considered retraining in the context of this discussion. Instead, the model has been modified by adding or removing signals and their corresponding data. With the data for these signals included or excluded, the model is again initially trained.
- Typical initial training data will characterize most, but not all, of the possible operating space. Depending on the model, it might be necessary to retrain with additional data that expand the operating space. Based on models developed at nuclear plants, operating outside the training space at some point during an operating cycle is common. One of the following situations can occur when a new operating state is outside of the trained space:
  - The new operating space might be a permanent change in the system operation. The model will require retraining.
  - The new operating space might represent a valid operating state that occasionally occurs and can continue in this space for an extended period. Once again, the model will require retraining.

  - The new operating space might be a short-term transient that rarely occurs. In this case, it might be preferable not to train for this event and allow spurious fault alarms during the event. Accounting for these transients in the training data set can desensitize the model, which is not desirable for rare transients. Additionally, notification that the process has moved to a possible transient condition may be useful.
  - Depending on the plant design, it might be difficult to accommodate all possible combinations of equipment operation in the training data. For example, a nuclear plant operating at low power might run one, two, or three feedwater pumps in various combinations and pump speeds. At lower power levels, there can be many different combinations of valid system operating states, and it is unlikely that the training data will ever adequately cover all of these states. A phase determiner should be used to exclude states with little or no data available for training. One possibility is to exclude all low-power data and focus mainly on the 100-percent power region. Notice that this approach does not require retraining. It simply excludes operating data for which the model has not been trained from fault-detection processing.
- For nuclear plant systems, some models might have a finite life before requiring retraining. The following are examples:
  - In some cases, subtle changes in only a few sensors in a model can have the appearance of drift when actually small process changes are occurring over time. These are the hardest cases to evaluate. After confirming that instrument performance is acceptable, it might be necessary to retrain with all new data or to retrain with supplemental data.
  - If the sensors or signal conditioning equipment associated with a model are recalibrated, there might be sufficient shift in the signal output to cause spurious fault alarms. It might be necessary to retrain with all new data.

As can be seen from the previous examples, retraining for operating space has a specific definition in model development. Starting with a model adequately trained with good-quality data, retraining for operating space either expands the operating space or redefines the operating space. If retraining expands the operating space, supplemental data were probably added to the existing training data. If the operating space is redefined, the existing training data were probably replaced with new training data. If a model is not trained for all possible operating states, such as some transients, additional guidance should be provided to users regarding the interpretation of fault alarms during these conditions.

8 FAULT DETECTION AND ALARM RESPONSE

Section 8 describes fault detection, provides guidance on how to assess identified failures, and discusses model retraining.

8.1 Fault Detection - Technical Overview

8.1.1 Background

Fault detectors operate on model residuals where a residual is the difference between the observed value and the estimate for a signal. The residuals for each monitored signal are used as the indicators for sensor and equipment faults. Instead of using simple threshold limits to detect fault indications (that is, declaring a fault when a signal's residual value exceeds a preset threshold), the fault-detection procedure employs statistical hypotheses test techniques to determine whether the residual error value is uncharacteristic of the learned model and, therefore, indicative of a sensor or equipment fault. This technique is a superior surveillance tool because it is sensitive not only to disturbances in the signal mean but also to very subtle changes in the statistical quality of the signals.

While changes in the residual mean, variance, skewness, and kurtosis can be monitored in this way, the only two tests that are regularly used in OLM applications are the mean and variance tests. For sudden gross failure of a sensor or item of equipment, the fault-detection method will annunciate the disturbance as fast as a conventional threshold limit check. However, for slow degradation, the procedure can detect the onset of the disturbance long before it would be apparent with conventional threshold limits. The fault-detection setup allows the user to specify false alarm and missed alarm probabilities, thereby allowing some control over the likelihood of false alarms or missed detection.

In general, the fault-detection procedure is accomplished by first establishing the expected statistical distribution of the residual values when the model is operating normally. This step is accomplished during the MSET model training procedure. After an MSET model is trained, the remaining (unselected) training data observations are processed through the model to characterize the expected distribution of the residual values.

Having characterized the expected distribution of the residual values when the model is operating normally, fault detection identifies those conditions that deviate from the learned MSET model. In operation, a time series of residual values are evaluated to determine whether

the series of values is characteristic of the expected distribution (the null hypothesis) or, alternatively, of some other specified distribution. The following four possible fault types are considered:

- The residual mean has shifted high (positive mean test)
- The residual mean has shifted low (negative mean test).
- The residual variance has increased (nominal variance test).
- The residual variance has decreased (inverse variance test).

Each of the four fault-detection tests is a binary hypothesis test. The residual signal is analyzed to determine whether the signal is consistent with normal behavior for each test. When a decision about current residual signal behavior is reached (either that the signal is behaving normally or abnormally), the decision is reported, and the test continues analyzing the data from the signal.

A user-configurable setting (referred to as the system disturbance magnitude setting in SureSense) is used to set the sensitivity for selecting between the expected (null-type) distribution and a fault-type distribution. This controls the crossover point at which a disturbance in the residual values is deemed uncharacteristic of the system normal operating states.

8.1.2 Mean Tests

The positive mean test and the negative mean test detect changes in the signal average value and are most commonly used for problems related to drift and calibration error as well as complete failures (open- or short-circuit failures). The system's disturbance magnitude setting controls the point at which a deviation in the signal mean value will be deemed abnormal. Figure 8-1 illustrates the change in the mean value required to produce a fault-detection event using the positive mean test and the negative mean test.

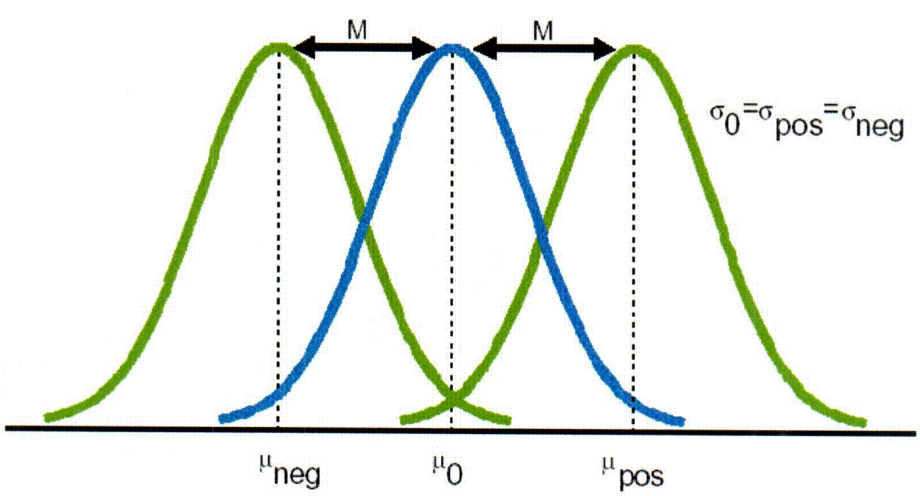


Figure 8-1
Mean Disturbance Magnitude Tests

C28

8.1.3 Variance Tests

The nominal variance test detects an increase in the signal's variance and is useful for detecting cable damage or loose connectors that can cause signal spiking in an otherwise normal signal. The inverse variance test detects a reduction in the signal's variance, which might be caused by a loss of response-type failure with a nonresponsive signal remaining in the normal expected range. In addition, stuck data occurring during monitoring will trip an inverse variance test. The system disturbance magnitude setting controls the point at which a deviation in the signal variance value will be deemed abnormal. Figure 8-2 illustrates the change in the variance value required to produce a fault-detection event using the nominal variance test and the inverse variance test.

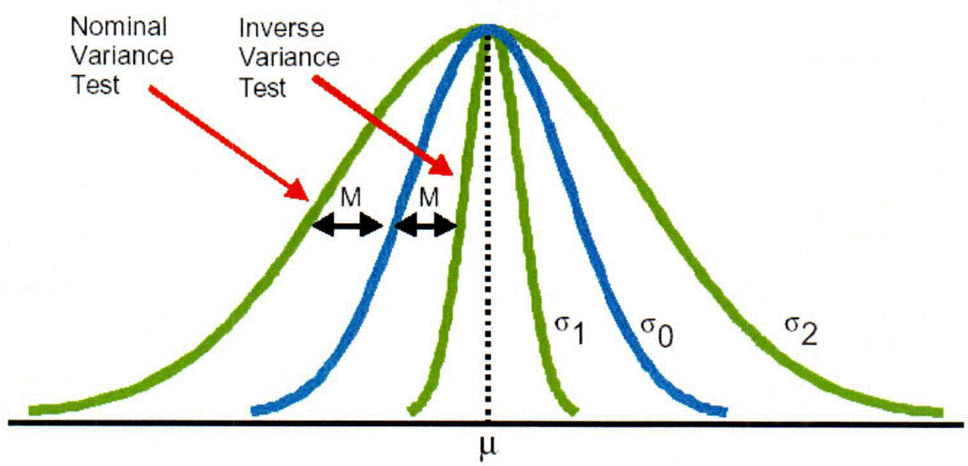


Figure 8-2
Variance Disturbance Magnitude Tests

8.1.4 Unique Probability Density Functions

Real-world data often have thicker tails (more outlying data) than would be predicted by a normal (Gaussian) distribution. A limitation of the SPRT technique is an underlying assumption that the residual data distribution is characterized by a normal probability density function. While many real-world models give residuals that are nearly normal, most have thicker tails (greater numbers of outlying points). These thicker tails can cause an undesirable increase in the number of false alarms that the SPRT procedure generates.

An adaptive sequential probability (ASP) fault-detection method is available in SureSense that can have fewer false alarms by fitting the data to a unique (but still approximately normal) probability density function (PDF). By allowing the PDF to contain more outlying data than would be expected for a purely normal distribution, the false alarm rate could be reduced.

ASP determines the PDF during training by evaluating the residuals of the training data. The residuals typically exhibit near-normal behavior with somewhat more outlying data than would be predicted by a standard normal distribution. This approach requires a large amount of training

C29

data to ensure that the unique PDF is properly described; at least 10,000 training data points are recommended. This amount of data is usually easy to obtain if the data sampling is performed at a 1-minute rate.

8.1.5 Applying Conditional Probability to Failure Declaration

Occasional false alarms are an inevitable consequence of any statistically based fault-detection test. For this reason, SureSense uses a conditional probability analysis of a series of fault-detection results to distinguish between real and false alarms. The SureSense Multicycle Event Filter (MEF) requires that a certain number of alarms occur in a short sequence of results before a failure is declared. In the MEF technique, each new decision reached by a fault-detection test is treated as a new piece of evidence about the state of the signal. The conditional probability of failure for the signal is updated on the basis of the new evidence. The conditional probability of failure is compared to a predefined limit. For probabilities below the limit, the signal is declared to be operating correctly even if occasional alarms are generated. For probabilities above the limit, the signal is declared to be abnormal. The MEF technique improves on a conventional multicycle voting approach by allowing the user to explicitly control the statistical confidence level used in the final fault decision.

8.2 Failure Evaluation

8.2.1 Summary of How Failures Are Determined

As data observations are processed through a model, estimates for the observations are calculated. Each residual (the difference between an observation and its corresponding estimate) is then evaluated by a series of statistical tests and, if appropriate, an alarm is generated. However, an alarm is only the first part of a failure determination. As described in Section 8.1, failure declaration is a two-step procedure involving the following steps:

- If a series of residuals presents a probability ratio that trips one of the SPRT tests, an alarm will be generated for that signal.
- Depending on the model settings, a failure declaration will be made based on the number of alarms generated within a sequence of observations. For example, it might take five alarms out of a series of 10 observations to achieve a failure declaration.

Even if everything is working well, occasional alarms will occur because of random variations in the data or minor process variations. This is why a single alarm is not considered a failure. It takes a certain number of alarms within a short sequence before a failure will be declared.

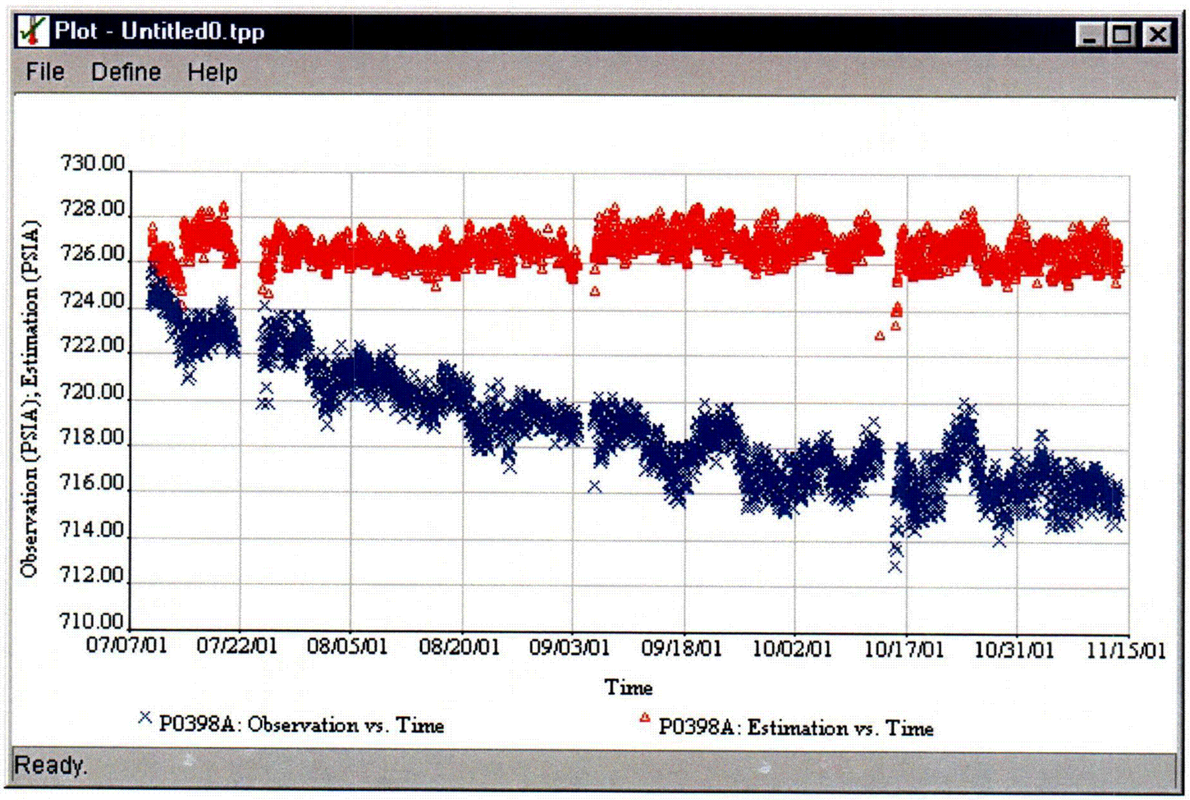
8.2.2 Recommended Response to an Identified Failure

Based on the experience to date, instruments used in nuclear plants are generally well behaved with only occasional occurrences of unacceptable drift or failure. Accordingly, most failures identified by the on-line monitoring system are not failures. They often represent operating states

that have not been adequately covered by the training data. Sometimes, the failure alarms are caused by overly sensitive fault-detection settings. The following summarizes the most likely causes of alarms or failure declaration:

- Data acquisition problems resulting in bad data
- Operating outside the training space—permanent or temporary changes in process values
- Operating outside the training space—short-term transients
- Overly sensitive fault detection

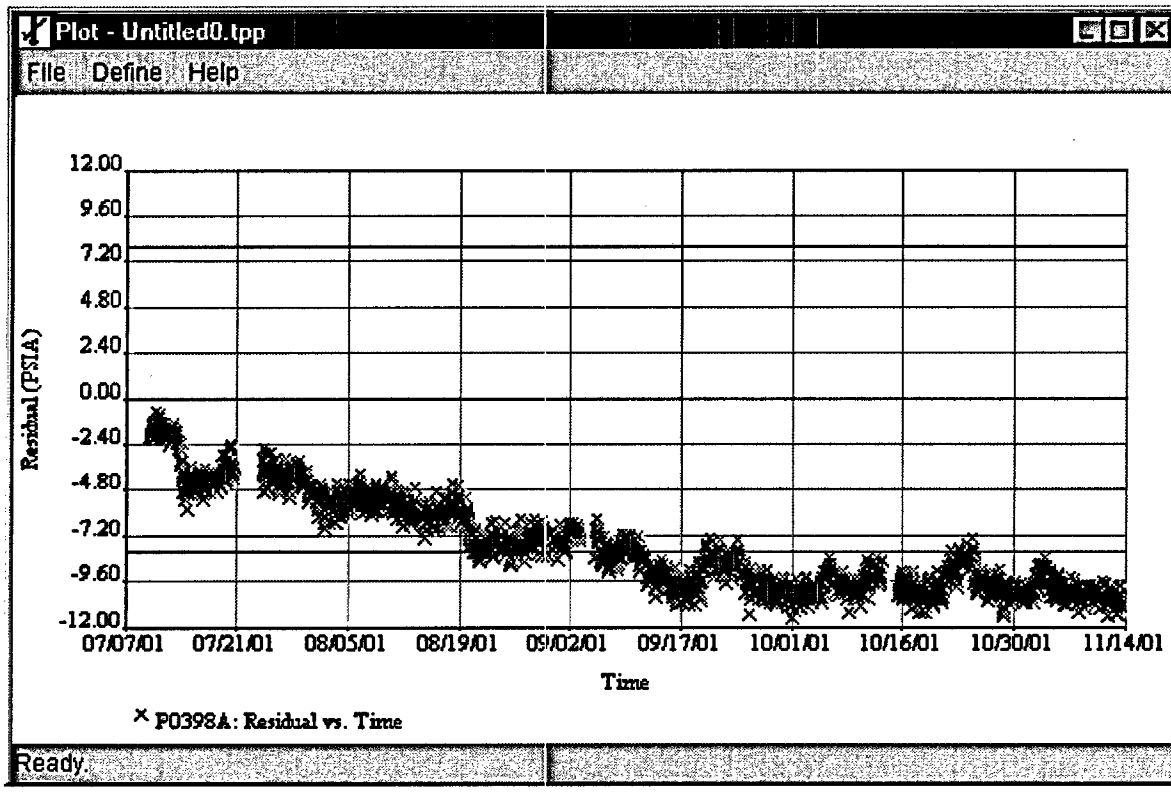
The EPRI On-Line Monitoring Implementation Project has developed dozens of models for hundreds of sensors at a variety of nuclear plants. Significant instrument drift or failure has rarely been observed in the models developed to date. Even if an instrument channel is drifting, the fault-detection method can be very sensitive, often producing failure alarms long before the drift is significant. Figure 8-3 shows an example of typical drift behavior (the observations shown as blue crosses, and the estimates as red triangles). It is apparent that this pressure transmitter is slowly drifting low and that it has drifted low by about 1.5 percent over a period of three months. Figure 8-4 provides the residual plot for this signal. The failure was first identified when the drift was about -3.0 psig (-20.7 kPa) or about 0.35 percent of span.



1 psi = 6.894757 kPa

Figure 8-3
Typical Drift Behavior

C30



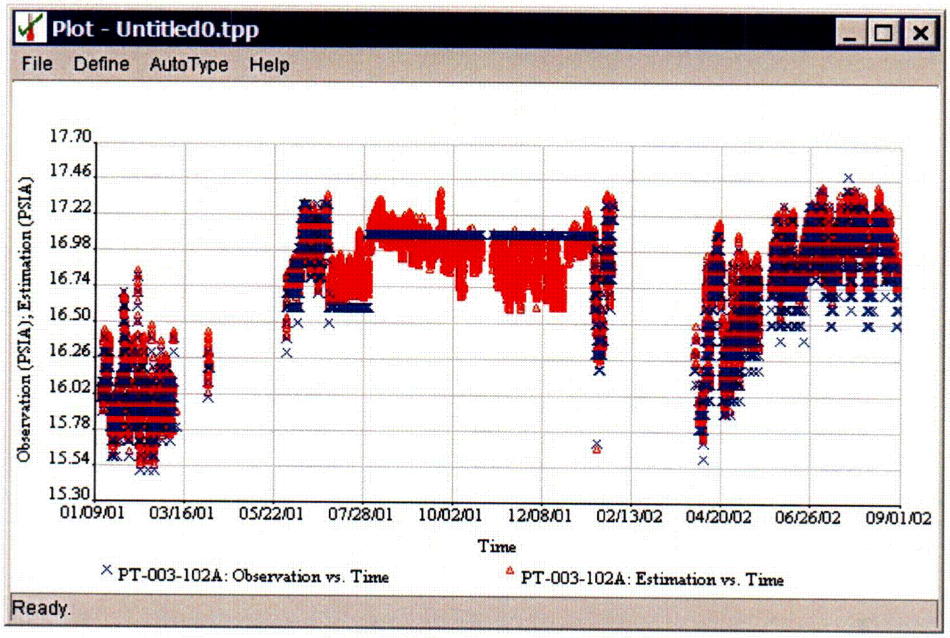
1 psi = 6.894757 kPa

Figure 8-4
Residual Plot Showing Drift Significance Upon Failure Alarm

The first response to a failure declaration is to determine if it is a false alarm. The experience to date is that most identified failures are caused by either data acquisition problems or operation outside the training space. The following sections discuss different types of spurious or false alarms and provide recommendations for how to respond to these alarms. These topics lead naturally into a discussion of retraining.

8.2.3 Failures Identified Because of Data Acquisition Problems

Data acquisition problems will cause occasional failure alarms. Data quality problems are often readily identifiable, particularly if there are redundant signals to allow direct comparison of the data. In extreme cases, a signal data quality problem might persist for months in archived data. Figures 8-5 and 8-6 show examples of data that were stuck for several months. In these examples, the actual process signals are not stuck; instead, the data stored in the data archive are incorrect. As a result, the data provided to the on-line monitoring system are incorrect and will be identified as such by a failure alarm.



1 psi = 6.894757 kPa

Figure 8-5
Extreme Example of Data Acquisition Error - Stuck Data

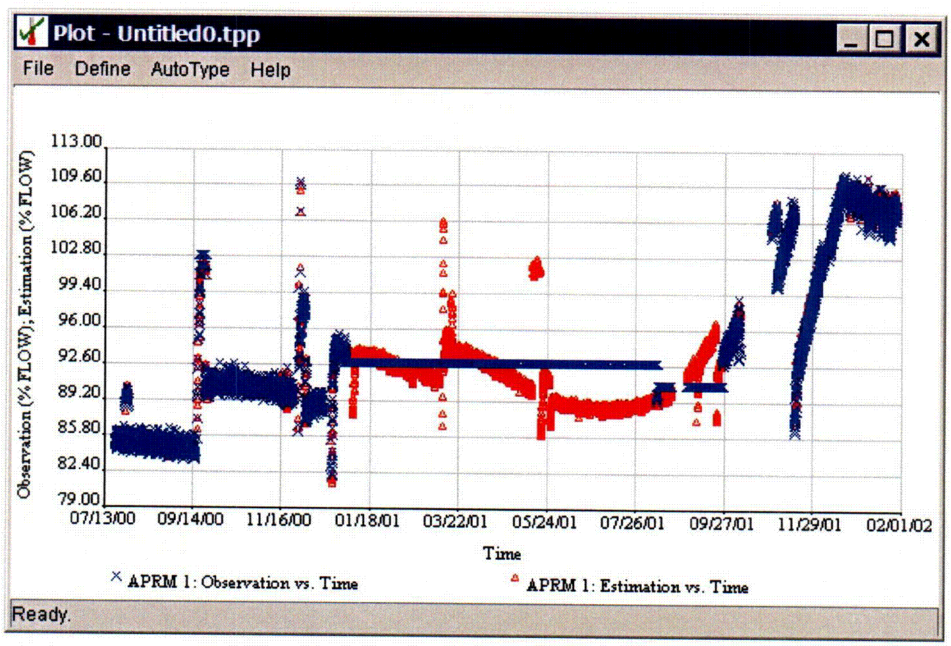


Figure 8-6
Extreme Example of Data Acquisition Error - Data Stuck for Almost One Year

In the cases shown in Figures 8-5 and 6-6, these stuck signals were left in the model, and it was recognized that the archived data were erroneous. The signals were not removed from the model

C31

so that signal validation could be performed again when acceptable data were obtained. Notice that the data storage problem in both examples was eventually corrected.

Not all failure alarms will be caused by long-term data problems as shown here. Many data-related problems will involve random spurious data values, occasional data dropouts, or other short-term data errors. Occasional short-term failure alarms caused by data acquisition problems can be handled in various ways that might include the following actions:

- Annotate the model with a note explaining that the identified failures are caused by bad data or a data acquisition problem. This approach leaves the bad data in historical data sets.
- Remove the bad data from the data set so that they do not continue to cause failure alarms whenever the data are rerun. This is the preferred approach for data files that are developed or archived for use as model training or verification testing data sets. Bad data must be removed from training data sets; otherwise, the model will be trained to recognize the bad data as normal.
- Annotate the database with data quality information that specifies the bad data as invalid. This is the preferred approach when the plant's operating database is used as the permanent repository for the data in question.

8.2.4 Operation Outside the Training Space

The estimation procedure depends on the quality and range of the training data. MSET cannot produce an estimate significantly outside the range of data stored in the training matrix. However, not every system operating state will be known or available in the data when developing a model. It is not unusual for some operating states to shift signals to outside the initial training boundaries. In these cases, these signals will be appropriately identified as failed by the model because they are outside the region defined as "normal and expected behavior." The model likely needs additional training data to describe these new operating states. For some models this will rarely happen; other models will have multiple operating states, and it might be difficult to initially train for all possible states. The following sections show instances in which model retraining might be necessary.

8.2.4.1 Permanent Change in Operating Space - Normal Process Changes

If the operating space shifts significantly, multiple failure alarms might be generated as shown in Figure 8-7. It should be noted that almost all steam pressure channels are identified as failed. This is a SureSense display in which the failed signals are highlighted in yellow rather than red (where yellow indicates that a group of signals are outside the training space). This type of fault-detection behavior indicates that the model is operating outside its training space, which is clearly shown for the actual data (refer to Figure 8-8). Notice that the model is trained for a steam pressure of about 980 psig (6757 kPa), but average pressure for all signals permanently increased to 985 psig (6791 kPa) during the evaluated time period. The signals are not faulty; the model is inadequately trained for the new operating condition. The model should be retrained with additional data that reflect this operating state.

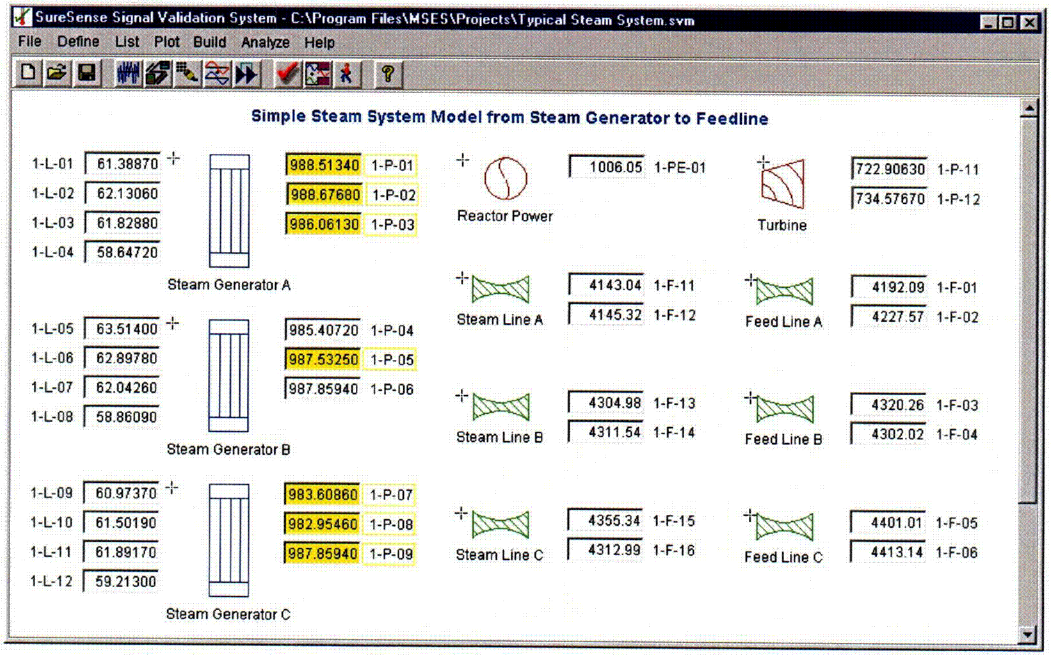
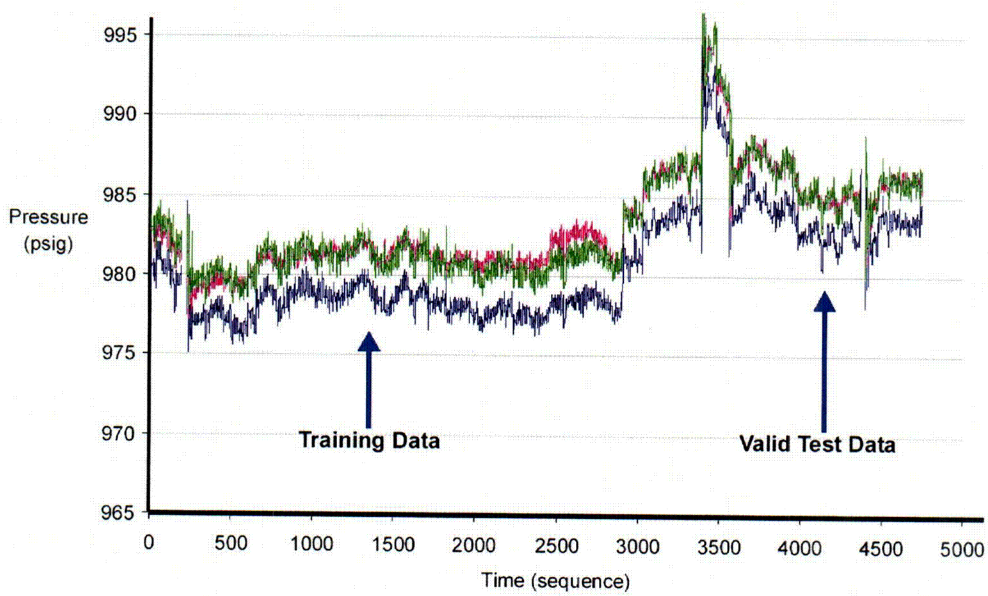


Figure 8-7
Test Data Outside the Training Range - SureSense Result



1 psi = 6.894757 kPa

Figure 8-8
Test Data Outside the Training Range - Actual Data

C32

8.2.4.2 Permanent Change in Operating Space - System Operation Changes

The initial training of a model is usually based on the best available understanding of how the system varies over time. Hopefully, the training data set bounds all normally expected operating states. However, there can be periods of plant operation during which system process parameters are constantly changing even if power is nearly constant. One example is the end-of-cycle coast-down period for a BWR. Figure 8-9 shows a typical example. The flow for this system is essentially constant until the end-of-cycle coast down. If the model was trained to recognize only the earlier period of system operation as normal, failure alarms would start as soon as the system operation changes.

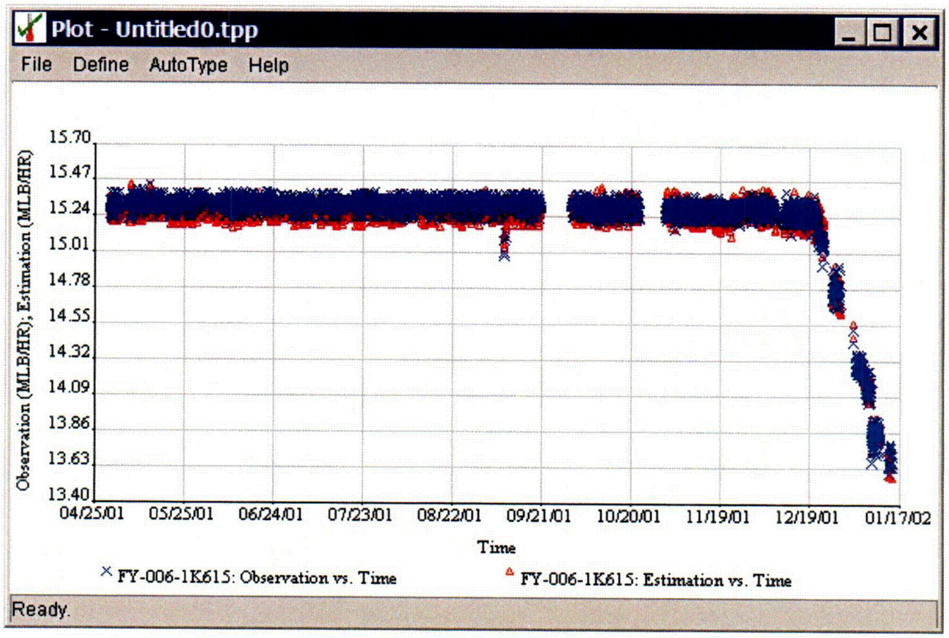


Figure 8-9
Test Data Outside the Training Range—System Operation

Despite the best efforts to acquire training that represent normal system operation, system operating changes can occur that invalidate all previous training data. As an example, consider the water storage tank level shown in Figure 8-10. This plot shows normal behavior in which the tank level exhibits daily level variations as the tank heats up during the day and cools down at night. Over the period of one to two weeks, tank level slowly falls to just below 98 percent, at which time operators refill the tank to almost 98.5 percent.

C33

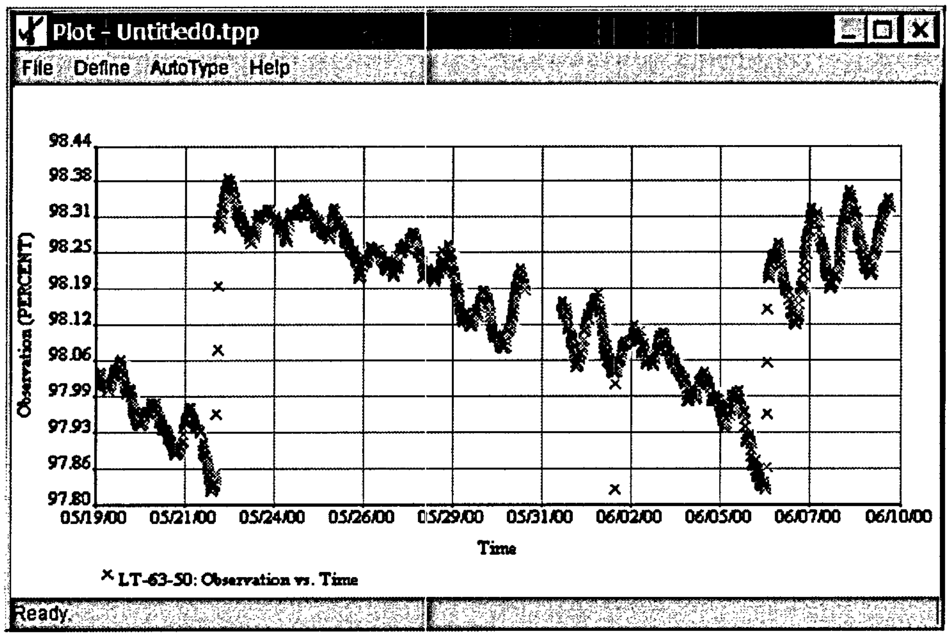


Figure 8-10
Tank Level Variation

Suppose operators change the way in which the tank level is controlled. For example, they might choose to let the tank level fall to 97 percent before refilling and then refill to over 99 percent. If this happens, the range of operation will have almost quadrupled from the range initially used for training. Once it is recognized that this has happened, it will be necessary to retrain the model with data that describe the new operating space. This type of change can occur with any system in which operators have some control in how the system is operated, including any adjustable control system.

8.2.4.3 Permanent Change in Operating Space – Equipment Repair

The process values that define a normal operating space can change because of an equipment repair or replacement. Figure 8-11 shows an example in which a reactor coolant pump impeller was repaired. Before the repair, measured flow was about 78 percent of span. After the repair, the measured flow increased to about 84 percent of span. As can be seen, there is an immediate step change in the measured flow. Figure 8-12 shows that MSET cannot produce an estimate anywhere near the new operating state (the observations shown as blue crosses, and the estimates as red triangles) and the estimates are flat-topped at the limit of the data used for training. The only approach that can be taken here is to retrain with data reflecting the new operating condition.

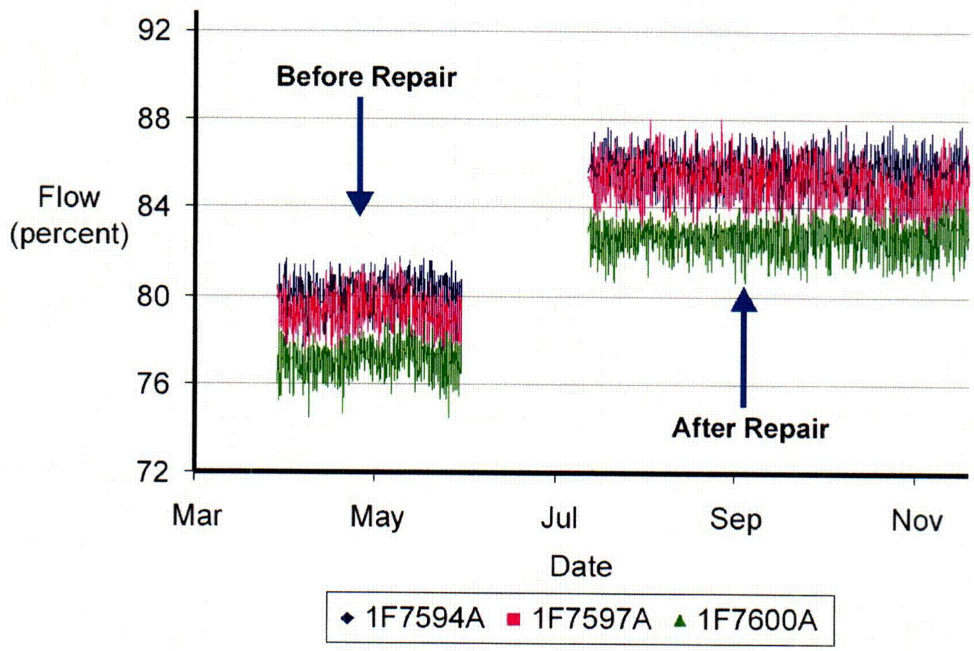


Figure 8-11
Test Data Outside the Training Range - Equipment Repair

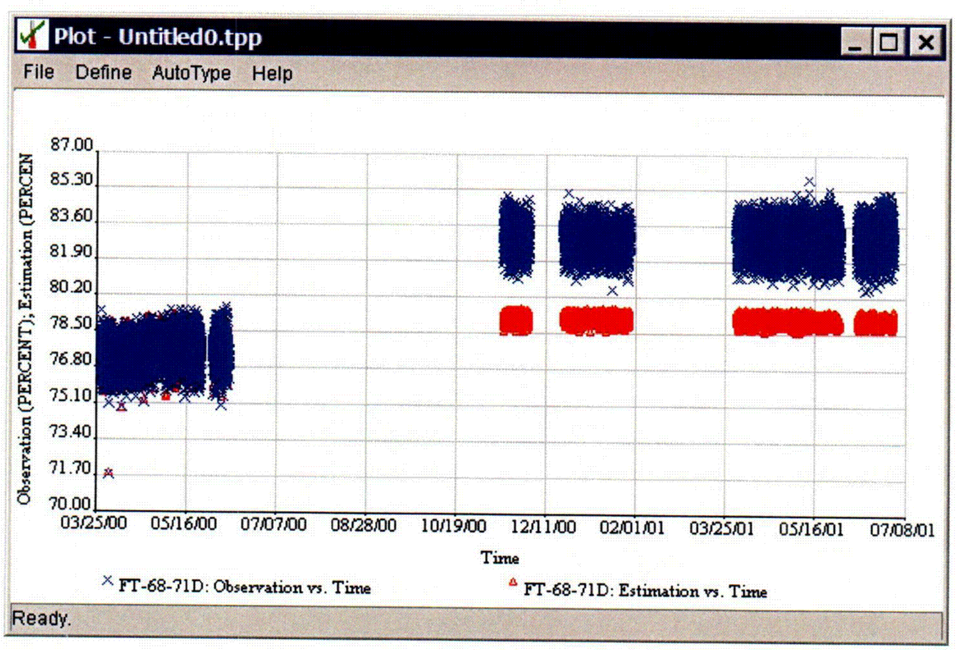


Figure 8-12
Test Data Outside the Training Range - MSET Results

Figure 8-13 shows a more subtle type of equipment repair or replacement (the observations are shown as blue crosses, and the estimates are shown as red triangles). Engineers wanted to monitor a signal at a higher data acquisition frequency, and they replaced the data acquisition

C34

card in the plant computer. The old card sampled at a 2-second rate, and the new card sampled at a 0.1-second rate. Several other signals are also passed through this module. Unfortunately, there is an obvious change in signal noise before and after the card replacement that cannot be explained by the higher sample frequency alone. Suddenly, relatively clean data appear to be very noisy and routinely wanders outside of the training range. The model recognizes this behavior as abnormal in comparison to its training data. In this example, the data acquisition card should probably be restored to its original configuration.

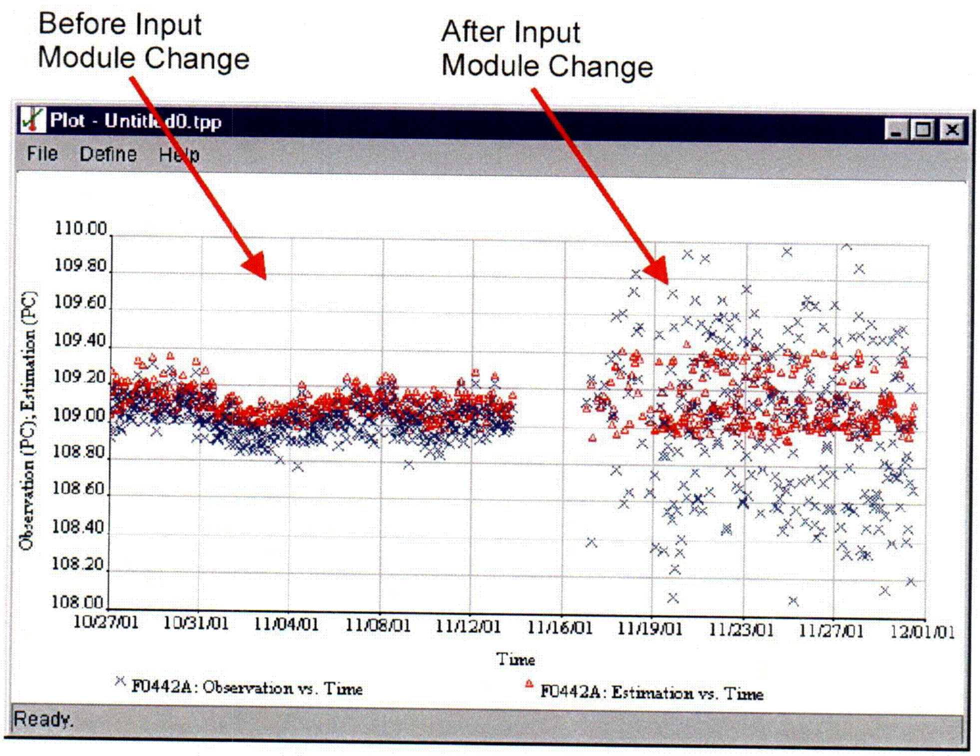


Figure 8-13
Test Data Outside the Training Range – Data Acquisition Card Replacement

8.2.5 Adjusting Phase-Determiner Settings for Transients

Section 5.4 explains the advantages of using submodels that are established by partitioning the overall model along the boundaries of specific operating states. The procedure used to define and partition these operating states is referred to as a *phase determiner,* and its result determines the phase in which the system is operating at any given time. For example, reactor power is commonly used to segregate high-power operation (near 100 percent power) from lower power operation. Even when a phase determiner is applied to a model, abnormal conditions might occur near the phase boundary between the submodels. Figure 8-14 shows an example in which turbine first-stage pressure operates for an extended period at about 735 psig (5068 kPa), but drops to almost 725 psig (4999 kPa) during a short transient (which is outside of the range of the data used for training).

C35

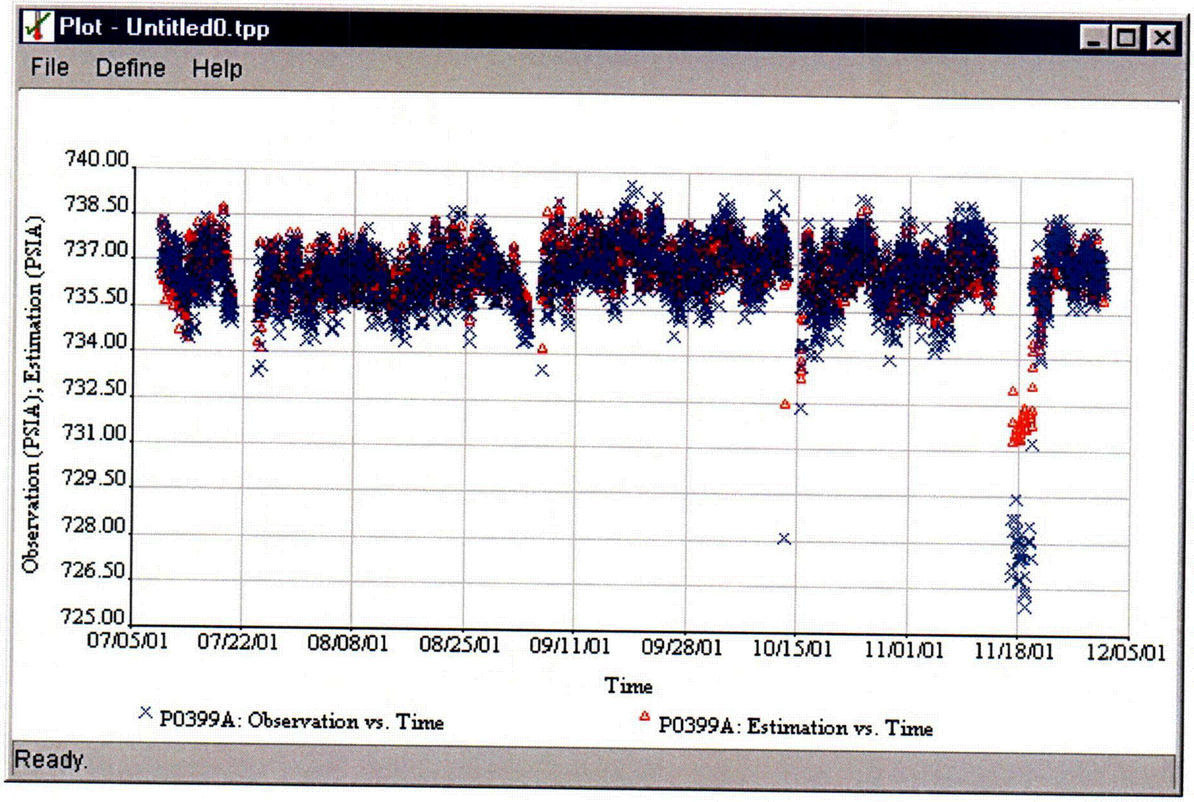


Figure 8-14
Signal Behavior During a Transient

Figure 8-15 shows another example in which a single transient occurs during an entire year of operation. The model was not trained to recognize this transient as normal behavior and, therefore, declared the channel as failed during the transient. Immediately after the transient, the channel returned to its expected operating state. In this instance, there is little benefit in training the model to recognize this transient behavior.

C36

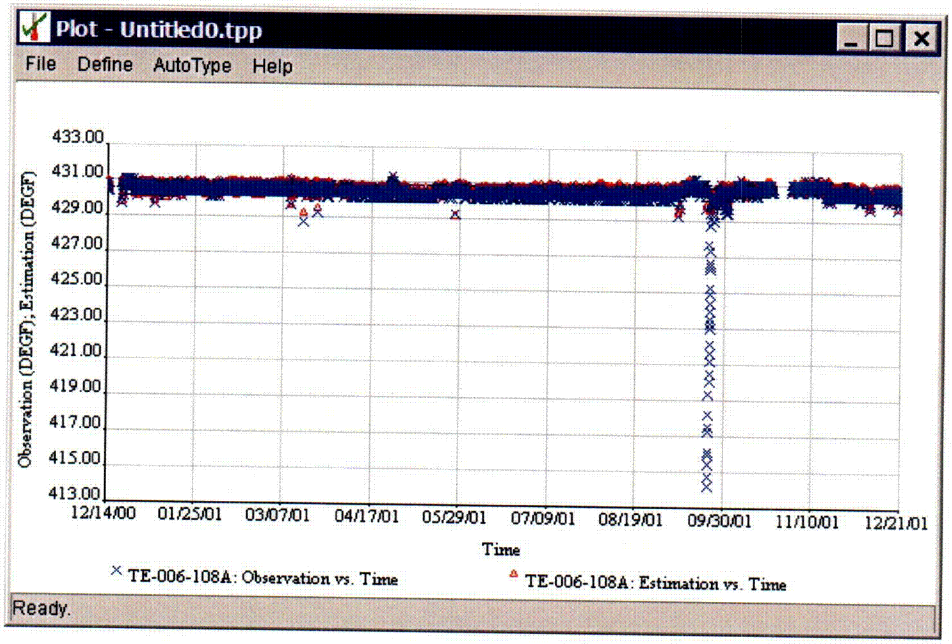


Figure 8-15
One Transient During an Extended Period of Operation

Short-term transients can occur that cause signal values to vary outside of the training space. As these transients occur, signal failures will appropriately be identified because the model does not recognize this operating state. Figure 8-16 shows another example of short-term transients. Notice that there are periodic spikes in the flow rate as pumps are adjusted. These flow spikes are normal and expected events. The training data included the first two spikes because this is a known and expected operating state. It should be noted, however, that the next three spikes exceed the limits of the training data, thereby resulting in alarms.

C37

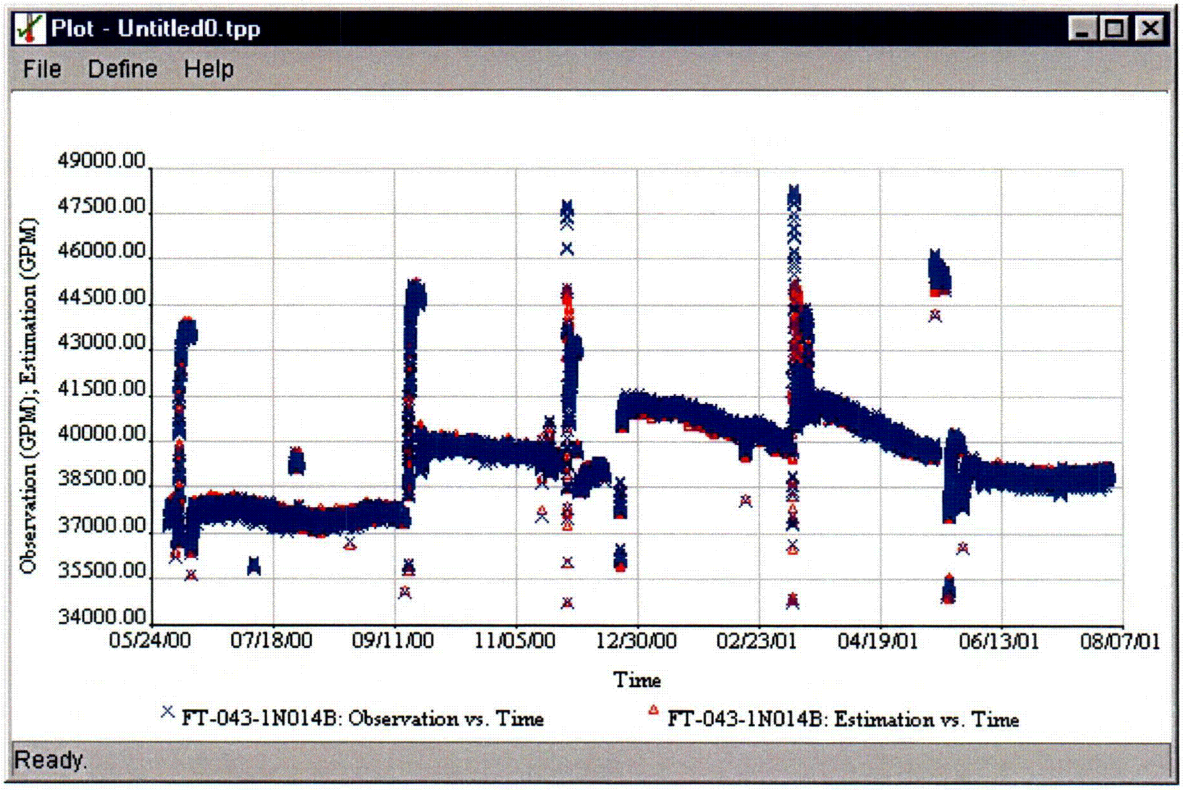


Figure 8-16
Example of Routine Transients Exceeding the Training Space

One of the three following approaches can be taken here:

- Retrain the model with additional data from the largest observed transient so that future smaller transients are not treated as signal failures.
- Do not retrain the model with additional data, and accept that failure declarations might occur during these transients. This is generally the preferred approach for short-term transients (as shown in Figure 8-16) after each transient appears normal with little deviation. This channel appears to be operating normally with no problems.
- Define a separate phase (using a phase determiner) for the transient behavior. This would cause transient data to be partitioned from the other data, effectively turning off all fault-detection processing while the system was in the transient phase and eliminating the false alarms.

If the transient rarely occurs, the preferred approach is to take no action. If the transient routinely occurs, retraining to include the transient or excluding the transient using the phase determiner is probably the preferred approach. Transients and the signal behavior should be reviewed at the phase boundaries. In some cases, the phase determiner should be modified to exclude untrained transients such as this.

C38

8.2.6 Occasional Outliers

Occasional data spikes or dips will occur that defy any explanation. Some of these occurrences are probably data acquisition problems, although the specific cause is often difficult to determine. Figure 8-17 shows an example in which two redundant steam pressure transmitters simultaneously dip from the normal operating range of about 830 psig (5723 kPa) to approximately 810 psig (5585 kPa). This change is sufficient to cause both channels to be simultaneously declared as failed. This dip occurred only once during the operating cycle. A review of other correlated data shows that no transient occurred during this period and that the data are most likely a partial data dropout caused by the data acquisition system.

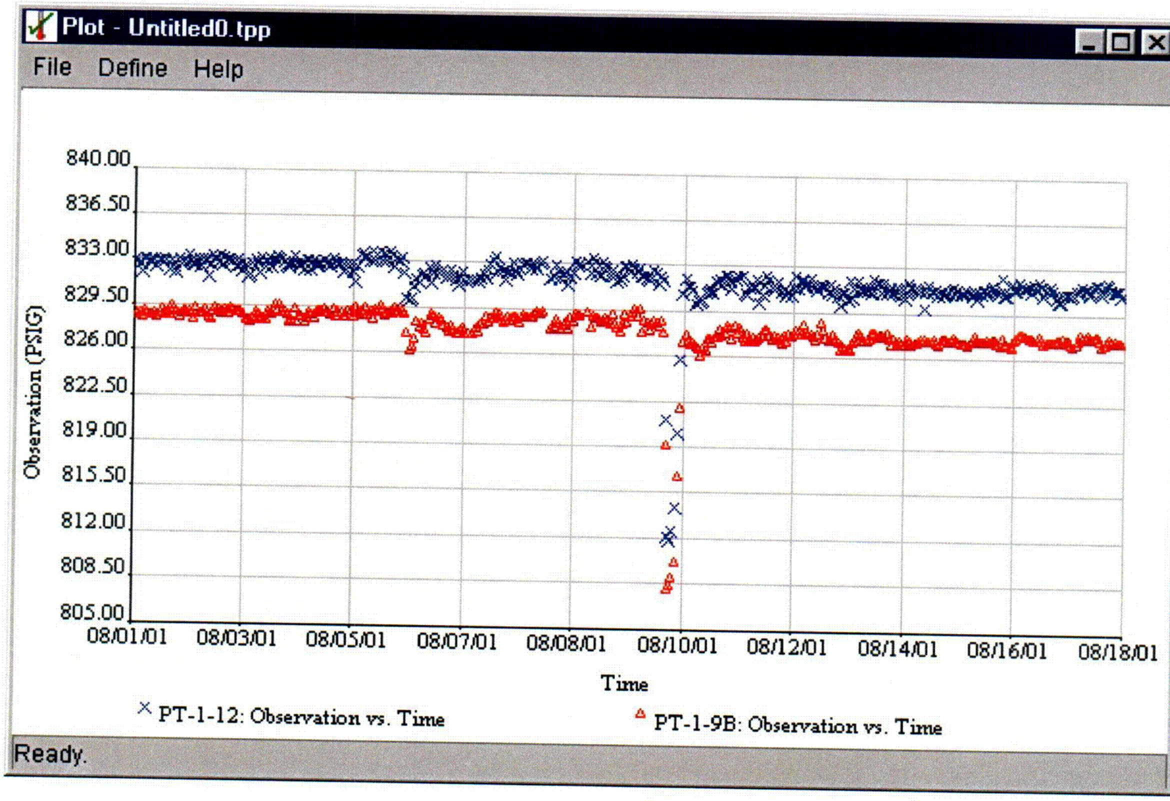


Figure 8-17
Short-Term Outlier

If an identified failure "recovers," the problem was probably not a sensor or instrumentation problem. What is important in terms of monitoring calibration and assessing drift is whether the channel behavior changes over time. Referring to Figure 8-17, it is apparent that a short transient of some sort simultaneously affected both redundant channels with a full recovery to normal conditions within a short period. Any failures identified during this period are not actual instrument failures; they are a consequence of either inadequate training for this event or data acquisition problems.

The signal behavior shown in Figure 8-17 is acceptable before and after the event. Rather than attempting to train the model to recognize an unexplained transient that only happened once, it is

C39

recommended that no action be taken. Rare and unexplained events will occur in real-world data. The model's recognition of these events as abnormal is an expected and desirable result. It is preferable to evaluate such events on a case-by-case basis rather than to include these rare cases in the training data and, therefore, risk desensitizing the model to failure events in general.

8.2.7 Using Threshold Settings for Overly Sensitive Alarms

8.2.7.1 Overview of Threshold Settings

Fault detection is based on a comparison of the difference or deviation between each observation and its corresponding estimate (referred to as the residual). Depending on 1) the signal's fault-detection settings and 2) the variance of the training data residual values, fault detection can be very sensitive, resulting in alarm generation for deviations of less than 0.1 percent. For many applications, this is more sensitive than necessary. It might be preferable to include simple threshold settings as a fault-detection tool.

Threshold settings are applied directly to the residuals for an analysis. Figure 8-18 shows an example of threshold settings applied to a signal. Two levels of drift have been specified—an allowable drift of ±1 percent and a maximum drift of ±1.5 percent. As the sensor starts drifting low, the residuals become increasingly more negative, eventually reaching the allowable drift limit.

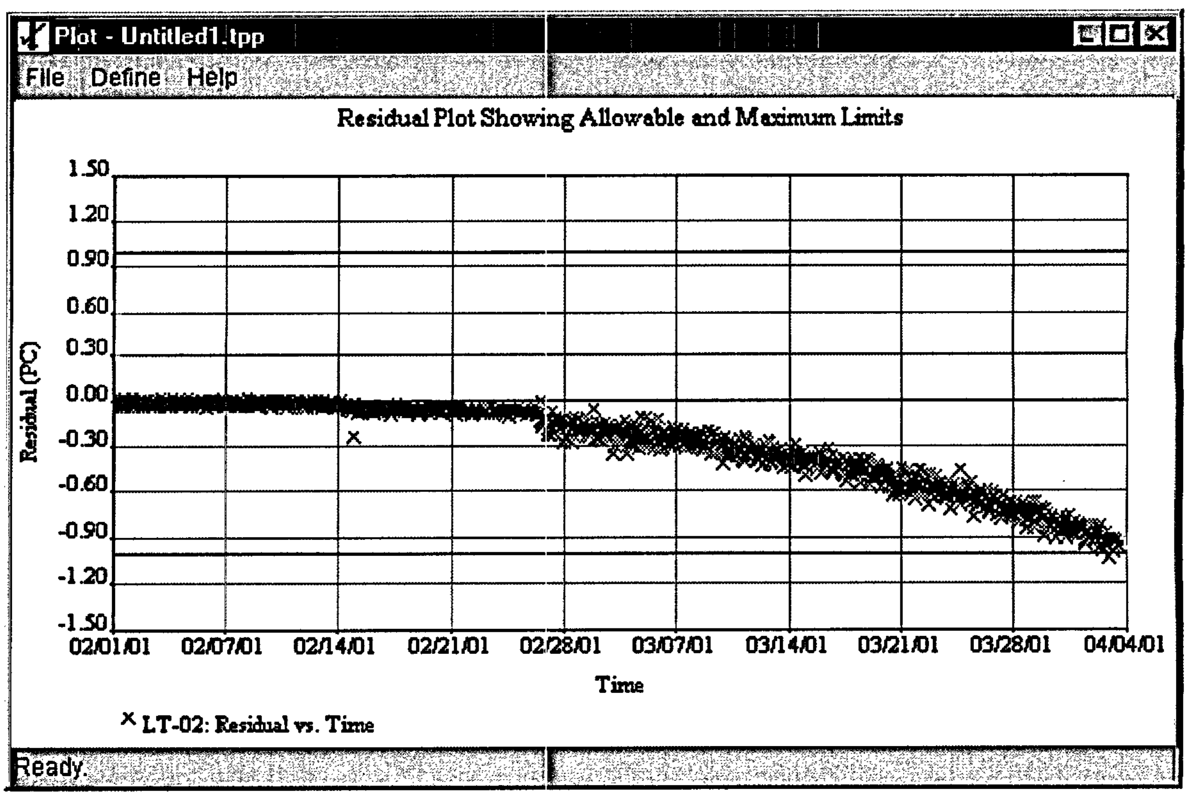


Figure 8-18
Threshold Settings Applied to Residual Plot

It should be noted that the channel shown in Figure 8-18 was identified as failed when the residual was only about -0.2 percent. This type of residual plot provides another tool to assess the urgency of any required corrective action. As shown in Figure 8-18, drift is occurring at a slow rate, which can allow for preplanning of any calibration activity.

Although threshold settings are less sensitive than statistically based fault-detection tools, it is recommended that the maximum acceptable drift be specified for each signal validated by the model. This will allow the user to retrieve residual plots during monitoring. Whenever fault alarms are received, a residual plot (such as the one shown in Figure 8-18) can provide additional insight into the severity of the detected drift.

8.2.7.2 Establishing Threshold Limits

Threshold limits are useful for evaluating fault detection during model development, but they should be adjusted as necessary before the model is placed in service. As part of model completion, threshold limits should be established based on the drift allowances of the monitored instrument channels. For safety-related channels, the threshold limits depend on the allowances specified in the corresponding set point study, if applicable. For non-safety-related channels, the threshold limits depend on the allowed channel variation for acceptable system performance.

Table 8-1 shows the typical elements of uncertainty that are considered important for most instrument loops. The uncertainty terms associated with the sensor are usually of the most interest. Rack-mounted signal-conditioning equipment, which usually performs quite well and often has a corresponding lower contribution to the channel uncertainty, should be considered whenever additional modules are in the instrument circuit.

Table 8-1
Instrument Channel Uncertainty Sources

Uncertainty Term	Present in On-Line Monitoring Path?	Present in Safety-Related Trip Path?	Included in Sensor Calibration?
Process measurement effect (PME)	X	X	
Process element accuracy (PEA)	X	X	
Sensor reference accuracy (SRA)	X	X	X
Sensor drift (SD)	X	X	X
Sensor temperature effect (SD) (normal variation)	X	X	X (partial)
Sensor pressure effect (SPE)	X	X	
Sensor vibration (SV)	X	X	
Sensor M&TE accuracy (SMTE)	X	X	X
Isolator reference accuracy (IRA)	X		
Isolator drift (ID)	X		
Isolator temperature effect (ITE)	X		
Isolator M&TE accuracy (IMTE)	X		
Computer input A/D accuracy (A/D)	X		
Bistable reference accuracy (BRA)		X	
Bistable drift (BD)		X	
Bistable temperature effect (BTE)		X	
Bistable M&TE accuracy (BMTE)		X	

With regard to on-line monitoring, the sensor uncertainty elements can be grouped according to whether they are associated with 1) process/environmental effects or 2) calibration effects. The uncertainty elements associated with process/environmental effects explain why redundant channels might not display the same value. There is some random variation in the measurements caused by these uncertainty elements. The uncertainty elements associated with calibration effects represent specifically what an on-line monitoring program is evaluating, and it is these terms that should relate directly to the specified threshold limits.

Suppose the following uncertainty values are provided (or are considered allowable) for a non-safety-related channel:

SA = ±0.5%, sensor reference accuracy

SD = ±1.5%, sensor drift

SMTE = ±0.25%, measurement and test equipment uncertainty

The combined uncertainty of these values can be calculated as follows:

$$SU = \pm\sqrt{SA^2 + SD^2 + SMTE^2}$$
$$SU = \pm\sqrt{0.5^2 + 1.5^2 + 0.25^2} = 1.6\%$$

The uncertainty of the MSET estimate is a consideration because it can also affect the drift allowance for the channel. For example, if the estimate uncertainty is ±0.25 percent, this uncertainty calculation would be adjusted as follows:

$$SU = \pm\sqrt{SA^2 + SD^2 + SMTE^2 - MSET_{Unc}{}^2}$$
$$SU = \pm\sqrt{0.5^2 + 1.5^2 + 0.25^2 - 0.25^2} = 1.58\%$$

For this channel, threshold limits might be established as follows:

- Allowable = 1.2 percent (arbitrarily selected)
- Maximum = 1.58 percent

The EPRI On-Line Monitoring Implementation Project has coordinated with two Department of Energy (DOE) Nuclear Energy Plant Optimization (NEPO) projects to review the estimation of uncertainty for various plant models and data types. Results from these related efforts are provided in *On-Line Monitoring of Instrument Channel Performance, Volume 3* [2].

Safety-related instruments have additional considerations for uncertainty that have been documented elsewhere under the tasks of this project [5].

8.2.8 Incorrect Initial Training

Figure 8-19 illustrates another type of potential problem in which a model was trained with one steam flow sensor already out of calibration (the observations are shown as blue crosses, and the estimates are shown as red triangles). The model has been trained to recognize the erroneously low signal of this steam flow sensor as normal. Eventually, the out-of-calibration condition was identified by conventional methods, and the transmitter was recalibrated. After calibration, the sensor was identified as failed because the model was trained with bad data. The model should be retrained with new data after the calibration. The prior training data must be excluded because it incorrectly identifies the out-of-calibration condition as normal.

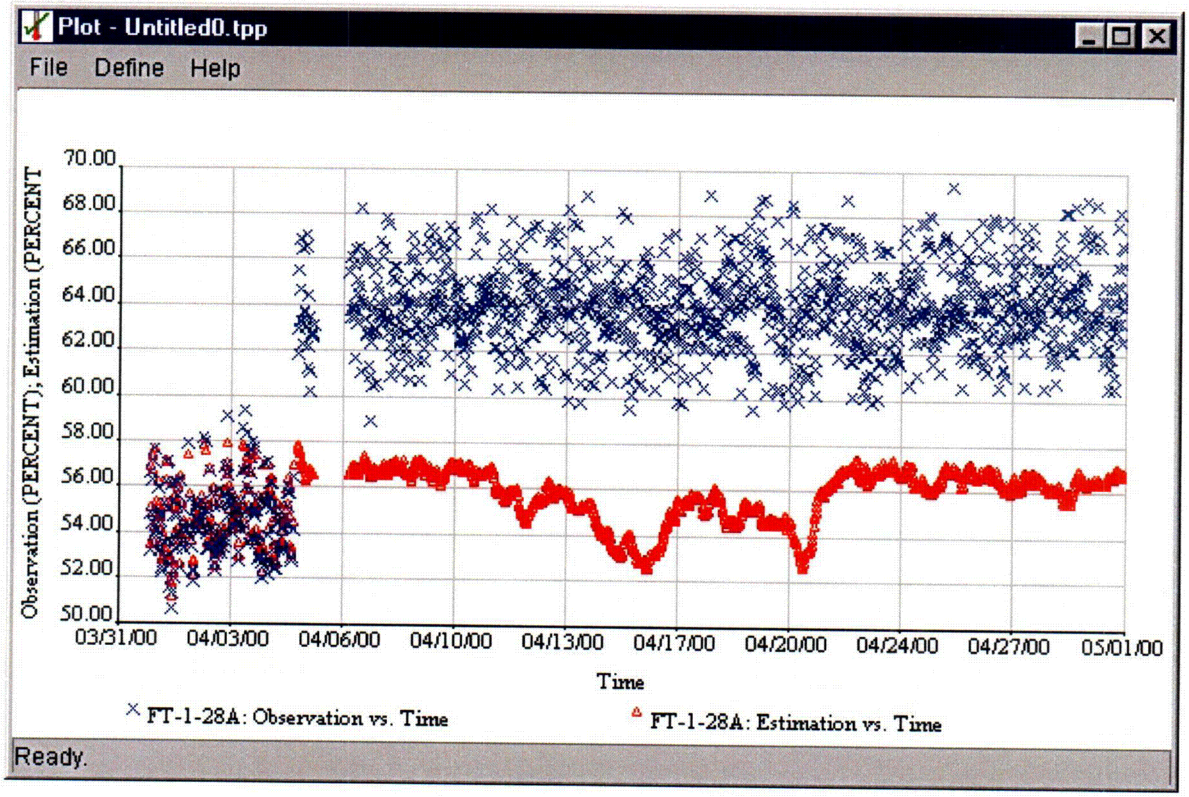


Figure 8-19
Recalibrated Sensor - Model Trained on Out-of-Calibration Data

Figure 8-20 shows a minor example of an incorrectly trained model (the observations are shown as blue crosses, and the estimates are shown as red triangles). This temperature sensor varied from other redundant sensors by about four degrees, but the model was trained to recognize this behavior as normal. After recalibration, this sensor matched the signals from the other redundant channels, but the model routinely declared this channel as failed because it was not trained for this new operating condition.

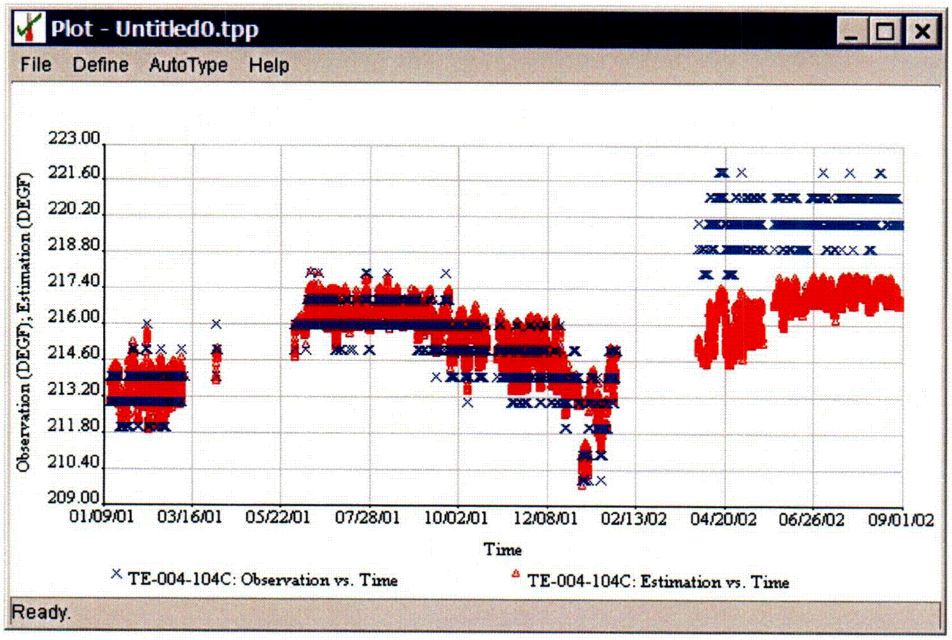


Figure 8-20
Recalibrated Sensor - Initially Out of Calibration

8.2.9 Equipment Operating States Not Covered by Available Training Data

There are likely to be some models for which it will be difficult to train for all possible operating states. Figure 8-21 provides a simple example. Depending on the system requirements, either one, two, or three pumps might be running.

C41

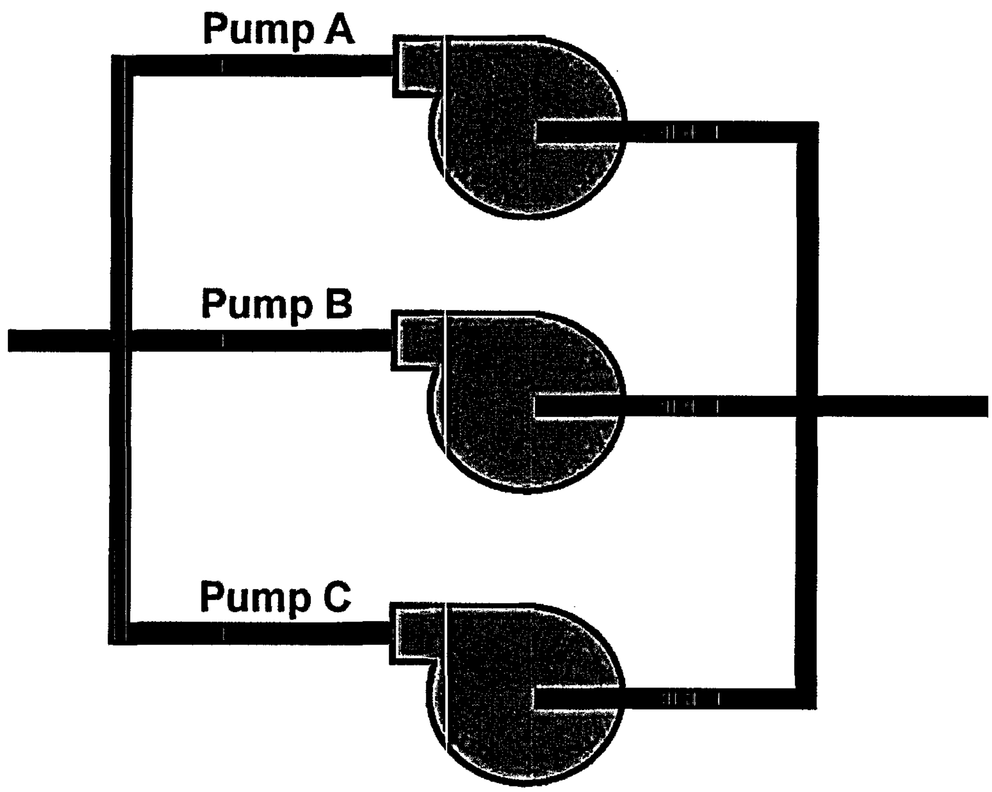


Figure 8-21
Many Possible Operating States

Referring to Figure 8-21, the following possible system flow conditions should be considered:

- Low flow—only one pump is running. Three different operating states are possible.
- Medium flow—two pumps are running. Three different operating states are possible.
- High flow—all three pumps are running. This is probably the easiest operating state to model.

As shown, there are at least seven different operating states if the three pumps are individually instrumented. This becomes even more complex if the pumps can operate at variable speed.

Some plants preferentially operate two specific pumps with a third pump off. Figure 8-22 shows an example of a condensate booster pump that almost always runs. While running, a pump bearing temperature measurement usually indicates about 150°F (65.6°C). When the pump motor is turned off, the temperature drops to about 75°F (24°C). During an 18-month period, this pump was stopped once for only two days. For the non-operating state, it is unlikely that adequate training data will be available for such an infrequent operating configuration.

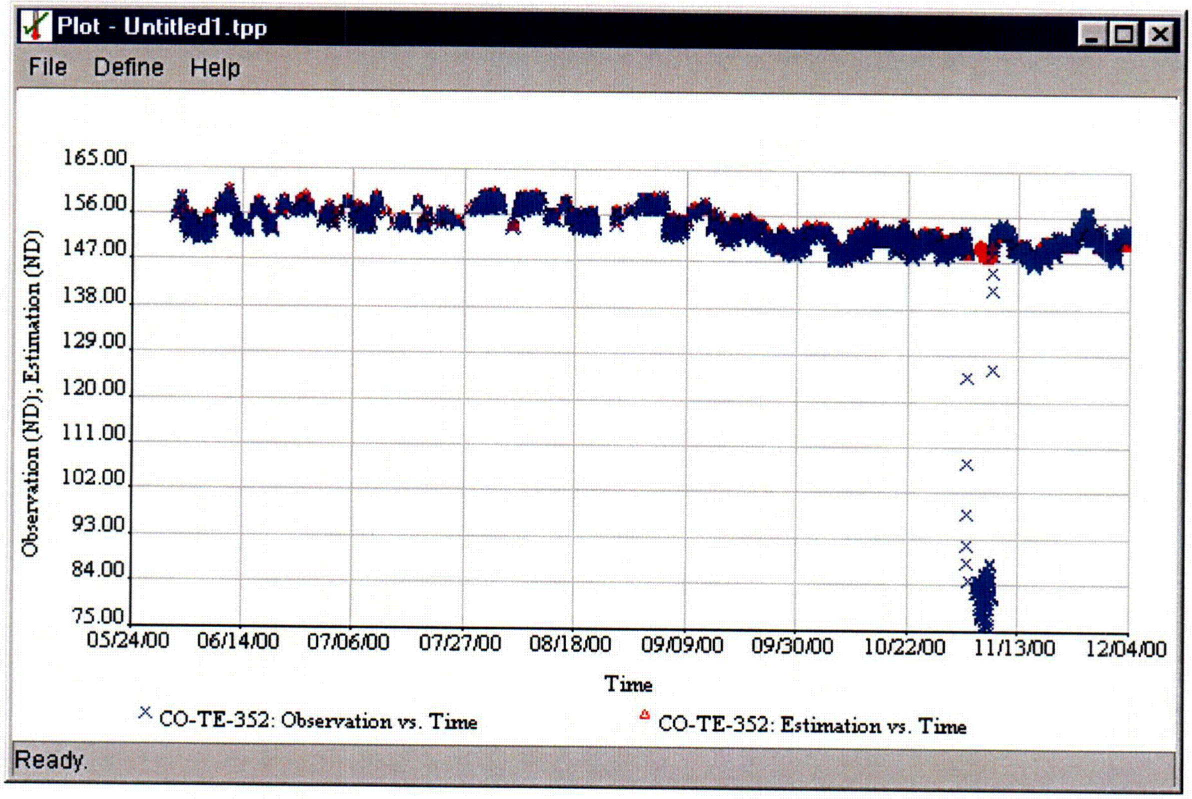


Figure 8-22
Example of a Pump That Almost Always Runs

Uncommon operating states are a real issue for training and operating a model. The important question to be resolved is which operating states will be validated and which operating states will not. For many models, it is unlikely that adequate training data can be acquired to train the model on all possible operating states. For that reason, phase determiners should always be considered so that untrained operating states are excluded from signal validation and subsequent failure alarms.

8.2.10 Inadequate Initial Training Within the Defined Operating Space

The MSET training vector selection method is not perfect. Even when adequate training data are available to define an operating space, estimation can be influenced by the combination of observations compared to the corresponding vectors contained in the training matrix. Figure 8-23 shows an example in which the observations (blue crosses) and the estimates (red triangles) track together very well for an extended period, followed by the estimations jumping while the observations remain almost unchanged. This change in the estimate was prompted by a small dip in the observations, which took some time before the estimates again tracked with the observations. The vectors selected for training do not adequately cover the operating space in this particular region, and additional vectors should be specified for the training matrix.

C42

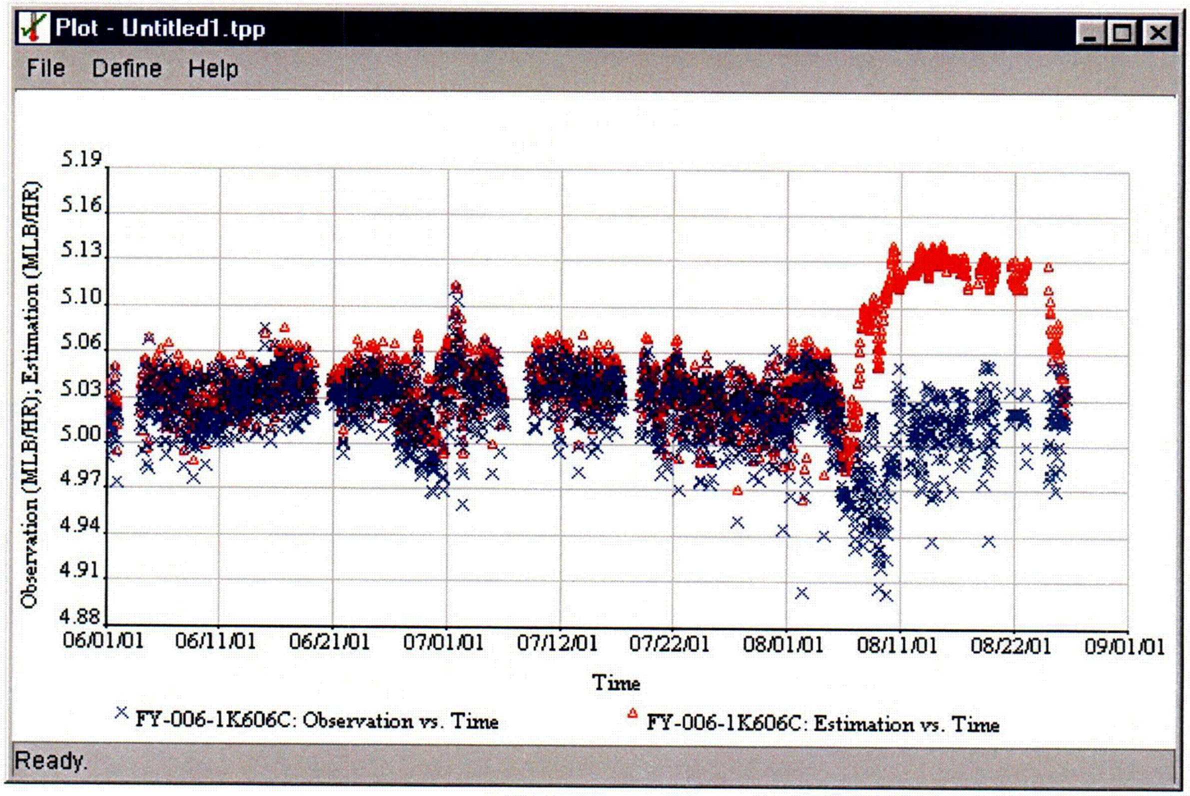


Figure 8-23
Unexpected Change in Estimate

Figure 8-24 shows a more dramatic example of a pump bearing temperature sensor (the observations are shown as blue crosses, and the estimates are shown as red triangles). The model was trained to recognize both running and off conditions. As trained, the model has trouble differentiating between the two states because of the influence of other signals in the model. The model needs to be retrained with additional training vectors. Or, if the pump is seldom off, it should probably be trained with either 1) a phase determiner based on the pump state or 2) without the pump off data so that it does not try to perform fault detection in both of the two possible states using a single model.

C43

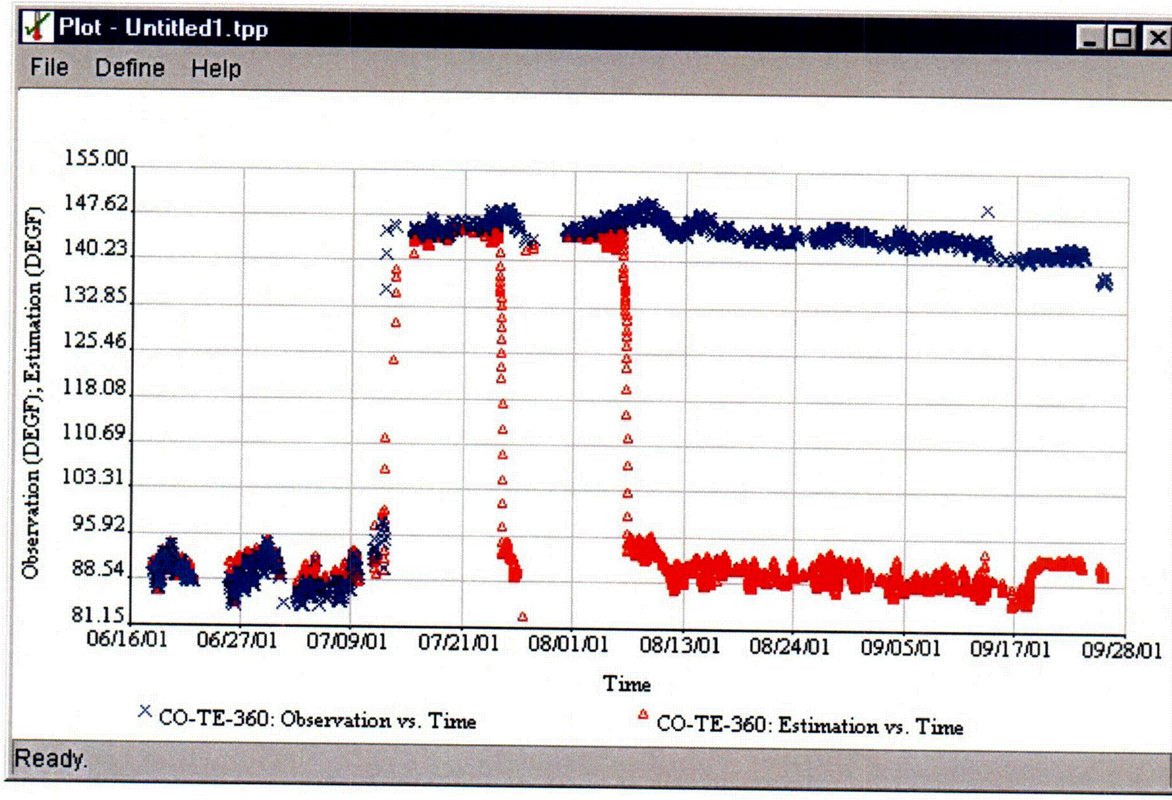


Figure 8-24
Inadequate Training for Pump On and Off Conditions

9 OPERATING IN ON-LINE MODE

An on-line monitoring system can provide timely assurance of signal data validity and equipment condition to a plant operator. The means of implementing an on-line monitoring system and the frequency at which the plant operator evaluates the monitoring system's results might vary significantly from application to application. In limited cases, it might be required to provide real-time monitoring with essentially continuous reporting to the operator. More commonly, on-line monitoring can be optimally implemented in a less-than-real-time mode. On-line monitoring systems can readily satisfy both real-time and less-than-real-time on-line monitoring requirements, thereby making them suitable for a wide variety of applications.

9.1 Modes of Operation

This section provides an overview of what is meant by on-line monitoring and describes the process by which a user would make the transition from a manual batch mode to an on-line mode of operation.

On-Line Monitoring of Instrument Channel Performance [5] defines the following possible options for an on-line monitoring system:

- An automated system that performs data acquisition and analysis essentially continuously in real time at the system-specified sample rate
- An automated system that performs data acquisition and analysis at discrete specified intervals
- An automated system that is normally off and is manually activated to perform data acquisition and analysis at a set interval (at least quarterly)
- A manual system in which data are acquired manually on at least a quarterly interval and entered manually into a computer program for the purpose of analysis

The differences among these options relate primarily to the degree of automated signal acquisition and the frequency of data collection and analysis. All but the first of the options are actually operating in a batch mode in which data are acquired and stored in files prior to performing the analysis.

The EPRI On-Line Monitoring Implementation Project has considered what is meant by on-line monitoring and has developed the following descriptions:

- Off-line – The software operates only in batch mode with training and testing data files prepared separately and evaluated on user command. This describes most on-line monitoring methods implemented to date.
- Periodic on-line – The data files are automatically updated with the latest information each time the software is run. This type of operation is effectively real-time and on-line in that it acquires the latest available data when the user accesses the model; however, this mode does not mean that each model is running continuously in the background. Also, data continue to be stored in data files rather than acquired in real time.
- Real-time on-line – The software is always running and sampling data at a specified frequency. Upon failure detection, the system notifies the user by some method. Upon notification, the user then opens a software viewer to review the cause of the failure alarm.
- Custom on-line – On-line monitoring as described previously with a unique user interface developed by the power plant. The software might be embedded in another application and might be inherently subservient to the host application.

The degree of automation in the on-line monitoring method can vary substantially among users. Some users will operate quite successfully in a batch mode and never operate in an on-line mode. It is possible, though arguably less cost effective, to meet most nuclear plant sensor and equipment condition-monitoring objectives while operating in batch mode. Other users will operate in a periodic on-line mode to optimize the method of fault detection, alarm, and graphical display of results.

On-line monitoring can provide an early indication of deteriorating sensors and equipment. With advanced warning of impending problems, plant personnel can take corrective action during periods of planned downtime, increasing the productive availability of the monitored equipment. To accomplish these objectives, the on-line monitoring system must integrate smoothly into the daily workflow of the I&C engineers and technicians, must be easy to operate, and should require only modest levels of specialized training.

In one EPRI project implementation, the on-line monitoring software was operated autonomously on a nightly basis so that the prior day's monitoring results were available each day for evaluation by the plant's I&C technicians. This periodic on-line monitoring approach eliminates the need for an I&C technician to manage or wait for the data extraction and diagnostic processing, thereby enabling the technicians to most efficiently complete their reviews on a daily basis. The approach integrates smoothly into the daily workflow of personnel involved in on-line monitoring at a nuclear power plant.

9.2 Making the Transition From Batch Mode to On-Line Mode

The primary difference between on-line mode and batch mode is the nature by which the plant data are acquired and processed. An on-line system will typically implement some form of direct or automated connection to the plant data historian. Further, the on-line system will typically

implement automated processing of the acquired data. A batch system will typically accomplish these objectives via several manual steps under the direction of the user.

Most users will initially operate their monitoring system in batch mode. Batch mode is typically a file-based mode of operation with a manually implemented interface to the plant data archive. Batch mode is useful for evaluating software capabilities, establishing plant-specific requirements, training software users, developing plant-specific monitoring models, and performing model acceptance testing. Several important roles for batch mode operation are described here. The remainder of this section will discuss on-line modes of operation.

9.2.1 Training as a Batch Operation

On-line monitoring relies on a user-provided set of historical operating data (known as the training data) to learn its internal model of the normal operation of the monitored signals and equipment. There are two important attributes of the data used to train a model. First, the data should contain all modes and ranges of operation that are to be considered normal operation of the monitored signals and equipment. Second, the data should not contain any operating anomalies, sensor failures, or equipment failures that would be considered as abnormal operation of the monitored signals and equipment. These criteria are prerequisites for training an effective model for use in on-line monitoring.

Ultimately, the quality of the estimates depends on the fidelity of the training data. For this reason, the training data require a careful review prior to use for model training. At the present, this is accomplished by a combination of automated screening techniques and engineering analysis. Data files evaluated in batch mode are generally more appropriate for this type of work than are on-line data sources. The experience at all participating power plants to date is that all training data files require an evaluation to identify and remove bad data.

In many cases, it might not be desirable to include all operating states (such as normal but infrequent transients) in the on-line monitoring model. In this case, the data for the unmodeled operating states should be removed from the training data. Some on-line monitoring software (such as SureSense) can provide the means to perform this type of operating state data, partitioning automatically without requiring manual modification of the training data files.

In all cases, it is important to maintain a configuration-controlled archive of the training data used as the basis for training an on-line monitoring model. The training data might be required for periodic model updating (retraining) and for model verification or acceptance testing. Data files are preferred over on-line sources for the purposes of archiving the cleaned-up and approved training data.

These considerations mean that training data will normally be contained in data files and the training process will likely be performed as a batch operation. Retraining considerations also suggest that new training data might be added to, rather than replace, existing training data.

9.2.2 Periodic On-Line Monitoring

Periodic on-line monitoring will be the most common mode of implementation for most power plant systems. The frequency at which the monitoring results are evaluated will depend on the criticality of the system, the rate at which the monitored failure modes can progress, and the availability of personnel resources. For example, an on-line monitoring program for sensor calibration reduction requires a monitoring frequency of at least once per calendar quarter. Software available for on-line monitoring will generally automate the periodic evaluation task to the extent that much higher frequencies (such as daily evaluation intervals) are cost effective.

An implementation of periodic on-line monitoring was completed as part of the EPRI project and is described here. The implementation was based on a Microsoft SQL Server plant data historian system in combination with the SureSense on-line monitoring software. The implementation included a data bridge for periodic data extraction and management (as described in Section 9.3). The configuration of the overall periodic on-line monitoring system is shown in Figure 9-1.

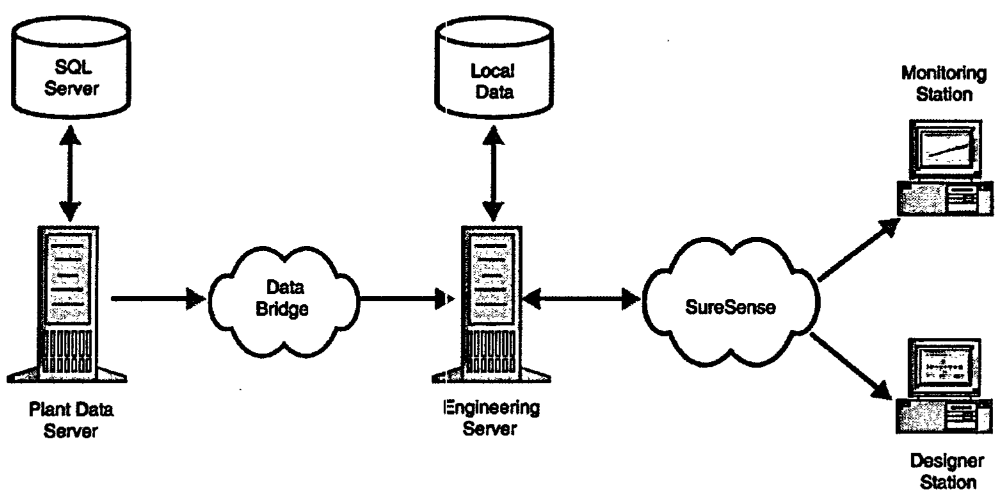


Figure 9-1
Reference Implementation for Periodic On-Line Monitoring

In the reference implementation, the on-line monitoring procedure followed this general approach:

- Plant operating data are acquired and archived to the SQL Server database with no change to existing plant software or procedures.
- The data bridge software runs automatically under a scheduler to extract the local data required for analysis by the on-line monitoring models. The data bridge is automatically run nightly during periods of low user loading on the network and database.
- The data bridge software automatically manages the local data archive to maintain a user-defined look-back history of plant operating data for the analysis.

- The on-line monitoring software runs automatically under a scheduler to perform a nightly evaluation of the current local plant operating data for each enabled model and automatically stores the run results in combination with the local data.
- The user accesses the daily run results at any time through either a designer interface or a monitoring interface. The designer interface is password protected and provides model design and modification features. The monitoring interface is also password protected and permits viewing of results and reports, but it does not permit model modifications.

This automated approach ensures that the user will spend a minimal amount of time reviewing the daily results. Experience shows that the data extraction from the plant data historian is the time-consuming step, often requiring several minutes for extraction of data for a 24-hour period at a 1-minute sampling rate. By precompiling the look-back data set and monitoring results, this implementation provides the user with nearly instantaneous access to the results and reports. When numerous models are placed into service, the accumulated time savings provided by instantaneous access to the results is appreciable.

In this implementation, the on-line monitoring software further provides the capability for a designer to query the SQL Server database directly and to acquire data for any valid time interval and signal subset. This capability uses the same database interfaces as the data bridge. Generally, data extraction time delays associated with the SQL Server are not problematic for the infrequent queries made by a design- or analysis-oriented user. Thus, the designer can perform a periodic assessment over any time interval of interest. This is considered an important feature for performing an operability assessment after a failure event has been identified.

An alternative approach supported by SureSense and other on-line monitoring software is to maintain the system in an always-on, real-time mode with a dedicated computer, a dedicated data connection, and a local data archive for look-back capability. This approach is outlined in the following section.

9.2.3 True On-Line Monitoring

The available on-line monitoring software available is readily capable of operating in a true on-line mode. Rather than evaluating data contained in a data file, the software receives data directly from the plant computer or data historian as a continuous data stream, sampled at a specified frequency. For example, the on-line monitoring software can acquire data in the following ways:

- Read data directly from one or more data files
- Read data directly from a data acquisition system in real time as a data stream
- Read data directly from a plant data historian in real time as a data stream
- Read data directly from any other network or internet-accessible source in real time as a data stream
- Read data simultaneously from any combination of the above

SureSense also provides an autocode compiler option that can produce stand-alone embeddable software capable of performing these data acquisition modes when combined with a user's custom software application.

Several technical decisions affect the method by which on-line monitoring is accomplished. In a real-time on-line mode, operating data would be evaluated as they are acquired. A real-time mode implies that the system is either continuously or regularly checked for alarm conditions. Certain performance-critical systems might merit this level of attention; however, a greater number of models can be adequately monitored on a daily or less frequent basis.

A real-time mode of operation is conceptually simple, but the method of implementation will require careful planning. The system will require dedicated computing resources and data connections. The acceptability of increasing the network load during heavy traffic periods should be confirmed. Procedures for continuous evaluation, results storage, and real-time notification to personnel should be established. Finally, procedures or software capability to capture a look-back window for engineering evaluation of a fault event should be provided. The ability to provide simultaneous look-back and engineering analysis capability while maintaining the real-time on-line monitoring function should be considered.

9.2.4 Look-Back Functions

An on-line monitoring system could function by evaluating a data stream (observation by observation) and identifying failed or degraded sensors as the failures occur. For some industries, a simple declaration of the failure might well be adequate. However, nuclear plant users will generally want to review historical data whenever a failure declaration is made. This historical review is referred to as a *look-back* function and is an important part of any on-line monitoring system. Specifically, the purpose is to:

- Evaluate current data in the context of recent historical data to verify the failure event
- Determine if the failure is a real instrument or equipment failure or if it is the result of inadequate model training for a new but valid operating mode of the equipment
- Identify the point of onset and character of the failure (which might be important for operability assessments)

Historical data are always contained in a data file or database, not sampled in an on-line fashion. This means that look-back capability is by its nature a batch mode of operation even in the most automated system. Some on-line monitoring systems will include a local look-back data cache to enhance the efficiency of their data plotting and analysis functions. This is deemed to be a highly desirable feature for most nuclear plant applications. Look-back periods on the order of 60–90 days are recommended. This is usually a long enough window to verify a failure event and to perform an operability assessment when a failure event occurs.

9.3 Data Bridge Description

9.3.1 General Description

When implementing an on-line monitoring system, plant-specific data issues must be addressed and resolved. A key issue is the level of automation and the efficiency of the on-line monitoring interface to the plant data archive. Several options that have been previously discussed include the following:

- An always-on, automated system that performs data acquisition and analysis continuously in real time at a specified sample rate
- A periodic automated system that performs data acquisition and analysis at discrete specified intervals
- An automated system that is normally off and is manually activated to perform data acquisition and analysis at a set interval (at least quarterly)
- A manual system in which data are acquired manually on at least a quarterly interval and entered manually into a computer program for the purpose of analysis

In power plant applications, the second or third approach is generally preferred. A data bridge can be an important element of the implementation approach in either case. A data bridge is defined as a middleware software application that manages the data transfer between the plant data historian and the on-line monitoring software application. A well-written data bridge will minimize or eliminate the need to modify existing plant data acquisition and data archiving systems and procedures.

Plant data are typically stored in one of a number of proprietary plant data historian software databases. Many such data historian databases are in use in the power generating industry today. All modern plant data historians provide a connectivity layer that enables data access by other software applications such as an on-line monitoring system. An open database connectivity (ODBC) driver or an application-programming interface (API) typically provides access to data stored in the database. It is important to understand the capabilities and limitations of the plant data historian software when planning for an on-line monitoring system implementation. Some data historian vendors provide connectivity modules as part of the base software, while others might charge separately for these modules. Data historian access module cost might be an important factor in budgeting for a project.

A data bridge functions as a connecting layer between the plant-specific data historian and the on-line monitoring software. The data bridge can be internal to the on-line monitoring software or can be operated as a stand-alone program. The SureSense software used in the EPRI project enables both approaches; however, there are advantages to each that should be considered.

The primary advantage of an internal data bridge between the plant data historian and the on-line monitoring software occurs during model training and acceptance testing. At this point, the designer will evaluate model performance over various historical periods. Implementing an

internal data bridge enables this development work to proceed with minimal human intervention to accomplish data gathering. The need to store and manage redundant copies of the data in multiple formats is also eliminated.

The downside of a direct connection between the plant data historian and the on-line monitoring software is that queries made to the plant data historian can be relatively slow to complete. Extracting large data sets can become unproductive if the queries made to the data historian take too long. A stand-alone data bridge can be used to mitigate this processing overhead and make optimal use of engineering resources.

A stand-alone data bridge provides a software tool to automatically extract the required information from the plant data historian and to format the data for highly efficient access by the on-line monitoring software. The data bridge can be provided in any number of configurations and capabilities dependent on the on-line monitoring software selected.

9.3.2 EPRI On-Line Monitoring Implementation Project Data Bridge

The data bridge used by the EPRI On-Line Monitoring Implementation Project is a stand-alone software program that provides an automated means to extract plant data from any data archive with a data access interface. The bridge automatically connects to the data archive, extracts model-specific data from the archive, and updates a local data file in binary signal data file (SDF) format. The bridge will automatically maintain a moving window of data beginning with the most recent data and extending for a user-specified look-back time interval. The bridge can be run in a scheduled and unattended mode. It is typically run when other network and database traffic is low.

Data extraction will typically be performed in a regularly scheduled fashion (such as once per day). The software can be configured to run unattended and will require human intervention only if the desired run behaviors need to be changed. The steps in the data extraction are listed here:

1. Select the set of signals to be acquired from the plant data historian.
2. Select the start and stop time for the data, or alternatively, select the current or initial time and a look-back interval.
3. Determine the overlap between the requested data interval and any previously extracted local data file.
4. Determine the time interval required from the plant data historian to update the local data file.
5. Connect to the plant data historian, and verify availability of the requested data.
6. Query the data historian for the requested data.
7. Time synchronize the newly acquired data with data contained in any previously extracted local data file.
8. Update the local data file with the requested data.

A reference data bridge implementation was completed as part of the EPRI project. The implementation was based on a Microsoft SQL Server plant data historian system in combination with the SureSense on-line monitoring software. The implementation accomplished a periodic on-line monitoring capability. The following requirements were specified for the reference data bridge implementation:

- Operating data will be extracted nightly from the plant's SQL Server database for SureSense processing.
- Operating data extraction will be an automatically scheduled batch procedure.
- Operating data will be extracted separately for any number of SureSense models.
- Extracted operating data will be saved to the engineering server in binary SDF format.
- Extracted operating data will include the data from the preceding 90 days; however, only the most recent day of data will be required to update a file archive on the engineering server.
- Extracted operating data will always overwrite the previous day's data in the same user-defined file location.
- SureSense operating data analysis will be performed nightly for any number of user-defined SureSense models.
- SureSense operating data analysis will be an automatically scheduled batch procedure.
- SureSense operating data analysis will include the data from the preceding 90 days.
- SureSense operating data analysis results will be saved to the engineering server in a binary format compatible with the SureSense data visualization and reporting tools.
- Operating data analysis results will always overwrite the previous day's results in the same user-defined file location.
- Operating data analysis results will be available for each model after processing.

The reference data bridge application was configured according to the these requirements. The following describes the reference implementation; however, variations on the implementation might be made to best accommodate plant-specific data processing environments.

The data bridge was implemented as a console application that performs data extraction from the SQL Server and saves the data in SDF file format to the engineering server. The data bridge implements the SureSense universal data access package to ensure compatibility with the SureSense application and to minimize development costs. The universal data access package is a plug-in data input and output software module that controls the data flow through the data bridge to the on-line monitoring application. Because the plant-specific data source characteristics are isolated in a plug-in software module, the universal data access package enables the data bridge and on-line monitoring software to connect to virtually any network-accessible data source in virtually any data format. A single plug-in module enables plant data system connectivity for both the data bridge software and for on-line monitoring system designers. SureSense users should consult their user documentation for more information about configuring these types of data access plug-ins.

The SureSense application was installed into a configuration-controlled directory under system administrator control. The installation created the default directory structure shown in Figure 9-2. The SureSense data bridge was installed in the top-level application directory for versions 1.4.1 and higher. The application directory and all subdirectories were assigned read-only user permissions; however, the system administrator was given full permission.

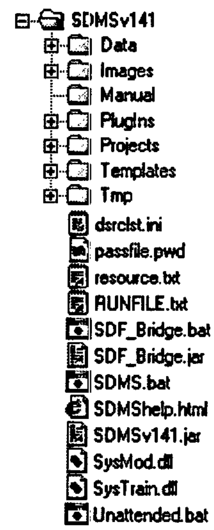


Figure 9-2
Directory Structure for a SureSense Data Bridge Implementation

9.3.3 Data Bridge Programming and Setup

The data bridge console application executable is named SDF_Bridge.jar. The application requires a single argument—the name of a resource file. An example resource file named resource.txt is provided with the installation. The application might be run manually from the command line, but more typically will be run under the control of a scheduler. In the reference implementation, the Microsoft Windows scheduler was used. The SDF_Bridge.bat file provides an example of an executable command file suitable for use with a scheduler.

The SureSense data bridge can access data from virtually any network-accessible data source using a built-in class loader that dynamically loads a data reader plug-in for each required data source. The data source location and the required reader plug-in for the data source are specified in the data bridge resource file (further described in the following paragraphs). The data bridge uses the plug-in to access the data source, to acquire the necessary data, and to reformat the data

to a local archive in SDF binary format. The data bridge resource file contains a series of data extraction run instructions wherein each extraction run is specified as shown here:

```
DATA SET
READER ODBC
SOURCE jdbc:odbc:PlantArchive
OUTPUT .\Data\Model_A_Recent.sdf
SAMPLE 6.9444444444E-4
LOOKBACK 90.0
NAMELIST Signal 1, Signal 2, Signal 3
```

Each entry in the resource file is made on a separate line. The DATA SET keyword is a delimiter that defines the beginning of each new data extraction specification.

The second entry begins with the READER keyword and defines the name of the data reader plug-in used to access the data source. This must the name of a class file located in the application's \PlugIns\reader\subdirectory. In this example, the reader plug-in is ODBC.class. Note that a different reader plug-in can be specified for each data extraction run.

The third entry begins with the SOURCE keyword and defines the name of the network-accessible data source. This could be the file path, database name, or COM/DCOM object name. If a database or other on-line source is used, the source must be registered with the operating system to be accessible. Note that a different data source can be specified for each data extraction run.

The fourth entry begins with the OUTPUT keyword and specifies the output file path name for the data extraction. The application's \Data\directory is the recommended location for the output files. Note that the data bridge will check for the existence of this file and will automatically merge the existing file with newly extracted data to minimize the plant data historian's processing overhead. As an example, consider the case where the data bridge runs daily with a 90-day look-back window. The first time the data bridge runs, the output file does not exist. The data bridge will run and extract the full 90-day period of data to the SDF file. The second time the data bridge runs, it will acquire only the necessary missing date (for one day) to fill in the time period from the current time back to the last time recorded in the previously extracted SDF file. It will then merge the data sets to update the SDF file for the most current 90-day look-back period. Note that a different output file should be specified for each data extraction run.

The fifth entry begins with the SAMPLE keyword and specifies the sampling interval for the data expressed in the native units of the data source. In this example, the SQL Server time units are based on days elapsed since January 1, 1900. Thus a specification of 6.9444444444E–4 days represents a 1-minute sampling interval. Note that a different sampling interval can be specified for each data extraction run.

The sixth entry begins with the LOOKBACK keyword and specifies the look-back window interval for the data expressed in the native units of the data source. In this example, the look-back window is 90 days. Note that a different look-back interval can be specified for each data extraction run.

The seventh entry begins with either the NAMELIST keyword or the NAMETABLE keyword and specifies the computer point names to be extracted in the current run. If the NAMELIST keyword is used, a comma-delimited list of names must be provided after the keyword. The comma-delimited list can extend over multiple lines. If the NAMETABLE keyword is used, the database table name for the name list must be provided after the keyword. Note that a different computer point name list can be specified for each data extraction run.

The computer points extracted will generally be coordinated with the computer points required as inputs to one or more on-line monitoring models. The recommended approach is to extract the computer points for each model into an individual SDF file. In other words, it is recommended that a separate extraction run be made for each model. However, it is also acceptable to place all computer points required by all models into a single extracted file.

As mentioned previously, it might be necessary to register the on-line data source with the operating system. For example, to register an ODBC data source on a Microsoft Windows platform, the ODBC Data Source Administrator is used. On a Windows platform the ODBC Data Source Administrator, "Data Sources (ODBC)," can be run from the Control Panel by clicking the Start button and then Settings, Control Panel. A new data source name (DSN) can be added, assuming that there is an ODBC driver for the database of interest. If there is not an ODBC driver for the database of interest, a driver needs to be installed before running the ODBC Data Source Administrator again. The facility's system administrator should be contacted for details for specific computer, network, and operating system environments.

After registering the data source and configuring the data bridge resource file, the data bridge can be run manually by entering the equivalent of the following on the command line:

```
java -cp .;.\SDF_Bridge.jar SDF_Bridge resource.txt
```

A command file should be configured with the necessary information and used to run the data bridge periodically under an automatic program scheduler. A sample command file named SDF_Bridge.bat is provided with the SureSense software. The Microsoft Windows operating system provides a suitable scheduler; however, any scheduler can be used. Each time the data bridge is run, it will update the SDF file archive for each data set specified in the resource file.

The data bridge manages the middleware interface between the on-line monitoring application and the plant data historian. Experience shows that data extraction from the data historian is the slowest step in a typical on-line monitoring implementation. The process of evaluating the data using the on-line monitoring software is typically much faster. Nonetheless, a 90-day archive of 1-minute interval data contains 129,600 observations of multiple signals. This is slightly more than 1 megabyte of data per signal or about 20 megabytes of data for a typical 20-signal model. On a typical desktop computer, it can take from tens of seconds to several minutes to process this amount of data through a model. For this reason, the on-line monitoring data processing step was also automated in the reference implementation so that users are provided with near instantaneous access to the analysis results.

9.4 On-Line Monitoring System Operation

The software used for the reference implementation can be operated in several modes with several types of user interface. The reference implementation takes advantage of two modes of SureSense operation—the unattended mode and the monitoring mode.

In the unattended mode, the SureSense software is operated as a command line process without a graphical user interface. This is the same SureSense software provided to users. However, operating in unattended mode without the overhead of a graphical user interface makes the processing more efficient. To invoke the unattended mode, simply a run control file name is added as the command line argument when launching the SureSense application. The SureSense command interpreter will open the named run control file and will operate based on the run specifications contained there. In the reference implementation, the unattended mode is used with a program scheduler to produce the daily run results.

In monitoring mode, the SureSense software is operated with a restricted graphical user interface that allows the user to access and review run results but that does not allow modifications to the underlying model or data set specifications. Monitoring mode is the preferred method of deployment in the reference implementation where it is used to evaluate the daily run results.

The software also offers a designer mode, which is more fully described in the *SureSense Diagnostic Monitoring Studio User's Guide* [3]. The designer mode will typically be used to prepare and evaluate models prior to deployment for on-line monitoring. As part of model preparation, it must be specifically configured for on-line operation. The designer mode might also be used to perform detailed engineering analysis of the run results for operability assessment after a fault event has been detected.

9.4.1 Producing Run Results

The SureSense program can be scheduled to run automatically at predetermined intervals and to archive monitoring run results for periodic review by technicians and engineers. This is the most popular approach because few organizations require or desire to dedicate resources for continuous monitoring with immediate notification of results. The periodic on-line monitoring approach is easily configured and is the subject of this section.

The following steps apply to automating periodic on-line monitoring data evaluation:

1. Open the model in designer mode, and configure the data set(s) that will be monitored for on-line operation. It is not necessary to enable all data sets for on-line monitoring. It might be preferable to set up a dedicated data set to be used for on-line monitoring. This step requires designer privileges.

2. Set up the run control files for use with an automated scheduling tool such as the Task Scheduler accessory available in Microsoft Windows NT, 2000, and XP Professional. The run control files are described later in this report.

3. Set up the automated scheduling tool such as the Task Scheduler accessory available in Microsoft Windows NT, 2000, and XP Professional.

4. Review and plot the results using either the designer mode or the monitoring mode user interfaces after the scheduler has run the program.

The SureSense monitoring interface is designed to use automatically stored run results for a model. Run results are based on the analysis of a specific data set. Automatic archiving of the run results for that data set is accomplished by checking the *Create Result Files* checkbox in the SureSense *Edit Data Set* window. Figure 9-3 shows the checkbox location. By setting up the data set to archive run results, any subsequent run of that data set will store the run results for later use. It is also recommended that the *Auto Select* check box in the *Edit Data Set* window be enabled. This will automatically select the first and last available time points for the data set.

It should be noted that a designer must set up the data set to store run results using the designer interface. The modified model project must be saved for the changes to be effective.

Apply the changes to the data set and Save the model to enable on-line monitoring. When this initial setup has been completed, the data set will refresh its dedicated archive with new results each time the Model>Data Set combination is run by a user or by an automated scheduler. It is recommended that this data set be run to initialize the archive by selecting Run from the main system window menu. Select Monitor from the submenu and select the modified data set to run in the Run Director window.

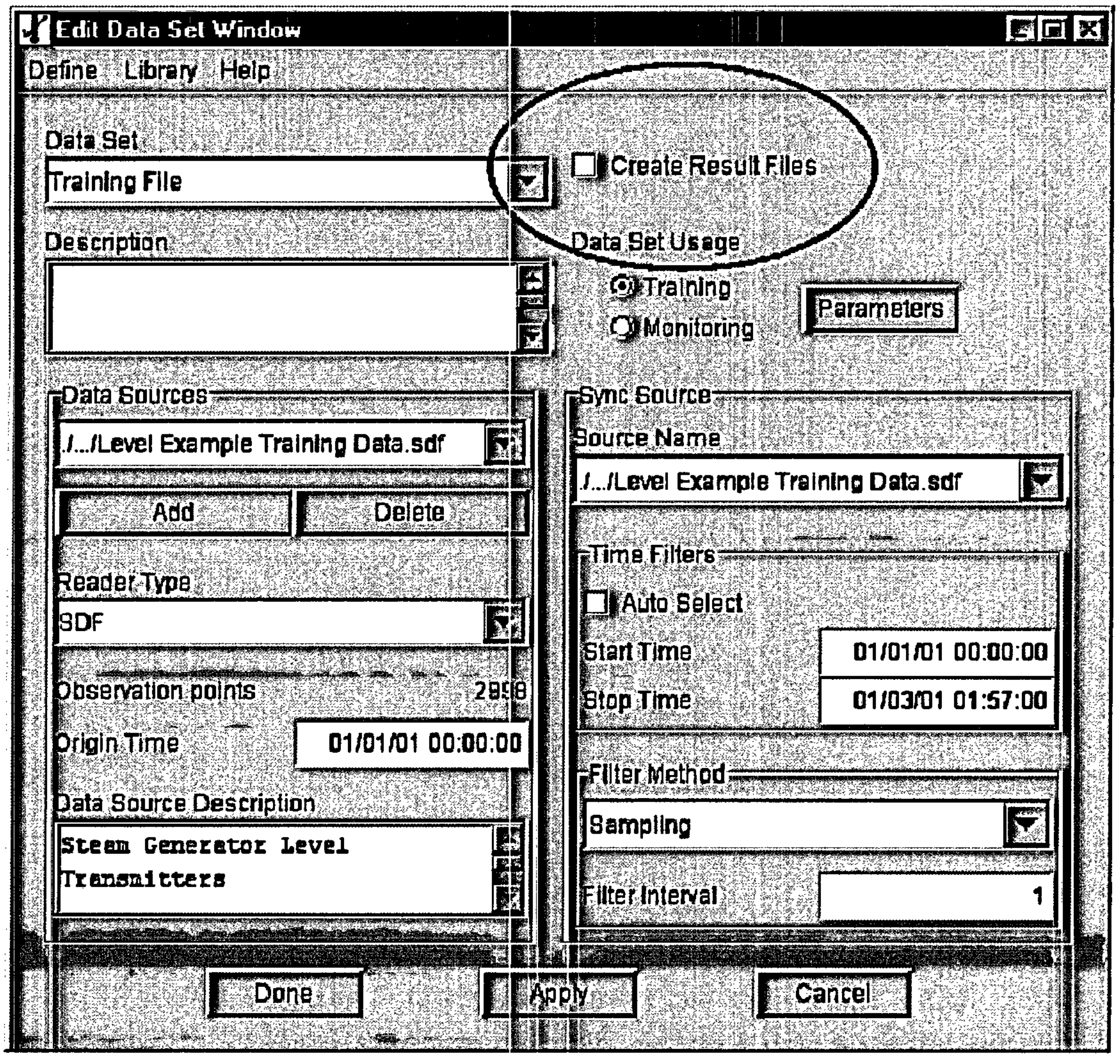


Figure 9-3
Setting Up a Data Set to Store Run Results

The run control files consist of a control file and a resource file. The control file is typically a batch command or .BAT file when running under Microsoft Windows operating systems. The resource file is a simple text file that specifies which models and data sets are to be run when the command file is executed. Any number of resource files can be run from a single command file. However, in this discussion it is assumed that a single resource file lists the models and data sets of interest. Contact the software provider or system administrator for instructions on setting up more sophisticated run control scenarios. Standard Microsoft Windows syntax is used in the following examples, but the concepts are equally applicable to other operating systems.

The command file will typically contain one or more command lines of the following format:

C:\PROGRA~1\JavaSoft\jre\1.3.1_03\bin\java -cp .;.\SDMSv140m.jar SDMS RUNFILE.txt

where

- C:\PROGRA~1\JavaSoft\jre\1.3.1_03\bin\java is the path to the Java virtual machine.
- -cp .;.\SDMSv140m.jar SDMS specifies the SureSense executable (presumes the command file is in the SureSense home directory denoted as .\).
- RUNFILE. txt specifies the path to the resource file (presumes the resource file is also in the SureSense home directory).

A sample command file can be found in the SureSense home directory with the name Unattended.bat. The name *unattended* means that SureSense will run automatically and without a graphical user interface. The presence of a command line argument specifying the resource file name instructs SureSense to run without the user interface.

The resource file is a plain text file and will typically contain multiple lines with the following format:

RUNFILE

USER Scheduler Daily

MODEL "./Projects/Model_A.svm" "Model_A_Recent"

MODEL "./Projects/Model_B.svm" "Model_B_Recent"

where

- *RUNFILE* is a keyword that instructs the SureSense command interpreter that this resource file implements an unattended run.
- *USER* is a keyword that instructs the SureSense command interpreter to expect the next two tokens to contain a user name and a password.
- *Scheduler* is a valid user name for a user with Monitor privileges. Only monitoring runs can be performed in unattended mode.
- *Daily* is a valid password for the monitoring user.
- *MODEL* is a keyword that instructs the SureSense command interpreter to expect the next two tokens to contain a model name and a data set name.
  - *"./Projects/Model_A.svm"* is the path name to a SureSense model file, which is always contained in quotes and can include spaces as necessary for the correct path.
  - *"Model_A_Recent"* is the data set name within the specified SureSense model file, which is always contained in quotes and can include spaces as necessary for the data set.

Any number of MODEL instructions can be listed in the resource file. Each instruction should be listed on a separate line in the file. The resource file should be saved as a plain text file. A sample resource file can be found in the SureSense home directory with the name RUNFILE.txt.

It should be noted that the specified model must be trained and the data set enabled for creating results files to run successfully in unattended mode. The unrun.log file in the SureSense home directory should be reviewed periodically to ensure that the models and data sets are running as expected. The .BAT file should be run manually prior to implementing the scheduler to be sure that its behavior is acceptable.

9.4.2 Evaluating Run Results

Models that have been previously developed are available for monitoring using either the SureSense designer interface or the monitoring interface. The monitor interface is provided for users to view run results without having to use the full designer interface. Figure 9-4 shows the arrangement.

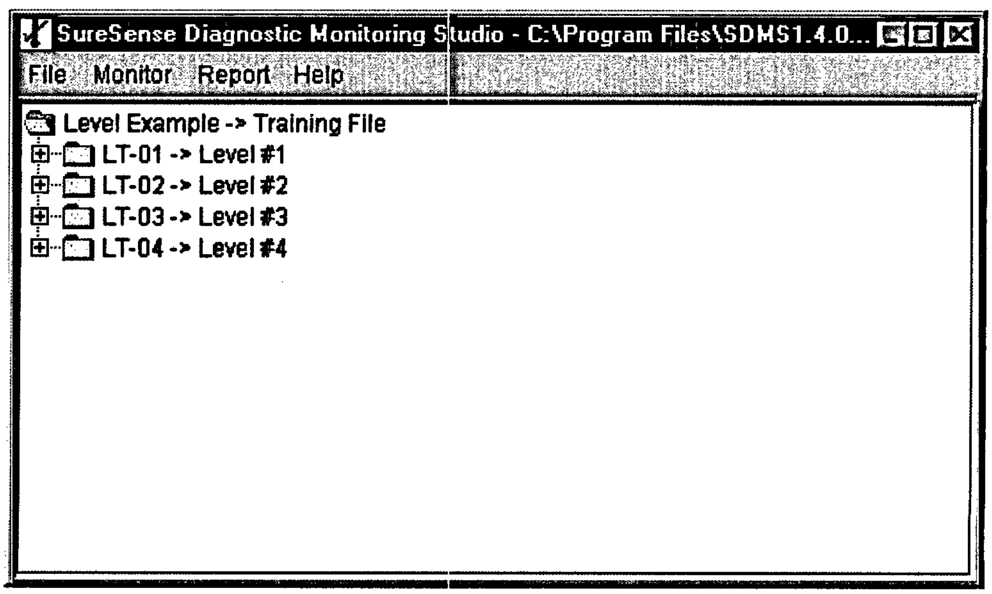


Figure 9-4
SureSense User Interface Options for Monitoring

The monitoring interface has the following desirable features:

- Not all personnel require detailed knowledge regarding how to design and maintain on-line monitoring models. The simpler monitoring interface allows personnel to access and view run results without needing extensive training in how to operate and navigate through the SureSense designer interface.
- The monitor interface is designed to support automated on-line monitoring in which data are collected, the model is updated, and run results are provided automatically at predefined times.

- The monitor interface has been designed to support remote user access. If desired, monitoring activities can be centralized for several facilities (with the monitor interface providing local access to on-line monitoring results).
- The monitoring interface displays results for a single data set within the model. When the model is first selected, SureSense automatically selects a default data set for the display. The preferred default is the first monitoring data set for which automated results file archiving is enabled. If no such data set is found, the first monitoring data set will be the second default choice. Open a different data set for display by selecting *Data Set* from the *Monitor* menu item.

As shown in Figure 9-5, three options are provided for each signal:

- Run information
- Observation estimate plot
- Residual plot

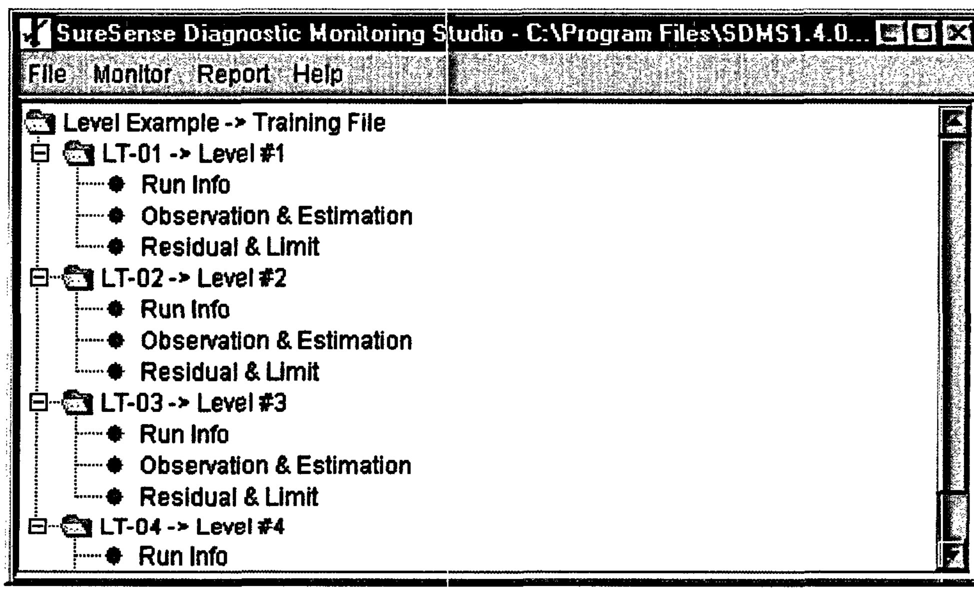


Figure 9-5
Monitor Window Options for Each Signal

Each of these options is obtained by double clicking on the desired item with the right or left button of the mouse. Figure 9-6 shows a typical signal report. Figure 9-7 shows a sample observation estimate plot, and Figure 9-8 shows a sample residual plot.

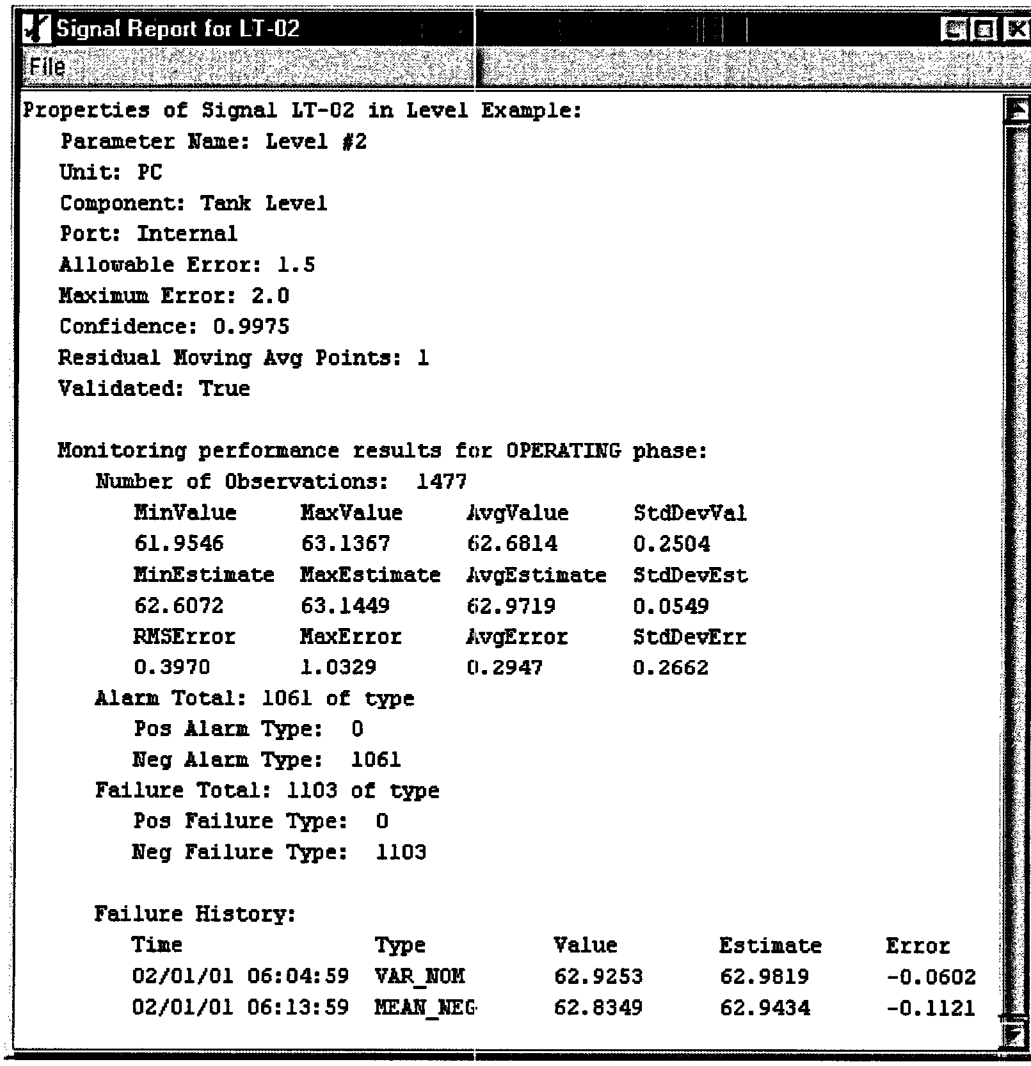


Figure 9-6
Sample Run Information for a Signal

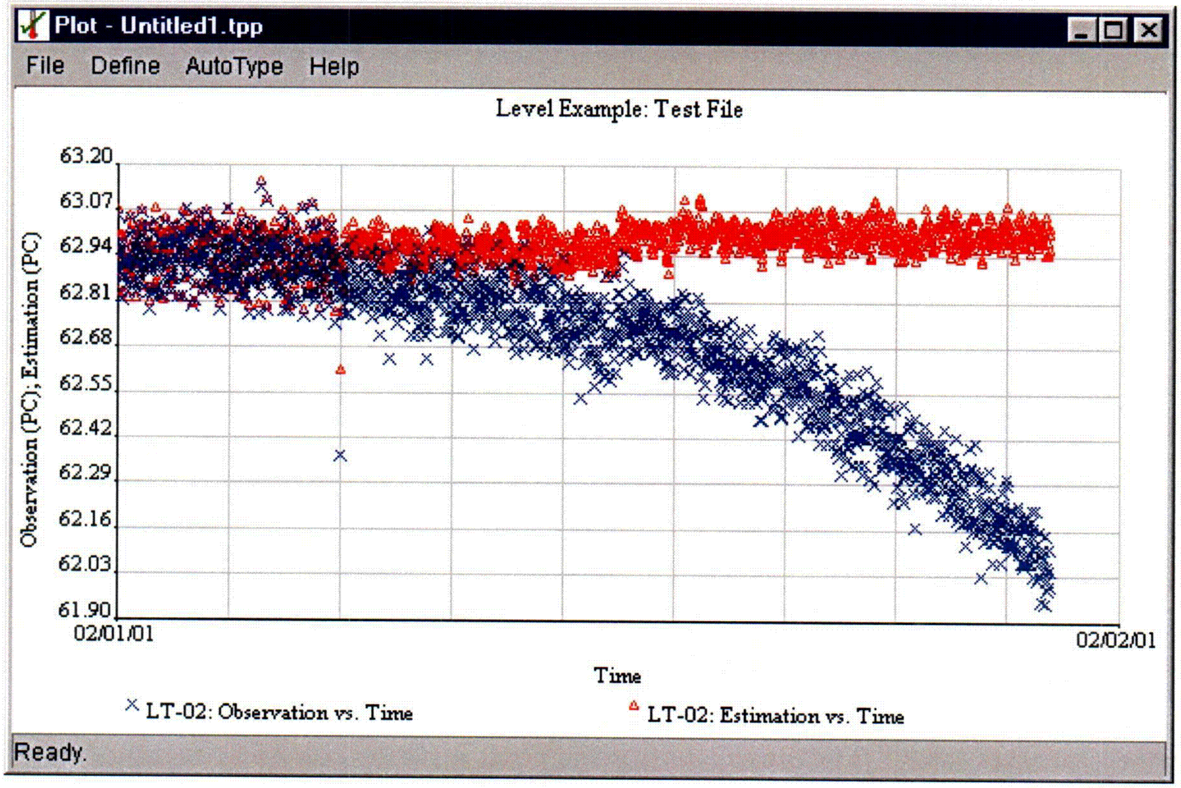


Figure 9-7
Sample Observation Estimate Plot for a Signal

C45

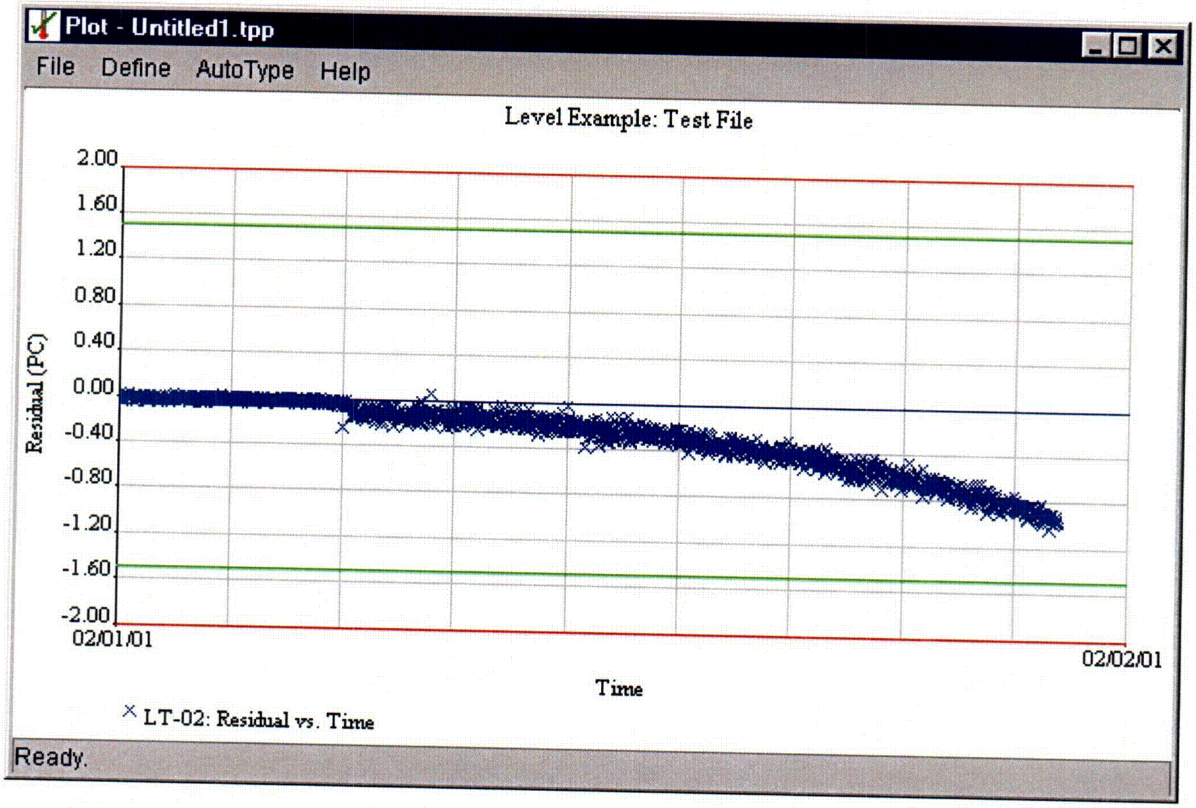


Figure 9-8
Sample Residual Plot for a Signal

These reports and plots will display very quickly if the selected data set is enabled for on-line monitoring. If the selected data set is not enabled for on-line monitoring, the result might still be available. However, the data set will have to be processed first to generate the results information. In this case, a *Data Analysis Required* dialog (similar to the one shown in Figure 9-9) will be displayed, requesting permission to connect to the data sources and run the data set prior to displaying the requested results. If the operator responds *Yes*, the specified data set will run in the background and display the requested information. Depending on the model or data set size, it might take several minutes to complete a run.

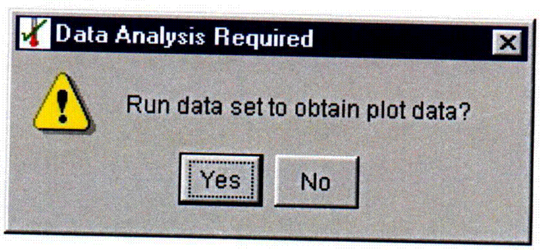


Figure 9-9
Option to Run a Data Set If Not Enabled for On-Line Monitoring

C46

Run summary information is also provided for the entire model. By selecting *Summary* from the Report menu, the Run Summary report is displayed as shown in Figure 9-10. If the selected data set is not enabled for on-line monitoring, the summary might still be available; however, the data set must be processed first to generate the summary information.

It should be noted that a SureSense model cannot be modified, trained, or saved from within the monitor window. A designer must have previously trained and saved the model before it can be used to display or generate monitoring results from within the monitor window. If results are requested from an untrained model, a message similar to the one shown in Figure 9-11 will be displayed.

```
Run Summary Report
File
Monitoring Results for Level Example -> Test File

Result Summary for OPERATING:
Data Points Processed: 1477
Average Processing Time:   1.9973 msec
Normalized RMS Error Percent:  0.1764 %
Single Cycle Alarm Total:  1064 alarms consisting of
     Pos Alarm Type:  1
     Neg Alarm Type:  1063
Failure Decision Total:  1103 failures consisting of
     Pos Decision Type:  0
     Neg Decision Type:  1103

Summary by Signal Parameter:
SignalName  MinValue     MaxValue     AvgValue     StdDevVal
            MinEstimate  MaxEstimate  AvgEstimate  StdDevEst
            RMSError     MaxError     AvgError     StdDevErr
            RMSError%    NumAlarms    NumFailures  Time1stFail
LT-01       62.8702      63.6775      63.4374      0.0797
            62.9672      63.6194      63.4347      0.0705
            0.0217       0.0951       0.0171       0.0133
            0.0342%      2            0
LT-02       61.9546      63.1367      62.6814      0.2504
            62.6072      63.1449      62.9719      0.0549
            0.3970       1.0329       0.2947       0.2662
            0.6305%      1061         1103         02/01/01 06:04:59
LT-03       61.4515      62.0803      61.8914      0.0668
            61.6583      62.0601      61.8908      0.0590
            0.0144       0.2054       0.0107       0.0096
            0.0232%      1            0
LT-04       58.5115      59.0244      58.7707      0.0845
            58.5236      59.0235      58.7679      0.0867
            0.0086       0.0313       0.0068       0.0052
            0.0147%      0            0
```

Figure 9-10
Model Run Summary Report

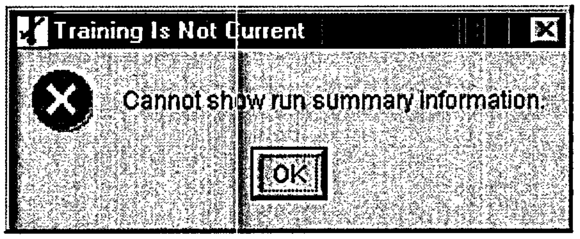


Figure 9-11
Notification That the Selected Model Is Not Currently Trained

9.5 Using the Microsoft Windows Scheduler

Any program scheduler can be used to initiate the command files described in the previous sections. Microsoft Windows provides a scheduler accessory with its Windows NT, 2000, and XP Professional operating systems. The following discussion illustrates the Windows 2000 scheduler. The NT and XP schedulers are nearly identical.

To set up the Windows scheduler, the following steps should be performed:

1. Select *Programs* from the *Start* menu, select *Accessories*, select *System Tools*, and select *Scheduled Tasks*. This will bring up the *Task Scheduler* window, similar to the one shown in Figure 9-12.

2. Double click on the *Add Scheduled Task* icon. This will start the *Scheduled Task Wizard,* similar to the one shown in Figure 9-13.

3. Click *Next* and select *Browse* in the window that is similar to the one shown in Figure 9-14. Browse to the command file (for example, Unattended.bat,), select it, and choose *Open.*

4. Name the scheduled task, and select the frequency for running the task in the window similar to Figure 9-15. Choose *Next.*

5. Enter the start time and date information in the next window similar to Figure 9-16. Choose *Next.*

6. Enter the operating system user name password, if necessary, in the window similar to Figure 9-17. This is *not* the SureSense user name or password but the user's valid login to the Windows operating system. The facility's system administrator should be contacted for this information if necessary. Choose *Next.*

7. Review the information in the final window, similar to Figure 9-18, then choose *Finish.* This task will now appear in the *Task Scheduler* window (Figure 9-12).

This procedure creates a scheduled task that will run SureSense in unattended mode as directed by the Scheduled Task Wizard. The facility's system administrator should be consulted for further details about configuring the Microsoft Windows Task Scheduler.

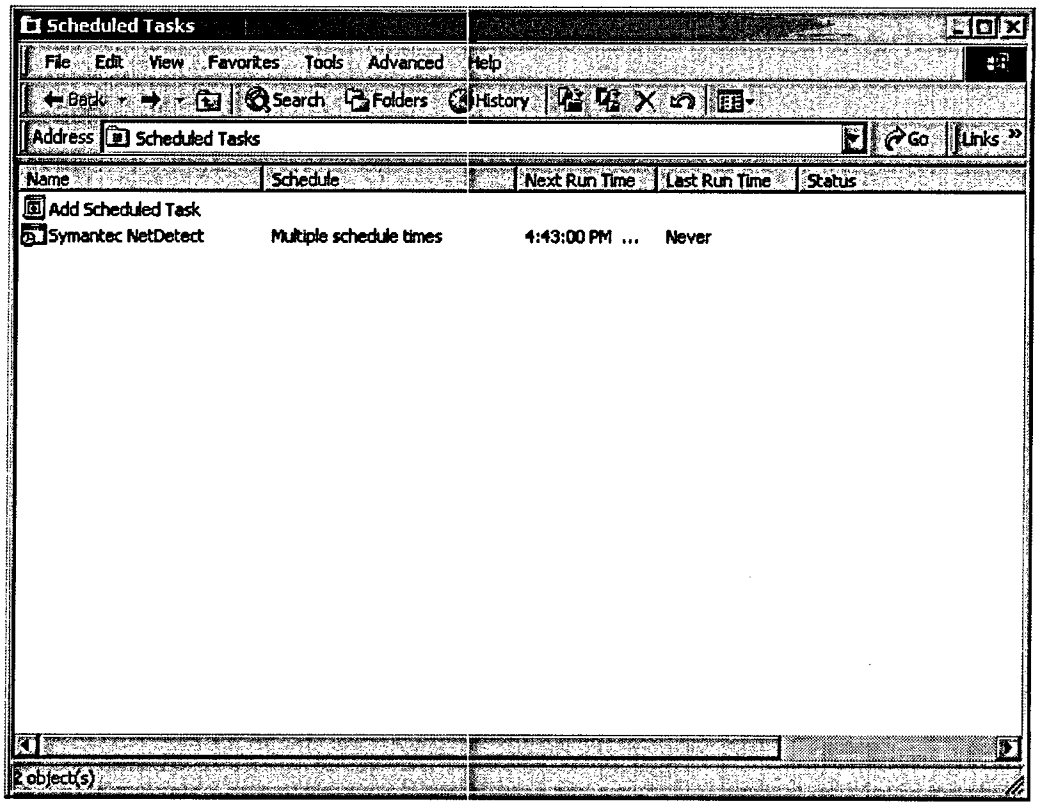


Figure 9-12
Microsoft Windows Task Scheduler

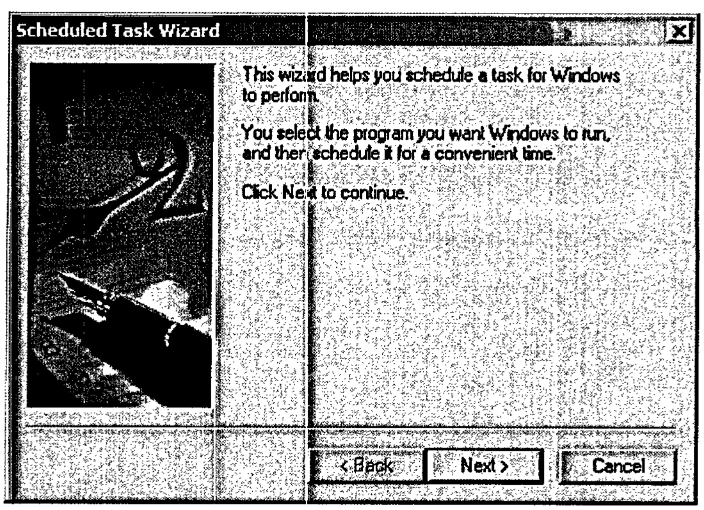


Figure 9-13
Microsoft Windows Scheduled Task Wizard Introduction

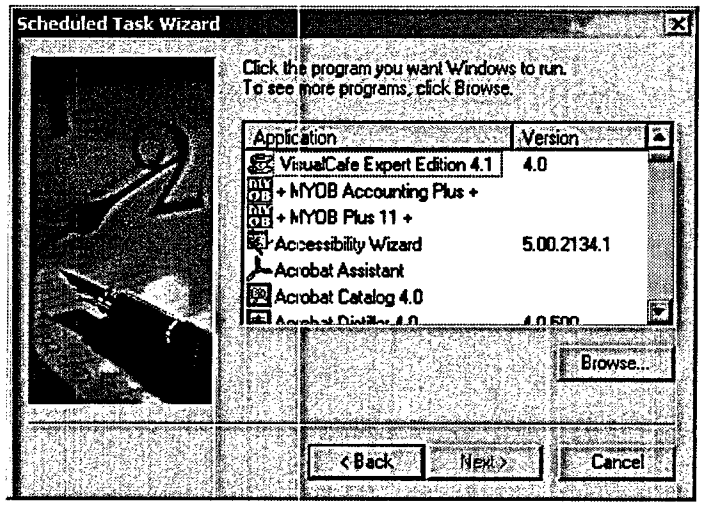


Figure 9-14
Select the SureSense Command File (*.BAT) Using the Browse Button

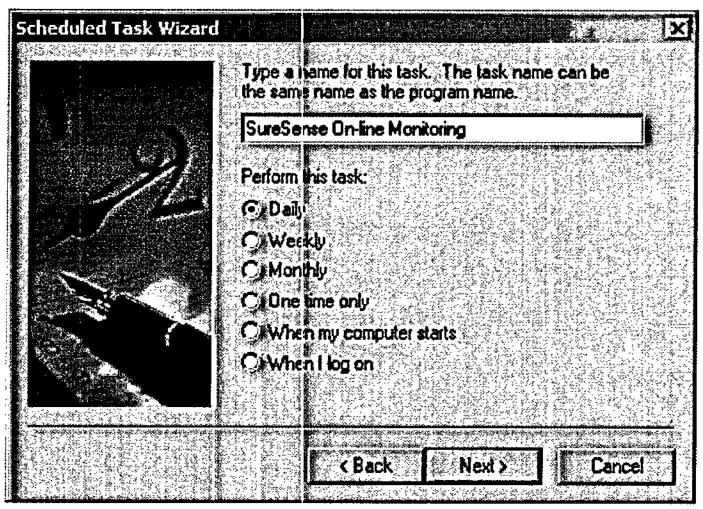


Figure 9-15
Name the Scheduled Task and Select the Run Frequency

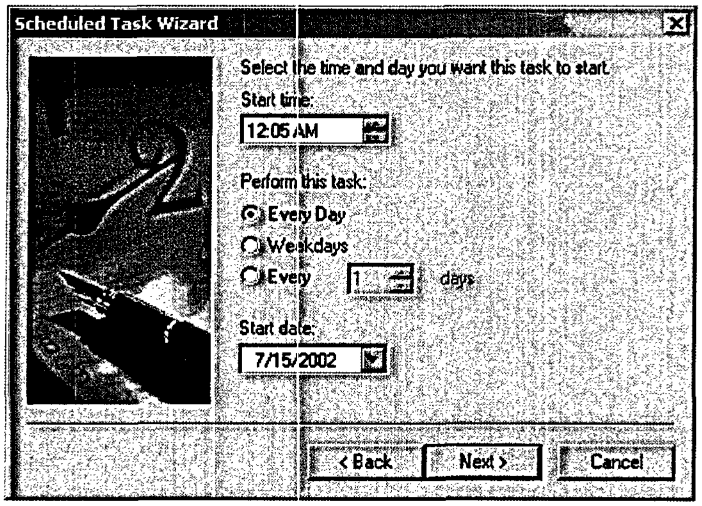


Figure 9-16
Provide Additional Scheduling Details

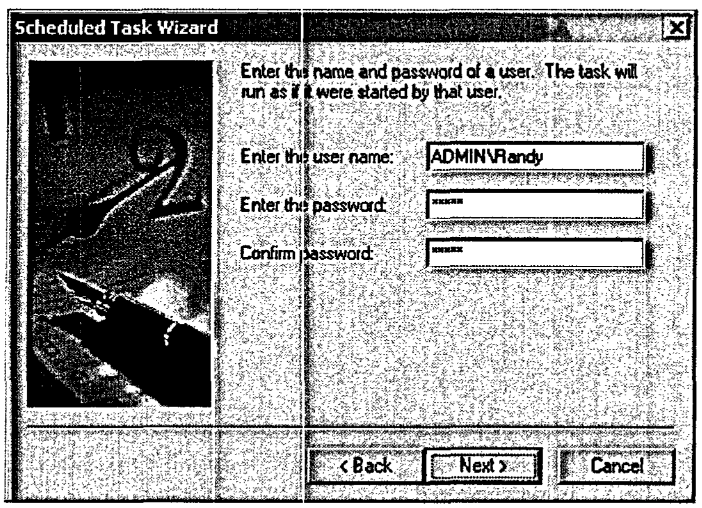


Figure 9-17
Provide Microsoft Windows Login Information

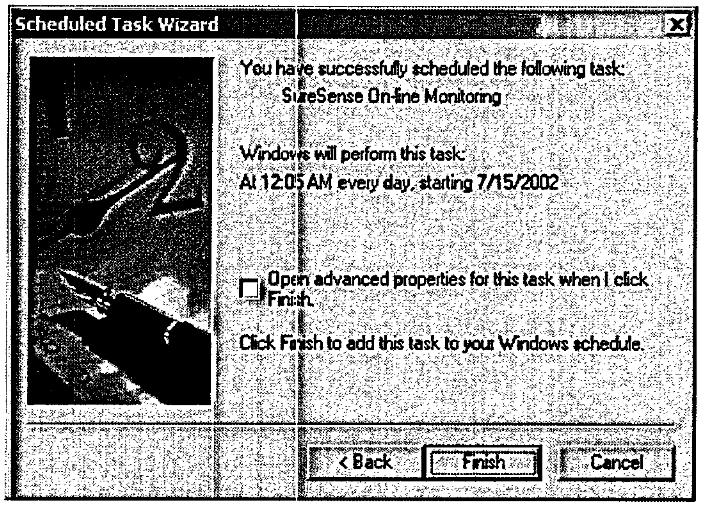


Figure 9-18
Review and Install the Scheduled Task

10
DECLARING THAT A MODEL IS READY FOR USE

Model development is prone to continual refinement and tinkering to optimize performance. In order for the model to be useful, however, it must eventually be placed into service. Section 10 addresses the steps necessary to declare that a model is ready for use, and Table 10-1 provides a short checklist to consider. The following sections discuss each of the items listed in Table 10-1. Some of the recommendations are model specific, and other recommendations apply equally to the entire on-line monitoring system.

Table 10-1
Checklist for Determining That a Model Is Ready for Use

Item	Recommendation	Ready?
Completing the Model		
1.	Confirm the signal selection.	
2.	Confirm the adequacy of training data.	
3.	Verify training adequacy.	
4.	Verify fault-detection settings.	
5.	Declare the model ready.	
Automating Data Acquisition		
6.	Set up ongoing data acquisition.	
7.	Assign responsibility for model testing	
Anticipating Failure Alarms		
8.	Establish an alarm response.	
9.	Determine retraining goals.	

10.1 Completing the Model

This topical report provides detailed modeling guidelines, including recommendations for approaching model development. At some point, each model's behavior will be adequate so that it can be relied on as an on-line monitoring tool. The following activities should be considered as part of the model completion process:

- Confirm the signal selection, Remove uncorrelated signals, and determine if other signals should be added to the model. Removing signals from an existing model is easy. Adding signals is more difficult because historical data will have to be acquired for these signals and the model must be retrained for the new configuration. For this reason, it is always better to evaluate the signal selection carefully before acquiring data.
- Confirm the accuracy of training data. Bad data should not be allowed to remain in training data sets even if model performance appears acceptable.
- Verify training adequacy. Test the model using historical data, and confirm that the system operating space is adequately bounded by the training space. Document any instances in which the training space does not bound transients. Evaluate the settings of the phase determiner and estimator as part of the training adequacy.
- Verify fault-detection settings. Fault detection should not be too sensitive. Occasional alarms should be evenly distributed across all signals in the training data sets.
- After completing the model, declare the model ready.

10.2 Automating Data Acquisition

As the model is placed in service, new data should be tested periodically using the model. Consider the following:

- Set up ongoing data acquisition. Set up the method by which data will be periodically made available to the model. This assumes that the model is operating in batch mode with data files periodically extracted from the plant computer's data storage system.
- Assign responsibility for model testing. Determine how new data will be tested and how the results will be evaluated.

10.3 Anticipating Failure Alarms

Signal (sensor) failures will be periodically identified by the on-line monitoring system. Based on the models developed to date, few of the identified failures will represent actual instrument drift or failure. More likely, the identified failures will represent process system excursions beyond the defined training space. Consider the following:

- Establish alarm response. In many cases, fault detection might be more sensitive than required by the instrumentation. Evaluate failure alarms, and determine if corrective action is required. Corrective action might involve instrument calibration or repair, or it might require changes to the on-line monitoring system model.
- Determine retraining goals. Decide under which conditions to retrain the model rather than accepting the periodic failure alarms.

11
REFERENCES

1. *On-Line Monitoring of Instrument Channel Performance, Volume 2: Model Examples, Algorithm Details, and Reference Information*, EPRI, Palo Alto, CA: 2004. 1003579.
2. *On-Line Monitoring of Instrument Channel Performance, Volume 3: Applications to Nuclear Power Plant Technical Specification Instrumentation,* EPRI, Palo Alto, CA: 2004. 1007930.
3. *SureSense Diagnostic Monitoring Studio User's Guide*, Version 2.0, Expert Microsystems, Inc., Orangevale, CA: 2004.
4. "Application of On-Line Performance Monitoring to Extend Calibration Intervals of Instrument Channel Calibrations Required by the Technical Specifications," *On-Line Monitoring of Instrument Channel Performance.* EPRI Topical Report TR-104965. Safety Evaluation by the Office of Nuclear Reactor Regulation, Project No. 669, U.S. Nuclear Regulatory Commission, July 2000.
5. *On-Line Monitoring of Instrument Channel Performance*, EPRI, Palo Alto, CA: 2000. 1000604.
6. *On-Line Monitoring Cost Benefit Guide*, EPRI, Palo Alto, CA: 2003. 1006777.

A
GLOSSARY

This glossary provides definitions for technical terms used in the report or otherwise applied to on-line monitoring. Abbreviations used in the body of the report are also included in the glossary.

A

accuracy (reference) – In process instrumentation, a number or quantity that defines a limit that error should not exceed when a device is used under specified operating conditions. Error represents the difference between the measured value and the standard or ideal value.

adaptive sequential probability – An inference procedure for determining a signal alarm condition based on the derived probability density function for the training data.

adjustment – The activity of physically adjusting a device to leave it in a state in which its performance characteristics are within acceptable limits.

ANL – Argonne National Laboratory.

ANN – Artificial neural network.

API – Application programming interface. A set of software functions or methods externally callable by another software program.

as-found – The condition in which a channel, or portion of a channel, is found after a period of operation and prior to any calibration.

as-left – The condition in which a channel, or portion of a channel, is left after calibration or surveillance check.

ASP – Adaptive sequential probability.

B

B&W – Babcock and Wilcox.

BART – Bounded angle ratio test.

Bayesian Belief Network – A mathematical method of specifying the probabilistic relationships among events. In the context of on-line monitoring, a belief network is an expression of the probabilistic knowledge of a system and its operation.

Bayesian conditional probability – An inference procedure for determining signal failure based on a preceding number of alarms.

Bayesian sequential probability – An inference procedure for determining a signal alarm condition based on the derived probability density function for the training data.

BCP – Bayesian conditional probability.

Belief Network – See Bayesian Belief Network.

BSP – Bayesian sequential probability.

BWR – Boiling water reactor.

C

calibration – The process of adjustment, as necessary, of the output of a device such that it responds within a specified tolerance to known values of input.

calibration interval – The elapsed time between the initiation or successful completion of calibrations or calibration checks on the same instrument, channel, instrument loop, or other specified system or device.

calibration (time-directed) – The calibration of an instrument at specified time intervals, without regard for the existing calibrated state of the instrument.

channel – An arrangement of components and modules as required to generate a single protective action signal when required by a generating station condition, a control signal, or an indication function.

channel calibration (typical Technical Specification definition) – The adjustment, as necessary, of the channel so that it responds within the required range and accuracy to known input. The channel calibration shall encompass the entire channel, including the required sensor, alarm, interlock, display, and trip functions. The channel calibration might be performed by means of any series of sequential, overlapping calibrations or total channel steps so that the entire channel is calibrated.

channel check – The qualitative assessment, by operator observation, of channel behavior during operation and includes, where possible, comparison of the channel indication to other indications from other redundant channels measuring the same parameter.

confidence interval – An interval that contains the population mean to a given probability.

CMP – Configuration management plan.

CSV – Comma delimited file format.

D

DDD – Design description document.

D/P – Differential pressure.

D-matrix – The matrix of vectors selected by the MSET training process. These vectors represent the model in terms of its recognition of "normal" system behavior. Also, referred to as the training matrix or the process memory matrix.

desired value – A measurement value with no error existing.

deviation – The difference between the parameter estimate and the monitored signal (more commonly referred to as the residual).

DOE – Department of Energy.

domain – The operating states that form the basis for training a model.

drift – An undesired change in output over a period of time, which is unrelated to the input, environment, or load.

E

EDF – Electricité de France.

error – The undesired algebraic difference between a value that results from measurement and a corresponding true value.

ESFAS – Engineered Safeguards Features Actuation System.

estimate – The best estimate of the actual process value; used interchangeably with parameter estimate.

F

field calibration – Performing the activities of surveillance and adjustment using an external reference source.

flat-topping – The tendency for an estimate to follow a signal disturbance to the upper or lower limit of the data used for training and remain at that limit.

FRD – Functional requirements document.

G

Gaussian probability – Normal probability.

GUI – Graphical user interface.

I

ICMP – Instrument Calibration and Monitoring Program.

IMC – Instrument monitoring and calibration.

Initial training – See Training.

instrument channel – An arrangement of components and modules as required to generate a single protective action or indication signal that is required by a generating station condition. A channel loses its identity where single protective action signals are combined.

L

linear – A straight-line relationship between one variable and another. When used to describe the output of an instrument, it means that the output is proportional to the input.

loop – See channel.

M

M&TE – Measuring (or measurement) and test equipment.

margin – An additional allowance added to the instrument channel uncertainty to allow for unknown uncertainty components. The addition of margin moves the set point further away from the analytical limit or nominal process limits.

mean – The average value of a random sample or population. For *n* measurements of x_i, where i ranges from 1 to *n*, the mean is given by.

$$\bar{x} = \frac{\sum x_i}{n}$$

median – The value of the middle number in an ordered set of numbers. Half the numbers have values that are greater than the median and half have values that are less than the median. If the data set has an even number, the median is the average of the two middle numbers.

MinMax – An algorithm that extracts vectors that bound a vector space defined by training data. (See vector ordering).

model – The group of signals that have been collected for an analysis.

module – Any assembly of interconnecting components that constitutes an identifiable device, instrument, or piece of equipment. A module can be removed as a unit and replaced with a spare. It has definable performance characteristics that permit it to be tested as a unit. A module can be a card, a drawout circuit breaker, or another subassembly of a larger device provided it meets the requirements of this definition.

monitoring – The activity of evaluating instrument channel performance to determine that it is performing within acceptable performance limits.

MSES – Multivariate State Estimation Studio.

MSET – Multivariate State Estimation Technique.

N

NEPO – Nuclear Energy Plant Optimization.

noise – An unwanted component of a signal or variable. It causes a fluctuation in a signal that tends to obscure its information content.

nonlinear – A relationship between two or more variables that cannot be described as a straight line. When used to describe the output of an instrument, it means that the output is of a different magnitude than the input.

normal distribution – The density function of the normal random variable X, with mean μ and variance σ^2 is

$$n(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

normalized – A term indicating that the data values for a group of disparate signals have been modified so that all signals have approximately equal weight in an analysis.

NRC – Nuclear Regulatory Commission.

O

ODBC – Open database connectivity.

OLM – On-line monitoring.

OLMS – On-line monitoring system.

on-line monitoring – An automated method of monitoring instrument performance and assessing instrument calibration while the plant is operating.

operating space – A defined region of operation.

operating state – A defined region of operation, often established by power level or equipment lineup. Often used interchangeably with operating space.

overfitting – The tendency for the estimate to follow a signal disturbance.

P

parameter estimate – The best estimate of the actual process value.

pattern recognition – The ability of a system to match large amounts of input information simultaneously and generate a categorical or generalized output.

PDF – Probability density function.

PEANO – Process Evaluation and Analysis by Neural Operators.

phase – A defined region of operation used to separate the model into submodels.

phase determiner – A software function used to determine which phase applies to a set of observations.

population – The totality of the observations with which we are concerned.

probability density function – An expression of the distribution of probability for a continuous function. The probability contained within a given interval can vary from 0 to 1 and is expressed by:

$$P(a < X < b) = \int_b^a f(x)dx$$

PWR – Pressurized water reactor.

R

random – Describing a variable whose value at a particular future instant cannot be predicted exactly but can only be estimated by a probability distribution function.

range – The difference between the minimum and maximum value in a set of data.

RCS – Reactor coolant system.

reference accuracy – A number or quantity that defines the limit that errors will not exceed when the device is used under reference operating conditions.

residual – The difference between the observation and the corresponding estimate for that observation. Also known as the residual error.

retraining – Any change made to the set of data originally selected as representative of system normal and expected behavior.

retraining for operating space – Retraining caused by modifying the data used for training. If the pool of data made available for training is modified, the vector selection for the training matrix will likely change, even if the model settings are unchanged.

retraining for settings – Retraining caused by adjusting model settings. Changing estimator settings, changing the number of signals, adjusting data limit filters, or modifying phase-determiner definitions for validation will require retraining and optimizes model performance for a given set of training data.

RPS – Reactor Protection System.

RTD – Resistance temperature detector.

S

S/G – Steam generator.

safety limit – A limit on an important process variable that is necessary to reasonably protect the integrity of physical barriers that guard against the uncontrolled release of radioactivity.

sample – A subset of a population.

SDF – Signal data file.

SDM – Signal disturbance magnitude.

SDMS – SureSense Diagnostic Monitoring System.

sensor – The portion of a channel that responds to changes in a plant variable or condition and converts the measured process variable into an electric or pneumatic signal.

set point – See trip set point.

signal – The output data from a channel.

signal conditioning – One or more modules that perform further signal conversion, buffering, isolation, or mathematical operations on the signal as needed.

span – The region for which a device is calibrated and verified to be operable.

spillover – The tendency for the estimate of one signal to follow a disturbance in a second highly correlated signal.

SPRT – Sequential probability ratio test (used with MSET to determine if a process is operating normally or abnormally).

SQL – Structured query language.

staggered test basis – Testing of one of the systems, subsystems, channels, or other designated components during the interval specified by the surveillance frequency, so that all systems, subsystems, channels, or other designated components are tested during *n* surveillance frequency intervals, where *n* is the total number of systems, subsystems, channels, or other designated components in the associated function.

standard deviation (population) – A measure of how widely values are dispersed from the population mean and is given by

$$\sigma = \sqrt{\frac{n\sum x^2 - (\sum x)^2}{n^2}}$$

standard deviation (sample) – A measure of how widely values are dispersed from the sample mean and is given by

$$s = \sqrt{\frac{n\sum x^2 - (\sum x)^2}{n(n-1)}}$$

state space – The operating states that form the basis for training a model.

steady-state – A characteristic of a condition, such as a value, rate, periodicity, or amplitude, exhibiting only a negligible change over an arbitrary long period of time.

SureSense – A commercially supported implementation of the MSET software originally developed by Argonne National Laboratory.

surveillance – The activity of checking a device to determine if it is operating within acceptable limits.

surveillance interval – The elapsed time between the initiation or successful completion of a surveillance or surveillance check on the same instrument, channel, instrument loop, or other specified system or device.

T

test interval – See calibration interval.

time-directed calibration – See calibration (time-directed).

training – For a pattern recognition system such as MSET, the selected vectors that describe the operating state for normal and expected behavior.

training matrix – The matrix of vectors selected by the MSET training process. These vectors represent the model in terms of its recognition of "normal" system behavior. Also, referred to as the D-matrix or the process memory matrix.

trip set point – A predetermined value at which a bistable device changes state to indicate that the quantity under surveillance has reached the selected value.

U

uncertainty – The amount to which an instrument channel's output is in doubt (or the allowance made therefore) due to possible errors either random or systematic that have not been corrected for. The uncertainty is generally identified within a probability and confidence level.

V

V&V – Verification and validation.

variance (population) – A measure of how widely values are dispersed from the population mean and is given by

$$\sigma^2 = \frac{n\sum x^2 - (\sum x)^2}{n^2}$$

variance (sample) – A measure of how widely values are dispersed from the sample mean and is given by

$$s^2 = \frac{n\sum x^2 - (\sum x)^2}{n(n-1)}$$

vector (of signals) – All data observations for a single time step. For example, if the data are contained in a spreadsheet, a single row of data is a vector.

vector ordering – An algorithm that adds representative vectors from the inner regions of a vector space to produce a more accurate process model. Vector ordering is differentiated from the MinMax algorithm in that it describes the interior of a space whereas the MinMax algorithm bounds the vector space.

vector pattern recognizer – An MSET estimation technique. The algorithm compares two vectors and defines their similarity as a function of the inverse of the Euclidean distance between the vectors.

vector similarity evaluation technique – An MSET estimation technique. The algorithm defines the similarity of two vectors as a function of the ratio between the Euclidean distance between the vectors and the sum of the root sum square (RSS) values of the vectors.

VPR – Vector Pattern Recognizer.

VSET – Vector Similarity Evaluation Technique.

Export Control Restrictions

Access to and use of EPRI Intellectual Property is granted with the specific understanding and requirement that responsibility for ensuring full compliance with all applicable U.S. and foreign export laws and regulations is being undertaken by you and your company. This includes an obligation to ensure that any individual receiving access hereunder who is not a U.S. citizen or permanent U.S. resident is permitted access under applicable U.S. and foreign export laws and regulations. In the event you are uncertain whether you or your company may lawfully obtain access to this EPRI Intellectual Property, you acknowledge that it is your obligation to consult with your company's legal counsel to determine whether this access is lawful. Although EPRI may make available on a case by case basis an informal assessment of the applicable U.S. export classification for specific EPRI Intellectual Property, you and your company acknowledge that this assessment is solely for informational purposes and not for reliance purposes. You and your company acknowledge that it is still the obligation of you and your company to make your own assessment of the applicable U.S. export classification and ensure compliance accordingly. You and your company understand and acknowledge your obligations to make a prompt report to EPRI and the appropriate authorities regarding any access to or use of EPRI Intellectual Property hereunder that may be in violation of applicable U.S. or foreign export laws or regulations.

About EPRI

EPRI creates science and technology solutions for the global energy and energy services industry. U.S. electric utilities established the Electric Power Research Institute in 1973 as a nonprofit research consortium for the benefit of utility members, their customers, and society. Now known simply as EPRI, the company provides a wide range of innovative products and services to more than 1000 energy-related organizations in 40 countries. EPRI's multidisciplinary team of scientists and engineers draws on a worldwide network of technical and business expertise to help solve today's toughest energy and environmental problems.

EPRI. Electrify the World

Program:

Nuclear Power

1003361

Printed on recycled paper in the United States of America

EPRI • 3412 Hillview Avenue, Palo Alto, California 94304 • PO Box 10412, Palo Alto, California 94303 • USA
800.313.3774 • 650.855.2121 • askepri@epri.com • www.epri.com