

# **Chapter 1**

## **Introduction to Process Supervision**

### **1.1 Process Supervision**

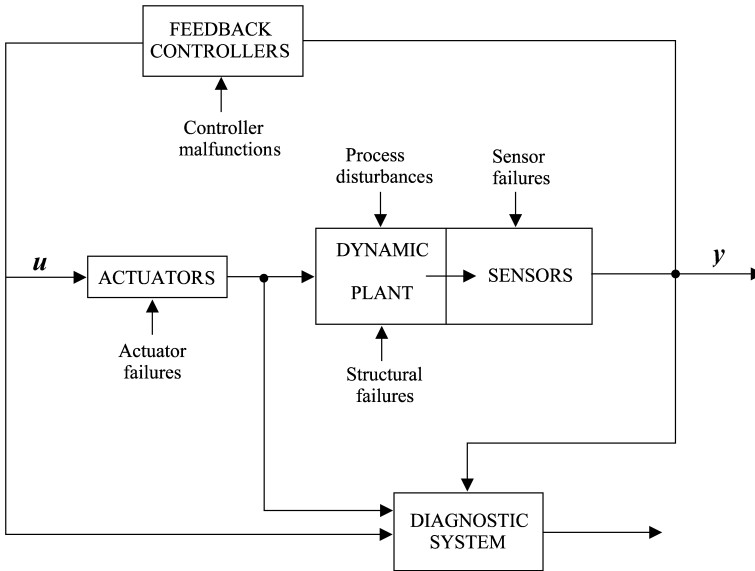
Modern process engineering plants use complex equipment, control laws, and myriads of operation sequences and instrumentation. In complex and safety critical systems, such as chemical plants, nuclear power plants and airplanes, total failure of a component can be extremely hazardous; the Chernobyl and Bhopal tragedies are indicators of the extent of such damage. Even partial failures or malfunctions of process components or instrumentation increase the operating costs of the plant. Gross failures, such as accidents, are definitely more serious. These failures could be due to faulty design, faulty operation and human errors, to which sabotage and acts of terrorism may be added these days.

Therefore, modern process engineering requires prompt detection and classification of process anomalies, which would minimize the overall repair time by assisting field and plant technicians in the diagnosis of system degradation. Moreover, it is essential to detect incipient faults and to locate the deteriorating or the deteriorated components as early as possible. In other words, some form of prediction of each device's condition and likelihood of its failure is necessary to undertake timely and appropriate maintenance work. The health (condition) of the system must be monitored at each and every instant and the purpose of the diagnostic system is to detect quickly any fault which could seriously degrade the performance of the system. A specialized field called condition monitoring and diagnosis is devoted to determine the life span of operating devices and helps in scheduling their maintenance. However, some devices tend to fail abruptly without showing any symptoms or warnings. Moreover, complex control laws often compensate for and conceal the faults. That is why real time process supervision assumes a central role in timely fault detection and isolation of complex systems. Real-time detection and diagnosis of faults while the plant is still operating in a controllable region can help avoid abnormal event progression and reduce productivity loss.

Automated Fault Detection and Isolation (FDI) procedures implemented in the supervision platform are meant to ensure safe operation of industrial processes, be-

cause faults in a process will often cause an undesired sequence of events and the consequences could be damaging to the plant, the personnel and the environment. Over the years, automated production systems have aided operators in controlling the process in order to improve the final production quality, the safety and the efficiency of industrial units. Recently, another challenge has appeared in the form of automation of the supervision aimed at providing assistance to the human operator in the management of alarming situations to increase reliability, availability and safety of the process. Therefore, the design of complex dynamic systems has to take into account additional functionality such as maintenance, management and supervision facilities.

Thus, FDI algorithms are central components of an Abnormal Event Management (AEM) system, which deals with the timely detection, diagnosis and correction of abnormal conditions in a process. Figure 1.1 depicts the components of a general fault diagnosis framework where a controlled system and the different sources of failures in it are indicated.



**Fig. 1.1** A general diagnostic framework

Process supervision is performed by means of a set of tools and methods, which ensure safe process operation in the normal situation as well as in the presence of failures or undesired disturbances. The primary activities concerned with process supervision are Fault Detection and Isolation (FDI level), diagnosis (determining the root cause of the fault) and the decision making to accommodate the fault, whenever possible (fault accommodation level). Fault accommodation is done through

reconfiguration, *i.e.* by using alternate standby devices called hardware redundancies, and/or Fault Tolerant Control (FTC), *i.e.* by changing the control laws and the associated control problem to maintain the desired output. The presence of a fault is detected at the monitoring level, which determines whether the process is in normal operation or not. Other tools associated with diagnosis and further high level tasks are executed only after detection of an abnormal process state.

### ***1.1.1 Basic Diagnosis Tasks***

The diagnosis problem can be divided into four distinct phases:

- Fault detection/monitoring: this concerns the activities to determine if the process dynamics has deviated beyond an acceptable limit from its normal operation model. If an unacceptable process behavior is detected then an alarm state is declared.
- Fault isolation: if there is an alarm then the objective of this stage is to locate one or more faulty component(s), called fault candidate(s). This stage further concerns filtering of the fault candidates such that they are kept to a minimum and also filtering out false alarms (due to process and measurement uncertainties). One or more decision procedures are used in parallel (multiple decision procedures, whose output are matched), series (a multi-stage procedure, where output of one decision procedure is filtered by another and so on), or hybrid (a network of parallel and series decision procedures).
- Fault quantification/identification: the aim of this stage is to determine the severity of the fault and its type. If the fault is due to a parametric change and it is not too severe, then the approximate values of the parameter has to be estimated. If the fault is due to a complete failure (hard failure), which changes the system morphology or its associated model structure, then the new system model has to be identified. There are several system identification and parameter estimation techniques available in control theory, which are readily used for fault quantification. If the results of fault quantification indicate that a component is so severely damaged that it cannot be further used even with acceptable degraded functionality, then the faulty component is removed from a database of available equipment maintained by the AEM. Note that as and when a previously faulty component is repaired or replaced with a new one, and it is ready for use, it has to be manually added to the equipment availability database.
- Fault accommodation: the decision regarding whether a fault can be accommodated or not is taken at this stage. If a fault cannot be accommodated, then the AEM system must determine the further sequence of events (*e.g.* to enter a controlled shut-down phase); if a fault can be accommodated, then the AEM system must determine how and to what extent it can be done. The degree of fault severity is used at this stage to determine whether to use process reconfiguration, FTC or manual supervision. Furthermore, development of control laws for FTC

requires an explicit model and approximate values of model parameters with limited uncertainty, both of which have to be available from the fault identification stage.

### 1.1.2 Fault, Failure and Safety

Fault is defined as a departure from an acceptable range of an observed variable or a calculated parameter associated with a process [265]. The underlying cause of a fault is called the basic event or the root cause. Unstructured uncertainties, *e.g.* process noise and measurement noise, are mainly faults that are not modeled *a priori* and are outside the scope of fault diagnosis. Process noise refers to the mismatch between the actual process and the predictions from model equations, whereas measurement noise refers to high frequency additive components in the sensor measurements. Usually, unstructured uncertainties are filtered out (*e.g.* Kalman filter) or decoupled (*e.g.* robust control) before they are used in a diagnosis system, or they are accommodated within the FDI and alarm filtering stages.

Different malfunctions can be classified into three types as follows:

- Gross parameter changes or parametric faults arise from disturbances to the process due to exogenous (independent) variables, whose dynamics are not provided with that of the process. An example of such a malfunction is a change in the heat transfer coefficient due to fouling in a heat exchanger. Here, the parameter for heat transfer coefficient is an exogenous variable. Another example of such a malfunction is a blockage in a pipeline resulting in modification of the overall discharge coefficient. This type of fault is also termed a multiplicative fault, because in the differential equation of the system, functions of system parameters are multiplied with the state variables.
- Structural changes refer to hard failures in equipment, which may change the model structure, *i.e.* disturb the information flow between various variables. The diagnostic system would require remodeling, *i.e.* the model equations have to be restructured in order to describe the present faulty state of the process. A failure of a controller, a broken transmission system, a stuck valve or a broken pipe may be categorized under this class of failure. Note that a model for a hybrid process is usually an ensemble of several models, each corresponding to a particular structure of the process. The system transits from one model structure to another depending on the evolution of its variables; these transitions and the corresponding conditions are generally represented in the form of a state-flow-graph. On the contrary, structural changes due to a fault results in new process dynamics, which may be known (*i.e.* a model exists for this kind of fault) or unknown (*i.e.* no corresponding model is immediately available for the new process structure).
- Malfunctioning of sensors and actuators may result due to a constant bias, intermittent disturbance, saturation, or an out of the range failure. Some feedback

signals, which are essential for the control of a plant, are provided by the measurement instruments. A failure in one of those instruments could cause the instability of the plant and seriously degrade the plant performance unless the failure is detected promptly and remedial actions are taken in time. These faults are also referred to as additive faults. In general, additive faults refer to undesirable change of inputs and outputs, and undesirable (often unknown) inputs, *e.g.* environmental effects.

A component's malfunction is called a fault when it is possible to take appropriate measures to recover from it without replacing the component, *i.e.* the fault can be accommodated through fault tolerant control. When the malfunctioning of the component is too severe, or there is a structural change to the system, we usually refer to it as a failure. A failure can be accommodated only through system reconfiguration, *i.e.* replacing the malfunctioning device by other devices rendering similar function if appropriate redundant hardware is/are available. Classification of different malfunctions as faults and failures depends on the specific process under consideration and its instrumentation architecture.

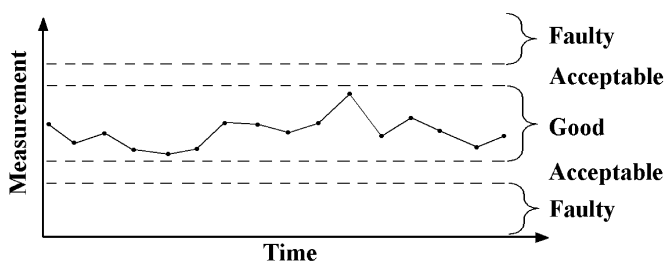
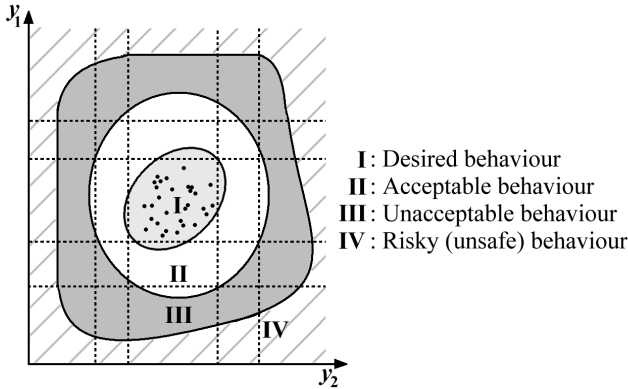


Fig. 1.2 One-dimensional limits on process variables

Because a fault is defined as departure from the acceptable behavior, it is necessary to fix the threshold for acceptable behavior. This threshold may put directly on process measurements as shown in Figure 1.2, where points marked by ‘.’ are process measurements acquired at fixed or variable intervals. Such a scheme would require separate thresholds on separate measurements and it becomes difficult to establish correlations between them. This is why a multi-dimensional threshold is more suitable.

A schematic representation of a cross-section of a multi-dimensional threshold and the various operating regimes of the process are shown in Figure 1.3, in which  $y_1$  and  $y_2$  are two independent measurements. The dotted lines in Figure 1.3 show the corresponding one-dimensional thresholds. During normal process behavior, the outputs (shown by ‘.’s) stay in zone-I. Faults and failures shift the outputs outside zone-I. If the outputs are in zone-II, then the fault can be accommodated to bring the process behavior inside zone-I. However, if the process behavior enters zone-III, then the fault cannot be accommodated. Zone-IV represents the unsafe process oper-



**Fig. 1.3** Threshold zones in a cross-section of the space spanned by the process variables

ation. The aim of the safety system is to continuously monitor the process behavior such that it does not drift into zone-IV from zone-III.

Controller actions often keep the output within desired ranges even when one or more faults are present in a process; *i.e.* controllers can mask some fault effects.

Therefore, the phase-space defining various zones of process behavior must include the controller outputs and process inputs. In such a multi-dimensional phase-space, cross-sections in two-dimension can be examined to determine the process state *vis-a-vis* various operating zones. Often, various variables in this phase-space (inputs and outputs) are correlated among themselves. Therefore, the dimension of the phase-space can be reduced by selection of a set of basis vectors called principal components. Principal Component Analysis (PCA) is a statistical approach to FDI, in which deviation of various principal components (a projection or map of inputs and outputs) are analyzed *vis-a-vis* various fixed or time varying multi-dimensional thresholds (zones in Figure 1.3).

Furthermore, occurrence of faults may be classified into four different types [16, 107]:

- Abrupt, *i.e.* all of a sudden a normally operating system starts behaving abnormally
- Progressive, *i.e.* a component's behavior is gradually drifting away from its normal behavior
- Intermittent, *i.e.* a component behaves abnormally for a small duration of time and this abnormality is repeated at uncorrelated time intervals
- Incipient, *i.e.* a component's behavior is normal, but is continuously in the borderline between zones I and II in Figure 1.3

## 1.2 Diagnostic System

### 1.2.1 Specification of Diagnostic Systems

A common set of requirements or standards that a diagnostic system should possess [16] may be identified as follows:

- Quick detection and diagnosis: the diagnostic system should respond quickly in detecting and diagnosing process malfunctions. However, designing a system for quick response attenuates noise sensitivity and hence some sort of trade off is usually made to achieve tolerable performance.
- Isolability: the diagnostic system should be able to differentiate between various failures. Note that fault isolability depends on availability of various redundant information. Thus, there is a trade off between desired degree of isolability and the cost incurred in providing too much instrumentation. Furthermore, if an abnormality is detected, but cannot be isolated or classified, then the diagnostic system must be able to classify the new fault and identify this type of fault on further occurrence.
- Robustness: the diagnostic system should be robust to various disturbances, noises and uncertainties. Robustness needs often compromise with process performance.
- Adaptability: process operating conditions rarely remain the same over the passage of time due to normal wear and tear, environmental degradations, changing climatic conditions, and periodic maintenance work, *etc.* Therefore, the diagnostic system should be able to adapt to newer situations and operating conditions; *i.e.* in other words the diagnostic algorithm should not be hard-coded and it should be flexible.
- Operator assistance: a diagnostic system should not only provide the list of faults and their place of occurrence, but it should also assist the operator in explaining the origin of the fault, *i.e.* the root cause, and the sequence of the fault propagation from the root cause to the presently detected malfunction. The diagnostic system must be able to do a qualitative reasoning, *i.e.* hierarchically organize different cause and effect relationships, and convey them to the operator in simple language. This means the diagnostic system should clearly state why certain fault hypothesis has been proposed and why other fault hypothesis are not proposed [265]. A quantitative structured FDI approach is good at selection of appropriate fault hypothesis while rejecting some others, but the operators must be trained to interpret the results obtained from such structural analysis.
- Resource intensiveness: a diagnostic system should have low modeling requirements and low computational and storage requirements. Furthermore, it should be easy to implement and be adaptable to various situations.

As of now, there is no single method which satisfies all of the above-mentioned desirable characteristics. Therefore, recent research focuses its attention on development of hybrid diagnostic systems, where a diagnostic system comprises of several

complimentary tools and the root cause analysis is done by using a voting scheme. European project CHEM [1] was one such attempt, under which several toolboxes were developed by using different diagnostic approaches and they were then used simultaneously to diagnose reliably faults in several industrial test cases.

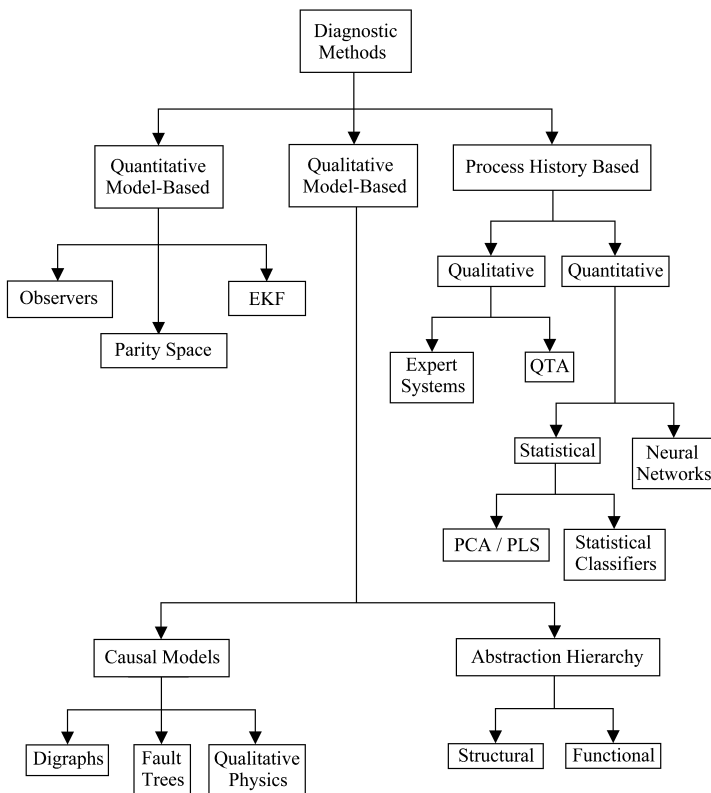
### 1.2.2 Classification of Diagnostic Systems

The first level in FDI is the fault detection step, wherein the objective is to decide whether the system is in normal operating condition or not and, if not, to generate the appropriate alarms. *A priori* knowledge representation forms the core of this step, which receives the process measurements as inputs and produces a set of alarms as outputs.

The second level in FDI deals with alarm interpretation, *i.e.* deciding the actual fault and what its characteristics are (occurrence time, fault size, consequences, *etc.*). The input to this step is a set of alarms while the output is the isolated fault with its characteristic. Fault characterization often relies on explicit availability of fault models as *a priori* knowledge to the diagnostic system. Fault characterization and quantification is required to determine the process state and to determine whether the fault can be accommodated such that the process is able to carry on its mission.

Diagnostic search strategy is usually a function of the knowledge representation scheme, which in turn is influenced by the kind of *a priori* knowledge available. The basic *a priori* knowledge that is needed for fault diagnosis is the set of failures and the relationship between the observations or symptoms and the failures, *i.e.* the knowledge representation forms a set of causal or inferential relationships. A diagnostic system may have them explicitly or it may be inferred from some source of domain knowledge. The *a priori* domain knowledge may be developed from a fundamental understanding of the process using first-principles knowledge. Such knowledge is referred to as deep, causal or model-based knowledge. On the other hand, it may be gleaned from past experience with the process. This knowledge is referred to as shallow, compiled, evidential or process history-based knowledge. The model-based *a priori* knowledge can be broadly classified as qualitative or quantitative. The model is usually developed based on some fundamental understanding of the physics of the process. In quantitative models this understanding is expressed in terms of mathematical functional relationships between the inputs and outputs of the system. On the other hand, in qualitative models, these relationships are expressed in terms of qualitative functions centered around different units in a process [266]. In contrast to the model-based approaches where *a priori* knowledge about the model (either quantitative or qualitative) of the process is assumed, in process history based methods only the availability of large amount of historical process data is assumed. There are different ways in which this data can be transformed and presented as *a priori* knowledge to a diagnostic system [76, 139, 267]. This is known as the feature extraction process from the process history data, and is done to facilitate later diagnosis. This extraction process can be quantitative or qualita-





**Fig. 1.4** Classification of diagnostic algorithms

tive feature extraction. In quantitative feature extraction one can perform either a statistical or non-statistical feature extraction.

The classification of diagnostic systems is shown in Figure 1.4. There is always an overlap between the various approaches. For instance, it is clear that all models need data for estimating some of the parameters in the model and all the methods based on process data need to extract some form of a model to perform fault diagnosis. Different approaches for the FDI procedures have been developed, depending on the kind of knowledge used to describe the process model. Each system or process has its own complexities, which need to be described by following a specific methodology. Venkatasubramaniam *et al.* [265–267] have classified various fault detection and diagnosis methods into quantitative model based methods, qualitative model and search based methods and process history based methods. Further sub-classifications and detailed reviews on each approach are given therein. Differentiation of quantitative, qualitative and process history methods, in the opinion of the authors of [265–267] and authors of this book, provides a classification in terms of the manner in which these methods approach the problem of fault diagnosis.

The quantitative approach relies on advance information processing techniques such as state and parameter estimation and adaptive filtering. The qualitative approach makes use of causal analysis (cause and effect/antecedents and consequences relationships), which links individual component malfunctions expressed in qualitative form with deviations in measured values. This approach can be used when precise numerical models are not available, especially in the design stage. We broadly classify FDI methods into two groups, namely, model based and non-model based.

Model based FDI methods require a mathematical model representing the behavior of the system. This mathematical model can be statistical or analytical. In this book, mainly analytical model based FDI methods are considered.

Modeling is an important and difficult step because of the complexity in representing the behavior of the monitored system along with its control equipment. Note that modeling errors seriously degrade performance of the FDI system. A model for FDI should neither be too complex (*i.e.* include minor or secondary dynamics) nor be too simple (*i.e.* exclude some essential dynamics). Therefore, there needs to be some trade-off to create a reduced order process model, which would reliably replicate essential process dynamics under given operating conditions.

Bond Graph modeling [20, 39, 46, 59, 127, 128, 172, 173, 199, 256, 257], which is a unified multi-energy domain representation and is also suitable for developing models of process engineering systems [258] involving various energetic couplings and control equipment, is used in this book. Note that bond graph modeling has attracted attention in the recent past for its application in various qualitative and quantitative FDI approaches [65, 72, 135, 150, 171, 189–192, 226, 233, 250, 284].

An important property of bond graph models relates to power conservation in a so-called junction structure, which neither consumes nor produces power. Therefore, when a modeler constructs a model from one viewpoint, *e.g.* dynamics (effort balance) or kinematics (flow balance), the other set is automatically satisfied by the constraints in a bond graph. This is why bond graph modeling is so useful in creating models of physical systems involving passive interactions, especially when different energy domains are interfaced and parts of the model can be modeled easily in some energy domains using effort constraints while other parts belonging to different energy domains may be modeled using flow constraints.

Online fault detection requires sensors, which capture the change in states of the system. This means that faults can be detected only when they occur in the observable subspace of the system's model. Moreover, to accommodate those faults, that subspace must be controllable. Therefore, controllability and observability properties assume central roles in the design of supervision systems. Controllability and observability properties of a system are elegantly derived from bond graph models.

## 1.3 Organization of the Book

This book presents the latest developments made in the field of bond graph modeling and its applications to process supervision. Fault detection and isolation of both linear and non-linear systems are considered.

There have been various recent developments in the field of control engineering by using various graphical models. This book utilizes those developments in the field of control engineering to develop FDI and fault accommodation algorithms. It is shown that the use of bond graphs helps to integrate many of the recently developed advances made in the field of control engineering into development of complex supervision systems. Special emphasis is given in this book to physical model based control and supervision.

This book follows a pedagogic approach to introduce various bond graph model based process supervision methods. The next four chapters are intended to develop a basic understanding of the subject. They may be part of an undergraduate course and therefore we have provided some unsolved problems at the end of these chapters. The rest of the chapters are on advanced topics and applications, which are more suitable for postgraduate level courses and as reference material for research.

The concept of bond graph modeling and the notion of causality are introduced in the second chapter. Emphasis is given to modeling of thermo-fluid and other process engineering systems.

The third chapter is devoted to introduce certain concepts in classical and modern control. We specifically introduce the principles of physical model based control and demonstrate the advantages offered by it during the design stage in development of complex control systems. Structural analysis and the notions of bicausality in bond graph models are introduced as tools for this purpose.

Fault detection by using qualitative reasoning methods is introduced in the fourth chapter. Therein, the importance of model order reduction is discussed followed by a few model order reduction schemes. The reduced order bond graph models are then used for fault detection and diagnosis through various qualitative analysis methods.

The fifth chapter forms the core of this book. In this chapter, the concepts of quantitative model based fault detection and isolation are introduced, which include observer based, parity relation based and frequency domain approaches. Thereafter, the analytical redundancy based fault detection and structural fault isolation are introduced. The analytical redundancy relations are first derived by following a classical approach and then the algorithm for their derivation from bond graph models is presented. Some examples are given to illustrate various steps involved in deriving the redundancy relations and to show how they can be used obtain residuals (fault indicators) for fault detection and isolation.

An industrial example is considered in the sixth chapter. The theory developed in the fifth chapter is used in this chapter to develop an actual supervision system. Practical results from an actual process are shown to illustrate the applicability of the developed approach. Moreover, the implementation issues (communication protocols, data storage, handling of multiple diagnosis tools, *etc.*) are also discussed.

The methodologies developed in the fifth chapter suffer from some inadequacies, which are particularly exposed when supervision of not so well instrumented processes is considered. Therefore, the concepts of the so-called diagnostic bond graphs and bicausal bond graphs are introduced in the seventh chapter. These two developments allow for direct use of bond graph models in online process supervision because they solve the fault detection and isolation problem in the numerical domain. Moreover, the problem of sensor placement for fault isolation is solved in the seventh chapter with the aid of bicausal bond graphs.

In the eighth chapter we deal with fault accommodation through reconfiguration. Management of process resources and selection of proper operating mode is discussed in this chapter. Operating modes are constructed by qualitatively synthesizing various functionalities offered by various healthy and redundant devices. Functional models, represented by external models, are integrated with bond graph models to add a quantitative/deterministic layer to operating mode management. Moreover, redundant hardware is determined through structural analysis and their availability in good health is determined from fault detection algorithms.

Sometimes it may not be possible to differentiate between two or more fault effects by using a purely structural approach. In this case, one way of fault disambiguation from a set of fault candidates is to use a bank of models and to simulate them with some fault hypotheses so that their outputs can be compared with the process outputs to isolate the actual fault. Such a way of diagnosis is developed in the ninth chapter by using bond graph models.

All the chapters leading up to the tenth chapter consider single fault hypothesis, *i.e.* how to detect and isolate a single fault occurring in a process. In a more general framework, more than one fault may occur in a process. While it is often possible to detect those faults (if the fault effects do not cancel each other), it is generally impossible to decouple the fault effects. One way of isolating multiple process faults is to estimate the process parameters by using the process response. This is formally called identification and parameter estimation based fault diagnosis. We introduce the basic parameter estimation techniques in the tenth chapter and show that the use of analytical redundancy relations for parameter estimation in a least-square-optimization problem is a better approach due to its speed and simplicity of implementation. Moreover, residual sensitivities may be obtained by representing the diagnostic bond graph of the process in its corresponding sensitivity bond graph form and it is shown to aid in quicker convergence of the optimization problem.

In the eleventh and final chapter, passive and active fault tolerant control problems are discussed. Bicausal bond graphs are used in this chapter for parameter estimation and system inversion. Bond graph model based implicit and explicit system inversion schemes are used to construct the input sequences, which are further tested against actuator capacities before being used for active fault tolerant control of the plant. As a passive approach, a bond graph model based robust overwhelming controller is designed in this chapter to control process variables in the presence of parametric uncertainties.



<http://www.springer.com/978-1-84800-158-9>

Model-based Process Supervision

A Bond Graph Approach

Samantaray, A.K.; Ould Bouamama, B.

2008, XX, 474 p., Hardcover

ISBN: 978-1-84800-158-9