1

Introduction

For more than three decades, the growing demand for safety, reliability, maintainability, and survivability in technical systems has created significant research interest in fault detection and diagnosis (FDD). Such efforts have led to the development of many FDD techniques. For a general exposure to the subject, the reader is directed to [1]–[5].

1.1 Overview

In the literature, fault detection and isolation or fault detection and identification are often used interchangeably and abbreviated as "FDI". To be precise and avoid further confusion, this book adopts the term "FDI" to stand for "fault detection and isolation"; "FDD" is used when the fault identification function is added to FDI. In FTCS designs, fault identification is important; therefore FDD is mainly used throughout this book to highlight the requirement of fault identification.

On a parallel path, research into reconfigurable fault-tolerant control systems has increased progressively since the initial research on restructurable control and self-repairing flight control systems began in the early 1980s (see [6]–[10]). More recently, fault-tolerant control has attracted more and more attention in both industry and academic communities due to increased demands for safety, high system performance, productivity and operating efficiency in wider engineering applications, not limited to traditional safety-critical systems. Several review or survey papers on FTCS have appeared since the 1990s including [11]–[16].

Fault tolerance is no longer limited to high-end systems and consumer products such as automobiles. However it is increasingly dependent on microelectronic and mechatronic systems, on-board communication networks, and software, thus requiring new techniques for achieving fault tolerance. Even though individual research on FTCS has been carried out extensively, systematic concepts, design methods, and even terminology are still not yet standardized. Recently, efforts have been made to unify some terminology [17]. In addition, for historical reasons and because of the complexity of the problem, most of the research on FDD and reconfigurable control (RC) has been treated as two separate fields. More specifically, most of the FDI techniques have been developed as a diagnostic or monitoring tool, rather

Analysis and Synthesis of Fault-Tolerant Control Systems, First Edition. Magdi S. Mahmoud and Yuanqing Xia.

than as an integral part of FTCS. As a result, some FDD methods may not satisfy the need of controller reconfiguration. On the other hand, most of the research on reconfigurable control is carried out assuming the availability of a perfect FDD. Little attention has been paid to analysis and design with the overall system structure and interaction between FDD and RC.

For example, the following questions are posed:

- From the viewpoint of RC design what are the needs and requirements for FDD?
- What information can be provided by existing FDD techniques for overall FTCS designs?
- How can we analyze systematically the interaction between FDD and RC?
- How can we design FDD and RC in an integrated manner for online and real-time applications?

Many other challenging issues still remain open for further research and development. One of the motivations of this book is to provide an overview of developments in FTCS and to address some challenging problems to attract the attention of future research.

1.2 Basic Concepts of Faults

The terminology used in this book is fairly standard. Below, some basic definitions of faults, failure, disturbances and uncertainties, fault detection, fault isolation, fault identification, and fault diagnosis are given. The interested reader is referred to [18, 19, 20] for more detailed explanation of the above mentioned terminology.

A "fault" is an unpermitted deviation of at least one characteristic property or parameter of a system from the acceptable (standard condition). The closely related term "failure" is regarded as a permanent interruption of a system's ability to perform a required function under specified operating conditions. *Failure* is used for the complete breakdown of a system, while *fault* is used to indicate a deviation from the normal characteristics. As far as detection is concerned, both faults and failures can be treated alike. Moreover, a fault can be treated as an external input or as a parameter deviation which changes the system characteristics. Similar to faults, "disturbances", "uncertainties", and "noises" can also be treated as external inputs. In fault detection and isolation (FDI) terminology, they are termed as "unknown inputs". Unlike faults, these unknown inputs are uncontrolled, unavoidable and are present during normal operation. The effect of the unknown inputs can be incorporated into the controller design and a process can perform well even in the presence of them. Faults, on the other hand, have very severe effects on the process and should be detected.

The process of *fault diagnosis* is referred to as the determination of the size, location, time of detection and type of fault in the process. Based on its performance, a fault diagnosis system (FDS) is regarded as a fault detection (FD), fault detection and isolation (FDI) or fault detection, isolation and analysis (FDIA) system [18]. An FD system is therefore the process of determining the fault in the process and its time of occurrence. An FDI system determines in addition the kind and location of the fault. Similarly, an FDIA, together with detection and isolation, also aims to determine the size and time behavior of the fault. It is worth noting that the existence conditions for fault isolation are more stringent than for fault detection, and even more so in the case of fault identification. Consequently, it is difficult to isolate or identify faults in most situations.

A fault detection system should ideally meet some general requirements. The most important desirable features are:

- early detection of faults (incipient and abrupt)
- successful detection of actuator, component, and sensor faults
- robustness against unknown inputs (external disturbances, measurement noises, and model uncertainties)
- differentiation of faults from unknown inputs so that false alarms are avoided
- less use of online computation so that it can be integrated into large-scale systems easily.

Besides the above important attributes, the design procedure of an FD scheme should be as simple as possible.

1.3 Classification of Fault Detection Methods

There exist a number of techniques used for fault detection (FD) in technical processes or dynamical systems. In this section, we present the widely accepted classification of these techniques.

1.3.1 Hardware redundancy based fault detection

The essence of this scheme is replication of the process component using identical hardware components. Figure 1.1 shows a schematic description of the hardware redundancy. Information about the fault is extracted if there is any deviation of the output of the process component from its redundant pair. Good reliability and the ability to isolate faults are the main advantages of this scheme. The major problems encountered with this scheme are the extra components, increased maintenance cost and additional space required to accommodate the redundant equipment. Thus, its use is limited to a number of key applications, for example, nuclear power plants and flight-control systems [18, 21].

1.3.2 Plausibility test

Figure 1.2 shows a schematic depiction of the plausibility test. The basic idea of this technique is to evaluate the measured process variable with regard to credible, convincing values and

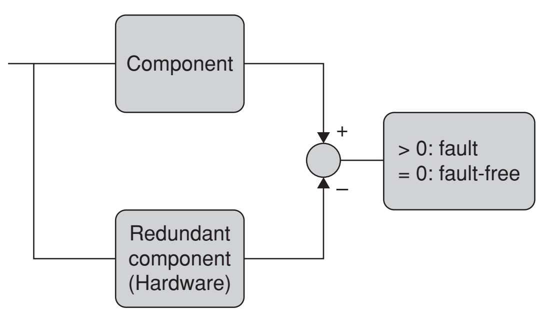

Figure 1.1 Hardware redundancy scheme

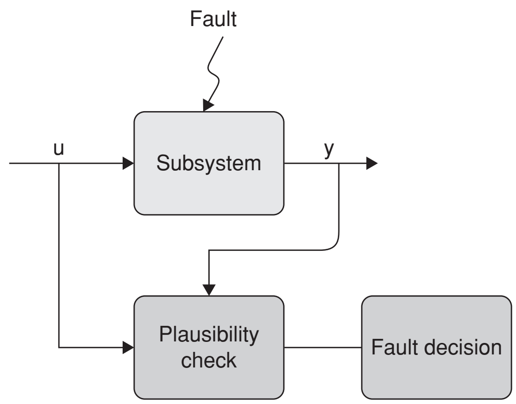

Figure 1.2 Plausibility test scheme

their mutual compatibility. On the assumption that a fault leads to the loss of plausibility, the presence of a fault in a certain variable can be determined using the plausibility check. It can be performed by simple rules with binary logic. The plausibility test is also a kind of limit checking but with a wider tolerance. This test can be viewed as a first step to model-based FD methods. However, it has limited efficacy for detecting faults in a complex process [18, 19].

1.3.3 Signal-based fault diagnosis

Figure 1.3 shows a conceptual depiction of the signal-based FD technique. The central idea of this scheme is to extract the fault information from the process signals. For this purpose, some signal properties (symptoms) are analyzed. These symptoms are generally divided into the time domain characteristics and the frequency domain characteristics of the process signal.

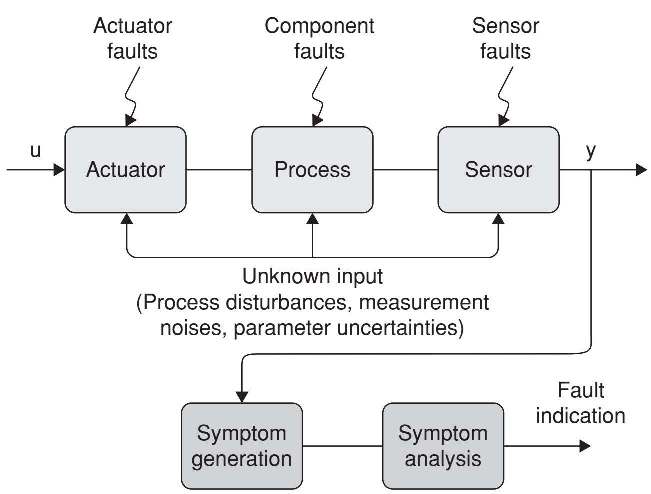

Figure 1.3 Signal-based FD scheme

The time domain characteristics comprise magnitude, mean (arithmetic or quadratic), limit values, trends, statistical moments of the amplitude distributions etc.; the frequency domain characteristics include spectral power densities and frequency spectral lines. Signal-based FD is used under steady-state operation of the process. The efficiency of this scheme is limited when the process is operating in a wide range due to the possible variation of input signals [18].

1.3.4 Model-based fault detection

The intuitive idea of the model-based FD technique is to replace the hardware redundancy by a process model which is implemented in software. The process model runs in parallel with the process itself and is driven by the same process inputs. In this way, the process behavior can be reconstructed online. Analogous to hardware redundancy, this technique is called "software redundancy" or "analytical redundancy" [18]. It is well-known that model-based FD techniques are more powerful than signal-based FD schemes [22, 23] because they use more information about the process.

In a typical model-based FD scheme, there are two stages: residual generation and residual evaluation. In residual generation, the "residual signal" is generated by comparing the process outputs with their estimates. The residual signal carries information about the faults. Since the residual signal, in a real process, is affected by the faults, disturbances, and measurement noises simultaneously, it is required to process the residual signal further to obtain possible information about faults. This is done in the residual evaluation stage.

It is widely accepted that a process model represents the qualitative and quantitative behavior of the process and can be obtained by utilizing well-established techniques from system modeling. The quantitative or analytical model of the process can be represented by a set of differential or difference equations while the qualitative model is expressed in terms of qualitative functions centered around different units in the process. The qualitative models are also known as "knowledge-based models", which include neural networks, petri nets, expert systems, fuzzy logic etc. [22, 23] Based on these arguments, model-based FD schemes can be divided into two classes: knowledge-based and analytical.

Knowledge-based FD techniques are useful where the precise model is not available or is very hard to obtain, for example, large-scale chemical processes and nuclear reactors. An extensive study of knowledge-based FD methods can be found in [22, 24, 25, 26, 27]. Analytical model-based FD techniques, on the other hand, make use of analytical models for the purpose of residual generation. The analytical techniques can be broadly classified as:

- Parity space FD
- Observer-based FD
- Parameter-identification-based FD.

The rest of this section describes these approaches.

1.3.4.1 Parity space approach

Figure 1.4 shows a conceptual diagram of the parity space approach to residual generation. The parity space approach makes use of a parity check on the consistency of the parity equation.

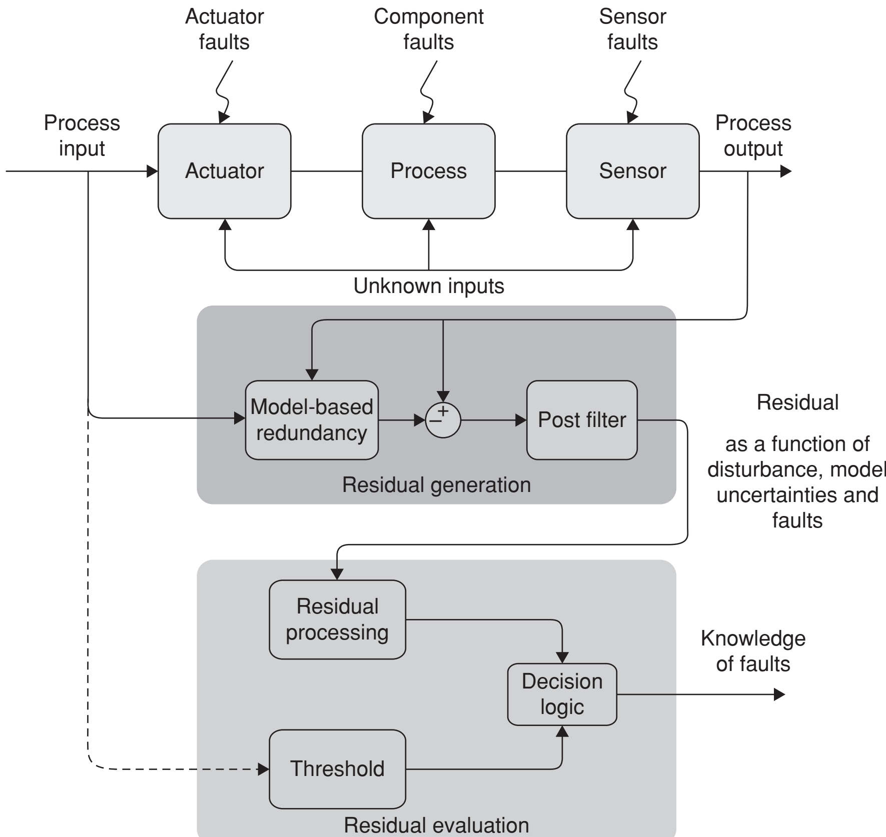

Figure 1.4 Parity space approach

A set of properly modified system equations (also called "parity relations") is derived based on measured signals from the process. These parity relations decouple the residuals from the system states and from each other. This enhances the ability to detect faults. Inconsistency in the parity relations indicates the presence of a fault. In [28], the parity relations were derived based on the state-space model of the system; later, they were derived using the system transfer function [29]–[32].

As mentioned in [18, 23], there exists a close relationship between the parity space approach and the observer-based approach. An extensive study on parity space D is presented in [18], where it is been shown that there exists a one-to-one mapping between the design parameters of observer and parity space based residual generation. Thus, given a set of parity relations, a diagnostic observer can be designed and *vice versa*.

1.3.4.2 Observer-based approach

The observer-based technique, see Figure 1.5, is one of the most commonly applied model-based schemes for detection of faults in a system. In this scheme, the residual signal is obtained

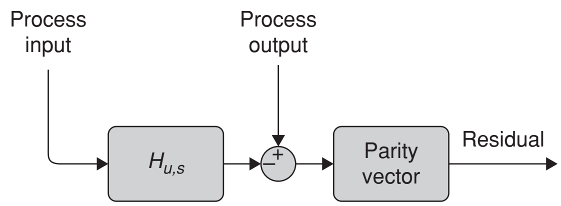

Figure 1.5 Observer-based residual generation

by comparing the process outputs with their estimates. It is worth noting that observers are mainly used by the control community in order to estimate the unmeasured states in the process, while the FDI community use them for diagnostic purposes. The existence conditions for diagnostic observers are more relaxed than for a state observer, however one particular class of diagnostic observer (the fault detection filter (FDF)) can be used for state estimation as well as diagnostic purposes.

1.3.4.3 Parameter identification approach

The parameter identification approach, see Figure 1.6, is also an important FDI technique [22, 33, 34]. In this approach, fault detection is performed based on online parameter estimation. Information about the fault can be extracted by comparing the estimated parameter with the nominal process parameter. Any discrepancy between the two gives an indication of fault. The advantages of this scheme are as follows:

- Several parameters can be estimated with less input and output from the process [19].
- It yields the size of the discrepancy, which is useful for fault analysis [22].

The disadvantage is that an excitation signal is necessary in order to estimate the parameter, which may cause problems in the case of processes running at stationary operating point. Further, the determination of a physical parameter from its mathematical model may not, in

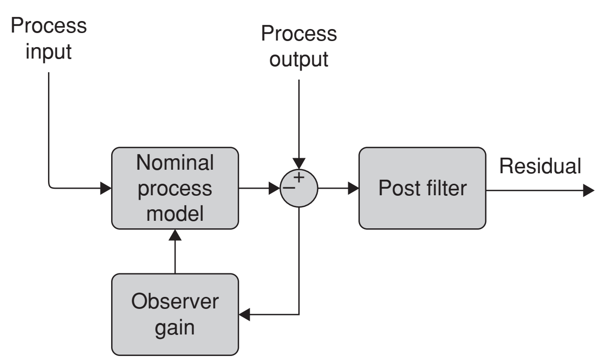

Figure 1.6 Parameter identification scheme

general, give a unique result and is only feasible if the system order is low [22]. There are several parameter estimation techniques available in the literature, among them are the least square (LS) method, the recursive least square(RLS) method, the extended least square (ELS) method etc.

1.4 Types of Fault-Tolerant Control System

Generally speaking, FTCS can be classified into two types: passive fault-tolerant control systems (PFTCS) and active fault-tolerant control systems (AFTCS). In PFTCS, controllers are fixed and are designed to be robust against a class of presumed faults [8]. This approach needs neither FDD schemes nor controller reconfiguration, but it has limited fault-tolerant capabilities. Discussions of PFTCS are beyond the scope of this book and interested readers are referred to [35, 36] and the references therein for recent developments. In the literature, PFTCS are also known as "reliable control systems" or "control systems with integrity".

In contrast to PFTCS, AFTCS react actively to system component failures by reconfiguring control actions so that the stability and acceptable performance of the entire system can be maintained. In certain circumstances, degraded performance may have to be accepted [37]. AFTCS are also referred to as "self-repairing", "reconfigurable", "restructurable", or "self-designing" control systems by some researchers. From the viewpoint of functionality in handling faults, AFTCS were also called fault detection, identification (diagnosis) and accommodation schemes by other researchers. In such control systems, the controller compensates for the impacts of the faults either by selecting a pre-computed control law or by synthesizing a new one online. To achieve a successful control system reconfiguration, both approaches rely heavily on real-time FDD schemes to provide the most up-to-date information about the true status of the system. Therefore, the main goal in a fault-tolerant control system is to design a controller with a suitable structure to achieve stability and satisfactory performance, not only when all control components are functioning normally, but also in cases when there are malfunctions in sensors, actuators, or other system components (for example, in the system itself, in control computer hardware or in software). This book focuses only on aspects pertaining to AFTCS.

1.5 Objectives and Structure of AFTCS

The design objectives for AFTCS include the transient and the steady-state performance for the system, not only under normal operations but also under fault conditions. It is important to point out that the emphasis on system behaviors in these two modes of operation can be significantly different. During normal operation, more emphasis should be placed on the quality of the system behavior. In the presence of a fault, however, how the system survives with an acceptable (probably degraded) performance becomes a predominant issue. Typically, an AFTCS can be divided into four subsystems):

- an FDD scheme;
- a reconfigurable controller;
- a controller reconfiguration mechanism;
- a command/reference governor.

Inclusion of both an FDD scheme and reconfigurable controllers within the overall system structure is the main feature distinguishing an AFTCS from a PFTCS. Key issues in AFTCS are how to design:

- a controller which can easily be reconfigured;
- an FDD scheme with high sensitivity to faults and robustness to model uncertainties, operating condition variations, and external disturbances;
- a reconfiguration mechanism which leads as much as possible to the recovery of the pre-fault system performance in the presence of uncertainties and time-delays in FDD, within the constraints of control inputs and system states.

The critical issue in any AFTCS is the limited amount of time available for the FDD and for the control system reconfiguration. Furthermore, in the case of failure, efficient utilization and management of redundancy (in hardware, software and communication networks), stability, and a transient and a steady-state performance guarantee are some of the important issues to consider in AFTCS.

The overall structure of a typical AFTCS is shown in Figure 1.7. In the FDD module, any fault in the system should be detected and isolated as quickly as possible, and fault parameters, system state/output variables, and post-fault system models need to be estimated online in real-time. Based on the online information about the post-fault system model, the reconfigurable controller should be designed to maintain automatically the stability, desired dynamic performance and steady-state performance. In addition, in order to ensure the closed-loop system tracks a command input trajectory in the event of faults, a reconfigurable feed-forward controller often needs to be synthesized. To avoid potential actuator saturation and to take into consideration the degraded performance after fault occurrence, a command/reference governor may also need to be designed to adjust command input or reference trajectory automatically.

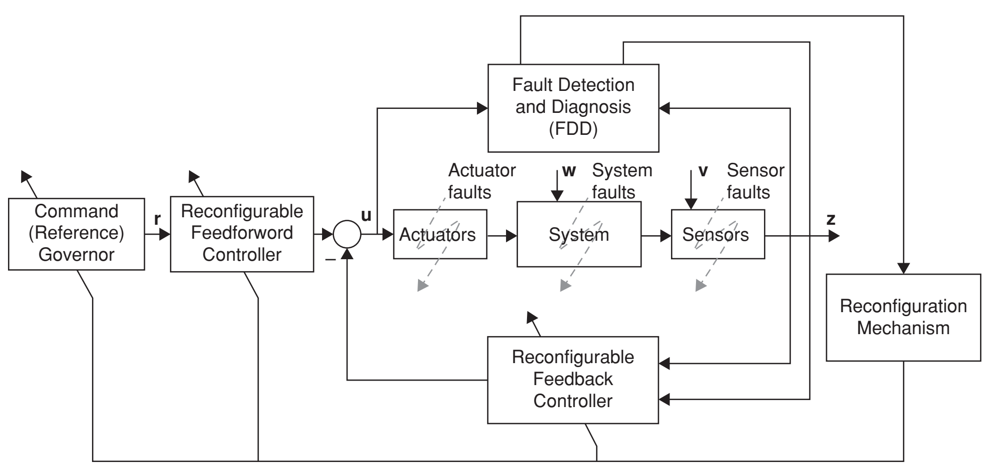

Figure 1.7 General structure of an AFTCS

Based on the described structure, the design objectives of an AFTCS can be stated as:

- to have an FDD scheme that provides, as precisely as possible, information about a fault (time, type and magnitude) and the post-fault model;
- to have a new control scheme (reconfigurable or restructurable) to compensate for the fault-induced changes in the system so that stability and an acceptable closed-loop system performance can be maintained.

It is important to point out that not only do the parameters of the controllers need to be recalculated, but also the structure of the new controllers (in terms of their order, number and type) might be changed. A corresponding AFTCS is often referred to as a "restructurable control system" to emphasize that the controller structure can change. Note that, in the literature, there are generally two ways of classifying AFTCS. One classifies them as reconfigurable versus restructurable; the other differentiates them as accommodation versus reconfiguration. In this book, we adopt the former definition. So long as there is no confusion, we use the term "reconfigurable control" in subsequent sections.

1.6 Classification of Reconfigurable Control Methods

Reconfigurable control methods can be broadly classified into several categories. The two most common categories are those based on control algorithms and those based on fields of application.

1.6.1 Classification based on control algorithms

In the literature, reconfigurable control design methods fall into one of the following approaches: linear quadratic; pseudo-inverse or control mixer; gain scheduling or linear parameter varying; (model reference) adaptive control or model following; eigenstructure assignment; multiple-model; feedback linearization or dynamic inversion; Hoo and other robust controls; model predictive control; variable structure and sliding mode control; generalized internal model control; and intelligent control using expert systems, neural networks, fuzzy logic and learning methodologies. Detailed classification can be carried out according to the following criteria:

- mathematical design tools: These include linear quadratic (LQ), intelligent control (IC), gain scheduling (GS)/linear parameter varying (LPV), adaptive control (AC), feedback linearization (FL)/dynamic inversion (DI), $\mathcal{H}_\infty$ and robust control, qualitative feedback theory (QFT), multiple model (MM), model predictive control (MPC), variable structure control (VSC)/sliding mode control (SMC)and generalized internal model control (GIMC);
- design approaches: These include pre-computed control laws (such as GS/LPV, MM, QFT and GIMC) or online automatic redesign (such as LQ, AC, FL/DI, VSC/SMC and MPC);
- reconfiguration mechanisms: These include optimization, switching, matching, following and compensation;
- types of system to be dealt with, whether linear or nonlinear.

An important criterion for judging the suitability of a control method for AFTCS is its ability to be implemented to maintain an acceptable (nominal or degraded) performance in the impaired system in an on-line real-time setting. In this regard, the following requirements should be satisfied:

- Control reconfiguration must be done under real-time constraints.
- The reconfigurable controller should be designed automatically with little trial-and-error or human interaction.
- The methods selected must provide a solution even if the solution is not optimal.

1.6.2 Classification based on field of application

A large amount of research has been carried out in the framework of aircraft flight control. Several reconfigurable flight control systems have been tested. With rapid advances in microelectronics, mechatronics, smart actuator and sensor techniques, and computing technologies, and motivated by increased demands for high requirements of system performance, product quality, productivity and operating efficiency beyond the conventional safety-critical aerospace and nuclear power systems, FTCS design is becoming an important feature to be considered in commercial product development and system design such as drive-by-wire automobiles. Recently, concepts and methodologies developed in fly-by-wire (FBW) fault-tolerant flight control systems have been extended to a wide range of engineering systems such as automobiles, railway vehicles, surface ships, autonomous underwater vehicles, automated highway systems, petrochemical plants, power systems, robots, medical systems and other industrial systems.

1.7 Outline of the Book

1.7.1 Methodology

Throughout the book, each chapter or section is composed of five parts:

- *mathematical modeling*: in which we discuss the main ingredients of the state-space model under consideration;
- *definitions or assumptions*: in which we state the definitions or constraints on the model variables to pave the way for subsequent analysis;
- *analysis and examples*: which is the core of the section and contains some solved examples for illustration;
- *results*: which are provided most of the time in the form of theorems, lemmas and corollaries.
- *remarks*: which are given to shed some light on the relevance of the developed results *vis-a-vis* published work.

Theorems, lemmas and corollaries are keyed to chapters, for example, *Theorem 3.4* means Theorem 4 in Chapter 3 and so on. Relevant notes and research issues are offered at the end of each chapter for the purpose of stimulating the reader. For convenience, the references are to be found at the end of each chapter. We hope that this way of articulating the information will attract the attention of a wide spectrum of readership.

This book aims to provide a rigorous framework for studying the analysis, stability and control problems of FTC while addressing the dominant sources of difficulties caused by: dimensionality; information structure constraints; parametric uncertainty and time delays. The primary objective is threefold: to review past methods and results from a contemporary perspective; to examine present trends and approaches; and to provide future possibilities, focusing on robust, reliable distributed design methods.

In brief, the main features of the book are:

- It provides an overall assessment of fault-tolerant control algorithms over the past few years.
- It addresses several issues that arise at the interaction of fault detection and fault accommodation.
- It presents key concepts with their proofs, followed by efficient computational methods.
- It considers representative industrial applications and provides the results of simulations tun using MATLAB®.

1.7.2 Chapter organization

Fault-tolerant control systems have been investigated for a long time in the control literature and have attracted increasingly more attention for more than three decades. The literature has grown progressively and quite number of fundamental concepts and powerful tools have been developed from various disciplines. Rapid technological progress raises many fundamental problems that call for further exploration. Among the core issues are those of representation, analysis, design and implementation. In particular, the field still lacks a unified framework that can cope with the core issues in a systematic way. This has motivated us to write the current book, which presents theoretical explorations of several fundamental problems with fault-tolerant control systems.

The book is primarily intended for researchers and engineers in the systems, control and communication community. It can also serve as complementary reading for elective courses for fault-tolerant control systems at the postgraduate level. The material of the book is divided into 10 chapters and an appendix:

- *Chapter 1* is an introductory chapter in which the different concepts and general ideas pertaining to fault-tolerant control (FTC) are presented.
- *Chapter 2* is devoted to a review of fault diagnosis and detection with extra emphasis on the unscented Kalman filter in an integrated design framework for process-fault processing within industrial applications.
- *Chapter 3* introduces robust fault detection methods including distributed fault diagnosis, data-driven detection and robust adaptive estimation.
- *Chapter 4* deals with designing a fault-tolerant control system as an essential component in an industrial process as it enables the system to continue robust operation under some conditions.
- *Chapter 5* outlines fault-tolerant control in nonlinear dynamical systems and presents a survey of various design approaches.
- *Chapter 6* establishes a detailed characterization of the problem of robust fault estimation filter design, encompassing multiconstrained fault estimation and adaptive tracking.

- *Chapter 7* provides results on methods for fault detection in dynamical systems operating over communication networks. This includes modified residual generator schemes, quantized fault-tolerant control, sliding-mode observer and linear-switched systems.
- *Chapter 8* presents some recent architectures and related concepts pertaining to industrial fault-tolerant detection.
- *Chapter 9* gives analysis and design methods of fault estimation for stochastic systems and their applications to unmanned air vehicles, real-time router fault accommodation and fault detection for Markov jump systems.
- *Chapter 10* is devoted to applications of fault-tolerant systems.
- *Appendix A* contains some relevant mathematical lemmas and basic algebraic inequalities that are used throughout the book.

Several illustrative numerical examples are provided within the individual chapters.

1.8 Notes

This book covers a wide spectrum of analysis and design tools for fault-tolerant control (FTC) in dynamical systems. The analytical tools developed in this volume are supplemented with rigorous proofs of closed-loop stability properties and simulation studies. For the purpose of clarity, the book is split into self-contained chapters with each chapter being equipped with illustrative examples, problems and questions. The book is supplemented by appropriate appendices and indexes. At the end of each chapter, a summary of the results with the relevant literature is given and pertinent discussions of future advances and trends are outlined. Some selected problems are left for the reader to solve. The bibliographical notes cannot give a complete review of the literature but indicate the most influential papers or books in which the results presented here have been described; they also provide interested readers with adequate entry points to the still-growing literature in this field.

While organizing the material, we planned that this book would be appropriate for use as graduate-level textbook in applied mathematics as well as other engineering disciplines (electrical, mechanical, civil, chemical, and systems engineering), a good volume for independent study and a suitable reference for graduate students, practicing engineers, interested readers and researchers from a wide spectrum of engineering disciplines, science and mathematics.

References

[1] Basseville, M. (1988) "Detecting changes in signals and systems: A survey", *Automatica*, **24**(3):309–326.

[2] Dash, S., and Venkatasubramanian, V. (2000) "Challenges in the industrial applications of fault diagnostic systems", *Computers and Chemical Engineering*, **24**(2):785–791.

[3] Frank, P. M., Ding, S. X., and Marcu, T. (2000) "Model-based fault diagnosis in technical processes", *Trans. Institute of Measurement and Control*, **22**(1):57–101.

[4] Dochain, D., Marquard, W., Won, S. C., Malik, O. P., and Kinnaert, M. (2006) "Monitoring and control of process and power systems: Towards new paradigms", *Annual Reviews in Control*, **30**(1):69–79.

[5] Witczak, M. (2007) *Modelling and Estimation Strategies for Fault Diagnosis of Nonlinear Systems: From Analytical to Soft Computing Approaches*, Lecture Notes in Control and Information Sciences, **354**, Berlin, Germany: Springer.

[6] Chandler, P. R. (1984) "Self-repairing flight control system reliability and maintainability program: Executive overview", in *Proc. IEEE National Aerospace and Electronics Conference*, 586–590.

[7] Chizeck, H. J., and Willsky, A. S. 1978 "Towards fault-tolerant optimal control", in *Proc. 1978 IEEE Conference on Decision and Control*, **1**, 19–20.
[8] Eterno, J. S., Weiss, J. L., Looze, D. P., and Willsky, A. S. 1985 "Design issues for fault tolerant-restructurable aircraft control", in *Proc. 24th IEEE Conference on Decision and Control*, **1**, 900–905.
[9] Montgomery, R. C., and Caglayan, A. K. (1976) "Failure accommodation in digital flight control systems by Bayesian decision theory", *J. Aircraft*, **13**(2):69–75.
[10] Montgomery, R. C., and Price, D. B. (1976) "Failure accommodation in digital flight control systems for nonlinear aircraft dynamics. *J. Aircraft*, **13**(2):76–82.
[11] Blanke, M., Izadi-Zamanabadi, R., Bøgh, S. A., and Lunau, C. P. (1997) "Fault-tolerant control systems: A holistic view", *Control Engineering Practice*, **5**(5):693–702.
[12] Blanke, M., Frei, C., Kraus, F., Patton, R. J., and Staroswiecki, M. (2000) "What is fault-tolerant control?", in *Proc. 4th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes*, 40–51.
[13] Blanke, M., Staroswiecki, M., and Wu, N. E. (2001) "Concepts and methods in fault-tolerant control", in *Proc. American Control Conference*, 2606–2620.
[14] Jiang, J. (2005) "Fault-tolerant control systems: An introductory overview", *Automatica SINCA*, **31**(1): 161–174.
[15] Staroswiecki, M., and Gehin, A. L. (2001) "From control to supervision", *Annual Reviews in Control*, **25**: 1–11.
[16] Steinberg, M. (2005) "Historical overview of research in reconfigurable flight control", in *Proc. IMechE, Part G: J. Aerospace Engineering*, **219**:263–275.
[17] Mahmoud, M., Jiang, J., and Zhang, Y. M. (2003) *Active Fault Tolerant Control Systems: Stochastic Analysis and Synthesis*, Lecture Notes in Control and Information Sciences, **287**, Berlin, Germany: Springer.
[18] Ding, S. X. (2008) *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*, Berlin, Heidelberg: Springer.
[19] Isermann, R. (2006) *Fault-Diagnosis Systems*, Springer, UK.
[20] Isermann, R., and Balle, P. (1997) "Trends in the application of model-based fault detection and diagnosis of technical processes", *Control Eng. Practice*, **5**, 709–719.
[21] Chen, J., and Patton, R. J. (1999) *Robust Model-Based Fault Diagnosis for Dynamic Systems*, Boston, Mass, USA: Kluwer Academic Publishers.
[22] Frank, P. M. (1996) "Analytical and qualitative model-based fault diagnosis: a survey", *Europ. J. Control*, **2**:6–28.
[23] Frank, P. M. (1994) "On-line fault detection in uncertain nonlinear systems using diagnostic observers: a survey and some new results", *Int. J. Systems Science*, **25**(12):2129–2154.
[24] Sauter, D., Mary, N., Sirou, F., and Thieltgen, A. (1994) "Fault diagnosis in systems using fuzzy logic", in *Proc. 3rd IEEE Conf. Control Application*, Glasgow, UK, 883–888.
[25] Venkatasubramanian, V., Rengaswamy, R., Yin, K., and Kavuri, S. N. (2003) "A review of process fault detection and diagnosis part II: quantitative model based methods", *Computers and Chemical Engineering.*, **27**: 313–326.
[26] Patton, K. (2008) *Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes* Springer, UK.
[27] Chiang, L. H., Russell, E. L., and Braatz, R. D. (2001) *Fault Detection and Diagnosis in Industrial Systems*, Springer, UK.
[28] Chow, E. Y., and Willsky, A. S. (1984) "Analytical redundancy and the design of robust failure detection systems", *IEEE Trans. Automatic Control*, **29**:603–614.
[29] Gertler, J. J., and Singer, D. (1990) "A new structural framework for parity equation based failure detection and isolation", *Automatica*, **26**:381–388.
[30] Gertler, J., and Kunert, M. (1995) "Optimal residual decoupling for robust fault diagnosis", *Int. J. Control.*, **61**:395–421.
[31] Delmaire, G., Cassar, J. P., and Staroswiecki, M. (1994) "Comparison of identification and parity space approach for failure detection in single-input single-output systems", in *Proc. IEEE Conf. Contr. A*, Glasgow, UK, 865–870.
[32] Staroswiecki, M., Cassar, J. P., and Cocquempot, V. (1993) "Generation of optimal structured residuals in the parity space", in *Proc. 12th IFAC World Congress*, Sydney, Australia, 535–542.
[33] Zhang, Y. M., and Jiang, J. (2002) "An active fault-tolerant control system against partial actuator failures", *IEE Proc. Control Theory and Applications*, **149**(1):95–104.

[34] Zhang, Y. M., and Jiang, J. (2002) "Design of restructurable active fault-tolerant control systems", *Preprints of the 15th IFAC World Congress*, Barcelona, Spain.

[35] Hsieh, C. S. (2002) "Performance gain margins of the two-stage LQ reliable control", *Automatica*, **38**:1985–1990.

[36] Jiang, J., and Zhao, Q. (2000) "Design of reliable control systems possessing actuator redundancies", *J. Guidance, Control, and Dynamics*, **23**(4):709–718.

[37] Stengel, R. F. (1991) "Intelligent failure-tolerant control", *IEEE Control Systems Magazine*, **11**(4):14–23.