Linköping Studies in Science and Technology.
Dissertations No. 591

Model Based Fault Diagnosis
Methods, Theory, and Automotive Engine Applications

Mattias Nyberg

Department of Electrical Engineering
Linköping University, SE-581 83 Linköping, Sweden

Linköping 1999

Model Based Fault Diagnosis:
Methods, Theory, and Automotive Engine Applications

Department of Electrical Engineering,
Linköping University,
SE–581 83 Linköping,
Sweden.

ISBN 91-7219-521-5
ISSN 0345-7524

Printed by Linus & Linnea AB, Linköping, Sweden, 1999.

Abstract

Model based fault diagnosis is to perform fault diagnosis by means of models. An important question is how to use the models to construct a diagnosis system. To develop a general theory for this, useful in real applications, is the topic of the first part of this thesis. The second part deals with design of linear residual generators and fault detectability analysis.

A general framework, for describing and analyzing diagnosis problems, is proposed. Within this framework a diagnosis method *structured hypothesis tests* is developed. It is based on general hypothesis testing and the task of diagnosis is transferred to the task of validating a set of different models with respect to the measured data. The procedure of deriving the diagnosis statement, i.e. the output from the diagnosis system, is in this method formalized and described by logic.

Arbitrary types of faults, including multiple faults, can be handled, both in the general framework and also in the method structured hypothesis tests. It is shown how well known methods for fault diagnosis fit into the general framework. Common methods such as residual generation, parameter estimation, and statistically based methods can be seen as different methods to generate test quantities within the method structured hypothesis tests.

Based on the general framework, a method for evaluating and comparing diagnosis systems is developed. Concepts from decision theory and statistics are used to define a performance measure, which reflects the probability of e.g. false alarm and missed detection. Based on the evaluation method, a procedure for automatic design of diagnosis systems is developed.

Within the framework, diagnosis systems for the air-intake system of automotive engines are designed. In one case, the procedure for automatic design is used. Also the methods for evaluation of diagnosis systems are applied. The whole design chain is described, including the modeling of the engine. All diagnosis systems are validated in experiments using data from a real engine. This application highlights the strengths of the method structured hypothesis tests, since a large variety of different faults need to be diagnosed. To the authors knowledge, the same problem can not be solved using previous methods.

In the second part of the thesis, linear residual generation is investigated by using a notion of *polynomial bases* for residual generators. It is shown that the order of such a basis doesn't need to be larger than the system order. Fault detectability, seen as a system property, is investigated. New criterions for fault detectability, and especially *strong* fault detectability, are given.

A new design method, the *minimal polynomial basis approach*, is presented. This method is capable of generating all residual generators, explicitly those of minimal order. Since the method is based on established theory for polynomial matrices, standard numerically efficient design tools are available. Also, the link to the well known Chow-Willsky scheme is investigated. It is concluded that in its original version, it has not the nice properties of the minimal polynomial basis approach.

Acknowledgments

This work has been carried out at the division of Vehicular Systems, Linköping University, with professor Lars Nielsen as supervisor. I would like to thank him for leading me into the area of model based diagnosis, his support, and inspiring discussions. I would also like to thank all staff at Vehicular Systems for creating a positive atmosphere.

I would like to thank NUTEK (Swedish National Board for Industrial and Technical Development) for financially supporting this work through the research center ISIS (Information Systems for Industrial Control and Supervision).

I'm indebted to SAAB Automobile and Mecel AB for providing experimental equipment and support for the experiments. Especially I would like to thank Thomas Gobl at SAAB Automobile for his engagement in the work.

My research colleague Erik Frisk is greatfully acknowledged for many insightful discussions, reading the manuscript, and help with LaTeX. For the experimental part, our research engineer Andrej Perkovic is acknowledged for help with experiments and keeping our lab running. Other people who I have enjoyed fruitful discussions with are Lars Eldén, Eva Enquist, Fredrik Gustavsson, and Magnus Larsson.

Finally I would like to thank my family and Maria for their encouragement and support during the work.

Linköping, May 1999

Mattias Nyberg

Contents

Some Notations Used

Θ	set of all fault states
Θ_γ	fault state space for fault mode γ
θ	fault state
$\mathcal{D}^i$	fault state space of component i
$\mathcal{D}^i_\psi$	fault state space of component i and component fault-mode ψ
θ_i	fault state of component i
θ_γ	free fault state parameter for fault mode γ
$\mathcal{M}(\theta)$	complete system model
$\mathcal{M}_\gamma(\theta) = \mathcal{M}_\gamma(\theta_\gamma)$	system model for fault mode γ

Chapter 1

Introduction and Overview of Thesis

Model based fault diagnosis is to perform fault diagnosis by means of models. An important question is how to use the models to construct a diagnosis system. To develop a theory for this, useful for real applications, is the topic of the first part of this thesis. The second part deals with design of linear residual generators and fault detectability analysis.

This chapter starts by, in Section 1.1, giving an introductory background and a general motivation to the field of fault diagnosis. In Section 1.2, some fundamental definitions are reviewed. Then Section 1.3 contains an overview and some criticism to some present approaches to fault diagnosis. Finally, Section 1.4 summarizes the thesis and gives the main contributions.

1.1 Introductory Background

From a general perspective, including for example medical and technical applications, fault diagnosis can be explained as follows. For a process there are observed variables or behavior for which there are knowledge of what is expected or normal. The task of fault diagnosis is to, from the observations and the knowledge, generate a *diagnosis statement*, i.e. to decide whether there is a fault or not and also to identify the fault. Thus the basic problems in the area of fault diagnosis is how the procedure for generating the diagnosis statement should look like, what parameters or behavior that are relevant to study, and how to derive and represent the knowledge of what is expected or normal.

This thesis focuses on diagnosis of technical systems, and typical faults considered are for example sensor faults and actuator faults. The observations are mainly output signals obtained from the sensors, but can also be observations made by a human, such as level of noise and vibrations. The knowledge of what is expected or normal, is derived from commanded inputs together with models

of the system. The term *model based* fault diagnosis refers to the fact that the knowledge of what is expected or normal, is represented in an explicit model of the system. The type of models considered is mainly differential equations.

Model based diagnosis of technical systems has gained much industrial interest lately. The reason is that it has possibilities to improve for example safety, environment protection, machine protection, availability, and repairability. Some important applications that have been discussed in the literature are:

- Nearly all subsystems of aircrafts, e.g. aircraft control system, navigation system, and engines
- Emission control systems in automotive vehicles
- Nuclear power plants
- Chemical plants
- Gas turbines
- Industrial robots
- Electrical motors

Manual diagnosis of technical systems has been performed as long as technical systems have existed, but automatic diagnosis started to appear first when computers became available. In the beginning of the 70's, the first research reports on model based diagnosis were published. Some of the earliest areas, that were investigated, were chemical plants and aerospace applications. The research on model based diagnosis has since then been intensified during both the 80's and the 90's. Today, this is still an expansive research area with many unsolved questions. Some references to books in the area are (Patton, Frank and Clark, 1989; Basseville and Nikiforov, 1993; Gertler, 1998; Chen and Patton, 1999).

Up to now, numerous methods for doing diagnosis have been published, but many approaches are more ad hoc than systematic. It is fair to say that few general theories exist, and a complete understanding of the relations between different methods has been missing. This is reflected in that few books exists and the fact that no general terminology has yet been widely accepted. However the importance of diagnosis is unquestioned. This can be exemplified by the computerized management systems for automotive engines. For these system, as much as 50% of the software is dedicated to diagnosis. The other 50% is for example for control.

1.1.1 Traditional vs Model Based Diagnosis

Traditionally diagnosis has been performed by mainly limit checking. When for example a sensor signal level leaves its normal range, an alarm is generated. The

normal range is predefined by using thresholds. This normal range can be dependent on the operating conditions. In for example an aircraft, the thresholds, for different operating points defined by altitude and speed, can be stored in a table. This use of thresholds as functions of some other variables, can actually be viewed as a kind of model based diagnosis.

Another traditional approach is duplication (or triplication or more) of hardware. This is usually called *hardware redundancy* and the typical example is to use redundant sensors. There are at least three problems associated with the use of hardware redundancy: hardware is expensive, it requires space, and adds weight to the system. In addition, extra components increase the complexity of the system which in turn may introduce extra diagnostic requirements.

Model Based Fault Diagnosis

Increased usage of explicit models in fault diagnosis has a large potential to have the following advantages:

- Higher diagnosis performance can be obtained, for example smaller and also more types faults can be detected and the detection time is shorter.
- Diagnosis can be performed over a larger operating range.
- Diagnosis can be performed passively without disturbing the operation of the process.
- Increased possibilities to perform isolation.
- Disturbances can be compensated for, which implies that high diagnosis performance can be obtained in spite of the presence of disturbances.
- Reliance on hardware redundancy can be reduced, which means that cost and weight can be reduced.

The model can be of any type, from logic based models to differential equations. Depending on the type of model, different approaches to model based diagnosis can be used, for example statistical approaches, AI-based approaches, or approaches within the framework of control theory. It is sometimes believed that model based diagnosis is very complex. This is not true since for example traditional limit checking is also a kind of model based diagnosis.

The disadvantage of model based diagnosis is quite naturally the need for a reliable model and possibly a more complex design procedure. In the actual design of a model based diagnosis system, it is likely that the major part of the work is spent on building the model. This model can however be reused, e.g. in control design. Someone may argue that an disadvantage of increasing the usage of models is that more computing power is needed to perform the diagnosis. However, this conclusion is not fair. Actually, for the same level of performance it can be the case that an increased used models is *less* computationally intensive than traditional approaches.

The accuracy of the model is usually the major limiting factor of the performance of a model based diagnosis system. Compared to the area of model based control, the quality of the model is much more important in diagnosis. The reason for is that the feedback, used in closed-loop control, tends to be forgiving against model errors. Diagnosis should be compared to open-loop control since no feedback is involved. All model errors propagates through the diagnosis system and degrades the diagnosis performance.

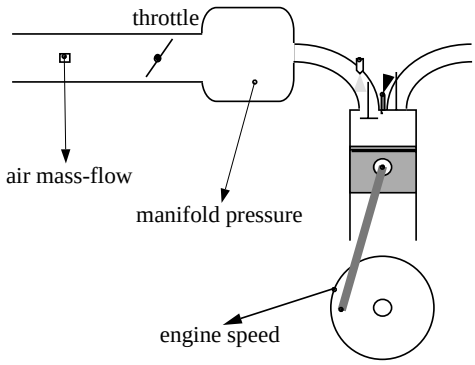

Figure 1.1: A principle illustration of an SI-engine.

Following is an example of a successful industrial application of model based diagnosis.

Example 1.1

Consider Figure 1.1, containing a principle illustration of a spark-ignited combustion engine. The air enters at the left side, passes the throttle and the manifold, and finally enters the cylinders. The engine in the figure have three sensors measuring the physical variables air mass-flow, manifold pressure, and engine speed.

The air flow $\dot{m}$ into the cylinders can be modeled as a function of manifold pressure p and engine speed n, i.e. $\dot{m} = g(p, n)$. The physics behind the function g is involved and it is therefore usually modeled by a black-box model. In engine management systems, one common solution is to represent the function g as a lookup-table. So by using this lookup-table an estimation of the air mass-flow can be obtained. When the measured air mass-flow significantly differs from the estimation, it can be concluded that a fault must be present somewhere in the engine. The fault can for example be that one of the three sensors are faulty or that a leakage have occured somewhere between the air mass-flow sensor and the cylinder. This is an example of model based diagnosis that is commonly used in production cars today. ■

1.2 Present Definitions

As a step towards a unified terminology, the IFAC Technical Committee SAFEPROCESS has suggested preliminary definitions of some terms in the field of fault diagnosis. Some of these definitions are given here as a way to introduce the field. Another reason is that most of these terms will be given a more formal definition later in this theses.

The following list of definitions is a subset of their list:

- **Fault**
  Unpermitted deviation of at least one characteristic property or variable of the system from acceptable/usual/standard behavior.
- **Failure**
  Permanent interruption of a systems ability to perform a required function under specified operating conditions.
- **Fault Detection**
  Determination of faults present in a system and time of detection.
- **Fault Isolation**
  Determination of kind, location, and time of detection of a fault. Follows fault detection.
- **Fault Identification**
  Determination of the size and time-variant behavior of a fault. Follows fault isolation.
- **Fault Diagnosis**
  Determination of kind, size, location, and time of detection of a fault. Follows fault detection. Includes fault isolation and identification.

For the definition of the term *fault diagnosis*, one slightly different definition also exists in the literature. This definition can be found in for example (Gertler, 1991) and says that *fault diagnosis* also includes *fault detection*. This is also the view taken in this thesis.

If fault detection is excluded from the term *diagnosis*, as in the SAFEPROCESS, one gets a problem of finding a word describing the whole area. This has partly been solved by introducing the abbreviation FDI (Fault Detection and Isolation), which is common in many papers.

In this context, it is also interesting to see how a general dictionary defines the word *diagnosis*. The following information can be found in the Webster Dictionary:

> **diagnosis**
> Etymology: New Latin, from Greek *diagnOsis*, from *diagignOskein* to distinguish, from *dia-* + *gignOskein* to know
> Date: circa 1681
> **1 a :** the art or act of identifying a disease from its signs and symptoms **b :** the decision reached by diagnosis

2 a : investigation or analysis of the cause or nature of a condition, situation, or problem <*diagnosis* of engine trouble>
b : a statement or conclusion from such an analysis

1.3 Present Approaches to Model Based Fault Diagnosis

This section is included because of two reasons. The first is to point out some problems with present approaches to fault diagnosis. The first part of the thesis is then devoted to present a new approach in which these problems are avoided. The second reason is to give newcomers to the field of fault diagnosis a short background to some of the approaches present in literature.

By reading recent books (Gertler, 1998; Chen and Patton, 1999) about fault diagnosis of technical processes, or survey papers (Patton, 1994; Gertler, 1991; Frank, 1993; Isermann, 1993), one can come to the conclusion that the two most common systematic approaches to fault diagnosis is to use a *"residual view"* or *parameter estimation*. Below these two approaches are presented shortly.

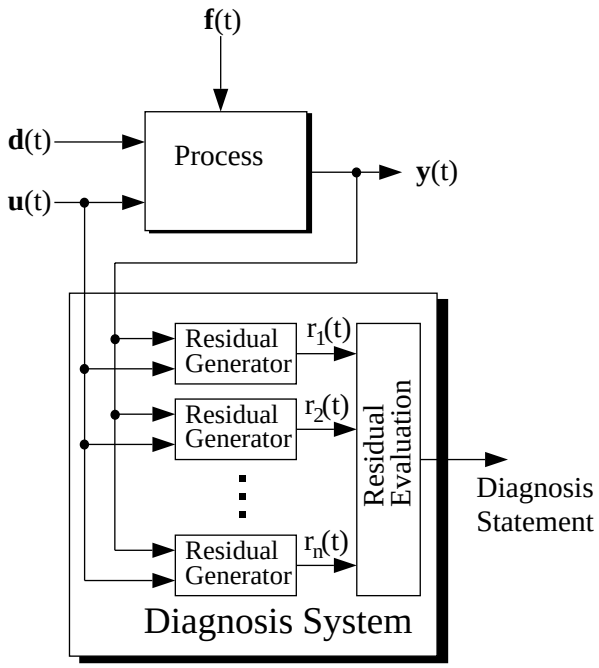

Figure 1.2: A diagnosis system based on the "residual view".

1.3.1 The "Residual View"

With this approach, faults are modeled by signals $f(t)$. Central is the residual $r(t)$ which is a scalar or vector signal that is 0 or small in the fault free case, i.e. $f(t) = 0$, and is $\neq 0$ when a fault occurs, i.e. $f(t) \neq 0$. The diagnosis system is then separated into two parts: *residual generation* and *residual evaluation*.

This view of how to design diagnosis system is well established among fault diagnosis researchers. This is emphasized by the following quotation from the most recent book (Chen and Patton, 1999) in the field:

> *"Chow and Willsky (1984) first defined the model-based FDI as a two-stages process: (1) residual generation, (2) decision making (including residual evaluation). This two-stages process is accepted as a standard procedure for model-based FDI nowadays."*

Almost equally well established is the following way of constructing the residual evaluation (also called decision logic) procedure. The method is often called *structured residuals* and is primarily an isolation method. A diagnosis system using structured residuals can be illustrated as in Figure 1.2. In this method, the first step of the residual evaluation is essentially to check if each residual is responding to the fault or not, often achieved via simple thresholding. By using residuals that are sensitive to different subsets of faults, isolation can be achieved. What residuals that are sensitive to what faults is often illustrated with a *residual structure*. An example of a residual structure is

	f_1	f_2	f_3
r_1	0	1	0
r_2	0	1	1
r_3	1	0	1

The 1:s indicates which residuals that are sensitive to each fault. For this residual structure, assume for example that residuals r_2 and r_3 are responding, and r_1 is not. Then the conclusion is that fault f_3 has occured.

A large part of all fault-diagnosis research has been to find methods to design residual generators. Of this large part, most results are concerned with linear systems.

A limitation with this approach to fault diagnosis is that faults are modeled as signals. This is very general and might therefore seem to be a good solution. However, the generality of this fault model is actually its drawback. Many faults can be modeled by less general models, and we will see in this thesis that to facilitate isolation this is necessary in many situations.

Another limitation is that the residual structure, with its 0:s and 1:s, places quite strong requirements on the residual generators. A 1 more or less means that the corresponding residual *must* respond to the fault. It can be understood that for small faults in real systems, with noise and model uncertainties present, this requirement is often violated.

A third limitation, related to the the previous limitation, is that the decision procedure, of how the diagnosis statement is formed from the real-valued

residuals, does not have a solid theoretical motivation. For example, in the context of deciding the diagnosis statement, what are the meanings of the 0:s and the 1:s, and what does it mean that a residual is above the threshold? It would be desirable to use a decision procedure for which we can find an intuitive formalism based on existing well-established theory, preferably mathematics if possible.

1.3.2 Parameter Estimation

The other main approach to model-based fault-diagnosis is to model faults as deviations in constant parameters. To illustrate the concept, consider a system with a model $\mathcal{M}(\theta)$, where θ is a parameter having the nominal (i.e. fault-free) value θ_0. By using general parameter estimation techniques, an estimate $\hat{\theta}$ can be formed and then compared to θ_0. If $\hat{\theta}$ deviates to much from θ_0, then the conclusion is that a fault has occured.

The most severe limitation with this approach is its quite restricted way of modeling faults. To model many realistic faults, more general fault models must be used.

Another limitation is that when the number of diagnosed faults grow, the parameter vector θ grows in dimension. This is a serious problem because the computations needed to calculate $\hat{\theta}$ can become quite difficult.

1.3.3 This Thesis

The first part of this thesis, i.e. Chapter 2 to 4, suggests a new approach to fault diagnosis. This approach does not have the limitations indicated above. Also, it includes both structured residuals and the parameter estimation approach as special cases.

1.4 Summary and Contributions of the Thesis

The summaries of the different chapters, given below, indicate the scope of the thesis and also give an idea of the contributions. In addition, a summary of the main contributions is included in the end of this section.

Chapter 2: A General Framework for Fault Diagnosis

In this chapter a new framework for describing and analyzing diagnosis problem is presented. The presentation is formal, and often used terms like “fault”, “isolation”, and “detectability” are defined. A connection to diagnosis based on logic (AI), is indicated.

In contrast to previous existing frameworks, e.g. the residual view, arbitrary fault models can be handled. Also multiple faults are naturally integrated so that no special treatment is needed. A diagnosis-system architecture, based on basic ideas from decision theory and propositional logic, is presented. We

introduce the idea that the output from a diagnosis system can be several possible faults. Finally, results that relates fault modeling with detectability and isolability properties, are developed.

Chapter 3: Structured Hypothesis Tests

The general diagnosis-system architecture presented in the previous chapter is refined to the isolation method *structured hypothesis tests*. It is based on general hypothesis testing and uses the general framework developed in Chapter 2. The task of diagnosis is transferred to the task of validating a set of different models with respect to the measured data. A main advantage with this method is that it can handle arbitrary types of faults. As a way to describe the structure of the diagnosis system we use an *incidence structure* and a *decision structure*. Also the relation to the method *structured residuals* is investigated.

Chapter 4: Design and Evaluation of Hypothesis Tests for Fault Diagnosis

This chapter discusses how to design hypothesis tests to be used with the method structured hypothesis tests. Three principles are described: the *prediction*, the *likelihood*, and the *estimate* principle. These three principles should be sufficient to solve most diagnosis problems.

In this chapter we see how well known methods for fault diagnosis fit in the general framework from Chapter 2 and structured hypothesis tests. This also clarifies conceptual links between different approaches to fault diagnosis, e.g. the connection between residual generation, parameter estimation, and a statistically based method for detection of abrupt changes. The importance of *normalization* is emphasized. Two special cases of this is adaptive thresholds and the likelihood ratio test.

Also discussed is how to evaluate hypothesis tests and for this, tools from statistics and decision theory are used. The evaluation scheme developed is applied to compare the estimate principle and the prediction principle, and it is concluded that the former has some optimality properties.

Chapter 5: Applications to an Automotive Engine

The methods and the theory developed in the previous chapters are applied to an automotive engine. Test quantities and diagnosis systems are designed and analyzed. The whole design chain is covered including the modeling of the engine. The results are validated in experiments using data from a real engine. The diagnosis system constructed highlights the strengths of the method structured hypothesis tests, since a large variety of different faults can be handled. To the authors knowledge, the same problem can not be solved using previous methods.

Chapter 6: Evaluation and Automatic Design of Diagnosis Systems

Based on decision theory, a method for evaluating and comparing diagnosis system is developed. Probability measures, such as probabilities of false alarm and missed detection, are used. One key result is the method to evaluate the performance of a complete diagnosis system by using probability measures of individual hypothesis tests.

Based on the evaluation method developed, a procedure for automatic design of diagnosis systems is proposed. The procedure is applied to a real automotive engine. The diagnosis system obtained is validated using experimental data from the engine and the results show both that the procedure is working and also that the evaluation method is sound.

Chapter 7: Linear Residual Generation

Design of linear residual generators, which is a special case of the prediction principle, is considered. A new method, the *minimal polynomial basis approach* has been developed in a joint work with Erik Frisk. This method is capable of generating all residual generators, explicitly those of minimal McMillan order. Since the method is based on established theory for polynomial matrices, standard numerically efficient design tools are available.

Also the well known Chow-Willsky scheme is investigated and it is concluded that in its original version, it has not the nice properties of the minimal polynomial basis approach. However, the Chow-Willsky scheme is modified so that it algebraically, although not numerically, becomes equivalent to the minimal polynomial basis approach.

The order of linear residual generators is investigated and it is concluded that to generate a *basis*, for all residual generators, it is sufficient to consider orders up to the system order. This result is new since previous related results only deal with the *existence* of residual generators and also only for some restricted cases.

Chapter 8: Criterions for Fault Detectability in Linear Systems

This chapter refines the general concepts of fault detectability from Chapter 2 to linear systems. The notion of bases, from the previous chapter, is used to investigate fault detectability seen as a system property, i.e. if there exists any residual generator in which a fault is detectable. New criterions for fault detectability and especially strong fault detectability are developed.

1.4.1 Main Contributions

- The general framework, for describing arbitrary faults, and describing and analyzing diagnosis problems, presented in Chapter 2.

- The diagnosis method *structured hypothesis tests* presented in Chapter 3.

- The methods to evaluate and compare diagnosis systems, presented in Chapter 4 and 6.
- Demonstration of the feasibility of the evaluation and design methods in real applications, presented in Chapter 5.
- The method to design linear residual generators, the *minimal polynomial basis approach*, presented in Chapter 7.
- The criterions for fault detectability and strong fault detectability in linear systems, presented in Chapter 8.

1.5 Publications

In the research work, leading to this thesis, the author has published the following conference and journal papers:

- Nyberg M. and Nielsen L. (1997), Model Based Diagnosis for the Air Intake System of the SI-Engine, SAE 1997 Transactions: Journal of Commercial Vehicles.
- Nyberg M. and Nielsen L. (1997), Design of a Complete FDI System based on a Performance Index With Application to an Automotive Engine, IFAC Fault Detection, Supervision and Safety for Technical Processes, Hull, United Kingdom, pp 812-817.
- Frisk M., Nyberg M. and Nielsen L. (1997), FDI with adaptive residual generation applied to a DC-servo, IFAC Fault Detection, Supervision and Safety for Technical Processes, Hull, United Kingdom, pp 438-443.
- Nyberg M. and Nielsen L. (1997), Parity Functions as Universal Residual Generators and Tool for Fault Detectability Analysis, IEEE Conf. on Decision and Control, San Diego, California, pp 4483-4489.
- Nyberg M. and Perkovic A. (1998), Model Based Diagnosis of Leaks in the Air-Intake System of an SI-Engine, SAE Paper 980514.
- Nyberg M. (1998), SI-Engine Air-Intake System Diagnosis by Automatic FDI-Design, IFAC Workshop Advances in Automotive Control, Columbus, Ohio, pp 225-230.
- Nyberg M. (1999), Model Based Diagnosis of Both Sensor-Faults and Leakage in the Air-Intake System of an SI-Engine, SAE Paper 1999-01-0860.
- Nyberg M. and Frisk E. (1999), A Minimal Polynomial Basis Solution to Residual Generation for Fault Diagnosis in Linear Systems, IFAC, Beijing, China.

- Nyberg M. and Nielsen L. (2000), A Universal Chow-Willsky Scheme and Detectability Criteria, IEEE Trans. Automatic Control.
- Nyberg M. (1999), Framework and Method for Model Based Diagnosis with Application to an Automotive Engine, ECC, Karlsruhe, Germany.
- Frisk E. and Nyberg M. (1999) Using Minimal Polynomial Bases for Fault Diagnosis, ECC, Karlsruhe, Germany.
- Nyberg M. (1999 or 2000), Automatic Design of Diagnosis Systems with Application to an Automotive Engine, accepted for publication in Control Engineering Practice.

Chapter 2

A General Framework for Fault Diagnosis

The author's experience and also other people's experience, e.g. Bøgh (1997), is that ad-hoc approaches to fault diagnosis give equally good or even better performance than present systematic approaches. One reason is that present approaches are too limited to special cases. For example, there is a large amount of systematic methods that are designed for linear systems. The problem is that almost no real systems are linear enough so that these methods often result in bad performance.

Previous attempts to introduce systematics have very much focused on systematic methods to design residual generators[1]. However, of all parts in a design chain, it is not sure that residual generation is the right thing to systematize. The reason is that systematic methods for residual generation tend to be either not general enough, so that they are not applicable to the specific application at hand, or *too* general, so that they can not utilize the special structure of each application. One further reason is that for many cases, residual generator design is actually not very difficult, and engineering intuition can often take us far. Instead of focusing on systemization of the residual generation, the approach in the following three chapters is to systematize other parts of the design, e.g. the architecture of the diagnosis system, and leaves the details of the residual generator design to the engineer. However, we will give some general principles also for the residual generation part.

The underlying philosophy of all this is that the engineer should do what he or she makes best, which is probably the residual generation, and the rest should be left to the design method. The goal has been to find a systematic approach that can utilize ad-hoc design of residual generators at the maximum. In this way, design solutions that have been previously considered to be ad-

[1] We use the term *residual generator* here in a quite broad meaning. This is because many readers have a quite good understanding of this term. However after this introductory section, we will switch to a more general terminology and *residual generator* will only be used for some specific cases.

hoc becomes part of a systematic method. Also previous methods that have been considered to be systematic, e.g. structured residuals, statistical methods, parameter estimation, are naturally included.

Although systematic, many previous diagnosis approaches are not based on any theoretical framework, as was exemplified in Section 1.3. On the contrary the approach suggested here is theoretically grounded in hypothesis testing (seen from either a statistical or decision theoretic standpoint) and to some extent also in propositional logic. Since many previous diagnosis methods are part of this framework, it also serves as a theoretical motivation to the methods that were previously not theoretically grounded. The approach presented is also strongly connected to how human beings would reason when performing diagnosis.

As said above, the description of this approach is distributed in the following three chapters. We start in this chapter by giving a general framework in which diagnosis problems can be described in a formalized and abstract manner. We will throughout this chapter, and also the following, not be restricted to *any* special types of faults and also, no restriction will be made regarding the multiplicity of faults. This is in contrast to almost all other works in which it is common that only one specific type of fault is considered and also only single faults. In fact the presented framework is valid for any arbitrary faults in any multiplicity.

Why is there a need for a general framework for fault diagnosis? One motivation is that in many situations we need to design diagnosis systems capable of diagnosing several different types of faults at the same time. One example of this is the automotive engine application investigated in Chapter 5. Another motivation is that, if we find design or analysis methods that can be described in terms of a general framework, then they are automatically valid for a large class of diagnosis problems. An example of such a design method is the *structured hypothesis tests* given in Chapter 3, and an example of such an analysis method is the method for diagnosis-system evaluation given in Chapter 6.

The first part of this chapter, i.e. Section 2.1, discusses fault modeling and then, in Section 2.2, the notion of *fault modes* will be introduced. Then a general architecture for a diagnosis system is given in Section 2.3. Section 2.4 defines a *submode* relation between fault modes and Section 2.5 contains definitions of isolability and detectability. Finally, Section 2.6 discusses what implications the submode relation has on isolability and detectability. All the formalism introduced in this chapter will be used in the next two chapters to describe more precise methods that can be used to perform diagnosis. Note that all notations introduced are summarized in the beginning of this thesis (and also in Appendix 2.A).

2.1 Fault Modeling

For constructing a model-based diagnosis system, a model of the system is needed. This model is the formal representation of the knowledge of possible faults and how they influence the process. In general, better models implies

better diagnosis performance, e.g. smaller faults can be detected and more different types of faults can be isolated. We will in this section describe a general framework for fault modeling. In this framework, practically all existing fault modeling techniques fit in naturally.

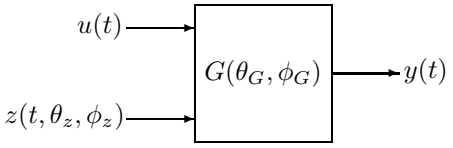

Figure 2.1: A general system model, linear or non-linear.

2.1.1 Fault State

The system model considered is illustrated in Figure 2.1. The model consists of a plant $G(\theta_G, \phi_G)$ and the vector valued signal $z(t, \theta_z, \phi_z)$. The parameters θ_G and θ_z describe faults and the parameters ϕ_G and ϕ_z describe disturbances.

The plant is modeled as an arbitrary system $G(\theta_G, \phi_G)$ described by differential equations. It has known inputs $u(t)$, e.g. control signals, and measurable outputs $y(t)$. In addition, the plant can be affected by other signals, which are collected in $z(t, \theta_z, \phi_z)$. These additional signals are assumed to be unknown or at least partially unknown. Some of the signals $z(t, \theta_z, \phi_z)$ may be modeled as stochastic processes. Note that the plant $G(\theta_G, \phi_G)$ is considered to be completely deterministic, and thus all stochastic parts of a model are collected in the signal $z(t, \theta_z, \phi_z)$. Except for this, there are cases in which a part of a model can be included in either $G(\theta_G, \phi_G)$ or $z(t, \theta_z, \phi_z)$. In such cases it is up to the user to decide what is most natural for the given application.

The constant parameter vector θ_G represents the true but unknown fault situation of the plant $G(\theta_G, \phi_G)$. The constant parameter vector θ_z represents the true but unknown fault situation of the signal $z(t, \theta_z, \phi_z)$. The parameter vector $\theta = [\theta_G \ \theta_z]$ is called the *fault state* and represents the fault situation of the complete system. One or possibly several fault states always corresponds to the fault-free case. The *fault state space*, i.e. the parameter space of θ, will be denoted Θ. Note that we have chosen the convention that θ is not dependent on time which corresponds to an assumption that the fault state of the system never changes. Even though this may seem to be a limitation, this is not the case as we will see later. We will be quite liberal regarding the definition of the parameter vector θ, e.g. we will allow elements that are functions.

Corresponding to θ there is the constant parameter vector $\phi = [\phi_G \ \phi_z]$, which represents disturbances affecting the system. However, this thesis will mostly *not* be focused on handling of disturbances. Therefore, the parameter ϕ will often be neglected and the system model then consists of $G(\theta_G)$ and $z(t, \theta_G)$.

Example 2.1

Consider a model of an amplifier:

$$y(t) = gu(t) + v(t) \qquad v(t) \sim N(0, \sigma)$$

where $u(t)$ is the input, $y(t)$ the output, g the amplifying gain, and $v(t)$ is a noise signal with variance σ^2. This means that the signal $z(t, \theta_z)$ in the general model here corresponds to $v(t)$ and the parameters θ_G and θ_z are:

$$\begin{aligned} \theta_G &= g \\ \theta_z &= \sigma \end{aligned}$$

Then the fault-free case can for example be assumed to correspond to the fault state

$$\theta = [g \ \sigma] = [10 \ 0.01]$$

and any deviation of θ from this fault state may be considered to be a fault. ■

2.1.2 Component Fault States

Besides to separate a system model into a plant $G(\theta_G)$ and a signal $z(t, \theta_z)$, it is natural to also separate a system into a number of *components*. For each of these components, a number of faults may occur. Parts of the system that are not directly affected by any fault are not considered to be components.

Each component i has a, possibly vector-valued, parameter θ_i which determines the exact fault state (which can be no fault) of the component. Assume that there is a total number of p components. Then the fault state θ of the whole system can be written

$$\theta = [\theta_1, \ldots \theta_p]$$

The parameter space of θ_i is denoted $\mathcal{D}^i$. Then parameter space Θ becomes

$$\Theta = \mathcal{D}^1 \times \cdots \times \mathcal{D}^p$$

2.1.3 Models

As was said above, the model consists of $G(\theta_G)$ and $z(t, \theta_z)$ (with ϕ_G and ϕ_z neglected). The whole system model will be denoted $\mathcal{M}(\theta)$ and thus

$$\mathcal{M}(\theta) = \langle G(\theta_G), z(t, \theta_z) \rangle$$

The model $\mathcal{M}(\theta)$ with a fixed value of θ then exactly specifies the system when a specific fault (or no fault) is present.

Example 2.2

Consider a system described by the following equations:

$$\dot{x} = f(x,u) \tag{2.1a}$$
$$y_1 = h_1(x) + b_1 \tag{2.1b}$$
$$y_2 = h_2(x) + b_2 \tag{2.1c}$$
$$b_1 \geq 0 \tag{2.1d}$$
$$b_2 \geq 0 \tag{2.1e}$$

The constants b_1 and b_2 represents sensor bias faults and it is assumed that only positive biases can occur.

The system can be considered to have two components: sensor 1 and sensor 2. Then $\theta_1 = b_1$ and $\theta_2 = b_2$. The corresponding fault-state spaces $\mathcal{D}^1$ and $\mathcal{D}^2$ are $\mathcal{D}^1 = [0, \infty[$ and $\mathcal{D}^2 = [0, \infty[$ respectively. This means that $\theta = [\theta_1 \ \theta_2] = [b_1 \ b_2]$ and the fault-state space Θ becomes

$$\Theta = \mathcal{D}^1 \times \mathcal{D}^2 = \{[b_1 \ b_2];\ b_1 \geq 0, b_2 \geq 0\}$$

■

2.1.4 Examples of Fault Models

We will in this section give some examples of common fault modeling principles, and see how they fit into the framework of this thesis. However, in a real application one should not be limited to the examples given here, but instead always choose the fault model that is "best suited" for the particular application, e.g. in terms of performance and computing power available. In practice only the fantasy sets the limit of what fault models that can be considered.

Fault Signals

Commonly faults are modeled as unrestricted arbitrary fault signals, e.g. (Gertler, 1998)(Chen and Patton, 1999). When fault signals are used, a specific fault is usually modeled as a scalar fault signal. Fault modeling by signals is very general and can describe all types of faults. However, as we will see later in this thesis, to use fault models that are *too* general may imply that it becomes impossible to isolate different faults.

Faults that are traditionally modeled as signals, are possible to describe also in the framework described above, where faults are described by the fault state parameter. To illustrate this, consider a general nonlinear system modeled as

$$\begin{aligned} \dot{x}(t) &= g\big(x(t), u(t), f(t)\big) \\ y(t) &= h\big(x(t), u(t), f(t)\big) \end{aligned}$$

The signal $f(t)$ here represents an arbitrary fault that can for example be an actuator fault or a sensor fault. There are several possibilities to include the fault signal $f(t)$ in the general framework:

1. The fault signal is seen as a parameter of the plant, i.e. $\theta_G = f(t)$. Note that θ_G is still constant and its value is the whole signal $f(t)$. If discrete time and finite data is considered, then θ_G becomes a vector $\theta_G = [f(t_1) \dots f(t_n)]$.
2. The fault signal is seen as an unknown input and $z(t, \theta_z)$ is chosen as $z(t) = f(t)$.
3. The fault signal is seen as an unknown input $z(t, \theta_z)$ where $\theta_z = f(t)$ and then $z(t, \theta_z) = \theta_z$. Note again that θ_z is constant.
4. The fault signal is seen as an unknown input and $z(t, \theta_z)$ is chosen as $z(t) = \theta_z f(t)$. The parameter θ_z can be binary (0 or 1), indicating only the presence of the fault, or real-valued, indicating the amplitude of the fault.

Remember that we want to describe the fault situation of the system with the fault state θ and that each possible fault corresponds to a point in the fault state space Θ. These desires can be met by using the first, third, or fourth alternative above, but the not the second.

It is also possible to include some more restrictions on the fault state parameter θ. An example of a natural restriction is that the value of a fault signal $f(t)$ is limited in range. Another example is that the bandwidth of $f(t)$ is limited to some value. In general it is advantageous to include restrictions into the fault models. The reason is that the isolation task get easier the more restrictive fault models we have.

Constant Plant Parameters

Another very common fault model is to model faults as deviations of constant plant parameters from their nominal value, e.g. (Isermann, 1993). It is obvious that such faults can in the general framework be modeled by the parameter θ_G. Faults that are typically modeled in this way are "gain-errors" and "off-sets" ("biases").

Fault modeling by constant plant parameters is exemplified in Example 2.1 where the parameter g is 10 in the nominal case and a fault is represented as a deviation from this nominal value. Another example is the parameters b_1 and b_2 in Example 2.2.

Also for this fault modeling principle, it is possible to include some restrictions on the fault state parameter θ. For example the size of a bias or a gain-error is usually limited by the system.

Constant Signal Parameters

In some cases, it is appropriate to model a fault as a deviation of a constant signal parameter from its nominal value. A typical example is a signal whose variance is constant and low in the fault-free case, and when a fault is present the variance is also constant but higher. These faults can in the general framework be modeled by the parameter θ_z.

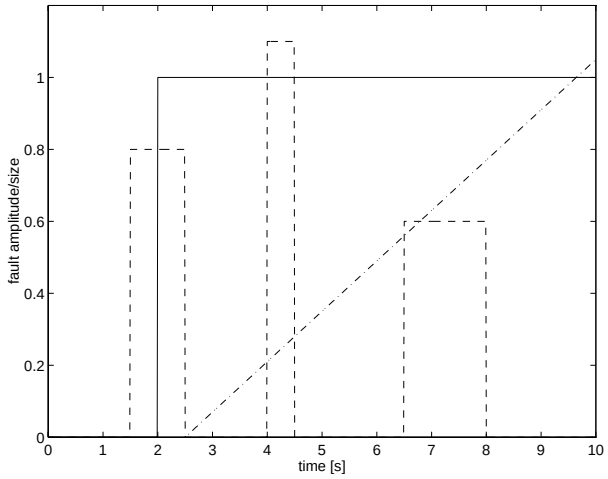

Figure 2.2: Some different types of time-variant behavior of faults.

Abrupt Changes

A quite common fault model is to consider abrupt changes of variables, e.g. see (Basseville and Nikiforov, 1993). This is illustrated in Figure 2.2 as the solid line. It is assumed that a variable or signal has a constant value θ_0 before an unknown change-time t_{ch} and then jumps to a new constant value θ_1. The parameters θ_0 and θ_1 can be unknown or known. The abrupt change model fit into the general framework by letting either θ_G or θ_z contain the three parameters θ_0, θ_1, and t_{ch}.

Example 2.3

Consider an electrical connector. One possible fault is a sudden "connection cut-off" at time t_{ch}. A model for this fault mode is

$$y_s(t) = (1 - c(t))x(t)$$

where

$$c(t) = \begin{cases} \theta_0 = 0 & t < t_{ch} \\ \theta_1 = 1 & t \geq t_{ch} \end{cases}$$

That is, the fault model is based on an abrupt change in the signal $c(t)$. Since the levels θ_0 and θ_1 are known at beforehand, this fault can be described by the single parameter t_{ch}, i.e. $\theta_G = t_{ch}$.

■

Note that the abrupt change model can also be used to model any abrupt change, and not only changes of the level of an signal. For example, we can assume that the derivative or the variance of a signal changes abruptly.

Incipient Faults

In some sense, the opposite of abrupt changes is *incipient* faults. Incipient faults are faults that gradually develops from no fault to a larger and larger fault. This is illustrated in Figure 2.2 as the dash-dotted line. An incipient fault could for example be a slow degradation of a component or developing calibration errors of a sensor. Modeling of incipient faults are exemplified in the two following examples:

Example 2.4

Let $c(t)$ represent the "size" of the fault. If the fault is incipient, then $c(t)$ becomes

$$c(t) = \begin{cases} 0 & t < t_{ch} \\ g(t - t_{ch}) & t \geq t_{ch} \end{cases}$$

Then the fault state could be $\theta = [t_{ch}\ g]$. This fault model can in fact be seen as special case of the abrupt change model. ■

Example 2.5

Consider a limited time window and assume that during this time window, either the no fault case is present or that an incipient fault has *already* started to develop, i.e. the starting-point is actually outside the range of the window. Then an appropriate fault model would be

$$c(t) = c_0 + gt$$

where t is the time *within* the window. Thus $\theta = [c_0\ g]$ and the fault free case would correspond to $\theta = [0\ 0]$. ■

Intermittent Fault

An *intermittent* fault is a fault that occurs and disappears repeatedly. This is shown in Figure 2.2 as the dashed line. A typical example of an intermittent fault is a loose connector.

Example 2.6

Consider a sensor measuring a state x. The model of this (sub-) system can be written

$$y_s(t) = c_1(t)x(t)$$

where y_s is the sensor output and x is the state. The function $c_1(t)$ is our model of the loose contact. For some t, there is no contact and therefore $c_1(t) = 0$. For other t, the contact is perfect and $c_1(t) = 1$. That is, $c_1(t)$ is a function that switches between 0 and 1 at unknown time instances. In terms of the general model description, $z(t, \theta_z)$ can be chosen as $z(t, \theta_z) = c(t)$ where the unknown time instances are collected in the vector θ_z. ■

2.2 Fault Modes

Different faults can be classified into different *fault modes*. For example, consider a system containing a water tank and leakages in the bottom of this tank. All such leakages, regardless of their area, belong to the same fault mode "water tank bottom leakage".

The classification of different faults into fault modes corresponds to a *partition* of the fault-state space Θ. This means that each fault mode γ is associated with a subset Θ_γ of Θ. One of the fault modes corresponds to the fault-free case and this fault mode will be denoted "no fault" or **NF**. Further, all sets Θ_γ are pairwise disjoint and

$$\Theta = \bigcup_{\gamma \in \Omega} \Theta_\gamma$$

where Ω is used to denote the set of all fault modes.

If fault mode γ is present in the system, then we know that $\theta \in \Theta_\gamma$. The fact that all sets Θ_γ are pairwise disjoint means that only one fault mode can be present at the same time. We will use the convention that one of the fault modes always corresponds to the no fault case.

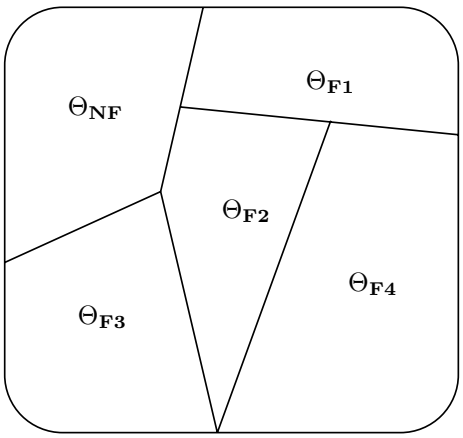

Figure 2.3: The fault state space divided into subsets corresponding to different fault modes.

For notational convenience we will to each fault mode associate an abbreviation, e.g. "no fault" was abbreviated **NF**. All this is illustrated in Figure 2.3 which shows how the whole set Θ has been divided into five subsets corresponding to fault modes **NF**, **F1**, **F2**, **F3**, and **F4**. It is now possible to formally define *fault*:

Definition 2.1 (Fault) *A fault state θ is a* fault *if $\theta \notin \Theta_{\mathbf{NF}}$.*

We have already used the term *fault* in a non strict sense and will also continue to do so in many not-so-formal parts of the thesis.

Example 2.7

Consider again Example 2.2. Four fault modes are considered:

NF	no fault
B1	bias in sensor 1
B2	bias in sensor 2
B1&**B2**	bias both sensor 1 and sensor 2

The sets Θ, $\Theta_{\mathbf{NF}}$, $\Theta_{\mathbf{B1}}$, $\Theta_{\mathbf{B2}}$, and $\Theta_{\mathbf{B1\&B2}}$ become

$$\Theta = \{[b_1\ b_2];\ b_1 \geq 0, b_2 \geq 0\} \tag{2.2a}$$
$$\Theta_{\mathbf{NF}} = \{[0\ 0]\} \tag{2.2b}$$
$$\Theta_{\mathbf{B1}} = \{[b_1\ 0];\ b_1 > 0\} \tag{2.2c}$$
$$\Theta_{\mathbf{B2}} = \{[0\ b_2];\ b_2 > 0\} \tag{2.2d}$$
$$\Theta_{\mathbf{B1\&B2}} = \{[b_1\ b_2];\ b_1 > 0, b_2 > 0\} \tag{2.2e}$$

■

The fault mode present in the system will frequently be denoted $\mathbf{F_p}$. Thus when the present fault mode is **F1**, we write this as $\mathbf{F_p} = \mathbf{F1}$. This further means for the present fault state θ it holds that $\theta \in \Theta_{\mathbf{F1}}$.

2.2.1 Component Fault-Modes

Besides defining fault modes for the whole system, it is natural to also consider *component fault-modes*. To emphasize the difference between component fault-modes and fault modes for the whole system, the latter will sometimes be called *system fault-modes*.

As was said in Section 2.1.2, a system can usually be separated into a number of components. The characteristic property of a component is that only one type of fault can be present at a time. The classification into different types of faults is made by introducing *component fault-modes*. Consider for example a valve with fault modes "no fault", "stuck open", and "stuck closed". Obviously no two of these fault modes can be present at the same time. In analogy with the system fault-modes, we use the convention that one of the component fault-modes is the no fault case.

Each component fault-mode ψ is associated with a subset $\mathcal{D}^i_\psi$ of $\mathcal{D}^i$. That is, if fault mode ψ is present in component i, then $\theta_i \in \mathcal{D}^i_\psi$. In analogy with the system fault-modes, the sets $\mathcal{D}^i_\psi$ form a partition of the component fault-state space $\mathcal{D}^i$. This means that the sets $\mathcal{D}^i_\psi$ are pairwise disjoint and

$$\mathcal{D}^i = \bigcup_{\psi \in \Omega_i} \mathcal{D}^i_\psi$$

where Ω_i is the set of all component fault-modes for component i.

Relation to System Fault-Modes

Let F_j^i denote the j:th component fault-mode of the i:th component. We will reserve the fault-mode F_0^i to be the "no fault" case of the i:th component. The fault-mode F_0^i will also be denoted NF^i. Let p be the number of components and n_i the number of different component fault-modes for the i:th component. All component fault-modes can then be collected in a table:

component number i	component fault-modes
1	$F_0^1 \equiv NF^1$, F_1^1, ... $F_{n_1}^1$
2	$F_0^2 \equiv NF^2$, F_1^2, ... $F_{n_2}^2$
$\vdots$	$\vdots$
p	$F_0^p \equiv NF^p$, F_1^p, ... $F_{n_p}^p$

A system fault-mode can then be composed by a vector of component fault-modes. Thus the length of this vector is p and the total number of possible system fault-modes is

$$\prod_{i=1}^{p} n_i \tag{2.3}$$

To distinguish between system fault-modes and component fault-modes, we have here used bold-face letters to denote system fault-modes. However, when it is clear from the context, we will later in the thesis often skip the bold-face notation. Some examples of system fault-modes are

$$\mathbf{NF} = [NF^1, NF^2, \dots NF^p] \tag{2.4a}$$
$$\mathbf{F_1^1} = [F_1^1, NF^2, \dots NF^p] \tag{2.4b}$$
$$\mathbf{F_1^2} = [NF^1, F_1^2, NF^3, \dots NF^p] \tag{2.4c}$$
$$\mathbf{F_2^1}\&\mathbf{F_1^2} = [F_2^1, F_1^2, NF^3, \dots NF^p] \tag{2.4d}$$

The first of these examples is the no-fault case of the whole system. For the other examples, we have used the convention that components, that have none of its component fault-modes included in the notation for the system fault-mode, are assumed to have component fault-mode NF^i. This means that from only the notation of the system fault-modes and the sets $\mathcal{D}_\psi^i$, we are able to uniquely infer the sets Θ_γ. For the examples (2.4) we have

$\mathbf{NF}$ $\quad \theta \in \Theta_{\mathbf{NF}} = \{\theta \in \Theta | \bigwedge_i \theta_i \in \mathcal{D}_{NF^i}^i\}$
$\mathbf{F_1^1}$ $\quad \theta \in \Theta_{\mathbf{F_1^1}} = \{\theta \in \Theta | \theta_1 \in \mathcal{D}_{F_1^1}^1 \bigwedge_{i \neq 1} \theta_i \in \mathcal{D}_{NF^i}^i\}$
$\mathbf{F_1^2}$ $\quad \theta \in \Theta_{\mathbf{F_1^2}} = \{\theta \in \Theta | \theta_2 \in \mathcal{D}_{F_1^2}^2 \bigwedge_{i \neq 2} \theta_i \in \mathcal{D}_{NF^i}^i\}$
$\mathbf{F_2^1}\&\mathbf{F_1^2}$ $\quad \theta \in \Theta_{\mathbf{F_2^1}\&\mathbf{F_1^2}} = \{\theta \in \Theta | \theta_1 \in \mathcal{D}_{F_2^1}^1 \wedge \theta_2 \in \mathcal{D}_{F_1^1}^2 \bigwedge_{i \neq 1} \theta_i \in \mathcal{D}_{NF^i}^i\}$

To clarify the relation between system fault-modes and component fault-modes, it may be useful to study a Venn diagram over the different fault modes of a system. This is illustrated in the following example.

Example 2.8

Consider again Example 2.7. Four component fault-modes are considered, i.e. $NF1$, $NF2$, $B1$, and $B2$, and they are defined by the sets $\mathcal{D}^i_\psi$ as follows:

$$\begin{aligned}\mathcal{D}^1_{NF1} &= \{0\}\\ \mathcal{D}^1_{B1} &= \{x > 0\}\\ \mathcal{D}^2_{NF2} &= \{0\}\\ \mathcal{D}^2_{B2} &= \{x > 0\}\end{aligned}$$

The sets Ω_i of component fault-modes implies that there are four possible system fault-modes:

$$\begin{aligned}\mathbf{NF} &= [NF1, NF2]\\ \mathbf{B1} &= [B1, NF2]\\ \mathbf{B2} &= [NF1, B2]\\ \mathbf{B1}\&\mathbf{B2} &= [B1, B2]\end{aligned}$$

The fault-state space and the different fault modes are shown in a Venn diagram in Figure 2.4. The whole area corresponds to the set Θ. The left circle represents all fault-states for which component fault-mode $B1$ is present, i.e. the set

$$\{\theta \mid \theta_1 \in \mathcal{D}^1_{B1}\}$$

Similarly the right circle represents all fault-states for which component fault-mode $B2$ is present. These two circles together divides the fault-state space into the four sets $\Theta_{\mathbf{NF}}$, $\Theta_{\mathbf{B1}}$, $\Theta_{\mathbf{B2}}$, and $\Theta_{\mathbf{B1\&B2}}$, which are shown in the figure. ■

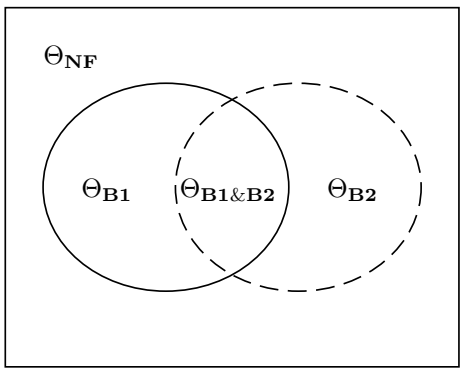

Figure 2.4: A Venn diagram showing the relation between the component and system fault-modes.

2.2.2 Single- and Multiple Fault-Modes

The system fault-modes in which only *one* of the component fault-modes is not NF^i are said to be *single fault-modes*. For example, **B1** and **B2** in the example above, are both single fault-modes. Usually also the no-fault system fault-mode, i.e. **NF**, is said to be a single fault-mode. The opposite are *multiple fault-modes* where more than one of the component fault-modes are not NF^i.

The terminology *single faults* and *multiple faults* are frequently used in the diagnosis literature. In the framework presented here, a fault θ is a *single fault* if it belongs to a single fault-mode, i.e. $\theta \in \Theta_\gamma$ and γ is a single fault mode. Similarly a fault θ is a *multiple fault* if it belongs to a multiple fault-mode. Note that with the formalism described here, multiple fault-modes comes in naturally and requires no special treatment.

A problem with considering multiple fault-modes is that the complexity of the diagnosis problem increases. When the number of components gets larger, the number of different system fault-modes grows exponentially, see (2.3). This further implies that a more complex and more expensive diagnosis system is needed. A solution is to consider only single fault-modes. This corresponds to an assumption that only one fault can be present at the same time. In that case, the number of system fault-modes grows linearly with the number of components, i.e. the number of possible system fault-modes becomes

$$1 + \sum_{i=1}^{p} (n_i - 1) \tag{2.5}$$

The assumption to only consider single fault-modes may seem to be unrealistic at first, but at least three practical considerations support this assumption.

- If a sufficiently small time scale is chosen it is probably the case that one fault has occured first even though several faults are present.
- In a system in which one fault is highly improbable (as it usually is), it is even more improbable that two or more faults occur.
- The specifications of a diagnosis system only require diagnosis of single faults. The reason can be that diagnosis systems capable of handling multiple fault modes would become to expensive because of increased sensor and hardware costs. In fact, the current diagnosis legislative regulations for automotive engines only require single fault diagnosis.

An alternative to only consider single fault-modes, but still not all multiple fault-modes, is to consider a subset of the multiple fault-modes. For example, one could choose to consider all system fault-modes where at maximum two component faults are present.

2.2.3 Models

Remember the system model $\mathcal{M}(\theta)$ that is capable of describing the system for all possible fault states $\theta \in \Theta$. By restricting θ to a subset Θ_γ, corresponding

to a fault mode γ, we get a "smaller" model. For especially single fault-modes, the models can get much smaller. To each fault mode γ, we can then associate a model $\mathcal{M}_\gamma(\theta)$ which we formally define as

$$\mathcal{M}_\gamma(\theta) = \mathcal{M}(\theta)|_{\theta\in\Theta_\gamma} \tag{2.6}$$

Thus the model $\mathcal{M}_\gamma(\theta)$ is capable of describing the system as long as fault mode γ is present.

For a specific fault mode γ, the constraint $\theta \in \Theta_\gamma$ usually fix a part of the vector θ to some constants. Then, as an alternative to the notation $\mathcal{M}_\gamma(\theta)$, we will use $\mathcal{M}_\gamma(\theta_\gamma)$, where θ_γ is the part of the θ-vector that is not fixed. If the θ-vector is completely fixed by the fault mode γ, the θ-argument becomes unnecessary and the corresponding fault model can be denoted $\mathcal{M}_\gamma$.

Example 2.9

The models corresponding to each fault mode are given by (2.1) and some additional constraints on b_1 and b_2 defined by (2.2). The models associated with the different fault modes are

NF:	$\mathcal{M}_{\mathbf{NF}}(\theta) = \mathcal{M}_{\mathbf{NF}}$
B1:	$\mathcal{M}_{\mathbf{B1}}(\theta) = \mathcal{M}_{\mathbf{B1}}(b_1)$
B2:	$\mathcal{M}_{\mathbf{B2}}(\theta) = \mathcal{M}_{\mathbf{B2}}(b_2)$
B1&**B2**:	$\mathcal{M}_{\mathbf{B1\&B2}}(\theta) = \mathcal{M}_{\mathbf{B1\&B2}}([b_1\ b_2])$

■

Note: As a reference, this sensor-bias example, that has been step-wise expanded in this and the previous section, is summarized in Appendix 2.A.

2.3 Diagnosis Systems

To perform fault diagnosis, a *diagnosis system* is needed. The general structure of an application including a diagnosis system is shown in Figure 2.5. Inputs to the diagnosis system are the signals $u(t)$ and $y(t)$, which are equal to, or a superset of, the control system signals. Except for control signals, the plant is also affected by faults and disturbances and these are not known to the diagnosis system. The task of the diagnosis system is to generate a *diagnosis statement* S, which contains information about which fault modes that can explain the behavior of the process. Note that it is assumed that the diagnosis system is *passive*, i.e. it can by no means affect the plant.

In terms of *decision theory* (e.g. see (Berger, 1985)), the diagnosis system is a *decision rule* $\delta(x)$, where $x = [u\ y]$, and S is the *action*. That is, the diagnosis system is a function of u and y and $S = \delta(x) = \delta([u\ y])$. Note that x can also contain several samples of u and y from different times.

One way of structuring a diagnosis system is shown in Figure 2.6. The whole diagnosis system $\delta(x)$ can be divided into smaller parts $\delta_i(x)$, which we

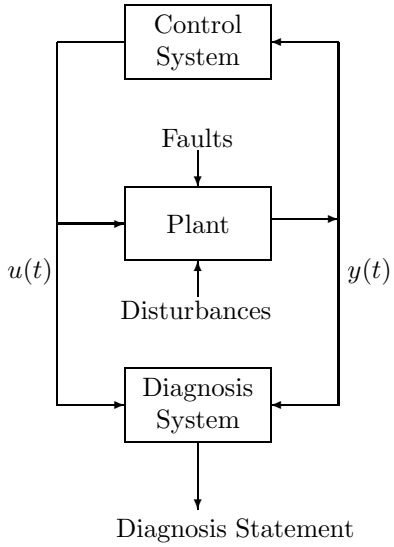

Figure 2.5: General structure of a diagnosis application.

will call *tests*. These tests are also decision rules. Assume that each of the tests $\delta_i(x)$ generates the diagnosis statement S_i, i.e. $S_i = \delta_i(x)$. The purpose of the *decision logic* is then to combine this information to form the diagnosis statement S.

The diagnosis statement S and the individual diagnosis statements S_i do all contain information about which system fault-modes that can explain the behavior of the system. We can represent and reason about this information in at least two ways. The first is to use a representation where the diagnosis statements S and S_i are sets of system fault modes. The second is to let the diagnosis statements be expressed as propositional logic formulas where the propositional symbols are component fault-modes. In the next two sections, these two alternatives will be investigated.

2.3.1 Forming the Diagnosis Statement by Using a Set Representation

An example of a diagnosis statement, represented by a set of system fault-modes, is

$$S = \{\mathbf{B1}, \mathbf{B2}\}$$

The interpretation here is that each of the fault modes **B1** and **B2**, can alone explain the behavior of the system. This can also be expressed as that each of the models $\mathcal{M}_{\mathbf{B1}}(\theta)$ and $\mathcal{M}_{\mathbf{B2}}(\theta)$ can explain the measured data x.

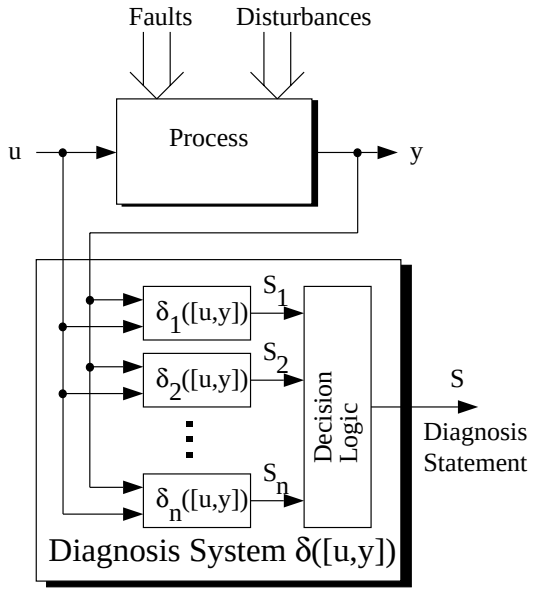

Figure 2.6: A general diagnosis system.

All individual diagnosis statements S_i contain information of which system fault-modes that can explain the data. To derive the diagnosis statement S, we want to summarize the information from all the individual diagnosis statements S_i. By using the set representation, this is done via an intersection operation, i.e. the diagnosis statement S is formed as

$$S = \bigcap_i S_i \tag{2.7}$$

Thus the decision logic of the diagnosis system can be seen as a simple intersection operation.

The following example illustrates this principle.

Example 2.10

Consider the system fault-modes **NF**, **B1**, **B2**, and **B1**&**B2**. Assume that the diagnosis system contains three individual tests. Assume further that the diagnosis system has collected and processed the input data, and the individual diagnosis statements S_i are

$$\begin{aligned} S_1 &= \{\mathbf{NF}, \mathbf{B1}\} \\ S_2 &= \{\mathbf{B1}, \mathbf{B1}\&\mathbf{B2}\} \\ S_3 &= \{\mathbf{B1}, \mathbf{B2}\} \end{aligned}$$

Then the diagnosis statement S becomes

$$S = \{\mathbf{NF}, \mathbf{B1}\} \cap \{\mathbf{B1}, \mathbf{B1}\&\mathbf{B2}\} \cap \{\mathbf{B1}, \mathbf{B2}\} = \{\mathbf{B1}\}$$

The result should be interpreted as **B1** is the only system fault-mode that can explain the behavior of the system. ■

In the example above, it happened that S only contained *one* system fault-mode. It can also happen that S contains several system fault-modes. If for example the individual diagnosis statements S_i are

$$\begin{aligned} S_1 =&\{\mathbf{NF}, \mathbf{B1}, \mathbf{B1}\&\mathbf{B2}\} \\ S_2 =&\{\mathbf{B1}, \mathbf{B1}\&\mathbf{B2}\} \\ S_3 =&\{\mathbf{B1}, \mathbf{B2}, \mathbf{B1}\&\mathbf{B2}\} \end{aligned}$$

Then the diagnosis statement S becomes

$$\begin{aligned} S =&\{\mathbf{NF}, \mathbf{B1}, \mathbf{B1}\&\mathbf{B2}\} \cap \{\mathbf{B1}, \mathbf{B1}\&\mathbf{B2}\} \cap \{\mathbf{B1}, \mathbf{B2}, \mathbf{B1}\&\mathbf{B2}\} = \\ =&\{\mathbf{B1}, \mathbf{B1}\&\mathbf{B2}\} \end{aligned}$$

This diagnosis statement should be interpreted as *both* the system fault-modes **B1** and **B1**&**B2** can explain the behavior of the system.

One special case is when the fault mode **NF** (no fault) is contained in the diagnosis statement. For example

$$S = \{\mathbf{NF}, \mathbf{B1}, \mathbf{B2}, \mathbf{B1}\&\mathbf{B2}\}$$

This means that the system fault-mode **NF** (and also some other system fault-modes) can explain the behavior of the system. Further this corresponds to that the fault free model $\mathcal{M}_{\mathbf{NF}}$ can explain the behavior of the system. In this case there is no reason to generate an alarm. On the other hand if the fault mode **NF** is not contained in the diagnosis statement S, some faults are probably present and an alarm should be generated.

The set representation of diagnosis statements will be used a lot in this thesis. One reason is that it is easy and intuitive to express that a system fault-mode γ is part of the diagnosis statement S. This is written $\gamma \in S$. For example the principle of when to generate an alarm can be expressed as

$\mathbf{NF} \in S$	NOT generate an alarm
$\mathbf{NF} \notin S$	generate an alarm

The diagnosis-system architecture presented here is based on the same principle as human beings are using when performing diagnosis. That is, a human being breaks down a complex diagnosis problem into smaller tasks (the tests). These smaller tasks are performed (can be to observe a special characteristic) and the outcome from all of them are combined to form the total diagnosis statement. This connection to human reasoning will be even more detailed in the next chapter in which the individual tests are seen as hypothesis tests.

Below follows a larger example, similar to one given in (Sandewall, 1991), of a diagnosis problem and a diagnosis system. In addition to illustrating general principles, also the connection to human reasoning will hopefully be realized.

Remember the symbol Ω which denotes the set of all system fault-modes. If a diagnosis statement is Ω, then this means that any fault mode can explain the system behavior.

Example 2.11

Assume that we want to diagnose a car. The following system fault-modes are considered:

NF	no fault
BD	battery discharged
SB	start motor broken
NG	no gasoline

Remember that only one of these fault modes can occur at the same time. An automated diagnosis system or a human being can perform the following tests:

δ_1: When the ignition key is turned on, observe if the start motor starts. The different conclusions are then

test is not performed	$S_1 = \Omega$
start motor starts	$S_1 = \{\mathbf{NF}, \mathbf{NG}\}$
start motor do not start	$S_1 = \{\mathbf{BD}, \mathbf{SB}\}$

The conclusion "test is not performed" means that the ignition key has not been turned on.

δ_2: When the ignition key is turned on, observe if the engine starts. The different conclusions are then

test is not performed	$S_2 = \Omega$
engine starts	$S_2 = \{\mathbf{NF}\}$
engine do not start	$S_2 = \{\mathbf{BD}, \mathbf{SB}, \mathbf{NG}\}$

δ_3: When the head-light switch is turned on, observe if the head-lights are turned on. The different conclusions are then

test is not performed	$S_3 = \Omega$
head-lights are turned on	$S_3 = \{\mathbf{NF}, \mathbf{SB}, \mathbf{NG}\}$
head-lights are not turned on	$S_3 = \{\mathbf{BD}\}$

Now assume that both the ignition key and the head-light switch are turned on and the following observations are made:

- start motor do not start
- engine do not start
- head-lights are turned on

This means that the diagnosis statement S becomes

$$S = S_1 \cap S_2 \cap S_3 = \{\mathbf{BD}, \mathbf{SB}\} \cap \{\mathbf{BD}, \mathbf{SB}, \mathbf{NG}\} \cap \{\mathbf{NF}, \mathbf{SB}, \mathbf{NG}\} = \{\mathbf{SB}\}$$

That is, the conclusion is that the only fault mode that can explain the behavior of the system is **SB**, i.e. start motor broken. ■

2.3.2 Forming the Diagnosis Statement by Using a Propositional Logic Representation

We will now investigate the case where the diagnosis statements S_i and S are expressed as propositional logic formulas, and the propositional symbols are component fault-modes. Note first that this representation is equivalent to using sets of system fault-modes. The diagnosis statement S is with this representation formed as

$$S = \bigwedge_i S_i$$

Thus the decision logic can be seen as a simple conjunction operation.

As noted in Section 2.2.2, a representation based on system fault-modes can be problematic since the number of system fault-modes grows exponentially with the number of components. The reasoning based on propositional logic and component fault-modes, does not have this problem. An additional advantage with this representation is that we obtain a closer connection to other diagnosis methods based on logic, e.g. (Reiter, 1987). It can also be argued that a representation based on component fault-modes is more natural.

The next example will illustrate reasoning based on propositional logic and component fault-modes. Also shown is the link to the equivalent representation based on sets of system fault-modes. In the example, we have assumed that each component has only two possible component fault-modes. In this case, standard "two-valued" propositional logic can be used. If some components have more than two possible component fault-modes, than some "multi-valued" propositional logic[2] must be used, e.g. see (Larsson, 1997).

Example 2.12

Assume that we want to diagnose the same car as in the previous example. Now we will consider multiple faults and it is natural to start defining the component fault-modes:

component name	component fault-modes
battery	NF^B, BD
start motor	NF^S, SB
gasoline	NF^G, NG

The abbreviations have the same meaning as in the previous example. This means that the set of all system fault-modes become:

$$\Omega = \{\mathbf{NF}, \mathbf{BD}, \mathbf{SB}, \mathbf{NG}, \mathbf{BD}\&\mathbf{SB}, \mathbf{BD}\&\mathbf{NG}, \mathbf{SB}\&\mathbf{NG}, \mathbf{BD}\&\mathbf{SB}\&\mathbf{NG}\}$$

We can now proceed as we did in Example 2.11, but instead we will choose to use a reasoning based on propositional logic and component fault-modes. Instead

[2] If such a multi-valued logic is adopted, we could in principle also use propositional logic to reason about system fault-modes, i.e. as an alternative to the set representation.

of for example NF^B we will write $\neg BD$. The symbol $\perp$ will be used to denote *falsity*. Then the three tests can be formulated as follows:

δ_1: When the ignition key is turned on, observe if the start motor starts. The different conclusions are then

test is not performed	$S_1 = \neg \perp$
start motor starts	$S_1 = \neg BD \wedge \neg SB$
start motor do not start	$S_1 = BD \vee SB$

Note that S_1 is now expressed with component fault-modes which is significantly different compared to the previous example where system fault-modes were used. For example, the last alternative conclusion of test δ_1, expressed by system fault-modes and the set representation, is

$$S_1 = \{\mathbf{BD}, \mathbf{SB}, \mathbf{BD}\&\mathbf{SB}, \mathbf{BD}\&\mathbf{NG}, \mathbf{SB}\&\mathbf{NG}, \mathbf{BD}\&\mathbf{SB}\&\mathbf{NG}\}$$

δ_2: When the ignition key is turned on, observe if the engine starts. The different conclusions are then

test is not performed	$S_2 = \neg \perp$
engine starts	$S_2 = \neg BD \wedge \neg SB \wedge \neg NG$
engine do not start	$S_2 = BD \vee SB \vee NG$

δ_3: When the head-light switch is turned on, observe if the head-lights are turned on. The different conclusions are then

test is not performed	$S_3 = \neg \perp$
head-lights are turned on	$S_3 = \neg BD$
head-lights are not turned on	$S_3 = BD$

Now assume that both the ignition key and the head-light switch are turned on and the following observations are made:

- start motor do not start
- engine do not start
- head-lights are turned on

This means that the diagnosis statement S becomes

$$\begin{aligned} S =& S_1 \wedge S_2 \wedge S_3 = (BD \vee SB) \wedge (BD \vee SB \vee NG) \wedge \neg BD = \\ =& \neg BD \wedge SB \end{aligned}$$

That is, the conclusion is that the behavior of the system corresponds to that the component fault-modes $\neg BD$ and SB are present. That is, the battery is not discharged *and* the start motor is broken. If we instead had used reasoning about the system fault-modes, the diagnosis statement would become

$$S = \{\mathbf{SB}, \mathbf{SB}\&\mathbf{NG}\}$$

■

Remark: In the above example, we considered multiple fault modes, in contrast to Example 2.10, in which only single fault-modes were used. If we want consider only single fault-modes, also when using reasoning based on components and propositional logic, we have to add a set of *premises* saying that no two component faults can be present at the same time, e.g. $\neg(BD \wedge SB)$. Such premises are not needed when the reasoning is based on system-fault modes and the set representation. That is, multiple fault-modes could have been introduced in Example 2.10, without any special considerations.

2.3.3 Speculative and Conclusive Diagnosis-Systems

As have been said above, a diagnosis statement S can in general contain more than one system fault-mode. This is in contrast to most fault diagnosis literature, in which the diagnosis statement can only contain *one* system fault-mode. The difference is fundamental and to distinguish between the two types of diagnosis system, we will use the terms *conclusive diagnosis-system* and *speculative diagnosis-system*.

A speculative diagnosis-system corresponds well to a desired functionality since in cases where it is difficult or even impossible to decide which fault mode that is present, it is very useful for a service technician to get to know that there are more than one fault mode that can explain the behavior of the process. If the diagnosis system was forced to pick out one fault mode in cases like this, it is highly probable that a mistake is made and wrong fault mode is picked out.

The diagnosis task of a conclusive diagnosis-system is to infer which one, of several fault scenarios (fault modes), that is present. On the other hand, the diagnosis task of a speculating diagnosis-system is to speculate which fault scenarios (possibly several) that *can* be present such that the collected data can explain the behavior of the system.

Formally, the conclusive diagnosis-system is a special case of a speculative diagnosis-system with the additional restriction that no matter the outcome of the different tests δ_i, the diagnosis statement S does always contain maximally one system fault-mode.

2.3.4 Formal Definitions

Now when faults, fault modes, and diagnosis systems have been formally defined, we are ready to introduce more formal definitions to the conceptually important terms in the SAFEPROCESS list from Section 1.2. These definitions are valid for the speculative as well as the conclusive diagnosis system.

Definition 2.2 (Fault Detection) Fault Detection *is the task to determine if the system fault-mode* **NF** *can explain the behavior of the system or not.*

Definition 2.3 (Fault Isolation) Fault isolation *is the task to determine which system fault-mode that can best explain the behavior of the system.*

Definition 2.4 (Generalized Fault Isolation) Fault isolation *is the task to determine which system fault-modes that can explain the behavior of the system.*

Definition 2.5 (Fault Identification) Fault identification *is the task to estimate the fault state θ that can best explain the behavior of the system.*

Now we define *fault diagnosis* as equivalent to the generalized fault isolation:

Definition 2.6 (Fault Diagnosis) Fault diagnosis *is the task to determine which fault modes that can explain the behavior of the system.*

Note that this definition of fault diagnosis is not in agreement with many other sources which define diagnosis as the combined task fault detection, fault isolation, and fault identification, e.g. compare with the definitions in Section 1.2. However, as we will see in Chapter 4, it can happen that fault identification must be implicitly performed when doing fault isolation.

Note also the difference between fault diagnosis and general system identification in which the single θ, that best explains the data, is sought. To use system identification directly to perform both fault detection, isolation, and identification, would in some cases theoretically be possible. However, for most cases the problem is that the vector θ is usually quite large and the identification therefore becomes difficult. In addition, it can very well be the case that the model is not *identifiable* with respect to θ. However, if one fault mode is assumed, the fault identification becomes much simpler and this is the motivation why we need to perform fault isolation before fault identification.

2.4 Relations Between Fault Modes

Because of for instance "over parameterization", it can happen that two different fault modes can describe the system behavior equally well. Consider for example a system modeled as

$$y = abu$$

where one fault mode $\mathbf{F_a}$ corresponds to that $a \neq 1$ and fault mode $\mathbf{F_b}$ corresponds to that $b \neq 1$. It is obvious that both $\mathbf{F_a}$ and $\mathbf{F_b}$ can equally well describe the system.

These kinds of relations between $\mathbf{F_a}$ and $\mathbf{F_b}$ are further investigated in this section. We will see later that for both analysis and design of a diagnosis system, these relations play a fundamental role. There is also a close relation to identifiability in system identification, e.g. (Ljung, 1987).

First a notion of *equivalent models* is established:

Definition 2.7 (Equivalent Models) *Two models $\mathcal{M}_1(\theta_1)$ and $\mathcal{M}_2(\theta_2)$, with fixed parameters θ_1 and θ_2 are equivalent, i.e.*

$$\mathcal{M}_1(\theta_1) = \mathcal{M}_2(\theta_2)$$

if for each initial state x_1 of $\mathcal{M}_1(\theta_1)$, there is an initial state x_2 of $\mathcal{M}_2(\theta_2)$ such that for all signals $u(t)$ and $z(t)$, the outputs $y_1(t)$ and $y_2(t)$ are equal, and vice versa.

Definition 2.8 (Submode) *We say that a fault mode γ_1 is a* submode *of another fault mode γ_2, i.e.*

$$\gamma_1 \preccurlyeq \gamma_2$$

if for each fixed value $\theta_1 \in \Theta_{\gamma_1}$, there is a fixed value $\theta_2 \in \Theta_{\gamma_2}$ such that $\mathcal{M}_{\gamma_1}(\theta_1) = \mathcal{M}_{\gamma_2}(\theta_2)$.

Definition 2.9 (Submode in the Limit) *We say that a fault mode γ_1 is a* submode in the limit *of another fault mode γ_2, i.e.*

$$\gamma_1 \preccurlyeq^* \gamma_2$$

if for each fixed value $\theta_1 \in \Theta_{\gamma_1}$, there is a fixed value θ^* such that

$$\mathcal{M}_{\gamma_1}(\theta_1) = \lim_{\substack{\theta_2 \to \theta^* \\ \theta_2 \in \Theta_{\gamma_2}}} \mathcal{M}_{\gamma_2}(\theta_2)$$

These relations are transitive which means that if $\gamma_1 \preccurlyeq \gamma_2$ and $\gamma_2 \preccurlyeq \gamma_3$, then $\gamma_1 \preccurlyeq \gamma_3$. Further if $\gamma_1 \preccurlyeq^* \gamma_2$ and $\gamma_2 \preccurlyeq^* \gamma_3$, then $\gamma_1 \preccurlyeq^* \gamma_3$ (at least under regularity conditions). Further we have that if $\gamma_1 \preccurlyeq \gamma_2$ then also $\gamma_1 \preccurlyeq^* \gamma_2$.

The submode relation between fault modes can quite easily arise when modeling systems and faults. Unfortunately they are undesirable since they, as we will see in the Section 2.5, imply that it becomes difficult or impossible to isolate different faults. Examples of how the submode relation can arise is given in the following example.

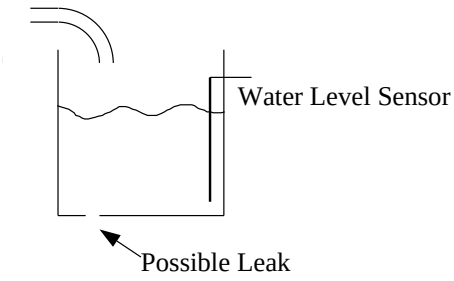

Figure 2.7: A water tank.

Example 2.13

Consider the water tank illustrated in Figure 2.7. Two types of faults can occur: there may be a leakage and the water-level sensor may fail. The diameter of

the leakage hole is assumed to be unknown but constant. For some reason, it is interesting to distinguish between three types of sensor faults: a simple calibration fault (i.e. a gain fault), a combination of a bias and a calibration fault, and an arbitrary fault. The component fault-modes can therefore be summarized as

component number i	component name	component fault-modes
1	Level Sensor	$NF1$, SCF, LSF, ASF
2	Tank	$NF2$, L

where L is "Leakage", SCF is "Sensor Calibration Fault", LSF is "Linear Sensor Fault", and ASF is "Arbitrary Sensor Fault". Thus the possible system fault-modes are

$$\begin{aligned}
\mathbf{NF} &= [NF1, NF2] \\
\mathbf{L} &= [L, NF2] \\
\mathbf{SCF} &= [NF1, SCF] \\
\mathbf{LSF} &= [NF1, LSF] \\
\mathbf{ASF} &= [NF1, ASF] \\
\mathbf{L\&SCF} &= [L, SCF] \\
\mathbf{L\&LSF} &= [L, LSF] \\
\mathbf{L\&ASF} &= [L, ASF]
\end{aligned}$$

The fault-complete model $\mathcal{M}_\Omega(\theta)$ of the tank is

$$\begin{aligned}
\dot{x}(t) &= u(t) - Ah(x(t)) \\
y(t) &= gx(t) + m + f(t)
\end{aligned}$$

where the state $x(t)$ is the water level, and $u(t)$ is the flow into the tank. The leakage flow is determined by the leakage area A times the nonlinear function $h(x(t))$. The sensor signal $y(t)$ is affected by different faults via the constants k and m, and the signal $f(t)$. The parameter vector θ becomes $\theta = [A, g, m, f(t)]$, and $\theta_1 = A$ and $\theta_2 = [g, m, f(t)]$.

The single fault-modes are defined by the following models:

$$\begin{aligned}
\mathcal{M}_{\mathbf{NF}} &= \{\mathcal{M}(\theta) \mid A = 0 \wedge g = 1 \wedge m = 0 \wedge \forall t. f(t) = 0\} \\
\mathcal{M}_{\mathbf{L}}(A) &= \{\mathcal{M}(\theta) \mid A > 0 \wedge g = 1 \wedge m = 0 \wedge \forall t. f(t) = 0\} \\
\mathcal{M}_{\mathbf{SCF}}(g) &= \{\mathcal{M}(\theta) \mid A = 0 \wedge g \neq 1 \wedge m = 0 \wedge \forall t. f(t) = 0\} \\
\mathcal{M}_{\mathbf{LSF}}([g\ m]) &= \{\mathcal{M}(\theta) \mid A = 0 \wedge (g \neq 0 \vee m \neq 0) \wedge \forall t. f(t) = 0\} \\
\mathcal{M}_{\mathbf{ASF}}(f(t)) &= \{\mathcal{M}(\theta) \mid A = 0 \wedge g = 1 \wedge m = 0 \wedge f(t) \neq 0\}
\end{aligned}$$

From these models it is easy to also derive the models for the multiple fault-modes.

When we have the models for all system fault-modes, we can identify the following relations:

$$\mathbf{NF} \preccurlyeq^* \mathbf{L}$$
$$\mathbf{NF} \preccurlyeq^* \mathbf{SCF} \preccurlyeq \mathbf{LSF} \preccurlyeq \mathbf{ASF}$$
$$\mathbf{NF} \preccurlyeq^* \mathbf{L\&SCF} \preccurlyeq \mathbf{L\&LSF} \preccurlyeq \mathbf{L\&ASF}$$

Even though most of these relations can be avoided, it is usually very difficult to avoid that **NF** is a submode of most other fault modes. ■

2.5 Isolability and Detectability

In this section we define and discuss isolability and detectability. The diagnosis statement is assumed to be expressed using the set representation. From now on, we skip the bold-face notation for system fault-modes. We start by defining what is meant by detecting and isolating a fault.

Definition 2.10 (Detected Fault) *Assume a fault $\theta \in \Theta_{F_1}$ is present. Then the fault θ is* detected *using a diagnosis system $\delta(x)$, if $NF \notin S$.*

Definition 2.11 (Isolated Fault) *Assume a fault $\theta \in \Theta_{F_1}$ is present. Then the fault θ is* isolated *using a diagnosis system $\delta(x)$, if $S = \{F_1\}$.*

Note that Definition 2.11 means that fault isolation implies fault detection.

Related to the above definitions, we also define the terms *false alarm*, *missed detection*, *missed isolation*:

Definition 2.12 (False Alarm) *Assume that no faults are present, i.e. $\theta \in \Theta_{NF}$. Then the diagnosis statement S represents a* false alarm *if $NF \notin S$.*

Definition 2.13 (Missed Detection) *Assume that a fault $\theta \in \Theta_{F_1}$ is present. Then the diagnosis statement S represents a* missed detection *if $NF \in S$.*

Definition 2.14 (Missed Isolation) *Assume that a fault $\theta \in \Theta_{F_1}$ is present. Then the diagnosis statement S represents a* missed isolation *if $S \neq \{F_1\}$.*

Next we define isolability and detectability for a given diagnosis system. We restrict the definitions to deterministic systems. This means that the system output y is completely determined by initial conditions x_0, the input u, faults θ, and disturbances ϕ. This further means that $S = \delta([y, u]) = \delta([y(x_0, u, \phi, \theta), u])$, i.e. also the diagnosis statement is deterministically determined by $[x_0, u, \phi]$ and θ. However, it is possible to generalize the definitions to the stochastic case.

The goal is to define what we mean when saying "the fault mode F_1 is isolable from the fault mode F_2" but we start with a simpler problem, namely what we mean by "the fault-state θ_1 is isolable from the fault-state θ_2":

Definition 2.15
(Fault-State Isolability Under $[x_0, u, \phi]$ in a Diagnosis System)
Given a fixed $[x_0, u, \phi]$ and a diagnosis system δ, we say that the fault state $\theta_1 \in \Theta_{F_1}$ is isolable from $\theta_2 \in \Theta_{F_2}$ under $[x_0, u, \phi]$ *if*

$$F_1 \in S = \delta\big(y(x_0, u, \phi, \theta_1), u\big) \wedge F_2 \notin S = \delta\big(y(x_0, u, \phi, \theta_1), u\big)$$

and

$$F_2 \in S = \delta\big(y(x_0, u, \phi, \theta_2), u\big)$$

Note that the definition is not symmetric, i.e. a fault θ_1 can be isolable from θ_2, without that θ_2 is isolable from θ_1.

Next, when defining what we mean by "F_1 is isolable from F_2", we have several choices. We can consider a single pair of fault states or all fault states in the fault modes. We can consider a single $[x_0, u, \phi]$, given or not given, or all possible $[x_0, u, \phi]$. All together, we end up with no less than six different definitions of fault isolability for a given diagnosis system. These six are illustrated in Table 2.1.

	Complete Isolability $\forall\theta$ $\Rightarrow$	Partial Isolability $\exists\theta$
Uniform Isolability $\forall[x_0, u, \phi]$ $\Downarrow$	F_1 is uniformly and completely isolable from F_2	F_1 is uniformly and partially isolable from F_2
Under $[x_0, u, \phi]$ $\Downarrow$	F_1 is completely isolable from F_2 under $[x_0, u, \phi]$	F_1 is partially isolable from F_2 under $[x_0, u, \phi]$
$\exists[x_0, u, \phi]$	F_1 is completely isolable from F_2	F_1 is partially isolable from F_2

Table 2.1: Definitions of fault-mode isolability.

If written out, the definitions from Table 2.1 become:

Definition 2.16
(Uniform (Complete) Fault-Mode Isolability in a Diagnosis System)
Given a diagnosis system δ, we say that F_1 is uniformly and completely isolable from F_2 *if*

$$\forall[x_0, u, \phi]\ \forall\theta_1 \in \Theta_{F_1}\ \forall\theta_2 \in \Theta_{F_2}\ .\quad \theta_1 \textit{ is isolable from } \theta_2 \textit{ under } [x_0, u, \phi]$$

Definition 2.17
((Complete) Fault-Mode Isolability in a Diagnosis System Under $[x_0, u, \phi]$)
Given a fixed $[x_0, u, \phi]$ and a diagnosis system δ, we say that F_1 is completely isolable from F_2 under $[x_0, u, \phi]$ *if*

$$\forall\theta_1 \in \Theta_{F_1}\ \forall\theta_2 \in \Theta_{F_2}\ .\quad \theta_1 \textit{ is isolable from } \theta_2 \textit{ under } [x_0, u, \phi]$$

Definition 2.18
((Complete) Fault-Mode Isolability in a Diagnosis System)
Given a diagnosis system δ, we say that F_1 is completely isolable from F_2 *if*

$$\exists[x_0, u, \phi]\ \forall\theta_1 \in \Theta_{F_1}\ \forall\theta_2 \in \Theta_{F_2}\ .\quad \theta_1 \textit{ is isolable from } \theta_2 \textit{ under } [x_0, u, \phi]$$

Definition 2.19
(Uniform Partial Fault-Mode Isolability in a Diagnosis System)
Given a diagnosis system δ, we say that F_1 is uniformly and partially isolable from F_2 *if*

$$\forall[x_0, u, \phi]\ \exists\theta_1 \in \Theta_{F_1}\ \exists\theta_2 \in \Theta_{F_2}\ .\quad \theta_1 \textit{ is isolable from } \theta_2 \textit{ under } [x_0, u, \phi]$$

Definition 2.20
(Partial Fault-Mode Isolability in a Diagnosis System Under $[x_0, u, \phi]$)
Given a fixed $[x_0, u, \phi]$ and a diagnosis system δ, we say that F_1 is partially isolable from F_2 under $[x_0, u, \phi]$ *if*

$$\exists\theta_1 \in \Theta_{F_1}\ \exists\theta_2 \in \Theta_{F_2}\ .\quad \theta_1 \textit{ is isolable from } \theta_2 \textit{ under } [x_0, u, \phi]$$

Definition 2.21 *[Partial Fault-Mode Isolability in a Diagnosis System] Given a diagnosis system δ, we say that* F_1 is partially isolable from F_2 *if*

$$\exists[x_0, u, \phi]\ \exists\theta_1 \in \Theta_{F_1}\ \exists\theta_2 \in \Theta_{F_2}\ .\quad \theta_1 \textit{ is isolable from } \theta_2 \textit{ under } [x_0, u, \phi]$$

Note the implications between the different isolability properties. These are indicated by arrows in Table 2.1.

The most weak property is partial fault-mode isolability. However *not* partial fault-mode isolability is quite strong; if we find that F_1 is not partially isolable from F_2, then F_1 is not isolable from F_2 in any other sense.

Next we define isolability also as a property of the system:

Definition 2.22 *[[Uniform] Complete/Partial Fault Mode Isolability] A fault mode F_1 is [uniformly] and completely/partially isolable from fault mode F_2 if there exists a diagnosis system δ in which fault mode F_1 is [uniformly] completely/partially isolable from fault mode F_2.*

We have here skipped the case isolability under $[x_0, u, \phi]$.

A special case of isolability is detectability. As with isolability, we can define detectability as a system property or not.

Definition 2.23 *[Fault Mode Detectability [in a Diagnosis System]] A fault mode F_1 is [uniformly] completely/partially detectable [in a diagnosis system δ] if F_1 is isolable from NF [in the diagnosis system δ].*

The isolability and detectability properties of a set of fault modes can be quite difficult to analyze by only using the definitions 2.16 to 2.23. However,

these properties are still important so therefore, we need some tools (i.e. theorems) by which isolability and detectability can be analyzed from more easily identified properties of the diagnosis system and the fault modes. Some tools, applicable in the general case, are presented in the next section, and some tools, applicable for linear systems, are presented in Chapter 8.

2.6 Submode Relations between Fault Modes and Isolability

Submode relations between fault modes, as defined in Section 2.4, can severely limit the possibility to perform fault isolation. This is formally explained by the following theorem:

Theorem 2.1 *Assume it holds that $F_1 \preccurlyeq^* F_2$, then*

a) F_1 is not completely isolable from F_2

b) F_2 is not completely isolable from F_1

c) if $\delta(x)$ is an ideal diagnosis system, i.e.

$$\gamma \in S \iff \mathcal{M}_\gamma(\theta) \text{ can explain data } x$$

and $\mathcal{M}(\theta)$ is a correct model, then F_1 is not partially or completely isolable from F_2 in $\delta(x)$.

Proof: For the (a)-part, assume that F_1 is completely isolable from F_2. Then from Definition 2.18 and 2.22 we know that there exists a diagnosis system and a $[x_0, u, \phi]$ such that for all $\theta_1 \in \Theta_{F_1}$ and all $\theta_2 \in \Theta_{F_2}$, it holds that

$$\theta_1 \text{ present} \quad \Longrightarrow \quad F_1 \in S \wedge F_2 \notin S \tag{2.8a}$$
$$\theta_2 \text{ present} \quad \Longrightarrow \quad F_2 \in S \tag{2.8b}$$

Assume that θ_1 is present. Since $F_1 \preccurlyeq^* F_2$, we know that there is a $\theta_2^* \in \Theta_{F_2}$ (or possibly in the limit) such that $\mathcal{M}_2(\theta_2^*) = \mathcal{M}_1(\theta_1)$. This means that the output y from the plant when θ_1 is present equals the output when θ_2^* is present. That is, the diagnosis statement when θ_2^* is present, equals the diagnosis statement when θ_1 is present. Therefore, when θ_2^* is present, we have, according to (2.8a), that $F_2 \notin S$. However, from (2.8b) we have that θ_2^* present implies $F_2 \in S$. This contradiction proves the (a)-part of the theorem.

For the (b)-part, assume that F_2 is completely isolable from F_1. Then we know that there exists a diagnosis system and a $[x_0, u, \phi]$ such that for all $\theta_2 \in \Theta_{F_2}$ and all $\theta_1 \in \Theta_{F_1}$, it holds that

$$\theta_2 \text{ present} \quad \Longrightarrow \quad F_2 \in S \wedge F_1 \notin S \tag{2.9}$$
$$\theta_1 \text{ present} \quad \Longrightarrow \quad F_1 \in S \tag{2.10}$$

The relation $F_1 \preccurlyeq^* F_2$ implies that there exists $\theta_1^* \in \Theta_{F_1}$ and $\theta_2^* \in \Theta_{F_2}$ (or possibly in the limit) such that $\mathcal{M}_2(\theta_2^*) = \mathcal{M}_1(\theta_1)$. These two θ_i^* give the same diagnosis statement S. From (2.9) we have that $F_1 \notin S$ and from (2.9), $F_1 \in S$. This contradiction proves the (b)-part of the theorem.

For the (c)-part, assume that F_1 is partially isolable from F_2 in an ideal diagnosis system δ. Then from Definition 2.21 we know that there exist $[x_0, u, \phi]$, $\theta_1 \in \Theta_{F_1}$ and $\theta_2 \in \Theta_{F_2}$ such that (2.8) holds.

Assume that θ_1 is present. With the same reasoning as for the (a)-part, we can then conclude that there is a $\theta_2^* \in \Theta_{F_2}$ which gives exactly the same diagnosis statement as θ_1, i.e. $F_2 \notin S$. Therefore, when θ_2^* is present, we have that $F_2 \notin S$. However, from the assumption of ideal diagnosis system and correct model, we know that θ_2^* present implies $F_2 \in S$. This contradiction proves the (c)-part of the theorem.

■

Note that since *not isolability* implies *not uniform isolability*, this theorem also proves that $F_1 \preccurlyeq^* F_2$ implies that F_1 is not uniformly completely/partially isolable from F_2.

The next theorem shows that when a fault mode is *not* related by the submode-relation to another fault mode, then we are able to prove at least partial isolability.

Theorem 2.2 *If it holds that $F_1 \not\preccurlyeq^* F_2$ and the model $\mathcal{M}(\theta)$ is correct, then F_1 is partially isolable from F_2 in an ideal diagnosis system.*

Proof: The relation $F_1 \not\preccurlyeq^* F_2$ means that there is a $\theta_1 \in \Theta_{F_1}$ such that for all $\theta_2 \in \Theta_{F_2}$ it holds that

$$\mathcal{M}_2(\theta_2) \neq \mathcal{M}_1(\theta_1) \tag{2.11}$$

Assume that θ_1 is present. Then the assumption of correct model and ideal diagnosis system implies that $F_1 \in S$. The relation (2.11) means that there must exist a $[x_0, u, \phi]$ such that $\mathcal{M}_2(\theta_2)$ can not explain the data for any θ_2. This further means that $F_2 \notin S$. Thus, we have shown that there exists a $[x_0, u, \phi]$ and a θ_1 such that $F_1 \in S \wedge F_2 \notin S$. From the assumption of correct model and ideal diagnosis system it also holds that for all $\theta_2 \in \Theta_{F_2}$, $F_2 \in S$. This proves that F_1 is partially isolable from F_2. ■

The following example illustrates some of the isolability properties and also how Theorem 2.1 and 2.2 can be used.

Example 2.14

Consider a valve whose position $x(t)$ is controlled by the signal $u(t)$ and measured with a sensor with output $y_s(t)$. Three system fault-modes are considered: NF (no fault), AF (actuator fault), and SF (sensor fault). The fault modes

are described by the following models:

$$\begin{array}{lll} \mathcal{M}_{NF}: & \mathcal{M}_{AF}(f(t)): & \mathcal{M}_{SF}: \\ x(t) = u(t) & x(t) = u(t) + f(t) & x(t) = u(t) \\ y_s(t) = x(t) & y_s(t) = x(t) & y_s(t) = 0 \end{array}$$

We also know that the input signal is limited as $1 < u < 2$.

By studying the models representing the different fault modes, we realize that the following relations hold:

$$\begin{array}{l} NF \preccurlyeq^* AF \\ NF \not\preccurlyeq^* SF \\ AF \not\preccurlyeq^* NF \\ AF \not\preccurlyeq^* SF \\ SF \not\preccurlyeq^* NF \\ SF \preccurlyeq^* AF \end{array}$$

Now we will use these relations together with Theorem 2.1 and 2.2, and assuming an ideal diagnosis system. Doing so we obtain the following facts:

NF not isol. from AF (Th. 2.1), AF not compl. isol. from NF (Th. 2.2)
NF part. isol. from SF (Th. 2.1)
AF part. isol. from NF (Th. 2.1)
AF part. isol. from SF (Th. 2.1)
SF part. isol. from NF (Th. 2.1)
SF not isol. from AF (Th. 2.1), AF not compl. isol. from SF (Th. 2.2)

By some more studying the models representing the different fault modes, it can be realized that some isolability properties are actually stronger than this. All isolability properties have been collected in the following table:

	NF	AF	SF
NF	-	not	uniformly completely
AF	uniformly partially	-	uniformly partially
SF	uniformly completely	not	-

The entries in the table shows the isolability of the fault mode of the row from the fault mode of the column. For example, the first row says that NF is not isolable from AF and NF is uniformly and completely isolable from SF. ■

Note that the isolability is not a symmetric property. For instance, in the example above, NF is not isolable from AF but AF is uniformly and partially isolable from NF.

From Theorem 2.1 and 2.2 it is clear that to facilitate isolation, we want to avoid that the fault modes are related with the submode-relation. One reason for the presence of submode-relations between the fault modes, is that faults have

been modeled by too general fault models. That is, too general fault models implies that it becomes difficult (or impossible) to isolate between different faults. When designing a model-based diagnosis-system, this fact implies that the following advice is of high importance:

> To facilitate fault isolation, fault models should be made as specific as possible.

In practice this means for example that when a fault can be modeled as a deviation in a constant parameter, then the fault should not be modeled with an arbitrary fault signal. Also, when parameters θ_i are known to be limited in range, this information should be incorporated into the fault model.

2.6.1 Refining the Diagnosis Statement

When fault modes are related by the submode relation, they are in accordance with Theorem 2.1 not isolable from each other. This means that if $A \preccurlyeq^* B$ and the fault mode present in the system is A, then if the diagnosis statement contains A, it is very likely to also contain B, i.e. $S = \{A, B, \dots\}$.

Now from another point of view, assume that we encounter a diagnosis statement $S = \{A, B\}$. This in principle means that both A and B can explain the data. However, since $A \preccurlyeq^* B$, i.e. A is more restricted than B, it is much more likely that the data has been generated by a system with fault mode A present. It is possible to extend the diagnosis system with this kind of reasoning, and in that case the fault statement would become the single fault mode A. In general, all fault modes in the diagnosis statements which are "supermodes" of other fault modes in the diagnosis statement, should be neglected. In this way we can produce a refined diagnosis statement $\bar{S}$ which becomes

$$\bar{S} = \{F_1 \in S \mid \forall F_2 \in S.\ F_2 \neq F_1 \rightarrow F_2 \not\preccurlyeq F_1\} \tag{2.12}$$

For example, NF is likely to be related to all other fault modes F_i as $NF \preccurlyeq^* F_i$. Because of this, even though NF is the present fault mode, it will never be the only fault mode in S. From a slightly different viewpoint, this was also discovered in Section 2.3.1. However, if the refined diagnosis statement (2.12) is used, it becomes $\bar{S} = \{NF\}$.

2.7 Conclusions

This chapter has introduced a general theoretical framework for describing and analyzing diagnosis problems. In contrast to other existing frameworks, e.g. the residual view, this framework is not limited to any special type of faults. We have shown how common types of fault modeling techniques fits into the framework, e.g. faults modeled as arbitrary signals, deviations in constants, and abrupt changes of variables. Also multiple faults are naturally integrated so that no special treatment is needed. The important term *fault mode* has been defined and it will be frequently used in all the following chapters of the thesis.

A general architecture for a diagnosis system has been introduced and a relation to methods based on propositional logic is indicated. We have introduced the idea that the output from a diagnosis system can be several possible faults.

Using the framework, many conceptually important terms have been defined, e.g. fault, fault diagnosis, fault isolation, detected fault, isolated fault, fault isolability, fault detectability, etc. The meanings of the terms isolability and detectability have been shown to have quite many nuances. A *submode* relation between fault modes have been defined. It has been shown that this relation has important consequences for isolability and detectability. An important conclusion is that fault models should not be made too general since then, it becomes difficult to isolate faults from each other.

Appendix

2.A Summary of Example

This section contains a summary of the sensor-bias example given in Sections 2.1 and 2.2.

Notation Summary

Θ	set of all fault states
Θ_γ	fault state space for fault mode γ
θ	fault state
$\mathcal{D}^i$	fault state space of component i
$\mathcal{D}^i_\psi$	fault state space of component i and component fault-mode ψ
θ_i	fault state of component i
θ_γ	free fault state parameter for fault mode γ
$\mathcal{M}(\theta)$	complete system model
$\mathcal{M}_\gamma(\theta) = \mathcal{M}_\gamma(\theta_\gamma)$	system model for fault mode γ

Sensor-Bias Example

The system is described by the following equations:

$$\dot{x} = f(x, u) \tag{2.13a}$$
$$y_1 = h_1(x) + b_1 \tag{2.13b}$$
$$y_2 = h_2(x) + b_2 \tag{2.13c}$$
$$b_1 \geq 0 \tag{2.13d}$$
$$b_2 \geq 0 \tag{2.13e}$$

The constants b_1 and b_2 represents sensor bias faults and it is assumed that only positive biases can occur.

The system contains two components: sensor 1 and sensor 2. The component fault-modes are summarized in the following table:

component number i	component name	component fault-modes	component fault-state
1	Sensor 1	$NF1$, $B1$	b_1
2	Sensor 2	$NF2$, $B2$	b_2

The fault mode $B1$ is a positive bias in sensor 1 and $B2$ is positive bias in sensor 2. The set of component fault-modes implies that there are four possible

system fault-modes:

$$\begin{aligned}\mathbf{NF} =&[NF1, NF2]\\ \mathbf{B1} =&[B1, NF2]\\ \mathbf{B2} =&[NF1, B2]\\ \mathbf{B1}\&\mathbf{B2} =&[B1, B2]\end{aligned}$$

The fault state of the system is described by the vector $\theta = [b_1\ b_2]$. The parameter spaces for b_1 and b_2 are defined by

$$\begin{aligned}b_1 \in&\mathcal{D}^1 = \mathcal{D}^1_{NF1} \cup \mathcal{D}^1_{B1}\\ b_2 \in&\mathcal{D}^2 = \mathcal{D}^2_{NF1} \cup \mathcal{D}^2_{B1}\\ \mathcal{D}^1_{NF1} =&\{0\}\\ \mathcal{D}^2_{NF2} =&\{0\}\\ \mathcal{D}^1_{B1} =&\{x > 0\}\\ \mathcal{D}^2_{B2} =&\{x > 0\}\end{aligned}$$

The parameter spaces for θ are defined by

$$\begin{aligned}\theta \in&\Theta = \mathcal{D}^1 \times \mathcal{D}^2 = \Theta_{\mathbf{NF}} \cup \Theta_{\mathbf{B1}} \cup \Theta_{\mathbf{B2}} \cup \Theta_{\mathbf{B1}\&\mathbf{B2}}\\ \Theta_{\mathbf{NF}} =&\{\theta | b_1 \in \mathcal{D}^1_{NF1} \wedge b_2 \in \mathcal{D}^2_{NF2}\} = \{\theta | b_1 = 0 \wedge b_2 = 0\}\\ \Theta_{\mathbf{B1}} =&\{\theta | b_1 \in \mathcal{D}^1_{B1} \wedge b_2 \in \mathcal{D}^2_{NF2}\} = \{\theta | b_1 > 0 \wedge b_2 = 0\}\\ \Theta_{\mathbf{B2}} =&\{\theta | b_1 \in \mathcal{D}^1_{NF1} \wedge b_2 \in \mathcal{D}^2_{B2}\} = \{\theta | b_1 = 0 \wedge b_2 > 0\}\\ \Theta_{\mathbf{B1}\&\mathbf{B2}} =&\{\theta | b_1 \in \mathcal{D}^1_{B1} \wedge b_2 \in \mathcal{D}^2_{B2}\} = \{\theta | b_1 > 0 \wedge b_2 > 0\}\end{aligned}$$

The model $\mathcal{M}(\theta) = \mathcal{M}([b_1\ b_2])$ is defined by (2.13).
The models associated with the four fault modes are

$$\begin{aligned}\mathcal{M}_{\mathbf{NF}}(\theta) =&\mathcal{M}(\theta)|_{\theta\in\Theta_{\mathbf{NF}}} = \mathcal{M}_{\mathbf{NF}}\\ \mathcal{M}_{\mathbf{B1}}(\theta) =&\mathcal{M}(\theta)|_{\theta\in\Theta_{\mathbf{B1}}} = \mathcal{M}_{\mathbf{B1}}(b1)\\ \mathcal{M}_{\mathbf{B2}}(\theta) =&\mathcal{M}(\theta)|_{\theta\in\Theta_{\mathbf{B2}}} = \mathcal{M}_{\mathbf{B2}}(b2)\\ \mathcal{M}_{\mathbf{B1}\&\mathbf{B2}}(\theta) =&\mathcal{M}(\theta)|_{\theta\in\Theta_{\mathbf{B1}\&\mathbf{B2}}} = \mathcal{M}_{\mathbf{B1}\&\mathbf{B2}}([b1\ b2])\end{aligned}$$

Chapter 3

Structured Hypothesis Tests

In this chapter, we will see how classical hypothesis testing can be utilized for model based diagnosis and especially fault isolation. The literature is quite sparse on this subject but some related contributions can be found in (Riggins and Rizzoni, 1990; Grainger, Holst, Isaksson and Ninnes, 1995; Bøgh, 1995; Basseville, 1997).

The formalism from the previous chapter will be used to define a new general approach called *structured hypothesis tests*. As its name indicates, the approach uses a structure of several hypothesis tests. Structured hypothesis tests may be seen as a generalization of the well known method *structured residuals* (Gertler, 1991), but have the additional advantage that it is theoretically grounded in classical hypothesis testing and also propositional logic.

As a result of this, the model of the system can be fully utilized in a systematic way. This implies that it is possible to diagnose a large variety of different types of faults within the same framework and same diagnosis system. For example both faults modeled as changes in parameters and faults modeled as additive signals are easily handled. Further, the approach is quite intuitive and very similar to the reasoning involved when humans are doing diagnosis. Several other principles for diagnosis can be seen as special cases, e.g. parameter estimation (Isermann, 1993), observer schemes (Patton et al., 1989), structured residuals (Gertler, 1991), and statistical methods (Basseville and Nikiforov, 1993).

The basics of structured hypothesis tests is given in Sections 3.1 and 3.2, and exemplified in Section 3.3. Design and analysis of the hypothesis tests is shortly mentioned, but the most of this discussion is left to Chapter 4. Section 3.4 discusses *incidence structures* and *decision structures*, which are related to the *residual structure*. This relation is then investigated in Section 3.5, which discusses the relation between structured hypothesis tests and the method structured residuals.

3.1 Fault Diagnosis Using Structured Hypothesis Tests

Using the principle of structured hypothesis tests, each of the individual tests δ_k are assumed to be *hypothesis* tests. Then the diagnosis system consists of a set of hypothesis tests, δ_1 to δ_n, and the *decision logic*. Except for this general connection to hypothesis testing, structured hypothesis tests has also a closer connection to the method *intersection-union test*, that can be found in statistical literature, e.g. (Casella and Berger, 1990).

The classical, statistical or decision theoretic, definition of *hypothesis test* is adopted, e.g. see (Berger, 1985; Lehmann, 1986; Casella and Berger, 1990). This means that a hypothesis test is a procedure to, based on sample data, select between exactly *two* hypotheses characterized by $\theta \in \Theta_0$ and $\theta \in \Theta_0^C$. This is in contrast to "multiple hypothesis testing" that is often found in literature, e.g. (Basseville and Nikiforov, 1993). Note that when using hypothesis testing, we can have a probabilistic (statistical) or a deterministic view. Therefore, the method structured hypothesis tests is valid either we have probabilistic knowledge, in terms of probability density functions of e.g. the signal z (described in Section 2.1.1) or measurement noise, or not.

As before, the test $\delta_k(x)$, now a hypothesis test, is a function of u and y and $S_k = \delta_k(x) = \delta_k([u\ y])$. The null hypothesis for the k:th hypothesis test, i.e. H_k^0, is that the fault mode, present in the process, belongs to a specific set M_k of fault modes. The alternative hypothesis H_k^1 is that the present fault mode does not belong to M_k. This means that if hypothesis H_k^0 is rejected, and thus H_k^1 is accepted, the present fault mode can not belong to M_k, i.e. it must belong to M_k^C. In this way, each individual hypothesis test contributes with a piece of informations about which fault modes that can be present. As before, the *decision logic* then combines this information to form the *diagnosis statement*.

Let F_p again denote the present system fault-mode. Then for the k:th hypothesis test, the null hypothesis and the alternative hypothesis can be written

$H_k^0 : F_p \in M_k$ "some fault mode in M_k can explain the measured data"
$H_k^1 : F_p \in M_k^C$ "no fault mode in M_k can explain the measured data"

An alternative is to use the definition of the sets Θ_γ to describe the hypotheses. This is done via the sets Θ_k^0 which are defined as

$$\Theta_k^0 = \bigcup_{\gamma \in M_k} \Theta_\gamma \tag{3.1}$$

The hypotheses can now be expressed as

$H_k^0 : \theta \in \Theta_k^0$ "some value of $\theta \in \Theta_k^0$ can explain the measured data"
$H_k^1 : \theta \notin \Theta_k^0$ "*no* value of $\theta \in \Theta_k^0$ can explain the measured data"

The convention used here and also commonly used in hypothesis testing literature, is that when H_k^0 is rejected, we *assume* that H_k^1 is true. Further, when H_k^0 is *not* rejected, we will for the present not assume anything. This latter fact will be slightly modified in Section 3.2, where we discuss how we also can assume something when H_k^0 is *not* rejected.

How the hypothesis tests are used to diagnose and isolate faults is illustrated by the following example.

Example 3.1

Assume that the diagnosis system contains the following set of three hypothesis tests:

$$\begin{aligned} H_1^0 &: F_p \in M_1 = \{NF, F_1\} \quad H_1^1 : F_p \in M_1^C = \{F_2, F_3\} \\ H_2^0 &: F_p \in M_2 = \{NF, F_2\} \quad H_2^1 : F_p \in M_2^C = \{F_1, F_3\} \\ H_3^0 &: F_p \in M_3 = \{NF, F_3\} \quad H_3^1 : F_p \in M_3^C = \{F_1, F_2\} \end{aligned}$$

Then if only H_1^0 is rejected, we can draw the conclusion that $F_p \in M_1^C = \{F_2, F_3\}$, i.e. the present system fault-mode is either F_2 or F_3. If both H_1^0 and H_2^0 are rejected, we can draw the conclusion that $F_p \in M_1^C \cap M_2^C = \{F_2, F_3\} \cap \{F_1, F_3\} = \{F_3\}$, i.e. the present system fault-mode is F_3. ■

We see that in this context, it is natural to let the diagnosis statement be represented by sets as was introduced in Section 2.3.1.

For the two possible decisions of a hypothesis test δ_k, we use the notation S_k^0 and S_k^1. This means that

$$S_k = \begin{cases} S_k^1 = M_k^C & \text{if } H_k^0 \text{ is rejected } (H_k^1 \text{ accepted}) \\ S_k^0 = \Omega & \text{if } H_k^0 \text{ is } \textit{not} \text{ rejected} \end{cases} \tag{3.2}$$

where Ω denotes the set of all fault modes. We will in Section 3.2 below, relax the definition of S_k^0 such that it may be a subset of Ω, i.e. $S_k^0 \subseteq \Omega$. Depending on how S_k^0 and S_k^1 are defined, a diagnosis system based on structured hypothesis tests can be either speculative or conclusive.

All together, the diagnosis-system architecture presented in Section 2.3, and the use of hypothesis tests, is closely related to human reasoning about diagnosis. A human being naturally speculates around a set of different hypotheses and then his/her diagnosis statement is composed of individual conclusions of how well his/her observations match the different hypotheses. An example of such reasoning is: "if it is the fuse that is broken, then no lamps in this room would be lighted". Then he/she may observe that there are lighted lamps and thus the hypothesis "the fuse that is broken" must be rejected.

Much of the engineering work involved in constructing a diagnosis system is to use the model $\mathcal{M}(\theta)$ to construct the individual hypothesis tests. The design of the hypothesis tests will be discussed in more detail in the next section and also in Chapter 4.

3.2 Hypothesis Tests

For each hypothesis test δ_k, we need to find a *test quantity* and a *rejection region*. The sample data $\mathbf{x}$ for each hypothesis is plant inputs u and outputs y. The sample data can further be all such data up to present time or a subset of this data. The test quantity is a function $T_k(x)$ from the sample data $\mathbf{x}$, to a scalar value which is to be thresholded by a threshold J_k. Thus δ_k will have a structure according to Figure 3.1.

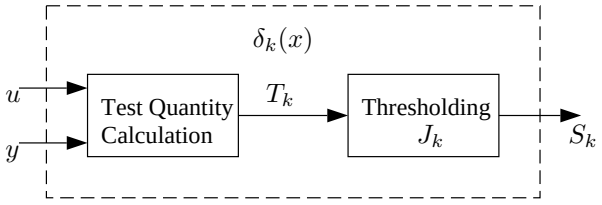

Figure 3.1: Hypothesis test $\delta_k(x)$.

The test quantity $T_k(x)$ is in many texts instead called a *test statistic*. However, the name *test statistic* indicates that $T_k(x)$ is a random variable which in general may not be a desired view. The test quantity $T_k(x)$ may for example be a *residual generator*[1] or a sum of squared prediction errors of a parameter estimator. In many applications, a deterministic view is taken and $T_k(x)$ is seen just as a function of the data and not as a random variable.

Formally the hypothesis test δ_k is defined as

$$S_k = \delta_k(x) = \begin{cases} S_k^1 & \text{if } T_k(x) \geq J_k \\ S_k^0 & \text{if } T_k(x) < J_k \end{cases} \tag{3.3}$$

The rejection region of each test is thereby implicitly defined.

The definition (3.3) means that we need to design a test quantity $T_k(x)$ such that it is low or at least below the threshold if the data x matches the hypothesis H_k^0, i.e. a fault mode in M_k can explain the data. Also if the data come from a fault mode not in M_k, $T_k(x)$ should be large or at least above the threshold. Using traditional terminology, the fault modes in M_k are said to be *decoupled*.

How well the hypothesis test meets these requirements is quantified by the *power function* $\beta_k(\theta)$ defined as

$$\beta_k(\theta) = P(\text{reject } H_k^0 \mid \theta) = P(T_k(x) \geq J_k \mid \theta)$$

We want the power function to be low for $\theta \in \Theta_k^0$ and large for $\theta \notin \Theta_k^0$. To be able to make the assumption that H_k^1 is true when H_k^0 is rejected, we need to design the hypothesis tests such that the significance level α, defined as

$$\alpha = \sup_{\theta \in \Theta_k^0} \beta_k(\theta)$$

[1] Here *residual generator* refers to specific filters used in the fault diagnosis literature, e.g. (Gertler, 1991), to indicate faults.

has a small value. This implies that the threshold J_k must be set relatively high. This in turn means that the value of $\beta_k(\theta)$ does not necessarily become large for all values $\theta \notin \Theta_k^0$. For instance, if the present fault mode is F_i and $F_i \in M_k^C$, then for some $\theta \in \Theta_{F_i}$, the probability to reject H_k^0 may be very small. This is the reason why we up to now, have assumed that $S_k^0 = \Omega$, i.e. we can not assume anything when H_k^0 is not rejected.

Now if it actually holds that the power function is large for all $\theta \in \Theta_{F_i}$, then we do not take any large risk if we assume that F_i has not occured when H_k^0 is not rejected. If this is the case, F_i should be excluded from S_k^0. The relation between the power function and the decisions S_k^0 and S_k^1 is further investigated in Section 4.7.2.

How the test quantities $T_k(x)$ are constructed depends on the actual case, and only for some specific classes of systems and fault models, general design procedures have been proposed, e.g. linear systems with fault modeled as inputs (Nyberg and Frisk, 1999).

To develop the actual hypothesis tests, we first need to decide the set of hypotheses to test. One solution is to use one hypothesis test for each fault mode. In this case, the set of hypothesis tests can be indexed by $\gamma \in \Omega$, i.e. δ_γ, and becomes

$$H_\gamma^0 : F_p \in M_\gamma \tag{3.4a}$$

$$H_\gamma^1 : F_p \in M_\gamma^C \tag{3.4b}$$

$$\gamma \in \Omega \tag{3.4c}$$

3.2.1 How the Submode Relation Affects the Choice of Null Hypotheses

The choice of null hypotheses is not a completely free choice but restricted by the submode relation defined in Section 2.4. The restriction can be expressed as:

> If $A \preccurlyeq^* B$, then the null hypotheses $F_p \in \{A, B\}$ and $F_p \in \{A\}$ are good choices but $F_p \in \{B\}$ is not.

The motivation is that if the null hypothesis is $F_p \in \{B\}$, then the test quantity is low for $F_p = B$ but since $A \preccurlyeq^* B$, the test quantity will be equally low for also $F_p = A$. Consider for example the fault modes "sensor bias" SB and NF. With the discussion of Example 2.13 in mind, we can expect that $NF \preccurlyeq^* SB$ and therefore we should never use $F_p \in \{SB\}$ as a null hypothesis but instead $F_p \in \{NF, SB\}$.

3.3 Examples

This section contains two examples that illustrates how hypothesis tests and especially test quantities can be constructed.

3.3.1 Faults Modeled as Deviations of Plant Parameters

Consider a process which can be modeled as

$$y(t) = \theta_1 u_1(t) + \theta_2 u_2(t) + \theta_3 u_3(t)$$

The fault state vector is $\theta = [\theta_1\ \theta_2\ \theta_3]$. Four fault modes are considered:

NF	$\theta = [1\ 1\ 1]$
F_1	$\theta_1 \neq 1,\ \ \theta_2 = \theta_3 = 1$
F_2	$\theta_2 \neq 1,\ \ \theta_1 = \theta_3 = 1$
F_3	$\theta_3 \neq 1,\ \ \theta_1 = \theta_2 = 1$

To diagnose this system, we use four hypothesis tests whose null hypotheses are defined by the sets M_k:

$$\begin{aligned} M_0 &= \{NF\} \\ M_1 &= \{NF, F_1\} \\ M_2 &= \{NF, F_2\} \\ M_3 &= \{NF, F_3\} \end{aligned}$$

The null and alternative hypotheses become

$$\begin{aligned} H_k^0 &: F_p \in M_k \\ H_k^1 &: F_p \in M_k^C \end{aligned}$$

for $k = 0, 1, 2, 3$. Then we have that $S_k^1 = M_k^C$ and S_k^0 is chosen as $S_k^0 = \Omega$.

As test quantities, we use the functions

$$T_0(x) = \sum_{t=0}^{N} \left(y - \hat{y}\right)^2 = \sum_{t=0}^{N} \left(y - u_1 - u_2 - u_3\right)^2 \tag{3.5a}$$

$$T_1(x) = \min_{\theta_1} \sum_{t=0}^{N} \left(y - \hat{y}\right)^2 = \min_{\theta_1} \sum_{t=0}^{N} \left(y - \theta_1 u_1 - u_2 - u_3\right)^2 \tag{3.5b}$$

$$T_2(x) = \min_{\theta_2} \sum_{t=0}^{N} \left(y - \hat{y}\right)^2 = \min_{\theta_2} \sum_{t=0}^{N} \left(y - u_1 - \theta_2 u_2 - u_3\right)^2 \tag{3.5c}$$

$$T_3(x) = \min_{\theta_3} \sum_{t=0}^{N} \left(y - \hat{y}\right)^2 = \min_{\theta_3} \sum_{t=0}^{N} \left(y - u_1 - u_2 - \theta_3 u_3\right)^2 \tag{3.5d}$$

Note that these functions are in principle parameter estimators and that $T_k(x)$ is the sum of squared prediction errors. It is obvious that the functions (3.5) are small when the present fault mode belongs to the corresponding set M_k. For example if F_1 is the present fault mode, then $T_1(x)$ will produce a good estimate of θ_1 which implies that the simulation error and $T_1(x)$ will become small. Also, for at least "large" faults and large inputs, the functions (3.5) are

large when the present fault mode does not belong to the corresponding set M_k. For example if F_1 is the present fault mode, and the fault is "large", then $T_0(x)$, $T_2(x)$, and $T_3(x)$ will all become large. All this means that the functions (3.5) satisfy our requirements on test quantities.

3.3.2 Faults Modeled as Arbitrary Fault Signals

Consider a process which can be modeled as

$$\begin{aligned} x(t+1) &= Ax(t) + B\big(u(t) + f_u(t)\big) \\ y_1(t) &= C_1 x(t) + f_1(t) \\ y_2(t) &= C_1 x(t) + f_2(t) \end{aligned}$$

where the signals f_u, f_1, and f_2 represent an actuator fault and faults in sensor 1 and 2 respectively. The fault state vector is $\theta = [f_u(t)\ f_1(t)\ f_2(t)]$. Four fault modes are considered:

$$\begin{array}{ll} NF & \theta = [0\ 0\ 0] \\ F_u & \theta = [f_u(t)\ 0\ 0], \quad f_u(t) \not\equiv 0 \\ F_1 & \theta = [0\ f_1(t)\ 0], \quad f_1(t) \not\equiv 0 \\ F_2 & \theta = [0\ 0\ f_2(t)], \quad f_2(t) \not\equiv 0 \end{array}$$

To diagnose this system, we use the two hypothesis tests

$$\begin{aligned} H_1^0 &: F_p \in M_1 = \{NF, F_1\} \quad H_1^1 : F_p \in M_1^C = \{F_u, F_2\} \\ H_2^0 &: F_p \in M_2 = \{NF, F_2\} \quad H_2^1 : F_p \in M_2^C = \{F_u, F_1\} \end{aligned}$$

To calculate the test quantities, we first use the following two observers

$$\hat{x}(t+1) = Ax(t) + Bu(t) - K(y_1(t) - \hat{y}_1(t)) \tag{3.6a}$$
$$\hat{y}_1(t) = C_1 x(t) \tag{3.6b}$$

$$\hat{x}(t+1) = Ax(t) + Bu(t) - K(y_2(t) - \hat{y}_2(t)) \tag{3.7a}$$
$$\hat{y}_2(t) = C_2 x(t) \tag{3.7b}$$

Then the test quantities can be defined as

$$\begin{aligned} T_1(x) &= |y_2(t) - \hat{y}_2(t)| \\ T_2(x) &= |y_1(t) - \hat{y}_1(t)| \end{aligned}$$

These test quantities $T_k(x)$ are zero or small if the present fault mode belongs to the corresponding sets M_k. For example, if F_1 is the present fault mode, then the observer (3.7) will produce a good estimate $\hat{y}_2(t)$ since the calculation of $\hat{y}_2(x)$ is not affected by a fault in sensor 1. This means that $T_1(x)$ will become small. Also when F_1 is present, it can be shown that $T_2(x)$ will become large or at least non-zero. This means that $T_1(x)$ and $T_2(x)$ serves well as test quantities. This configuration of observers, in which each observer is fed by only one of the output signals, is called a *dedicated observer scheme* (Clark, 1979).

3.4 Incidence Structure and Decision Structure

This section describes the concept of *incidence structure* and *decision structure* which can be seen as generalizations of the well known *residual structure* (Gertler, 1998). We here introduce a distinction between the incidence structure, describing how the faults affects the test quantities, and the decision structure, describing how the fault decision depend on the thresholded test quantities. We will also see that the decision structure relates to structured hypothesis tests in the same way as the residual structure relates to the isolation method method *structured residuals* (Gertler and Singer, 1990).

3.4.1 Incidence Structure

To get an overview of how faults in different fault modes *ideally* affect the test quantities, it is useful to set up an *incidence structure*. With *ideally*, we mean that the system behaves exactly in accordance with the model and all stochastic parts have been neglected, e.g. no unmodeled disturbances exists and there is no measurement noise. An incidence structure is a table or matrix containing 0:s, 1:s, and X:s. The X:s will be called *don't care*. An example of an incidence structure is

	NF	F_1	F_2	F_3
$T_1(x)$	0	0	1	0
$T_2(x)$	0	0	1	1
$T_3(x)$	0	X	0	1

(3.8)

A 0 in the k:th row and the j:th column means that if the system fault-mode present in the system, is equal to the system fault-mode of the j:th column, then the test quantity $T_k(x)$ will not be affected, i.e. it will be exactly zero. A 1 in the k:th row and the j:th column means that for *all*[2] faults belonging to the fault mode of the j:th column, $T_k(x)$ will always be affected, i.e. it will be non-zero. An X in the k:th row and the j:th column means that for *some* faults belonging to the fault mode of the j:th column, $T_k(x)$ will under some operating conditions be affected, i.e. it will be non-zero.

As said above, although a distinction has not been made between incidence structures and decision structures in previous literature, the basic idea of using incidence structures (or residual structures) is not new. However, compared to previous works involving incidence structures, a major difference is that we have here added the use of don't care.

The incidence structure is derived by studying the equations describing the process model and how the test quantities $T_k(x)$ are calculated. This is illustrated in the following example:

[2] As noted in (Wünnenberg, 1990), we may have to relax the requirement to *almost* all faults; e.g. when faults are modeled as arbitrary signals, we can not require that faults that are solutions to the differential equation $T_k(x) = 0$, affects test quantity.

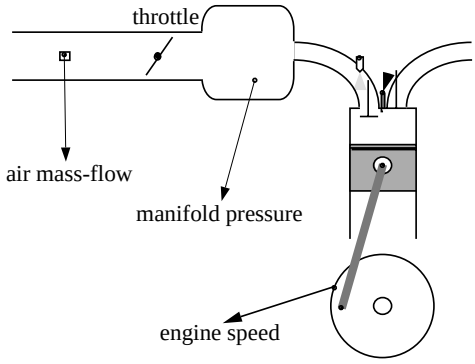

Figure 3.2: A principle illustration of an SI-engine.

Example 3.2

Consider Figure 3.2, containing a principle illustration of a spark-ignited combustion engine. The air enters at the left side, passes the throttle and the manifold, and finally enters the cylinders. The engine in the figure have sensors measuring the physical variables air mass-flow, throttle angle, and manifold pressure.

The air flow $\dot{m}$ past the throttle can be modeled as a non-linear function of the throttle angle α and the manifold pressure p:

$$\dot{m} = (1 - \cos\alpha)\Phi(p) \tag{3.9}$$

where the $d\Phi(p)/dp = 0$ for supersonic air-speeds which occurs for all $p < 53$kPa (Heywood, 1992). The throttle angle α is always between 0 and $\pi/2$.

Three system fault modes are considered: no fault NF, air mass-flow sensor fault M, and manifold pressure sensor fault P. For both M and P, the faults are modeled as an arbitrary signal added to the sensor signals:

$$\dot{m}_s = \dot{m} + f_{\dot{m}} \tag{3.10a}$$
$$p_s = p + f_p \tag{3.10b}$$

where the index s indicates sensor signals. As test quantity, we can use

$$T(x) = T([\dot{m}_s, \alpha_s, p_s]) = \dot{m}_s - (1 - \cos\alpha_s)\Phi(p_s) \tag{3.11}$$

To see how the faults affects the test quantity, we can substitute (3.9) and (3.10) into (3.11):

$$\begin{aligned} T(x) &= \dot{m} + f_{\dot{m}} - (1 - \cos\alpha)\Phi(p + f_p) = \\ &= f_{\dot{m}} + (1 - \cos\alpha)\Phi(p) - (1 - \cos\alpha)\Phi(p + p_f) \end{aligned}$$

We see that a fault in M will always affect $T(x)$. Also, a fault in P will affect $T(x)$ if and only if $p > 53$kPa or $p + p_f > 53$kPa.

This means that the incidence structure for the test quantity $T(x)$ becomes

	NF	M	P
$T(x)$	0	1	X

(3.12)

■

Let s_{kj} denote the entry in the k:th row and the j:th column of an incidence structure. Then the interpretation or semantics of 0:s, 1:s, and X:s can be formalized as

$$F_p = F_j \ \rightarrow \ T_k(x) = 0 \qquad \text{if } s_{kj} = 0 \tag{3.13a}$$

$$F_p = F_j \ \rightarrow \ T_k(x) \neq 0 \qquad \text{if } s_{kj} = 1 \tag{3.13b}$$

where F_p, as before, denotes the present system fault-mode. Note that the implication, denoted by the arrow, is not symmetric. Note also that the interpretation of X is implicitly contained in these two formulas.

In the next section, we will also define interpretations of 1:s, 0:s, and X:s for the decision structure. To the author's knowledge, no such strict interpretation has been defined in previous literature. The motivation for these strict definitions, is that we can discuss relations to for example propositional logic and hypothesis testing. In addition, these interpretations of 1:s, 0:s, and X:s alone, also defines the function of the whole diagnosis system.

By using the formulas (3.13), it is possible to formally describe the interpretation of a whole incidence structure. We will exemplify this below, by giving the interpretation of the incidence structure (3.8), but note first that $F_p \notin \{F_2\} \equiv F_p \in \Omega - \{F_2\}$. The symbol $\Longleftrightarrow$ will be used to denote tautological equivalence. Now, the interpretation of the incidence structure (3.8) becomes

$$\begin{aligned}
&T_1 = 0 \leftarrow F_p \in \{NF, F_1, F_3\} && \Longleftrightarrow && T_1 \neq 0 \rightarrow F_p = F2 \\
&T_1 \neq 0 \leftarrow F_p = F2 && \Longleftrightarrow && T_1 = 0 \rightarrow F_p \in \{NF, F_1, F_3\} \\
&T_2 = 0 \leftarrow F_p \in \{NF, F_1\} && \Longleftrightarrow && T_2 \neq 0 \rightarrow F_p \in \{F_2, F_3\} \\
&T_2 \neq 0 \leftarrow F_p \in \{F_2, F_3\} && \Longleftrightarrow && T_2 = 0 \rightarrow F_p \in \{NF, F_1\} \\
&T_3 = 0 \leftarrow F_p \in \{NF, F_2\} && \Longleftrightarrow && T_3 \neq 0 \rightarrow F_p \in \{F_1, F_3\} \\
&T_3 \neq 0 \leftarrow F_p = F_3 && \Longleftrightarrow && T_3 = 0 \rightarrow F_p \in \{NF, F_1, F_2\}
\end{aligned}$$

By using *if-and-only-if* relations, these formulas can be written on a slightly shorter form:

$$\begin{aligned}
&T_1 = 0 \leftrightarrow F_p \in \{NF, F_1, F_3\} && \Longleftrightarrow && T_1 \neq 0 \leftrightarrow F_p = F2 \\
&T_2 = 0 \leftrightarrow F_p \in \{NF, F_1\} && \Longleftrightarrow && T_2 \neq 0 \leftrightarrow F_p \in \{F_2, F_3\} \\
&T_3 = 0 \leftarrow F_p \in \{NF, F_2\} && \Longleftrightarrow && T_3 \neq 0 \rightarrow F_p \in \{F_1, F_3\} \\
&T_3 \neq 0 \leftarrow F_p = F_3 && \Longleftrightarrow && T_3 = 0 \rightarrow F_p \in \{NF, F_1, F_2\}
\end{aligned}$$

As seen, the *if-and-only-if* relation can only be used with rows, in the incidence structure, which have no X:s.

3.4.2 Decision Structure

The incidence structure corresponds to the case where ideal conditions holds. If this were the case, we could derive the diagnosis statement S by using the incidence structure, the formulas (3.13), and the values of the test quantities $T_k(x)$. In practice, the model is not perfect, unmodeled disturbances affects the process, and there is measurement noise. All this means that the formulas (3.13) are not valid and can therefore not be used to form the diagnosis statement.

In practice, we have to relax the assumptions of ideal conditions and the formulas (3.13) can be replaced by a formulation based on the use of thresholds, i.e hypothesis testing. Doing this, we obtain a *decision structure*. Still letting s_{kj} denote the entry in the k:th row and the j:th column, the new interpretation or semantics of 0:s, 1:s, and X:s becomes

$$F_p = F_j \;\rightarrow\; T_k(x) < J_k \qquad \text{if } s_{kj} = 0 \tag{3.14a}$$

$$F_p = F_j \;\rightarrow\; T_k(x) \geq J_k \qquad \text{if } s_{kj} = 1 \tag{3.14b}$$

or by using the terminology of hypothesis testing:

$$F_p = F_j \;\rightarrow\; \text{not rej. } H_k^0 \qquad \text{if } s_{kj} = 0 \tag{3.15a}$$

$$F_p = F_j \;\rightarrow\; \text{reject } H_k^0 \qquad \text{if } s_{kj} = 1 \tag{3.15b}$$

The implications are not completely true, but we assume that they holds. This corresponds to the basic assumptions, discussed in Section 3.2, that when H_k^0 is rejected, we assume that H_k^1 holds. However, there is a conflict between the two rules (3.15a) and (3.15b). To make the assumption that (3.15a) holds reasonable, the significance level α_k of all tests must be low. This means that the thresholds must be chosen relatively high. Further, this violates the assumption that (3.15b) holds. To achieve reasonable assumptions, some or probably most 1:s from the incidence structure must be replaced by X:s. It might seem that another choice is to replace 0:s by X:s, but the problem with this is that for all small faults, the assumption of (3.15b) still not becomes reasonable. We will see later that representing a diagnosis system with a decision structure, is equivalent to a representation using the sets M_k, S_k^0, and S_k^1.

An example of a decision structure is obtained by considering the incidence structure (3.8) which can be transformed to, for instance the following decision structure:

	NF	F_1	F_2	F_3
$\delta_1(x)$	0	0	X	0
$\delta_2(x)$	0	0	X	1
$\delta_3(x)$	0	X	0	X

(3.16)

Because the decision structure is related to the whole hypothesis tests and not only the test quantities, we use δ_k to label the rows instead of T_k.

The process of replacing 1:s with X:s is further illustrated by the following example:

Example 3.3

Consider again Example 3.2. When the fault mode M is present, we have that

$$T(x) = f_{\dot{m}} + v$$

where v is a signal that represents model errors, disturbances, and measurement noise. Even for fault mode NF, which implies $f_{\dot{m}} = 0$, the test quantity $T(x)$ will *not* be zero. This means that the threshold J must be raised above zero. Then for small $f_{\dot{m}}$, $T(x)$ will not reach the threshold.

If the incidence structure (3.12) would be used as decision structure, we would have the rule

$$M \rightarrow T(x) \geq J$$

However, according to what was said above, the implication will not hold for a small $f_{\dot{m}}$. This means that to obtain the decision structure, the 1 in (3.12) must be replaced by an X, i.e.

	NF	M	P
δ	0	X	X

■

A decision structure together with the formulas (3.14) can be used to derive the diagnosis statement. Consider for example the decision structure (3.16), which have the interpretation

$$\begin{aligned}
T_1 < J_1 \leftarrow F_p \in \{NF, F_1, F_3\} \quad &\Longleftrightarrow \quad T_1 \geq J_1 \rightarrow F_p = F2 \\
T_2 < J_2 \leftarrow F_p \in \{NF, F_1\} \quad &\Longleftrightarrow \quad T_2 \geq J_2 \rightarrow F_p \in \{F_2, F_3\} \\
T_2 \geq J_2 \leftarrow F_p = F_3 \quad &\Longleftrightarrow \quad T_2 < J_2 \rightarrow F_p \in \{NF, F_1, F_2\} \\
T_3 < J_3 \leftarrow F_p \in \{NF, F_2 \quad &\Longleftrightarrow \quad T_3 \geq J_3 \rightarrow F_p \in \{F_1, F_3\}
\end{aligned}$$

Now if $T_1 < J_1$, $T_2 \geq J_1$, and $T_3 \geq J_1$, we know by using the rules, that $F_p \in \{F_2, F_3\}$ and $F_p \in \{F_1, F_3\}$. This means that F_3 must be the present fault mode. It is clear that there must be a strong relationship between this procedure, i.e. forming the diagnosis statement S by using the decision structure, and how the diagnosis statement S is formed by using the individual diagnosis statements S_k.

The relationship between the decision structure and the sets S_k^0 and S_k^1 is as follows. A 0 in the k:th row for δ_k and the j:th column means that the set S_k^0 contains the fault mode of the j:th column and S_k^1 does *not* contain this fault mode. A 1 in the k:th row and the j:th column means that the set S_k^1 contains the fault mode of the j:th column and S_k^0 do *not* contain this fault mode. An X in the k:th row and the j:th column means that both S_k^0 and S_k^1 contain the

fault mode of the j:th column. For example, the sets S_k^0 and S_k^1 for the decision structure (3.16), are

$$\begin{aligned} S_1^0 &= \{NF, F_1, F_2, F_3\} & S_1^1 &= \{F_2\} \\ S_2^0 &= \{NF, F_1, F_2\} & S_2^1 &= \{F_2, F_3\} \\ S_3^0 &= \{NF, F_1, F_2, F_3\} & S_3^1 &= \{F_1, F_3\} \end{aligned}$$

In this way, the decision structure can be seen as an overview of a diagnosis system based on structured hypothesis tests. In accordance with the formulas (3.15), we can read out that when the result of a test is S_k^0, then the fault modes with 0:s and X:s in the decision structure, are the possible present fault modes. When the result is S_k^1, then the fault modes with 1:s and X:s are the possible present fault modes.

Still in accordance with the formulas (3.15), we can from a decision structure also read out which tests that will respond, i.e. which null hypothesis that will be rejected, when a particular fault mode is present. For the decision structure (3.16), we know that if NF is the present fault mode, then *no* tests will respond, because the corresponding column has only zeros. Also, if F_3 is the present fault mode, then test δ_1 will *not* respond, test δ_2 *will* respond, and test δ_3 *may* respond.

3.5 Comparison with Structured Residuals

This section contains a comparison between the well known isolation method *structured residuals* (Gertler, 1991) and structured hypothesis tests. Isolation with structured residuals is based on a *residual structure* which in principle is a combined incidence and decision structure.

A residual structure contains only 0:s and 1:s and an example is

	f_1	f_2	f_3
r_1	0	1	0
r_2	0	1	1
r_3	1	0	1

(3.17)

A minor notational difference between the residual structure and the decision structure is that usually r_i is used to label the rows instead of δ_i and also that the column related to the case no fault is usually not included in the residual structure. Further, when using structured residuals, faults are usually modeled as arbitrary fault signals. These fault signals f_j are then used to "label" the columns instead of fault modes. Usually one fault signal is used for each component which means that, as long as only single fault-modes are considered, there is a one-to-one correspondence between the fault modes F_j and the fault signals f_j.

The residual structure can be interpreted as an incidence structure in accordance with the formulas (3.13). In addition, the residual structure is also used to form the diagnosis statement. That is, it is interpreted as a decision structure

in accordance with the formulas (3.14) and (3.15). Thus a 1 in the k:th row and the j:th column means that we *assume* that for *all* faults belonging to the fault mode of the j:th column, $T_k(x)$ will be above the threshold J_k. However this assumption is mostly far from the truth. In reality, a 1 in the k:th row and the j:th column means that for *some* faults belonging to the fault mode of the j:th column, $T_k(x)$ will under some operating conditions be above the threshold J_k. Thus a more correct interpretation would be obtained by replacing most 1:s with X:s.

Usually it is required that the residual structure must be *isolating*, which means that all columns must be distinct. This together with the fact that there are only 1:s in the residual structure, implies that the fault statement always contain at the maximum *one* fault mode. That is, a diagnosis system using the principle of structured residuals with an isolating residual structure, is always *conclusive* (remember the definition from Section 2.3.3). This is illustrated in the following example:

Example 3.4

Consider the following two structures

	NF	F_1	F_2	F_3
r_1	0	0	1	0
r_2	0	0	1	1
r_3	0	1	0	1

	NF	F_1	F_2	F_3
$\delta_1(x)$	0	0	X	0
$\delta_2(x)$	0	0	X	1
$\delta_3(x)$	0	X	0	X

Assume that the left structure is a residual structure and the right is a decision structure for the same set of test quantities and thresholds. Then Table 3.1 contains a comparison between the diagnosis statement generated from the residual structure and the diagnosis statement generated from the decision structure. The leftmost column lists all possible results of thresholding the test quantities. For example, the second row 001 means that $T_1 < J_1$, $T_2 < J_2$, and $T_3 > J_3$. Note the diagnosis statements $S = \{\}$, meaning that no fault modes can explain the behavior of the system. ■

			Struct. res.	Struct. hyp. tests
1	2	3	S	S
0	0	0	$\{NF\}$	$\{NF, F_1, F_2\}$
0	0	1	$\{F_1\}$	$\{F_1\}$
0	1	0	$\{\}$	$\{F_2, F_3\}$
0	1	1	$\{F_3\}$	$\{F_3\}$
1	0	0	$\{\}$	$\{F_2\}$
1	0	1	$\{\}$	$\{\}$
1	1	0	$\{F_2\}$	$\{F_2\}$
1	1	1	$\{\}$	$\{\}$

Table 3.1: The diagnosis statement using structured residuals compared to structured hypothesis tests.

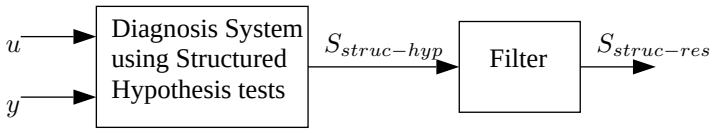

Figure 3.3: A diagnosis system using structured residuals as a filtered version of structured hypothesis tests.

As seen in Example 3.4, the "unnatural" 1:s, in the residual structure, make the diagnosis statement empty in many situations, where the diagnosis statement from structured hypothesis tests is *not* empty, e.g. study the third row. This difference is fundamental. The diagnosis system using structured hypothesis tests is in general speculative, i.e. it gives *possible* fault modes that can explain the system behavior. As we said above, a diagnosis system using structured residuals, is on the other hand conclusive.

Regardless of what diagnosis method that is used, it may be the case that several different fault modes can explain the system behavior. This information is contained in the behavior of the thresholded test quantities also when using structured residuals. However the diagnosis system neglects this information and in principle says that *no* faults can explain the system behavior. This in turn, is usually interpreted as no faults are present and no alarm is therefore generated. All this means that structured residuals can be viewed as a filtered version of structured hypothesis tests. This view is illustrated in Figure 3.3. The filter filters out useful information that could have been utilized in some way. On the other hand, there may be situations where we want to limit the information from the diagnosis system, which in that case would motivate such a filter.

As was said above, the empty diagnosis statement is usually interpreted as no faults are present. For example, in the fault free case, it might happen that one test quantity is above the threshold by mistake. A diagnosis system using structured residuals would in this case *not* generate an alarm but on the contrary, structured hypothesis tests would generate an alarm. It might therefore be argued that structured residuals is more robust to false alarms than structured hypothesis tests. This conclusion is however not fair since structured hypothesis tests is more powerful than structured residuals in the sense that the diagnosis statement contains more information. In addition, the same level of robustness can be achieved in also structured hypothesis tests by raising the thresholds.

As mentioned above, the interpretation of the 1:s is in most cases unrealistic. This implies that it may often happen that some test quantities, that according to the residual structure should reach the thresholds, are below the threshold. The effect is serious since it can happen that wrong fault is isolated. To compensate for this, it is often required that the residual structure should be *strongly isolating*. This means that when a test quantity is not above the threshold, even though it should, there should be no other column that matches the

thresholded test quantities. For example, consider the residual structure (3.17), and assume that fault f_3 is present. Especially for small faults, it can very well happen that $T_1 < J_1$, $T_2 < J_2$, and $T_3 > J_3$. However, this last fact conflicts with the rule (3.14b) and this has the consequence that the thresholded test quantities matches the column for fault f_1. Thus the residual structure (3.17) is not strongly isolating. Note that in the framework of structured hypothesis tests, we do not need to introduce requirements of a strongly isolating decision structure as a way to compensate for an unrealistic interpretation of the 1:s.

We end this section by discussing the last major difference between structured residuals and structured hypothesis tests. As seen in Section 3.4.2 above, there is a one-to-one correspondence between the representation based on the decision structure and a representation based on hypothesis tests, i.e. the sets S_k^0 and S_k^1. When using structured hypothesis tests, the interpretation of the 1:s corresponds well to standard conventions within general hypothesis testing literature. This makes it easy to relate to other traditional areas of fault diagnosis, e.g. statistical views, logic based methods. The structured residuals on the other hand, have an interpretation of 1:s that is not compatible with these standard conventions.

Concluding Remarks

We have concluded that in the method structured residuals, the 1:s in a residual structure, are interpreted as the 1:s in the decision structure, using the method structured hypothesis tests. This interpretation is however unrealistic since it claims that even small faults results in that the test quantity becomes above the threshold. The "unnatural" 1:s in structured residuals has three main consequences, which were all discussed above: (1) useful information is unnecessarily neglected, (2) the "ad-hoc" compensation of strongly isolating residual structure must be used, and (3), the thresholded test quantities can not be interpreted as standard hypothesis tests.

3.6 Conclusions

This chapter has refined the general diagnosis-system architecture from Chapter 2 by saying that the tests δ_k are *hypothesis* tests. We have formalized the procedure of how the diagnosis statement is formed from the real-valued test quantities (or residuals). This is achieved by using a standard interpretation of the functionality of each hypothesis tests. The formation of the diagnosis statement is then obtained in accordance with the function of the general diagnosis-system architecture from Chapter 2.

We have seen that the choice of null hypothesis in each hypothesis test is not a completely free choice, but is restricted by the submode relation between fault modes. Structured hypothesis tests can be used with arbitrary types of faults and this has been indicated in some examples. This topic will be further investigated in the next chapter where the design of the test quantities will be discussed.

In contrast to structured residuals, we have introduced a distinction between the incidence structure, describing how faults ideally affect the test quantities, and the decision structure, describing how the faults affect the formation of the diagnosis statement. By doing so, we have been able to define meanings of the 0:s, 1:s, and X:s, present in the incidence/decision structure. We have motivated that an introduction of X:s (don't care) in the incidence/decision structure is necessary since only using 0:s and 1:s often places unrealistic requirements on the test quantities (or residuals).

Chapter 4

Design and Evaluation of Hypothesis Tests for Fault Diagnosis

In the previous chapter, the diagnosis-system architecture *structured hypothesis tests* was proposed. To get a complete diagnosis system, the engineer has also to construct the individual hypothesis tests. In fact, this is a large portion of the total engineering work involved when constructing a diagnosis system. The question is how to use the model of the system, including the fault models, to design the best possible individual hypothesis tests. The topic of this chapter is to try to find some answers this question.

Design of hypothesis tests has been extensively discussed in general hypothesis testing literature, e.g. see (Lehmann, 1986). In this chapter we try to collect some general principles that are particularly useful for the purpose of model based diagnosis. We will see that the general framework of hypothesis testing brings structure to the field. Links between several different methods will become clear, for example: the likelihood principle from statistics vs residual generation, adaptive thresholds vs likelihood ratio, and parameter estimation methods vs residual generation.

Since the goal is to find "good" or "best" test quantities, we have to know what "good" or "best" means. Therefore we also discuss measures to evaluate hypothesis tests. Although many specific cases will be exemplified, the general principles, of how to design and evaluate the hypothesis tests, are valid for all kinds of fault models.

We start in Sections 4.1 to 4.4 to discuss general principles for test-quantity design. Three main principles are identified: the *prediction*, the *estimate*, and the *likelihood* principle. Then the issue of robustness is approached via *normalization* in Section 4.5. In Section 4.6, the measures for evaluating hypothesis tests are discussed. These measures are then used in Section 4.7 to select the parameters J_k, S_k^0, and S_k^1 of a hypothesis test. The evaluation measures are also used in Section 4.8 to compare the prediction and the estimate principle.

4.1 Design of Test Quantities

From the previous chapter, we realize that the assumption (or conclusion) we make when performing a hypothesis test δ_k, can be written

$$F_p \in \begin{cases} M_k^C & \text{if } T_k(x) \geq J_k \\ \Omega & \text{if } T_k(x) < J_k \end{cases} \tag{4.1}$$

where F_p denotes the present fault mode. In (4.1), we have again assumed that $S_k^0 = \Omega$. As said before, the test quantity $T_k(x)$ should be designed such that if the data x come from a system, whose present fault mode belongs to M_k^C, then $T_k(x)$ should be large. On the other hand, if the data x matches the hypothesis H_k^0, i.e. a fault mode in M_k can explain the data, then $T_k(x)$ should be small. This can be restated by using the notation of the model (2.6):

> The test quantity $T_k(x)$ should be small if the data x matches any of the models $\mathcal{M}_\gamma(\theta)$, $\gamma \in M_k$, and large otherwise.

Thus the test quantity can be seen as a measure of the validity of some models $\mathcal{M}_\gamma(\theta)$.

Several principles for constructing such measures exists and we will here discuss three of them: the *prediction principle*, the *estimate principle*, and the *likelihood principle*. These principles should be sufficient to solve most diagnosis problems. Note that although these principles are different, it can very well happen that, in some specific cases, the derived expressions for $T_k(x)$ equal each other.

4.1.1 Sample Data and Window Length

One way to define the sample data $\mathbf{x}$ is as a matrix:

$$\mathbf{x}(t) = \begin{bmatrix} u(t-N) & u(t-N+1) & \dots & u(t) \\ y(t-N) & y(t-N+1) & \dots & y(t) \end{bmatrix} \tag{4.2}$$

This corresponds to the use of a finite time window and as seen, the data $\mathbf{x}$ becomes a function of time t. This time window can be a sliding window, which means that consecutive data sets are overlapping. Another choice is to let consecutive data sets be non-overlapping.

The time window can also be infinite, at least conceptually. This corresponds to that $N = \infty$ in (4.2). In reality this means that all available data are used from the time-point when the diagnosis started (i.e. the window length is actually growing). An example of when an infinite time window is desirable, is when recursive techniques are used to calculate the test quantities. Another example is general residual generation which can be seen as a special case of the prediction principle. This will be further discussed in Section 4.2.2.

Theoretically, the optimal choice of window length is always infinite. This since it makes no sense to throw away any data, no matter what kind of data we have. However, if computational aspects are considered, it is often advantageous to use a finite window length.

4.2 The Prediction Principle

We will now discuss the prediction principle. In addition to giving general methods that can be used for test-quantity design, one purpose of this section is also to show how some well known approaches to fault diagnosis fit into the general framework proposed in this thesis.

Using the *prediction principle*, the calculation of the test quantity is based on a model validity measure $V_k(\theta, \mathbf{x})$ which in turn is based on a comparison between signals and/or predictions (or estimates) of signals. Typically an output signal y is compared with an estimate $\hat{y}$, but it is also possible to for example compare two estimates of the same signal.

To get a more precise definition, recall first the definition of Θ_k^0:

$$\Theta_k^0 = \bigcup_{\gamma \in M_k} \Theta_\gamma$$

Consider now the case where Θ_k^0 consists of several values θ. Using the prediction principle, the test quantity can be written as

$$T_k(x) = \min_{\theta \in \Theta_k^0} V_k(\theta, x) \tag{4.3}$$

The function $V_k(\theta, x)$, where θ is fixed, is a measure of the validity of the model $\mathcal{M}(\theta)$, for a fixed θ, in respect to the measurement data x. The test quantity $T_k(x)$ then becomes a measure of the validity of any the models $\mathcal{M}_\gamma(\theta)$, $\gamma \in M_k$, where θ is assumed free.

If Θ_k^0 consists of only one value θ_0, the test quantity becomes

$$T_k(x) = V_k(\theta_0, x) \tag{4.4}$$

and thus no minimization is needed.

To calculate (4.3), we need in principle to perform a parameter estimation. The prime interest here is fault isolation but it is obvious that this parameter estimation means that fault identification implicitly becomes a part of fault isolation. Note that the term *decoupling* in principle corresponds to estimation. The faults (or fault modes) that are decoupled are the fault modes described by the parameters we estimate.

Note that although the model validity measure $V_k(\theta, x)$ in (4.3) is indexed by k, meaning that it is specific for the hypothesis test δ_k, it is often possible (and also quite elegant) to use the same $V(\theta, x)$ for all hypothesis tests. In that case, the only thing that differs test quantities in different tests, is the set Θ_k^0 over which the minimization is performed. This approach will be discussed more in Chapter 5.

In *adaptive model based diagnosis*, we need to use *adaptive test quantities*. This means that the set of parameters we need to estimate is expanded to include also the unknown or uncertain parameters that we want to adapt to. Another case where the set of estimated parameters needs to be expanded, is when disturbances must be handled. From Section 2.1.1, we remember the

parameter ϕ which describes the disturbances, and decoupling of disturbances is therefore achieved by replacing (4.3) with

$$T_k(x) = \min_{\theta \in \Theta_k^0, \phi \in \Phi} V_k(\theta, \phi, x)$$

where Φ is the space of possible disturbances.

In general, one could think of several types of model validity measures, but the characteristic property of the prediction principle is that we let $V_k(\theta, x)$ be based on comparisons between signals and/or predictions of signals. One choice is to compare an output $y(t)$ with its prediction $y(t|\theta, x)$, derived from an assumption of a specific θ and the measured data x. That is, the model validity measure becomes the *prediction error* $y(t) - y(t|\theta, x)$. The principle to use the prediction error to calculate the test quantity is very natural and a so common choice, that we will denote it by its own name name: the *prediction error principle*. From now on, the focus will be mostly on this principle.

To reduce the sensitivity to noise and unmodeled disturbances it is advantageous to weight together several prediction errors. One possibility is to use a mean of some measure of prediction errors. This means that the function $V_k(\theta, x)$ becomes

$$V_k(\theta, x) = \frac{1}{N} \sum_{t=1}^{N} \|y(t) - \hat{y}(t|\theta, x)\| \tag{4.5}$$

For notational convenience, we have here assumed unit time. The measure $\|\cdot\|$ can for example be the quadratic norm. Another possibility is to first apply the sum operation and then the measure $\|\cdot\|$. Then the function $V_k(\theta, x)$ becomes

$$V_k(\theta, x) = \| \sum_{t=1}^{N} y(t) - \hat{y}(t|\theta, x) \ \| \tag{4.6}$$

It is also possible to use a measure dependent on time and/or the data itself. One reason would for example be that the model accuracy varies with the operating point of the system. Another case is when recursive parameter estimation is used. Recursive techniques implies that an infinite time-window is used and old data is by means of a time-dependent measure usually weighted less. These issues are thoroughly discussed in the general system identification literature, e.g. (Ljung, 1987).

The following four examples illustrates the prediction error principle for different types of fault modeling.

Example 4.1

Consider a system that can be modeled as

$$y(t) = gu(t) + b + v(t) \qquad v(t) \in N(0, \sigma) \qquad \theta = [b, g]$$

Assume that we want to consider three fault modes:

NF	$g = 1, b = 0$	"no fault"
F_b	$g = 1, b \neq 0$	"bias fault"
F_g	$g \neq 1, b = 0$	"gain fault"

Further we want to design a test quantity for the hypotheses

$$H^0 : F_p \in \{NF, F_b\}$$
$$H^1 : F_p = F_g$$

For these hypotheses, Θ^0 becomes $\Theta^0 = \{[b, g] \mid g = 1\}$. By using the formulas (4.3) and (4.5), we get

$$T(x) = \min_{\theta \in \Theta_k^0} \frac{1}{N} \sum_{t=1}^{N} \|y(t) - \hat{y}(t|\theta, x)\| = \min_b \frac{1}{N} \sum_{t=1}^{N} \big(y(t) - \hat{y}(t|b, x)\big)^2 \qquad (4.7)$$

The estimate $\hat{y}(t|b)$ (we have skipped the argument x) can be obtained as

$$\hat{y}(t|b) = u(t) + b$$

Inserting this expression into (4.7) means that the test quantity becomes

$$T(x) = \min_b \frac{1}{N} \sum_{t=1}^{N} (y(t) - u(t) - b)^2 \qquad (4.8)$$

The minimization is simple since it can be shown that the minimizing value of b is

$$\hat{b} = \frac{1}{N} \sum_{t=1}^{N} y(t) - u(t)$$

The test quantity (4.8) will be small under H^0 and thus the bias fault is decoupled in $T(x)$. ■

The following example illustrates how the prediction error principle can be applied to a change detection problem.

Example 4.2

Consider a signal $y(t)$ which can be modeled as

$$y(t) = v(t) + a(t)$$

where $v(t)$ is independent and $N(0, \sigma)$. The function $a(t)$ is $a(t) \equiv \mu_0 = 0$ in the fault free case, but can contain an abrupt change to an unknown value μ_1 if a fault occurs.

Assume that we want to consider three fault modes:

NF	"no fault"
F_μ	"an abrupt change in $a(t)$ at the time t_{ch}"
F_σ	"an abrupt change in standard deviation σ at the time t_{ch}"

This means that the fault-state vector can be described as $\theta = [t_{ch}, \mu, \sigma]$.

Further we want to design a test quantity for the following hypotheses:

$$H_0 : F_p \in \{NF, F_\mu\}$$
$$H_1 : F_p \in \{F_\sigma\}$$

By using the general expression (4.3), the test quantity becomes

$$T(x) = \min_{\theta \in \Theta^0} V(\theta, x) = \min_{[t_{ch}, \mu]} \sum_{t=1}^{N} (y(t) - \hat{y}(t|t_{ch}, \mu))^2$$

where

$$\hat{y}(t|t_{ch}, \mu) = \begin{cases} 0 & \text{if } t < t_{ch} \\ \mu & \text{if } t \geq t_{ch} \end{cases}$$

The test quantity can further be rewritten as

$$T(x) = \min_{t_{ch}} \Big(\sum_{t=1}^{t_{ch}} (y(t))^2 + \min_{\mu} \sum_{t=t_{ch}+1}^{N} (y(t) - \mu)^2 \Big)$$

■

The next example illustrates how test quantities can be designed in the case where one fault is modeled as an arbitrary input and another fault is modeled as a constant parameter. Also illustrated is how the submode relation from Section 2.4 affects the design.

Example 4.3

Consider a system that can be modeled as

$$x(t+1) = ax(t) + u(t)$$
$$y(t) = x(t) + f(t)$$

Assume that we want to consider three fault modes:

NF	$a = 0.5$	no fault
F_a	$a \neq 0.5$, $f(t) \equiv 0$	a fault in the dynamics
F_f	$a = 0.5$, $f(t) \not\equiv 0$	an arbitrary sensor fault

This definition of fault modes implies that the three fault modes are related as $NF \preccurlyeq^* F_a \preccurlyeq F_f$. According to the discussion in Section 3.2.1, the only possible choices of M_k are then $\{NF\}$, $\{NF, F_a\}$, and $\{NF, F_a, F_f\}$. The last one is useless for fault isolation and therefore we decide to design test quantities for two hypothesis tests with the hypotheses

$$H_1^0 : F_p \in \{NF, F_a\} \qquad H_1^1 : F_p = F_f$$
$$H_2^0 : F_p = NF \qquad H_2^1 : F_p \in \{F_f, F_a\}$$

The test quantity for the first test becomes

$$T_1(x) = \min_a \frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t|a))^2 = \frac{1}{N}\sum_{i=1}^{N}(y(t) - \hat{a}y(t-1) - u(t-1))^2$$

where $\hat{a}$ is the least square estimate of a. For the second test, the set Θ_2^0 contains only one element. Thus, the test quantity using the formula (4.4) becomes

$$T_2(x) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t))^2 = \frac{1}{N}\sum_{i=1}^{N}(y(t) - 0.5y(t-1) - u(t-1))^2$$

Now assume that the present fault mode is F_a and H_2^1 is accepted but H_1^0 is not rejected, i.e. $T_1 < J_1$ and $T_2 > J_2$. This will imply that the diagnosis statement becomes

$$S = \{NF, F_f, F_a\} \cap \{F_f, F_a\} = \{F_f, F_a\}$$

That is, both F_f and F_a can explain the process behavior. However, it is quite unlikely that the arbitrary fault signal $f(t)$ behaves in such a way that the process output matches the model $\mathcal{M}_{F_a}(\theta)$. Therefore, using a refined diagnosis statement in accordance with Section 2.6.1, we may draw the conclusion that the fault mode F_a is the one present in the process. ■

The following example shows how traditional in-range monitoring can be fitted into this framework using the prediction principle.

Example 4.4

Assume that under a no-fault situation, a state x is limited in range, $c_l < x < c_h$. Assume further that x is measured using a sensor y as $y(t) = x(t)$. If no more models are available, a prediction of $y(t)$ can in any case be written

$$\hat{y}(t|c) = c \qquad c_l < c < c_h$$

By using the general expression (4.3), the test quantity becomes

$$T(x) = \min_{c_l<c<c_h} V(c, x) = \min_{c_l<c<c_h} |y(t) - \hat{y}(t|c)|$$

This shows that traditional in-range testing can be seen as a special case of the prediction error principle. ■

The above example is also a clear illustration on how knowledge of range limitations of θ should be incorporated into the fault model to improve diagnosis performance. More specifically, without the knowledge $c_l < c < c_h$, the sensor y can not be diagnosed.

4.2.1 The Minimization of $V_k(\theta, x)$

The procedure to compute (4.3), i.e. to minimize $V_k(\theta, \mathbf{x})$, has not been addressed so far. The technical details are not going to be discussed here, but the interested reader is referred to general literature on optimization, e.g. (Luenberger, 1989), and system identification, e.g. (Ljung, 1987). In many cases the minimization procedure required in (4.3) is quite straightforward. However, in some cases, the computational load of doing the actual minimization in (4.3) can be quite heavy. One solution can be to use a *two-step approach*:

1. Find a $\hat{\theta}$ that minimizes another function $\bar{V}_k(\theta, \mathbf{x})$, i.e.

$$\hat{\theta} = \arg \min_{\theta \in \Theta_k^0} \bar{V}_k(\theta, \mathbf{x})$$

2. Calculate the test quantity as

$$T_k(\mathbf{x}) = V_k(\hat{\theta}, \mathbf{x}) \tag{4.9}$$

The point with this two-step approach is that $\bar{V}_k(\theta, \mathbf{x})$ can be chosen such that it is much easier to minimize compared to $V_k(\mathbf{x})$. Further, let $\bar{V}_k(\theta, \mathbf{x})$ be chosen such that the minimizing value $\hat{\theta}$, under H_k^0, is close to the value that minimizes $V_k(\theta_k, \mathbf{x})$. Then in the case H_k^0 holds, it is reasonable to assume that

$$\min_{\theta \in \Theta_k^0} V_k(\theta, \mathbf{x}) \approx V_k(\hat{\theta}, \mathbf{x})$$

This means that if we use the test quantity $T_k(\mathbf{x}) = V_k(\hat{\theta}_k, \mathbf{x})$, we can expect approximately the same result compared to if (4.3) was used.

Example 4.5

Consider a system that can be modeled as

$$y_1 = u + f_1 \tag{4.10}$$
$$y_2 = 2u + f_1 + f_2 \tag{4.11}$$

Assume that we want to consider three fault modes:

$$\begin{array}{ll} NF & f_1(t) \equiv 0, f_2(t) \equiv 0 \\ F_1 & f_1(t) \not\equiv 0, f_2(t) \equiv 0 \\ F_2 & f_1(t) \equiv 0, f_2(t) \not\equiv 0 \end{array}$$

Further we want to design a test quantity for a hypothesis test with the hypotheses

$$H_0 : F_p \in \{NF, F_1\}$$
$$H_1 : F_p \in \{F_2\}$$

Let $\mathbf{y} = [y_1\ y_2]^T$ and also let the predictions of y_1 and y_2 be $\hat{y}_1(\hat{f}_1) = u + \hat{f}_1$ and $\hat{y}_2(\hat{f}_1) = 2u + \hat{f}_1$. Then using the prediction error principle, the test quantity can be constructed as

$$\begin{aligned} T(x) &= \min_{f_1} V(f_1, x) = \min_{f_1} \big(\mathbf{y} - \hat{\mathbf{y}}(f_1)\big)^T \big(\mathbf{y} - \hat{\mathbf{y}}(f_1)\big) = \\ &= \min_{f_1} \big(y_1 - \hat{y}_1(f_1)\big)^2 + \big(y_2 - \hat{y}_2(f_1)\big)^2 \end{aligned} \tag{4.12}$$

The minimizing value of f_1 is $\hat{f}_1 = f_1 + f_2/2$. This implies that

$$\begin{aligned} T(x) &= \big(y_1 - u - f_1 - \frac{f_2}{2}\big)^2 + \big(y_2 - 2u - f_1 - \frac{f_2}{2}\big)^2 = \\ &= \big(u + f_1 - u - f_1 - \frac{f_2}{2}\big)^2 + \big(2u + f_1 + f_2 - 2u - f_1 - \frac{f_2}{2}\big)^2 = \frac{f_2^2}{2} \end{aligned}$$

Even though the minimization required in (4.12) is very simple, let us now consider a test quantity using the two-step approach. The estimate $\hat{f}_1$ is first found as

$$\hat{f}_1 = \arg\min_{\hat{f}_1} \bar{V}(\hat{f}_1, x) = \arg\min_{\hat{f}_1} \big(y_1 - \hat{y}_1(\hat{f}_1)\big)^2 = \arg\min_{\hat{f}_1} \big(u + f_1 - u - \hat{f}_1\big)^2$$

It is obvious that this will result in that $\hat{f}_1 = f_1$. The test quantity then becomes

$$\begin{aligned} T_{\text{2-step}}(x) &= V(\hat{f}_1, x) = V(f_1, x) = \big(y_1 - u - f_1\big)^2 + \big(y_2 - 2u - f_1\big)^2 = \\ &= 0 + f_2^2 = f_2^2 \end{aligned}$$

Under H_0, the minimizing value of $V(f_1, x)$ equals the minimizing value of $\bar{V}(f_1, x)$. Under H_0 it also holds that $T(x) = T_{\text{2-step}}(x)$. ■

From the above example it is clear that for $\theta \notin \Theta^0$, it can happen that

$$T(x) = \min_{\theta \in \Theta_k^0} V(\theta, x) < T_{\text{2-step}}(x) = V(\hat{\theta}, x) \tag{4.13}$$

and the difference can be significant. Note that this is acceptable as long as $T(x) \approx T_{\text{2-step}}(x)$ or, as in the example, $T(x) = T_{\text{2-step}}(x)$ for $\theta \in \Theta^0$. Moreover, this is actually an advantage of the two-step approach, since we want the test quantity to become as large as possible for $\theta \notin \Theta^0$. Thus the two-step approach has the potential to improve the test quantities.

4.2.2 Residual Generation

The term *residual generation*, as it is most often used in fault diagnosis literature, is a special case of the prediction error principle. Also the following restrictions are made:

- The faults are modeled as arbitrary inputs which are zero in the fault free case.
- A new value of the test quantity is calculated at every sample time-point. Also continuous time is often considered.
- A sliding time window is used and the length is finite or infinite.

When using residual generation, the test quantity is called *residual* (or residual generator). Linear residual generation is illustrated in the following two examples and will be further studied in Chapters 7 and 8.

Example 4.6

Consider a system that can be modeled as

$$y_1 = \frac{1}{q^{-1}+1}(u+f_1) \tag{4.14}$$

$$y_2 = \frac{1}{q^{-1}+2}(u+f_1)+f_2 \tag{4.15}$$

Assume that we want to consider three fault modes:

NF	$f_1(t) \equiv 0, f_2(t) \equiv 0$	no fault
F_1	$f_1(t) \not\equiv 0, f_2(t) \equiv 0$	actuator fault
F_2	$f_1(t) \equiv 0, f_2(t) \not\equiv 0$	fault in sensor 2

Further we want to design a test quantity for a hypothesis tests with the hypotheses

$$H_0 : F_p \in \{NF, F_1\}$$
$$H_1 : F_p \in \{F_2\}$$

A linear residual generator that can be used as a test quantity is

$$r = \frac{(q^{-1}+2)y_2 - (q^{-1}+1)y_1}{q^{-1}+3} \tag{4.16}$$

It will now be shown how the same test quantity can be obtained by using the general expression (4.5) for the prediction error principle.

We use the two-step approach and this means that we first have to estimate the parameter (now a signal) $f_1(t)$. From the model (4.14), the fault signal $f_1(t)$ can be estimated as

$$\hat{f}_1 = \arg\min_{f_1} \left(y_1 - \frac{1}{q^{-1}+1}(u+f_1)\right)^2 = (q^{-1}+1)y_1 - u \tag{4.17}$$

With this estimate and by using the general expression (4.5), the test quantity, using an infinite window length, can be formed as

$$T_1(x) = V(\hat{f}_1, x) = \sum_{t=0}^{\infty} \|y_2(t) - \hat{y}_2(t|\hat{f}_1)\|$$

By means of the estimate (4.17), the prediction error can be expressed as

$$y_2 - \hat{y}_2(\hat{f}_1) = y_2 - \frac{1}{q^{-1}+2}(u + \hat{f}_1) = y_2 - \frac{q^{-1}+1}{q^{-1}+2}y_1$$

Then choose the measure $\|\cdot\|$ as

$$\sum_{n=0}^{\infty} c_n q^{-1}(\,\cdot\,)$$

where

$$\sum_{n=0}^{\infty} c_n q^{-1} = \frac{q^{-1}+2}{q^{-1}+3}$$

This means that

$$T_1(x) = \sum_{n=0}^{\infty} c_n q^{-1}\big(y_2(t) - \hat{y}_2(t|\hat{f}_1)\big) = \frac{q^{-1}+2}{q^{-1}+3}\big(y_2(t) - \frac{q^{-1}+1}{q^{-1}+2}y_1(t)\big) = r$$

We have thus shown how the residual generator (4.16) can be obtained in the framework of the prediction error principle. Note that the sign of $T_1(x)$ can be negative and thus, it is the absolute value of $T_1(x)$ that should be thresholded. ■

Example 4.7

Assume that we have a non-linear model

$$\dot{x} = f(x,u) \tag{4.18}$$
$$y_1 = h_1(x,u) + f_1 \tag{4.19}$$
$$y_2 = h_2(x,u) \tag{4.20}$$

Here f is a signal modeling a fault in sensor 1. Then assume that an observer for x can be constructed as

$$\dot{\hat{x}} = f(\hat{x},u) + K\big(y_2 - h_2(\hat{x},u)\big) \tag{4.21}$$

Then

$$r = y_2 - \hat{y}_2 = y_2 - h_2(\hat{x},u) \tag{4.22}$$

is a residual generator which will be insensitive to faults in sensor 1. This means that the corresponding null hypothesis is described by $M_k = \{NF, F_1\}$ where F_1 is the fault mode for $f_1 \not\equiv 0$. Obviously, r is also a test quantity that is naturally constructed with the prediction error principle in accordance with formula (4.5).

According to the expression (4.3), the parameter f_1 should be implicitly estimated when calculating the test quantity. This is not the case for the test quantity (4.22). However, it is possible to derive the expression (4.22) by using (4.3) and the two-step approach. First let f_1 be estimated as

$$\hat{f}_1 = \arg\min_{f_1}(y_1 - \hat{y}_1)^2 = \arg\min_{f_1}(y_1 - h_1(x,u) + f_1)^2 = y_1 - h_1(\hat{x},u)$$

Then in accordance with the formula (4.5), the test quantity becomes

$$\begin{aligned} T(x) = V(\hat{f}_1, x) = |\mathbf{y} - \hat{\mathbf{y}}| = \left| \begin{matrix} y_1 - \hat{y}_1(\hat{f}) \\ y_2 - \hat{y}_2 \end{matrix} \right| = \left| \begin{matrix} y_1 - h_1(\hat{x},u) - \hat{f} \\ y_2 - h_2(\hat{x},u) \end{matrix} \right| = \\ = \left| \begin{matrix} y_1 - h_1(\hat{x},u) - y_1 + h_1(\hat{x},u) \\ y_2 - h_2(\hat{x},u) \end{matrix} \right| = \left| \begin{matrix} 0 \\ y_2 - h_2(\hat{x},u) \end{matrix} \right| = |y_2 - h_2(\hat{x},u)| = |r| \end{aligned}$$

■

4.3 The Likelihood Principle

When the probability density functions of the noise is known, or can be assumed to be known, it is possible to use the likelihood principle. The likelihood principle is based on the likelihood function which is defined as

Definition 4.1 (Likelihood Function) *Let $f(\mathbf{x}|\theta)$ denote the probability density function of the sample $\mathbf{X} = [X_1,\ X_2,\ \ldots\ X_n]$. Then, given that $\mathbf{X} = \mathbf{x}$ is observed, the function of θ defined by*

$$L(\theta|\mathbf{x}) = f(\mathbf{x}|\theta) \tag{4.23}$$

is called the likelihood function.

Given a model, it is possible to set up a likelihood function which become a measure for how well the measured data matches the model. Recall from Section 4.1 that this is exactly what we want when constructing test quantities. This is also the reason why likelihood functions are a common choice for test quantities in general statistical hypothesis testing. Thus, using the likelihood principle, the measure $V_k(\theta, x)$ in (4.3) corresponds to $L(\theta|\mathbf{x})$. In contrast to the prediction error principle, the likelihood function becomes large when measurement data matches the model and small when the data does not match the model. When using the likelihood principle, the null hypothesis H_k^0 is rejected if $T_k(x) < J_k$. Note that $>$ has been changed to $<$, compared to previous cases.

If the set Θ_k^0 consists of only one element, then the likelihood function (4.23) can be used directly as a test quantity. When Θ_k^0 consists of several elements, we

have to use optimization in accordance with (4.3). However, since the likelihood function becomes large when measurement data matches the model, the minimization must be replaced by maximization. The test quantity then becomes

$$T_k(\mathbf{x}) = \max_{\theta \in \Theta_k^0} L(\theta|\mathbf{x}) \tag{4.24}$$

This principle is usually called the *maximum likelihood principle*. The two-step approach described in the context of the prediction principle is of course possible to use also for the maximum likelihood.

Often it is assumed that the data are independent and identically distributed such that

$$f(\mathbf{x}|\theta) = \prod_{i=1}^{N} f(x_i|\theta)$$

Here x_i means $x(t_i)$. This means that the likelihood function becomes

$$L(\theta|\mathbf{x}) = \prod_{i=1}^{N} f(x_i|\theta)$$

and thus, much simpler to calculate.

A further simplification is obtained by using the *log-likelihood function* defined as

$$l(\theta|\mathbf{x}) = \ln\ L(\theta|\mathbf{x})$$

If the assumption about independent data is used, we get

$$l(\theta|\mathbf{x}) = \ln\ L(\theta|\mathbf{x}) = \ln \prod_{i=1}^{N} f(x_i|\theta) = \sum_{i=1}^{N} \ln\ f(x_i|\theta)$$

Note that since the logarithm function $\ln(x)$ is monotone, a hypothesis test based on the log-likelihood function $l(\theta|\mathbf{x})$ is equivalent to a test based on the basic likelihood function $L(\theta|\mathbf{x})$.

Example 4.8

Consider again Example 4.1 but instead of (4.7), we use the likelihood principle to obtain the test quantity. Let x_i denote $y(i) - u(i)$ which means that $x_i \sim N(b, \sigma)$. The test quantity then becomes

$$T(x) = \max_{\theta \in \Theta^0} L(\theta|x) = \max_b \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\{-\frac{(x_i - b)^2}{2\sigma^2}\}$$

The log-likelihood version of this test quantity becomes

$$T'(x) = \max_{\theta \in \Theta^0} l(\theta|x) = \max_b \sum_{i=1}^{N} \ln\ \frac{1}{\sigma\sqrt{2\pi}} \exp\{-\frac{(x_i - b)^2}{2\sigma^2}\} =$$

$$= \max_b -N\ \ln\ \sigma\sqrt{2\pi} - \frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - b)^2$$

Note that the last expression contains a constant term. This term can be neglected and the remaining expression is then equivalent to.

$$\min_{\theta\in\Theta_k}\sum_{i=1}^{N}(x_i-\theta)^2$$

Now note that it happens to be the case that this expression is equal to the expression obtained in Example 4.1. This means in this particular problem, the prediction error principle and the likelihood principle are equivalent. ■

The drawback with the likelihood principle, compared to the prediction error principle, is that to make the calculations tractable, we must usually assume that the data are independent and normally distributed. On the other hand the likelihood principle is very universal. It can for example easily handle faults that are modeled as an increase in the variance of a signal.

4.4 The Estimate Principle

Both the prediction error end the likelihood principle are based on the idea that the test quantity should be a model validity measure. A somewhat different approach to construct the test quantity is the estimate principle. We have seen that in both the prediction error and the likelihood principle, it is common that a parameter estimation is involved. The idea of the estimate principle is to construct a test quantity that more directly uses the estimated parameter. Note however that the principle goal of that the test quantity should be a model validity measure, is still the same.

One solution is to estimate a component fault state θ_i and then compare it with the nominal value θ_i^0. This means that the set $\mathcal{D}_{NF}^i$ must contain only one element, i.e. $\mathcal{D}_{NF}^i=\{\theta_i^0\}$. First consider the case where the set Θ_k^0 consists of only one element. Then a test quantity can be constructed as

$$T_k(x)=\|\hat{\theta}_i-\theta_i^0\| \qquad \hat{\theta}_i=\arg\min_{\theta_i\in\mathcal{D}^i} V'(\theta_i,x) \tag{4.25}$$

where $V'(\theta_i,x)$ is some model validity measure. This is a common solution used in literature, e.g. (Isermann, 1993). The measure $\|\cdot\|$ can for example be the quadratic norm.

When the set Θ_k^0 consists of more than one element, the test quantity can be constructed as

$$T_k(x)=\|\hat{\theta}_i-\theta_i^0\| \qquad \hat{\theta}_i=\arg\min_{\theta_i}\min_{\theta\in\Theta_k^0\cup\bar{\Theta}_i} V'(\theta,x) \tag{4.26}$$

where $\bar{\Theta}_i=\{\theta \mid \theta_i\in\mathcal{D}^i,\theta_{j\neq i}\in\mathcal{D}_{NF}^j\}$. That is, in addition to estimate the parameter θ_i we also have to estimate the free parameters in Θ_k^0, i.e. the ones corresponding to faults that are decoupled. For an illustration of this technique, see the test quantity (4.49) in Section 4.8.1.

Note that compared to when using the prediction and likelihood principle, one extra parameter must be estimated. That is, in addition to all parameters that we want to decouple, we also need to estimate the parameter that is used in the test quantity. This implies that decoupling might be more difficult for the estimate principle.

We will see later in Section 4.5 that test quantities based on estimates may imply that the test quantity under H_0 is dependent on u. In that case, it must be *normalized* which is also described in Section 4.5.

The estimate principle has both advantages and disadvantages compared to the prediction error end the likelihood principle. Test quantities based on estimates can have very good performance for the fault mode corresponding to the estimated parameter. However for other fault modes, the performance might be quite bad and also highly dependent on the input signal. This is investigated more in Section 4.8.

4.5 Robustness via Normalization

When constructing test quantities, a goal is that they should be insensitive to *uncontrolled effects* such as changes in inputs u and state x, disturbances d, model errors, etc. Sometimes, the constructed test quantities meet these goals but often they do not. The reasons why the test quantities become sensitive to uncontrolled effects are

- Approximate decoupling. Because of fundamental limitations it is sometimes impossible to completely decouple disturbances and effects of faults (i.e. the faults belonging to fault modes in the null hypothesis).
- Model Errors. Most unmodeled disturbances, incorrect model structure, and unmodeled noise etc. implies that the performance of the test quantities is degraded. The most serious problem is usually that the significance level is raised.
- Modeled noise. Even though noise terms are included in the model, it is mostly impossible to avoid that the noise is going to affect the test quantities.

The discussion above is closely related to the issue of *robustness*. More exactly, robustness can be defined as the ability of the test quantities to satisfy some specific performance goals while the uncontrolled effects are present to a certain degree. In connection with linear residual generation, methods to achieve and analyze robustness have been extensively studied, e.g. see (Chen and Patton, 1999)(Frisk and Nielsen, 1999). In many of these methods, the robustness issue is hardly integrated as a part of the design process for the test quantities. A somewhat different approach is to first design the test quantity without robustness considerations and then afterwards consider robustness as an additional design step by adjusting and compensating the originally designed test quantity. It is interesting to note that there are experimental results showing

the advantage of the latter robustness approach, e.g. (Höfling and Isermann, 1996), while the literature is very sparse on experimental experience with the former robust method.

As a way to achieve and improve robustness by adjusting and compensating already designed test quantities, we will here consider *normalization*. Normalization is to compensate the test quantity for unmodeled effects by multiplying it with a cleverly chosen variable that is a function of the measured data x. Here we investigate normalization for the estimate principle, prediction principle and the likelihood principle.

4.5.1 The Estimate Principle

The discussion here will be limited to an example.

Example 4.9

Consider a system which can be modeled as

$$y(t) = bu(t) + v(t)$$

where $v(t) \sim N(0, \sigma_v)$. The nominal (i.e. corresponding to the no fault case) value of b is b_0. We will use the notation U, Y, and V to denote column vectors of u, y, and v respectively.

Assume a test quantity based on the estimate principle:

$$T_2(x) = (\hat{b} - b_0)^2 \qquad \hat{b} = \frac{1}{U^T U} U^T Y \tag{4.27}$$

where $\hat{b}$ is the least square estimate of b. Consider the fault free case, i.e. $b = b_0$, which means that

$$\hat{b} - b_0 = \frac{1}{U^T U} U^T (U + V) - 1 = b_0 - 1 + \frac{1}{U^T U} U^T V = \tag{4.28}$$

$$= \frac{1}{Np} U^T V \quad \sim N(0, \frac{\sigma_v}{\sqrt{Np}}) \tag{4.29}$$

where $Np = U^T U$, and p is the mean power of u. We see that $\hat{b} - 1$ has a standard deviation that is dependent on u. If the mean power of u varies, this is undesirable since the significance level of a hypothesis test will then depend on u. The solution is to use normalization and we multiply therefore (4.28) with $\sqrt{Np}$. Then we have that

$$\sqrt{Np}(\hat{\theta} - 1) \ \sim N(0, \sigma_v) \tag{4.30}$$

The corresponding normalization for the test quantity (4.27) becomes

$$T_2'(x) = Np(\hat{\theta} - 1)^2 \qquad \hat{b} = \frac{1}{U^T U} U^T Y \tag{4.31}$$

Thus, using (4.31) means that a fixed threshold will imply a fixed significance level independent on u. ■

In terms of robustness, a hypothesis test based on the normalized test quantity (4.31) and with a fixed threshold, will satisfy the performance goal that the significance level must not be above a certain level. This will hold for any u. However, there is no guarantee that other performance goals, such as the probability of $T_2'(x) < J_2$ when a fault is present, are satisfied.

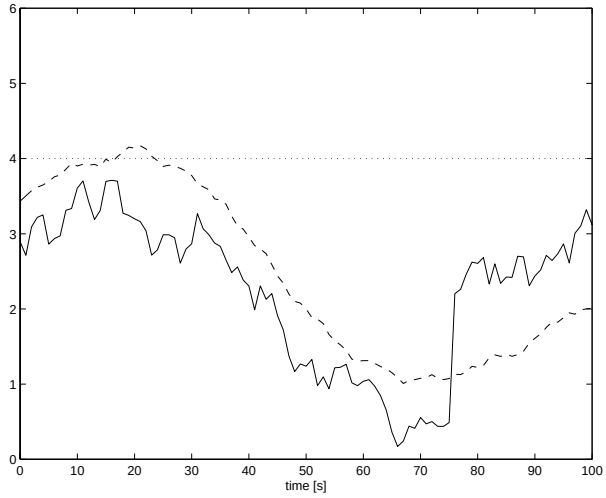

Figure 4.1: An example of the use of an adaptive threshold, with the test quantity (solid), the adaptive threshold (dashed), and as a comparison, the fixed threshold (dotted).

4.5.2 The Prediction Principle and Adaptive Thresholds

The basic idea of adaptive thresholds is that since disturbances and other uncontrolled effects vary with time, also the thresholds should vary with time instead of being fixed to a constant value. An example is shown in Figure 4.1. The solid line represents the a test quantity, the dashed line is the adaptive threshold, and the dotted line the fixed threshold. There is a fault occuring at time $t = 75$ s, but because of disturbances, the test quantity is above zero also before this time-point. To avoid false alarm, the fixed threshold has been set high. This means that the fault is missed if the fixed threshold is used. The adaptive threshold "adapts" to the disturbances and therefore follows the test quantity as long as there are no faults. When the fault occurs, the residual crosses the threshold and the fault is detected.

One technique for computing adaptive thresholds in connection with linear residual generation, is presented in (Ding and Frank, 1991). Consider a system which can be described as

$$y = \big(G(s) + \Delta G(s)\big)u + G_d(s)d + G_f(s)f + v$$

where $\Delta G(s)$ is a model error, u is the input, d is the disturbance, f is the fault, and v is measurement noise. Consider then a residual described by

$$\begin{aligned} r =& H_y(s)y + H_u(s)u = \\ &= H_y(s)\big(G(s)u + \Delta G(s)u + G_d(s)d + G_f(s)f + v\big) + H_u(s)u \end{aligned}$$

If measurement noise v is neglected and it is assumed that the input u and the disturbance d are perfectly decoupled, then in the fault free case, the residual becomes

$$r = H_y(s)\Delta G(s)u$$

It is seen that the size of the residual in the fault free case depends on the absolute size of the model error $\Delta G(s)$ and the input $u(t)$. If $\delta > \|\Delta G(s)\|$ denotes a known bound of $\Delta G(s)$, the adaptive threshold can be selected as

$$J_{adp}(t) = \delta\|H_y(s)u\| \tag{4.32}$$

This approach relies on that a bound on the model uncertainty can be determined with confidence. If this is the case, it is guaranteed that no false alarm, caused by model uncertainties, will be generated.

Another approach is proposed in (Höfling and Isermann, 1996). This approach is more ad-hoc because the computation of the adaptive threshold is determined by tuning some design parameters. On the other hand, it is probably more generally applicable because it can handle more kinds of disturbances, i.e. not only model uncertainties. A generalized description of how the threshold is computed is the non-linear expression

$$J_{adp}(t) = kH_{LP}(s)\big(|H_d(s)u(t)| + c\big) \tag{4.33}$$

where $H_{LP}(s)$ and $H_d(s)$ are linear filters, and k and c constants. The filter $H_d(s)$ functions as a weighting, in the frequency domain, of model uncertainties. For frequency ranges where the model uncertainty is high, the filter gain should be high and vice versa. For example if the model is good for low frequencies but uncertain for higher frequencies, the filter $H_d(s)$ should be a high-pass filter. The value of the constant c is determined by the amount of other kinds of disturbances, such as measurement noise, and makes the threshold become greater than zero even though the input is zero. Finally $H_{LP}(s)$ is a low-pass filter for smothering of the threshold.

By using adaptive thresholds according to the principles described above, it is possible to get a nearly fixed significant level, independent on changes in the input signal. In this sense, the adaptive threshold is similar to the normalization described for the estimate principle. Robustness is achieved in the sense that a certain significant level can be guaranteed independently of the input. Note however that if overall performance gains are desirable, these robustness techniques are never a substitute for using better models.

Both kinds of adaptive thresholds, i.e. (4.32) and (4.33), can be written on the more general form

$$J_{adp} = c_1 W(u, y) + c_2 \tag{4.34}$$

where $W(u, y)$ is some measure of the model uncertainty present for the moment.

To use an adaptive threshold is equivalent to *normalize* the test quantity. Consider the use of a test quantity $T(x)$ in combination with the threshold (4.34):

$$T(x) = T(x) \geq J_{adp} \qquad \text{(reject } H_0\text{)}$$

By using normalization, this relation can instead be written as

$$T'(x) = \frac{T(x)}{c_1 W(u, y) + c_2} \geq 1 \qquad \text{(reject } H_0\text{)}$$

where $T'(x)$ is the normalized test quantity. The new threshold becomes $J = 1$.

Two measures $W(u, y)$ of the model uncertainty are implicitly given in the expressions (4.32) and (4.33). Another alternative is to use a minimized sum of prediction errors:

$$W(u, y) = \min_{\theta \in \Theta} V(\theta, x) = \min_{\theta \in \Theta} \sum_{t=1}^{N} (y(t) - \hat{y}(t|\theta))^2 \tag{4.35}$$

Note that the minimization is over *all* possible θ. The expression 4.35 might seem to be difficult to calculate but if the same $V(\theta, x)$ is used for all hypothesis tests, as was described in Section 4.2, then the calculation of 4.35 becomes easy. This will be demonstrated in Section 5.8.

Now assume that $c_2 = 0$. Then an adaptive threshold becomes

$$J_{adp} = \min_{\theta \in \Theta} W(\theta, x)\ c_1 \tag{4.36}$$

With this adaptive threshold, the normalized version of a test quantity based on the expression (4.3) becomes

$$T'(x) = \frac{\min_{\theta \in \Theta_k^0} V(\theta, x)}{\min_{\theta \in \Theta} V(\theta, x)} > c_1 \qquad \text{(reject } H_0\text{)}$$

We will see that this expression has strong similarities with the likelihood ratio described next.

4.5.3 The Likelihood Principle and the Likelihood Ratio

Now consider the likelihood principle and an adaptive threshold similar to the one defined by (4.35) and (4.36):

$$J_{adp} = \max_{\theta \in \Theta} L(\theta|x)\ c_1$$

Thus H_0 is rejected if

$$T(x) = \max_{\theta \in \Theta^0} L(\theta|x) < \max_{\theta \in \Theta} L(\theta|x)\ c_1$$

By using normalization, we get a new test quantity $T'(x)$ and H_0 is now rejected if

$$T'(x) = \frac{\max_{\theta\in\Theta^0} L(\theta|x)}{\max_{\theta\in\Theta} L(\theta|x)} < c_1 \tag{4.37}$$

The test quantity $T'(x)$ is called the *likelihood ratio* test quantity (or statistic). To emphasize that a maximization is involved, the term *maximum likelihood ratio* or *generalized likelihood ratio* is also used in the literature.

A number of different variations of the maximum likelihood ratio exists. One variation is to switch the numerator and the denominator. Another is to make the maximization in the denominator of (4.37) over $\Theta^1 = \Theta^{0^C}$ instead of Θ. It can be shown that in this case, it is equivalent to make the the maximization over Θ (Lehmann, 1986). Further, the maximization is often replaced by supremum. Two examples of variations are

$$T(x) = \frac{\sup_{\theta\in\Theta} L(\theta|x)}{\sup_{\theta\in\Theta^0} L(\theta|x)} \tag{4.38}$$

$$T(x) = \frac{\max_{\theta\in\Theta^1} L(\theta|x)}{\max_{\theta\in\Theta^0} L(\theta|x)} \tag{4.39}$$

The likelihood ratio test quantity is widely used in statistics. The reason is partly that it is the optimal test quantity in the case both the null hypothesis and the alternative hypothesis are simple, i.e. Θ^0 and Θ^1 consists each of only one element (Neyman-Pearson lemma). Optimality proofs also exists for many other cases where H^0 is simple (P.H.Garthwaite, 1995). For many cases where a theoretical justification is missing, the likelihood ratio has still been shown to be very good in practice (Lehmann, 1986). However, there are also cases for which the likelihood ratio is *not* good (Lehmann, 1986).

Commonly the *maximum log-likelihood ratio* is used. This together with a change detection application is illustrated in the following example:

Example 4.10

Consider a signal $x(t)$ which can be modeled as

$$x(t) = v(t) + \theta(t)$$

where $v(t)$ is independent and $N(0, \sigma)$. Before the change-time t_{ch}, $\theta(t) = 0$ and after the change time, $\theta(t) = \mu$.

The following two hypotheses are considered:

H_0 :	"no change in mean μ of $x(t)$ occurs"
H_1 :	"an abrupt change in mean μ occurs"

By using the assumption of independent data, the likelihood ratio test quantity

on the form (4.38) becomes

$$T(x) = \frac{\sup_{\theta\in\Theta} L(\theta|x)}{\sup_{\theta\in\Theta_0} L(\theta|x)} = \frac{\sup_{[t_{ch},\mu]} L([t_{ch},\mu]|x)}{L([N,0]|x)} =$$
$$= \frac{\sup_{[t_{ch},\mu]} \prod_{i=0}^{t_{ch}-1} f(x(i)|0) \prod_{i=t_{ch}}^{N} f(x(i)|\mu)}{\prod_{i=0}^{N} f(x(i)|0)} = \sup_{[t_{ch},\mu]} \frac{\prod_{i=t_{ch}}^{N} f(x(i)|\mu)}{\prod_{i=t_{ch}}^{N} f(x(i)|0)}$$

Now by using the assumption of Gaussian data, and switching to the log-likelihood ratio, we get the following test quantity:

$$T'(x) = \sup_{[t_{ch},\mu]} \ln \frac{\prod_{i=t_{ch}}^{N} f(x(i)|\mu)}{\prod_{i=t_{ch}}^{N} f(x(i)|0)} =$$
$$= \sup_{[t_{ch},\mu]} \sum_{i=t_{ch}}^{N} \ln f(x(i)|\mu) - \sum_{i=t_{ch}}^{N} \ln f(x(i)|0) =$$
$$= \sup_{[t_{ch},\mu]} -\frac{1}{2\sigma^2} \sum_{i=t_{ch}}^{N} (\mu - 2x(i))\mu =^{*} \sup_{t_{ch}} \sup_{\mu} -\frac{1}{2\sigma^2} \sum_{i=t_{ch}}^{N} (\mu - 2x(i))\mu =$$
$$= \frac{1}{2\sigma^2} \sup_{t_{ch}} \sup_{\mu} -(N - t_{ch} + 1)\mu^2 + 2\mu \sum_{i=t_{ch}}^{N} x(i)$$

The equality marked with $=^*$ can be shown to hold in special cases, including this one, but is not generally valid. ■

Note the relation between this example and Example 4.2, where a similar problem was solved by using the prediction error principle.

4.6 Evaluation of Hypothesis Tests Using Statistics and Decision Theory

The basic concepts presented in this section are probably well known to statisticians and decision theorists. However, because of their usefulness for fault diagnosis problems, especially in the view of this thesis, they deserve some attention. The performance measures used for evaluation here are *risk functions* and power functions. There exists also other performance measures, e.g. the ARL function (Basseville and Nikiforov, 1993).

When the null hypothesis H_k^0 is true, we want to *not* reject H_k^0. The mistake to reject H_k^0 when H_k^0 is true is called a TYPE I error. Similarly, to not reject H_k^0 when the alternative hypothesis H_k^1 is true is called a TYPE II error. In fault diagnosis, there is a connection between these errors and the probability of false alarm, missed detection, and missed isolation. We will not go into these details here but this connection will be discussed in Chapter 6. At this point, it is at least clear that to achieve low probabilities of false alarm, missed detection,

and missed isolation, we need to keep the probabilities of the TYPE I and II errors low.

Thus the probabilities of the TYPE I and II is a kind of performance measure for a single hypothesis test. However a more precise measure is the power function or more generally a *risk function* from decision theory (Berger, 1985). The risk function is obtained by first defining a *loss function*. A loss function $\mathcal{L}(\theta, S_k)$ should reflect the "loss" for a given specific fault state and a specific decision S_k of the hypothesis test δ_k. The loss function for the hypothesis test δ_k can be defined as

$$L_k(\theta, S_k) = \begin{cases} 0 & \text{if } \theta \in \Theta_k^0 \text{ and } S_k = S_k^0 \\ 0 & \text{if } \theta \notin \Theta_k^0 \text{ and } S_k = S_k^1 \\ c_I(\theta) & \text{if } \theta \in \Theta_k^0 \text{ and } S_k = S_k^1 \\ c_{II}(\theta) & \text{if } \theta \notin \Theta_k^0 \text{ and } S_k = S_k^0 \end{cases} \tag{4.40}$$

where the functions $c_I(\theta)$ and $c_{II}(\theta)$ are chosen by the user to for example indicate that some faults are more important to detect that other. In Section 6.1.1, we will use the functions $c_I(\theta)$ and $c_{II}(\theta)$ to distinguish between *significant* and *insignificant* faults.

From decision theory, the definition of risk function is as follows:

Definition 4.2 (Risk Function) *The* risk function $R(\theta, \delta)$ *of a decision rule* $\delta(x)$ *is*

$$R(\theta, \delta) = E_\theta\{\mathcal{L}(\theta, \delta(X))\}$$

where E_θ *denotes expectation for a fixed* θ*.*

With the loss function (4.40), the risk function $R(\theta, \delta_k)$ becomes

$$R(\theta, \delta_k) = \begin{cases} 0 \cdot P(S_k = S_k^0|\theta) + c_I(\theta)P(S_k = S_k^1|\theta) & \text{if } \theta \in \Theta_k^0 \\ c_{II}(\theta)P(S_k = S_k^0|\theta) + 0 \cdot P(S_k = S_k^1|\theta) & \text{if } \theta \notin \Theta_k^0 \end{cases}$$

Recall from the previous chapter, the definition of power function:

$$\beta_k(\theta) = P(\text{reject } H_k^0 \mid \theta) = P(T_k(x) \geq J_k \mid \theta)$$

By using the power function, the risk function can be written as

$$R(\theta, \delta_k) = \begin{cases} c_I(\theta)\beta_k(\theta) & \text{if } \theta \in \Theta^0 \\ c_{II}(\theta)\big(1 - \beta_k(\theta)\big) & \text{if } \theta \notin \Theta^0 \end{cases} \tag{4.41}$$

Thus to get a usable performance measure of δ_k, we need to define the functions $c_I(\theta)$ and $c_I(\theta)$ and also know the power function $\beta_k(\theta)$. Commonly a so called 0-1 loss is considered. This means that $c_I(\theta) = c_I(\theta) = c$.

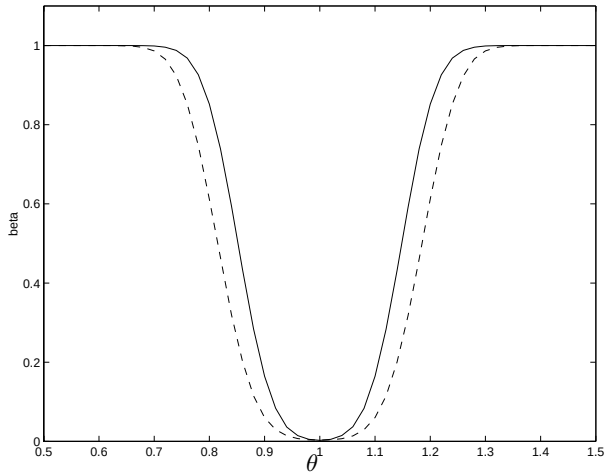

Figure 4.2: Two power functions.

4.6.1 Obtaining the Power Function

Either we want to use the risk function (4.41) as a performance measure or alternatively, the power function alone, we need the power function. As was said in Section 3.2, the power function is also used to calculate the significance level α.

The power function can in rare cases be derived analytically. Commonly we need to assume independent data which is also Gaussian distributed. An analytical derivation of a power function is demonstrated in the following example.

Example 4.11

Consider again Example 4.9. To use (4.31) in a hypothesis test is equivalent to using (4.30). Since the distribution of (4.30) $\hat{b}$ is known, it is easy do derive the power function for a test based on the test quantity (4.31). In Figure 4.2 this power function is plotted as a solid line. ■

In cases where it is not possible to derive the power function analytically, but the distribution of the measured data is known, we can use (Monte Carlo) simulations. Another method is to estimate the power function $\beta(\theta)$ by using measurements on the real process. The method is similar to simulations but we do not need to know or assume any distribution of the measured data. This method can be described as follows:

1. To calculate $\beta(\theta)$ for a specific θ, we manipulate the process such that the fault state θ is obtained.

2. Collect a number of measurement series x_i, $i = 1, \ldots N$. Each x_i is an instance of x, i.e. a matrix of inputs u and outputs y from different times in accordance with Section 4.1.1.

3. For each data series x_i, calculate the value t_i of the test quantity, i.e. $t_i = T(x_i)$.

4. Collect all the N values t_i in a histogram. This histogram is now an estimation of the probability density function $f(t|\theta)$.

5. By using a fixed threshold J_k, $\beta(\theta)$ can now be estimated.

This procedure can be repeated for a number of different θ:s, and thereby the power function $\beta(\theta)$ can be obtained as a sampled function of θ.

4.6.2 Comparing Test Quantities

The risk function alone can be used to compare different hypothesis tests. However it is needed that thresholds are defined. Thus to compare test quantities we must first specify thresholds. When a 0-1 loss is chosen, the performance of a hypothesis test can equally well be described by its power function. However if a more general loss is used, we may have to consider the risk function.

Consider first the case of a 0-1 loss and that we want to compare two test quantities $T_1(x)$ and $T_2(x)$. A hypothesis test for each test quantity is constructed and the thresholds are chosen such the significance levels equal each other. Both the power functions $\beta_1(\theta)$ (dashed) and $\beta_2(\theta)$ (solid) can then be calculated and studied. In Figure 4.2, an example of two power functions are plotted. The set Θ_k^0 is assumed to be $\Theta_k^0 = \{1\}$. From this plot we can conclude that the test based on $\beta_2(\theta)$ is better than the test based on $\beta_1(\theta)$. This is because $\beta_2(\theta) > \beta_1(\theta)$ for all θ except $\theta = 1$.

Now assume that $c_I(\theta)$ and $c_{II}(\theta)$ are not constants. Then for a case where $\beta_2(\theta) \geq \beta_1(\theta)$ for all θ, the decision theoretic view of studying the risk $R(\theta, \delta_k)$ is equivalent to only studying the power function. The reason is that for each value θ, the functions $c_I(\theta)$ and $c_I(\theta)$ only affects as a scaling factor. However, if it is the case that $\beta_2(\theta) \geq \beta_1(\theta)$ for only some θ, we could not tell which test is the best. Then other principles have to be used and the functions $c_I(\theta)$ and $c_{II}(\theta)$ may then play a more important role. This will to some extent be discussed in Chapter 6.

4.7 Selecting Parameters of a Hypothesis Test

Except for constructing the test quantity, we need to select the threshold. For a complete hypothesis test δ_k we need also to define the decisions S_k^0 and S_k^1. These "parameter" choices are discussed in this section.

4.7.1 Selecting Thresholds

The selection of thresholds in each test, largely affects the performance of the hypothesis tests and the diagnosis system. To analyze this, we will study how the risk function is affected when varying the thresholds. The threshold J_k will be regarded as a design parameter of the hypothesis test and to denote the hypothesis test, we will therefore use the notation $\delta_k(J_k)$. The risk function then becomes $R(\theta, \delta_k(J_k))$, i.e a function of two variables, θ and J_k. If a 0-1 loss is used, the risk function will for a specific θ and a threshold J_k indicate the probability that test δ_k does not responds according to a desired response[1].

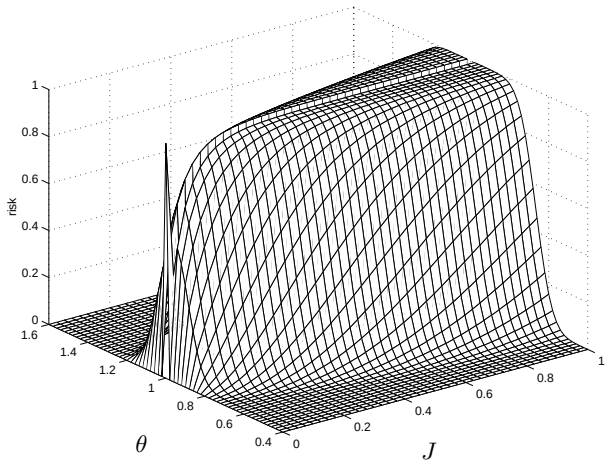

Figure 4.3: A risk function $R(\theta, \delta_k(J_k))$ as a function of two variables, θ and J_k.

Now assume that we use a 0-1 loss and consider a risk function $R(\theta, \delta_k(J_k))$ for the test quantity (4.31) from Example 4.9. In Figure 4.3, this risk function is plotted as a function of θ and J. This plot should be compared with the solid plot in Figure 4.2, which is the corresponding power function for one fixed threshold. The "peak" and the "valley" visible in Figure 4.3, corresponds to $\theta = \theta_0 = 1$ and thus the fault free case.

In Figure 4.4, the same risk function is plotted as a function of J for seven different values of θ. The dashed line corresponds to the case $\theta = 1$ and because a 0-1 loss is used, this is the probability of a TYPE I error, i.e. significance level, as a function of the threshold. This means that this kind of plot is useful to determine the significance level of a test. At the same time, we see how the probability of a TYPE II error for different θ:s varies as the threshold changes. The dash-dotted lines represents small faults, i.e. θ close to 1. It is obvious that

[1] An exact definition of *desired response* will be given in Section 6.1.4.

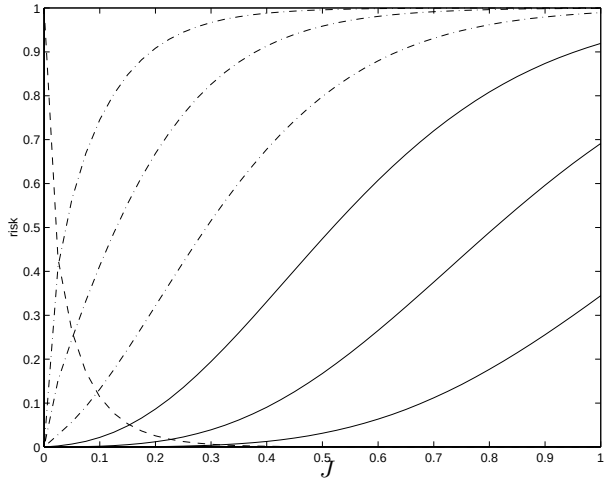

Figure 4.4: The same risk function $R(\theta, \delta_k(J_k))$ as in Figure 4.3.

for these small faults, the probability of a TYPE II error will be quite large for any reasonable low significance level. As said in the previous section, it is easy to make such a plot based on real measurement data. For an example of this, see Figure 6.10 in Section 6.4.5.

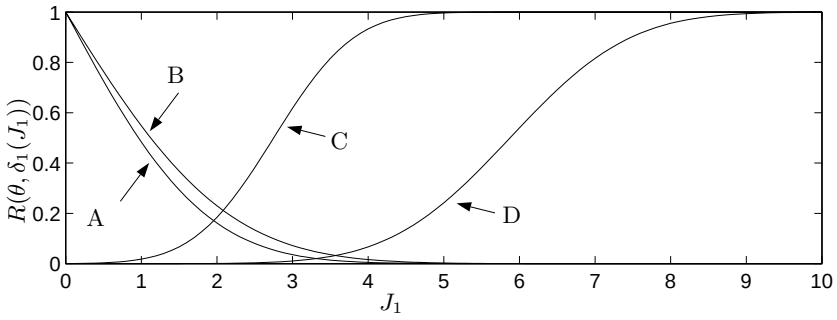

Figure 4.5: The risk $R(\theta, \delta_1(J_1))$ as a function of the threshold level J_1.

So far, we have studied how the risk function varies along one axis of the fault state space Θ. This usually corresponds to that only one fault mode, in addition to NF, is considered. However, since it is mostly interesting to investigate the performance for more than one fault mode, the risk function must be studied along several axes. In Figure 4.5, the risk function, for a test δ_1, is plotted as a function of the threshold for four different θ:s, which we denote θ_A, θ_B, θ_C, and θ_D. These four θ:s are assumed to belong to the fault modes F_A, F_B, F_C,

and F_D respectively. This could for example correspond to a row, in a decision structure, like

	F_A	F_B	F_C	F_D
δ_1	0	0	X	1

For each threshold level, the plots show the probability of an *un*desirable response. For example if the threshold $J_{thresh} = 2$, the probability of an undesirable response is at the maximum about 0.2. The tradeoff between false alarms and missed detections are clearly visible in this plot. If F_A is the fault mode NF, it is obvious that the threshold must be chosen above 3 to get a low probability of "false firing" and also a false alarm of the whole system. On the other hand, this will result in that test δ_1 will with high probability miss that $\theta = \theta_C$.

Choice of Threshold

In this section we have mainly discussed how the choice of threshold affects the performance the hypothesis test, and also how to represent this information by plotting the risk function. However, we have not discussed which threshold value to choose. This problem is difficult since the choice of threshold in each test, is dependent on the choice of thresholds in the other tests. In addition, the relation between the performance of a single hypothesis test and the whole diagnosis system is quite complex, as we will see in Chapter 6.

If non-constant functions $c_I(\theta)$ and $c_{II}(\theta)$ are defined, then an ad-hoc choice can be to use the minimax principle (see Section 6.2.2). This corresponds to selecting the threshold as

$$J_k = \arg\min_J \max_\theta R(\theta, \delta_k(J))$$

For example, consider again Figure 4.4. If $c_I(\theta) \equiv 1$, and $c_{II}(\theta)$ is chosen as $c_{II}(\theta) \equiv 0$ for the small faults (dash-dotted lines), and $c_{II}(\theta) \equiv 1$ for the large faults (solid lines), then the threshold, chosen with the minimax principle, would be $J = 1.6$.

Another threshold choice (still a bit ad-hoc) is to choose the threshold such that a specific significance level is obtained. For example, if the significance level 0.025 is desirable in Figure 4.4, then the threshold should be chosen as $J = 0.2$. If the thresholds are chosen such that all hypothesis tests get the same fixed significance level, then the analysis of the diagnosis system becomes particularly simple, as we will see in Chapter 6.

4.7.2 Specifying Hypothesis Tests

From the discussion around Figure 4.5, it should be clear that there is a close relation between the the risk or power function and the decision structure or equivalently the choice of S_k^0 and S_k^1. We will here describe how the power function can be used to specify a hypothesis test δ_k, i.e. to choose the sets M_k,

S_k^0, and S_k^1. The basic principle is that if the power function $\beta_k(\theta)$ is low for all θ belonging to a fault mode γ, then we should choose M_k such that $\gamma \in M_k$. As said before, this also gives the set S_k^1 since $S_k^1 = M_k^C$. This means that S_k^1 can be described as

$$S_k^1 = \Omega - \{\gamma \mid \forall \theta \in \Theta_\gamma.\ \beta_k(\theta) \text{ small}\} \tag{4.42}$$

Further, if the power function $\beta_k(\theta)$ is large for all θ belonging to a fault mode γ, then we should choose S_k^0 such that $\gamma \notin S_k^0$. This means that S_k^0 can be described as

$$S_k^0 = \Omega - \{\gamma \mid \forall \theta \in \Theta_\gamma\ .\beta_k(\theta) \text{ large}\} \tag{4.43}$$

Remember that this also completely specifies the contents of the decision structure, i.e. where to put 0:s, 1:s, and X:s. How small "small" is and how large "large" is, depends on the actual case, but these sizes are related to the probability of taking wrong decisions for the diagnosis system. Also the significance level α_k is related to this probability and the sizes "small" and "large" should therefore be chosen to be around α_k and $1 - \alpha_k$ respectively.

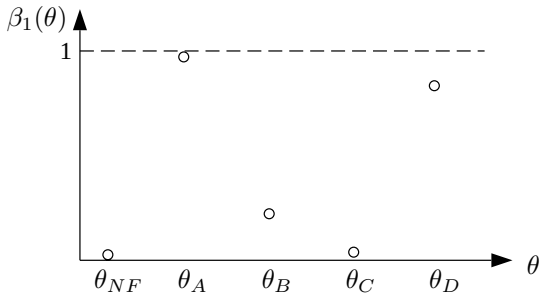

Figure 4.6: The power function.

Example 4.12

Assume we want to consider five fault modes:

NF	$\theta = \theta_{NF}$
A	$\theta = \theta_A$
B	$\theta = \theta_B$
C	$\theta = \theta_C$
D	$\theta = \theta_D$

Further assume that we have designed a test quantity $T_1(x)$ which together with a specific threshold J_1 gives the (discrete) power function shown in Figure 4.6.

Then using the expressions (4.42) and (4.43), the sets S_1^0 and S_1^1 becomes

$$S_1^1 = \{NF, A, B, C, D\} - \{NF, C\} = \{A, B, D\}$$
$$S_1^0 = \{NF, A, B, C, D\} - \{A\} = \{NF, B, C, D\}$$

■

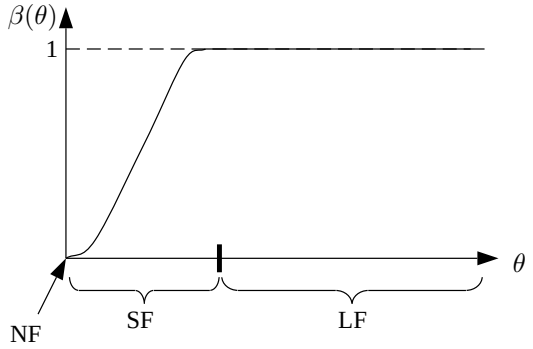

Figure 4.7: The power function.

Consider a specific fault mode γ. As has been said before, it is often difficult or impossible to construct a power function that is large for all θ belonging to Θ_γ. Still it may be the case that $\beta(\theta)$ is large for a subset Θ_L of Θ_γ, i.e. $\Theta_L \subseteq \Theta_\gamma$. The set Θ_L typically corresponds to "large" fault sizes. If H_0 is not rejected in a case like this, we would be tempted to draw the conclusion that $\theta \notin \Theta_L$, i.e. the fault is not large. If this kind of reasoning is desired, the fault mode γ can be splitted into two: γ-*small* and γ-*large*. This is further illustrated by the following example:

Example 4.13

Assume we want to consider three fault modes:

NF	$\theta = 0$	no fault
SF	$0 < \theta < c$	small fault
LF	$c \geq \theta$	large fault

Further assume that we have designed a test quantity $T_1(x)$ which together with a specific threshold J_1 gives the power function shown in Figure 4.7. Then the decision structure becomes

	NF	SF	LF
$\delta_1(x)$	0	X	1

Thus if H_1^0 is not rejected, the diagnosis statement becomes $S = \{NF, SF\}$, i.e. a large fault has not occured. Also if H_1^0 is rejected, $S = \{SF, LF\}$, i.e. some fault (small or large) has definitely occured. ■

4.8 A Comparison Between the Prediction Error Principle and the Estimate Principle

When constructing test quantities, it is often difficult to know which one of the prediction, likelihood, or estimate principle that is the best choice. The definite answer can of course be found by studying the risk functions for each specific case, but it is nevertheless interesting to also have a more general discussion. We will here discuss how a test quantity based on the estimate principle performs compared to a test quantity based on the prediction error principle. This will be done by studying an example.

Consider a system which can be modeled as

$$y(t) = b|u|^{\varphi}\operatorname{sgn} u + a + v \tag{4.44}$$

where $v(t) \sim N(0, \sigma_v)$ and $\operatorname{sgn} u$ is the sign of u, i.e. -1, 0, or 1. The nominal (i.e. corresponding to the no fault case) values for the three parameters are $b_0 = 1$, $a_0 = 0$, and $\varphi_0 = 1$. The four fault modes considered are

NF	$b = 1, a = 0, \varphi = 1$
F_b	$b \neq 1, a = 0, \varphi = 1$
F_φ	$b = 1, a = 0, \varphi \neq 1$
F_a	$b = 1, a \neq 0, \varphi = 1$

We will start by comparing the two test quantities

$$T_1(x) = \sum_{1}^{N}(y - u)^2 \tag{4.45}$$

$$T_2(x) = Np(\hat{b} - b_0)^2 \qquad \hat{b} = (U^T U)^{-1} U^T Y \tag{4.46}$$

where $\hat{b}$ is the least square estimate of b. The comparison study will be made by using the power function, as was described in Section 4.6.2. In Example 4.9, we saw that $\sqrt{Np}(\hat{b} - b_0)$ is $N(0, \sigma_v)$ under H_0. This implies that $T_2(x)/\sigma_v^2$ is $\chi^2(1)$-distributed. Similarly it can be shown that $T_1(x)/\sigma^2$ is $\chi^2(N)$-distributed under H_0. The knowledge of these distributions can be used to find thresholds J_1 and J_2 such that a specific significant level is obtained.

To evaluate the test quantities (4.45) and (4.46), two tests are constructed, δ_1 based on $T_1(x)$ and δ_2 based on $T_2(x)$. The standard deviation σ_v is assumed to be 0.2 and then the thresholds are chosen such that the significance level for both tests becomes $\alpha = 0.0034$.

4.8.1 Studying Power Functions

We will now compare the the test quantities in three different cases: when fault mode F_b is present, when fault mode F_φ is present, and when fault mode F_b is present and the test quantities are modified such that fault mode F_a is decoupled.

Fault Mode F_b Present

This case corresponds to that the power functions for the case $a = a_0$, and $\varphi = \varphi_0$ are studied, i.e. along the b-axis of the fault state space. This means that the system model becomes linear and can be written as

$$y(t) = bu + v \tag{4.47}$$

Further, the power functions becomes functions of b, i.e. $\beta_1(b)$ and $\beta_2(b)$.

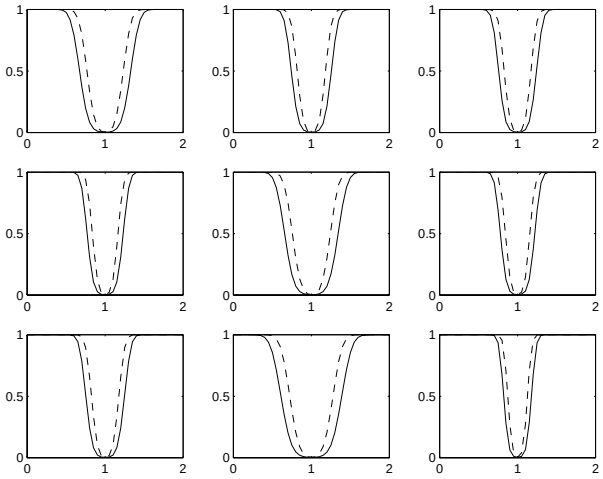

Figure 4.8: The power functions $\beta_1(\theta)$ (solid) and $\beta_2(\theta)$ (dashed) for two tests based on $T_1(x)$ and $T_2(x)$. The result for 9 different input signals u is shown.

The power function for δ_2 can be obtained analytically in accordance with Example 4.11. However the power function for δ_1 can not be so easily obtained. Instead, simulations have to be used. The power functions $\beta_1(b)$ and $\beta_2(b)$ for 9 different input signals u, estimated by means of simulations, are plotted in Figure 4.8.

In the figure, it is seen that for all 9 different u:s, the two power functions are equal for large deviations from θ_0 but for many other values, $\beta_2(\theta)$ (dashed) is greater than $\beta_1(\theta)$ (solid), i.e. $T_2(x)$ is better than $T_1(x)$. In other words, the estimate principle, with the estimated parameter the same as the one modeling the fault, here outperforms the prediction error principle.

Fault Mode F_φ Present

Now consider the fault mode F_φ, which means that φ is a free variable while $b_0 = 1$ and $a_0 = 0$. The model (4.44) now becomes

$$y(t) = |u|^{\varphi} \operatorname{sgn} u + v$$

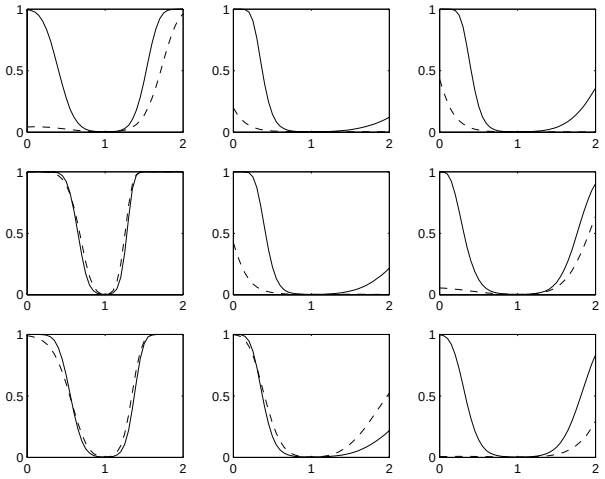

Figure 4.9: The power functions $\beta_1(\varphi)$ (solid) and $\beta_2(\varphi)$ (dashed) for two tests based on $T_1(x)$ and $T_2(x)$. The result for 9 different input signals u is shown.

The two power functions $\beta_k(\varphi)$ for $T_1(x)$ and $T_2(x)$ respectively, are plotted in Figure 4.9. As before 9 different input signals u have been considered. In contrast to Figure 4.8, there are large differences between the different plots. This holds for both power functions $\beta_1(\varphi)$ and $\beta_2(\varphi)$. However, it is clear that $T_2(x)$ is very sensitive to different u:s while $T_1(x)$ is more robust. Also, in all plots it no longer holds that $\beta_2(\varphi) \geq \beta_1(\varphi)$ for all φ. In most of the plots, $\beta_1(\varphi)$ is actually larger than $\beta_2(\varphi)$. It is obvious that the overall performance of $T_1(x)$ is much better than $T_2(x)$. Thus, in this case where the estimate principle uses an estimate of a parameter not modeling the fault, the prediction error principle outperforms the estimate principle.

The incidence structure for the two test quantities and for the fault modes NF, F_b, and F_φ, is

	NF	F_b	F_φ
$T_1(x)$	0	1	1^*
$T_2(x)$	0	1	1

From the discussion above it is clear that the 1 marked 1^* is much "weaker" than the other 1:s. However in a diagnosis system containing several hypothesis tests, it is enough if the power function of a specific test is high in only one or a few directions. For example, when using the two tests based on $T_1(x)$ and $T_2(x)$ described here, it is enough if only $T_2(x)$ has high power for the fault mode corresponding to φ. The reason is that either H_1^1 or H_2^1 or both are accepted, the diagnosis statement will become the same, namely $S = \{F_b, F_\varphi\}$.

Decoupling of F_a and Fault Mode F_b Present

Next we will investigate how the test quantities $T_1(x)$ and $T_2(x)$ are affected by decoupling of the fault mode F_a. To do this, we construct two new test quantities $T_{1a}(x)$ and $T_{2a}(x)$ in accordance with (4.3) and (4.26) respectively:

$$T_{1a}(x) = \min_a \sum_{t=1}^{N} \big(y(t) - \hat{y}(t|a)\big)^2 = \min_a \sum_{t=1}^{N} \big(y(t) - u(t) - a\big)^2 \tag{4.48}$$

$$T_{2a}(x) = Np(\hat{b} - 1)^2 \qquad \hat{b} = \arg\min_b \min_{b,a} \sum_{t=1}^{N} \big(y(t) - bu(t) - a\big)^2 \tag{4.49}$$

The least square estimate of a that minimizes (4.48) is

$$\hat{a} = \bar{y} - \bar{u} = \frac{1}{N}\sum_{t=1}^{N} y(t) - \frac{1}{N}\sum_{t=1}^{N} u(t)$$

The least square estimate of a and b that minimizes (4.49) is

$$\hat{b} = \frac{\sum_{t=1}^{N}(u(t) - \bar{u})(y(t) - \bar{y})}{\sum_{t=1}^{N}(y(t) - \bar{y})^2} \tag{4.50}$$

$$\hat{a} = \bar{y} - \hat{b}\bar{u} \tag{4.51}$$

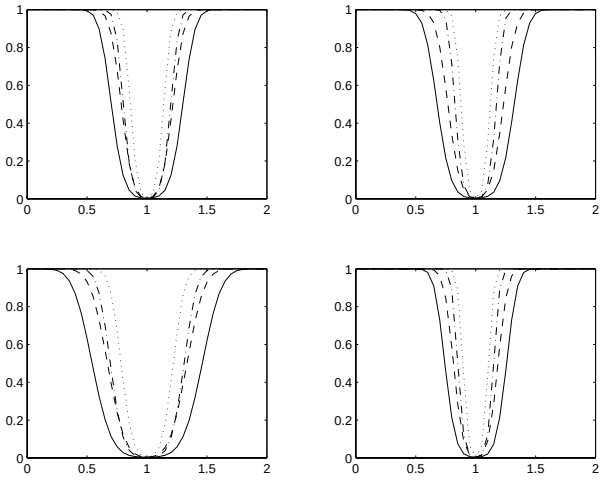

Figure 4.10: The power functions $\beta_{1a}(b)$ (solid), $\beta_{2a}(b)$ (dashed), $\beta_1(b)$ (dash-dotted), and $\beta_2(b)$ (dotted) for tests based on $T_{1a}(x)$, $T_{2a}(x)$, $T_1(x)$, and $T_2(x)$ respectively. The result for 4 different input signals u is shown.

Tests using $T_{1a}(x)$ and $T_{2a}(x)$ are constructed with the significance level $\alpha = 0.0034$ (the same as before). The parameter a is chosen as $a = 1$. The

resulting power functions $\beta_k(b)$ corresponding $T_{1a}(x)$, and $T_{2a}(x)$ are then estimated via simulations. Included in the study are also $T_1(x)$ and $T_2(x)$, i.e. (4.45) and (4.46), but here with data compensated for the non-zero a, i.e. $y' = y - 1$. The power functions for $T_{1a}(x)$, $T_{2a}(x)$, $T_1(x)$, and $T_2(x)$ are plotted in Figure 4.10. Here 4 different input signals u have considered. We can see that the estimate principle also for this case, i.e. including decoupling, outperforms the prediction error principle. It should be remembered though, that the estimate principle implies that one extra parameter must be estimated, and this can in general be a substantial problem.

Also seen in the plots is that the dotted line is above the dashed, meaning that the test quantity $T_2(x)$ performs better than $T_{2a}(x)$. However, this is the expected result since one less parameter has to be estimated using $T_2(x)$ compared to $T_{2a}(x)$. Theoretically this can be explained by comparing the distribution of the estimate (4.50), i.e.

$$\hat{b} \sim N(b, \frac{\sigma_v}{\sqrt{\sum_{t=1}^{N}(u(t) - \bar{u})^2}})$$

with the distribution (4.28). It holds that

$$\sum_{t=1}^{N}(u(t) - \bar{u})^2 \leq \sum_{t=1}^{N}(u(t))^2$$

and therefore the variance of $\hat{b}$ obtained via (4.50) and corresponding to $T_{2a}(x)$, is greater than the variance of $\hat{b}$ corresponding to $T_2(x)$. This explains the difference between the power functions $\beta_{1a}(b)$ and $\beta_{2a}(b)$.

4.8.2 A Theoretical Study

To find a theoretical motivation to why the estimate principle is better than the prediction error principle, we will here study a somewhat simplified case. Consider the model (4.47) but assume that $b \geq 0$ and the no fault case corresponds to $b = b_0 = 0$. We will consider two test quantities: $T_1(x)$ from (4.45) and $T_2''(x)$ which we define as

$$T_2''(x) = \sqrt{Np}\,\hat{b} = \sqrt{Np}\,(U^T U)^{-1} U^T Y$$

Power functions for corresponding tests are plotted in Figure 4.11. The result is the same as in Figure 4.8, i.e. the test quantity based on the estimate principle, i.e. $T_2''(x)$, is better than the test quantity $T_1(x)$ based on the prediction principle.

Now consider the following theorem (Casella and Berger, 1990):

Theorem 4.1 *If $f(\mathbf{x}|\theta)$ is the joint probability density function of $\mathbf{X}$, and $q(t|\theta)$ is the probability density function of $T(\mathbf{X})$, then $T(\mathbf{X})$ is a sufficient statistic for θ if, and only if, for every $\mathbf{x}$ in the sample space, the ratio $f(\mathbf{x}|\theta)/q(T(\mathbf{x})|\theta)$ is constant as a function of θ (i.e. independent of θ).*

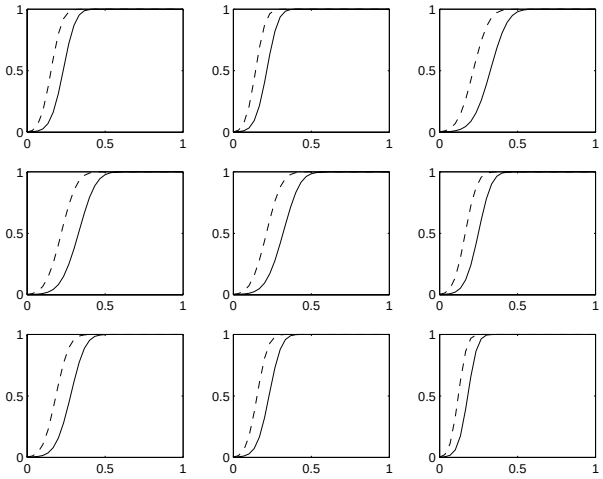

Figure 4.11: The power functions $\beta_1(b)$ (solid) and $\beta_2(b)$ (dashed) for two tests based on $T_1(x)$ and $T_2''(x)$. The result for 9 different input signals u is shown.

With this theorem it can be shown that $T_2''(x)$, is a sufficient statistic for b. Next consider the following theorem (Casella and Berger, 1990):

Theorem 4.2 *Consider testing $H_0 : \theta \in \Theta_0$ versus $H_1 : \theta \in \Theta_0^C$. Suppose a test based on a sufficient statistic T with rejection region $\mathcal{S}$, satisfies the following three conditions:*

a. The test is a level α test.

b. There exists a $\theta_0 \in \Theta_0$ such that $P(T \in \mathcal{S} \mid \theta_0) = \alpha$.

c. Let $g(t|\theta)$ denote the probability density function of T. For the same θ_0 as in (b), and for each $\theta' \in \Theta_0^C$, there exists a $k' \geq 0$ such that

$$t \in \mathcal{S} \text{ if } g(t|\theta') > k'g(t|\theta_0) \quad \text{and} \quad t \in \mathcal{S}^C \text{ if } g(t|\theta') < k'g(t|\theta_0)$$

Then this test is a UMP[2] level α test of H_0 versus H_1.

The conditions (a) and (b) are trivially fulfilled and to show condition (c), we must show that

$$\forall t > J \; . \; g(t|b) > \frac{g(J|b)}{g(J|0)} g(t|0) \tag{4.52}$$

[2] A test with power function $\beta(\theta)$ is a UMP (uniformly most powerful) level α test if there exist no other test with the same significance level α and with a power function $\beta'(\theta)$ such that $\beta'(\theta) > \beta'(\theta)$ for any θ.

where J is the threshold of the test, and $g(t|b)$ is the probability density function of $T_2''(x) \sim N(\sqrt{Np}\, b, \sigma_v)$. It is easy to realize that (4.52) holds and therefore we have the result that a hypothesis test based on $T_2''(x)$ is a UMP test. This means that there can not exist any test quantity better than $T_2''(x)$ for this hypothesis test.

4.8.3 Concluding Remarks

Even though the discussion has mainly focused on specific examples, we are able to summarize the following conclusions:

- Test quantities based on estimates can have very good performance for the fault mode corresponding to the estimated parameter.
- For other fault modes, the performance might be quite bad and also highly dependent on the input signal.
- Decoupling degrades the performance of both the prediction error principle and the estimate principle but the relation that the estimate principle is better than the prediction error principle still holds.

4.9 Conclusions

In Chapters 2 to 4, a new general framework for fault diagnosis has been proposed. We have seen that we do not need separate frameworks for statistical vs deterministical approaches to fault diagnosis. Both views are contained in the general framework presented here.

The framework is also general with respect to what types of faults that can be handled. Many papers in the field of fault diagnosis discuss decoupling of faults modeled as additive arbitrary signals. It is realized that the principle of decoupling has in this chapter been generalized to include decoupling of faults modeled in arbitrary ways, e.g. as deviations of constant parameters or abrupt changes of parameters.

For the design of test quantities, we have identified three different principles: the prediction, the likelihood, and the estimate principle. For all three principles we have discussed how robustness can be achieved by means of normalization. The known techniques *adaptive threshold* and *likelihood ratio tests* are in fact shown to be special cases of normalization. The importance of normalization, when using the estimate principle, has been emphasized.

Statistics and decision theory is used to define measures to evaluate hypothesis tests and test quantities. We have also discussed how these measures can be used to select the threshold and the sets S^0 and S^1 of a hypothesis test. Finally we applied the evaluation measures to compare the prediction and the estimate principle in some cases. The conclusion was that the estimate principle is, in at least one common case, superior to the prediction principle.

Chapter 5

Applications to an Automotive Engine

In the field of automotive engines, environmentally based legislative regulations such as OBDII (On-Board Diagnostics II) (*California's OBD-II Regulation*, 1993) and EOBD (European On-Board Diagnostics) specifies hard requirements on the performance of the diagnosis system. This makes the area a challenging application for model-based fault-diagnosis. Other reasons for incorporating diagnosis in vehicles are repairability, availability and vehicle protection. The importance of diagnosis in the automotive engine application is highlighted by the fact that up to 50% of the code in present engine-management systems are dedicated to diagnosis.

Model-based diagnosis for automotive engines, has been studied in several works, e.g. (Gertler, Costin, Fang, Hira, Kowalalczuk, Kunwer and Monajemy, 1995; Krishnaswami, Luh and Rizzoni, 1994; Nyberg and Nielsen, 1997*b*). Although the techniques in these papers are not fully developed, it is obvious that there is much to gain by using a *model based* approach to diagnosis of automotive engines.

In this chapter, the framework, theory, and methods from the previous chapters are demonstrated on a real application: the air-intake system of a turbocharged automotive engine. Design of diagnosis systems is discussed, as well as theoretical issues and results of practical experiments. First, the modeling work is presented in Sections 5.1 to 5.3. Then diagnosis of leakage is discussed in Sections 5.4 and 5.5. Finally, diagnosis of leakage *and* sensor faults is investigated in Sections 5.6 to 5.8.

Diagnosis of leakage is an important problem. This is because a leakage can cause increased emissions and drivability problems. If the engine is equipped with an air-mass flow sensor, a leakage will result in that this sensor does not correctly measure the amount of air entering the combustion. This in turn will result in a deviation in the air-fuel ratio. A deviation in the air-fuel ratio is serious because it causes the emissions to increase since the catalyst becomes less efficient. Also misfires can occur because of a too lean or rich mixture.

In addition, drivability will suffer and especially in turbo-charged engines, a leakage will result in loss of horsepowers.

The above requirements imply that it is important to detect leaks with an area as small as some square millimeters. For the engine management system, it is also important to get an estimate of the size of the leakage. This is to know what appropriate action that should be taken, e.g. give a warning to the driver. Additionally if the size of the leak is known, it is possible to reconfigure the control algorithm so that at least the increase in emissions, caused by the leak, will be small. We will see that the diagnosis principles developed in this chapter fulfills these requirements.

As said above, we will also discuss the diagnosis of sensors connected to the air-intake system. For the same reasons as in the leakage case, this is also an important diagnosis problem. Faults in the sensors degrade the performance of the engine control system, which in turn is likely to cause increased emissions and drivability problems. One of the interests is to investigate how to diagnose *both* leakage and different types of sensor faults at the same time. For instance, a leakage can easily be mis-interpreted as a air-mass flow sensor fault if not extra care is taken. The presented solution to this problem is a good illustration of the usefulness of the general principle of structured hypothesis tests and related theory.

Note that the purpose of this chapter is not to present complete and good designs of diagnosis systems, but rather to exemplify the techniques presented in the previous chapters in a real application.

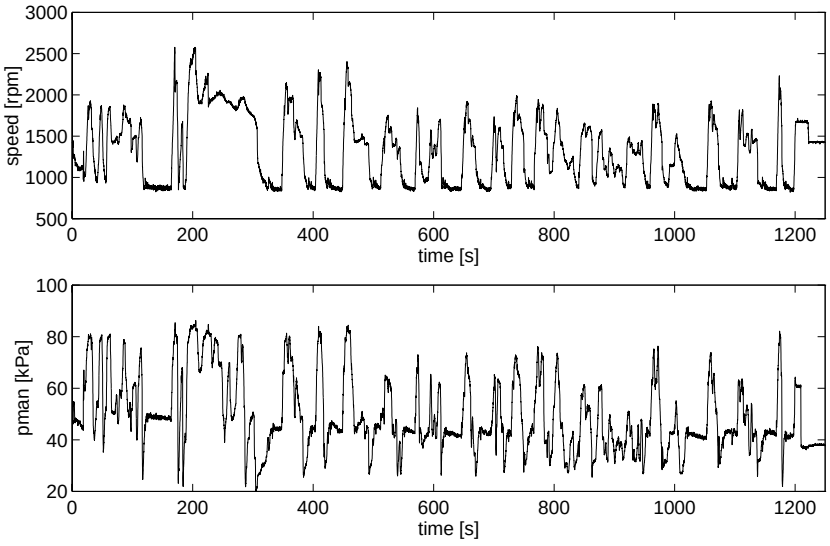

Figure 5.1: Engine speed and manifold pressure during the FTP-75 test-cycle for a car with automatic transmission.

5.1 Experimental Setup

All experiments in this chapter were performed on a 4 cylinder, 2.3 liter, turbo-charged, spark-ignited SAAB production engine. It is constructed for the SAAB 9-5 model. The engine is mounted in a test bench together with a Schenck "DYNAS NT 85" AC dynamometer. Both during the model building and the validation, the engine was run according to Phase I+II of the FTP-75 test-cycle. The data for the test cycle had first been collected on a car with automatic transmission. This resulted in the engine speed and manifold pressure shown in Figure 5.1. In addition, static tests were performed in 172 different operating points defined by engine speed and manifold pressure.

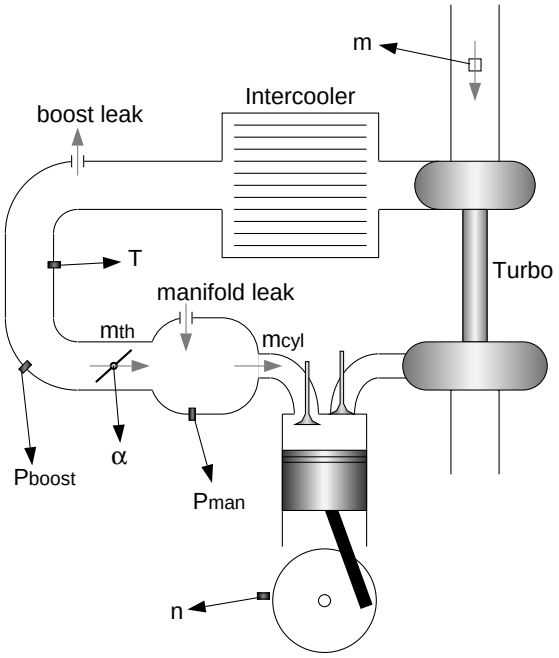

Figure 5.2: The turbo-charged engine. Air-mass flows that are discussed in the text are marked with gray arrows.

A schematic picture of the air-intake system is shown in Figure 5.2. Ambient air enters the system and an air-mass flow sensor measures the air-mass flow rate m. Next, the air passes the compressor side of the turbo-charger and then the intercooler. This results in a *boost* pressure p_b and a temperature T that are both higher than the ambient pressure and temperature respectively. Next, the air passes the throttle and the flow m_{th} is dependant on p_b, T, the throttle angle α, and the manifold pressure p_m. Finally the air leaves the manifold and enters the cylinder. This flow m_{cyl} is dependant on p_m and the engine speed n. Also shown in the figure are the two possible leaks: the *boost leak* somewhere between

the air-mass flow sensor and the throttle, and the *manifold leak* somewhere in the manifold.

Leaks were applied by using exchangeable bolts. One bolt were mounted in the wall of the manifold and the other in the wall of the air tube 20 cm in front of the throttle. The exchangeable bolts had drilled holes of different diameters ranging from 1 mm to 8 mm.

Data were collected by a DAQ-card mounted in a standard PC. All data were filtered with a LP-filter with a cutoff frequency of 2 Hz.

5.2 Model Construction - Fault Free Case

For the purpose of fault diagnosis, a simple and accurate model is desirable. In this work, the air-intake system is modeled by a mean value model (Hendricks, 1990). This means that no within-cycle variations are covered by the model. The automotive engine is a non-linear plant and it has been indicated in a pre-study that diagnosis based on a linear model is not sufficient for the engine application. This has also been concluded by other authors (Gertler, Costin, Fang, Hira, Kowalczuk and Luo, 1991; Krishnaswami et al., 1994). This motivates the choice of a non-linear model in this work.

A model is first developed for the case when no leakage is present. Because there is no need for extremely fast detection of leakage, it is for the model sufficient to consider only static relations. The model for the fault-free air-intake system is described by the following equations

$$m = m_{th} \tag{5.1a}$$

$$m_{th} = m_{cyl} \tag{5.1b}$$

These equations say that the measured intake air-flow is equal to the air-flow past the throttle which in turn is equal to the air-flow into the cylinders. The models for the air-flows m_{th} and m_{cyl} are presented next.

5.2.1 Model of Air Flow Past the Throttle

The air-mass flow past the throttle m_{th} is described well by the formula for flow through a restriction (Heywood, 1992) (Taylor, 1994):

$$m_{th} = \frac{C_d A_{th} p_{boost}}{\sqrt{RT}} \Psi\left(\frac{p_{man}}{p_{boost}}\right) \tag{5.2}$$

where A_{th} is the throttle plate open area, C_d the discharge coefficient, and $\Psi(\frac{p_{man}}{p_{boost}})$ is

$$\Psi(\frac{p_{man}}{p_{boost}}) = \begin{cases} \sqrt{\frac{2\kappa}{\kappa-1}\left\{\left(\frac{p_{man}}{p_{boost}}\right)^{\frac{2}{\kappa}} - \left(\frac{p_{man}}{p_{boost}}\right)^{\frac{\kappa+1}{\kappa}}\right\}} \\ \qquad\qquad \text{if } \left(\frac{p_{man}}{p_{boost}}\right) \geq \left(\frac{2}{\kappa+1}\right)^{\frac{\kappa}{\kappa-1}} \\ \sqrt{\kappa\left(\frac{2}{\kappa+1}\right)^{\frac{\kappa+1}{\kappa-1}}} \qquad \text{otherwise} \end{cases}$$

By defining the coefficient K_{th} as

$$K_{th} = \frac{C_d A_{th}}{\sqrt{R}} \tag{5.3}$$

and

$$\beta(T, p_{boost}, p_{man}) = \frac{p_{boost}}{\sqrt{T}}\,\Psi(\frac{p_{man}}{p_{boost}})$$

the flow model (5.2) can be rewritten as

$$m_{th} = K_{th}\beta(T, p_{boost}, p_{man}) \tag{5.4}$$

From m-, T-, p_{boost}-, and p_{man}-data collected during the FTP-75 test-cycle, the K_{th} coefficient can for each sample be computed as

$$K_{th} = \frac{m}{\beta(T, p_{boost}, p_{man})}$$

if dynamics is neglected and therefore $m_{th} = m$. This calculated K_{th} coefficient is plotted against throttle angle in Figure 5.3. It is obvious that the throttle angle by its own describes the K_{th} coefficient well. From Equation 5.3, we see that the K_{th} coefficient is dependant on the throttle plate open area A_{th}. A physical model of this area is

$$A_{th} = A_1(1 - \cos(a_0\alpha + a_1)) + A_0 \tag{5.5}$$

where A_1 is the area that is covered by the throttle plate when the throttle is closed and A_0 is the *leak area* present even though the throttle is closed. The parameters a_0 and a_1 are a compensation for that the actual measured throttle angle may be scaled and biased because of production tolerances.

If values of $C_dA_0/\sqrt{R}$, $C_dA_1/\sqrt{R}$, a_0, and a_1 are identified from the data shown in Figure 5.3, this results in a model of the K_{th} coefficient as function of the throttle angle α. In Figure 5.3, this model is plotted as a dashed line and we can see that the match to measured data is almost perfect except for some outliers for low throttle angles. It should be noted the these outliers are very few compared to the total amount of data. The reason for the outliers are probably unmodeled dynamic effects. The good fit obtained means that it is possible to assume that the discharge coefficient C_d is constant and independent of the throttle angle. In conclusion, the K_{th} coefficient together with equation (5.4) defines the model of the air-mass flow past the throttle.

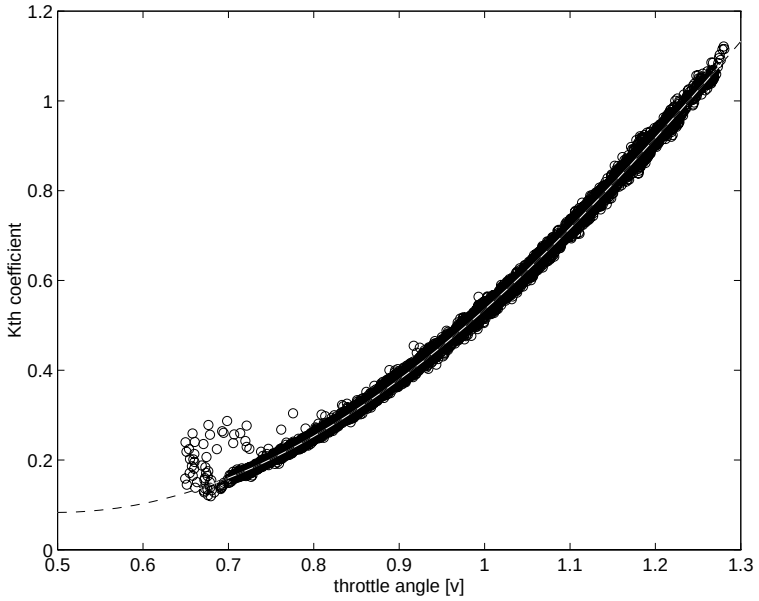

Figure 5.3: The K_{th} coefficient for different throttle angles. It is obvious that the throttle angle by its own describes the K_{th} coefficient well.

5.2.2 Model of Air Flow into Cylinders

There are no accurate and simple physical models describing the flow from the manifold into the cylinders. Therefore a black box approach is chosen. From the mapping data, the air-mass flow is, in Figure 5.4, plotted against engine speed and manifold pressure. The preliminary model of the air flow into the cylinder m_{cyl} consists of a linear interpolation of the data in Figure 5.4. It is assumed that the manifold temperature variation do not affect the flow. In the indoor experimental setup used, with the engine operating at approximately constant temperature, there was no way to validate this assumption.

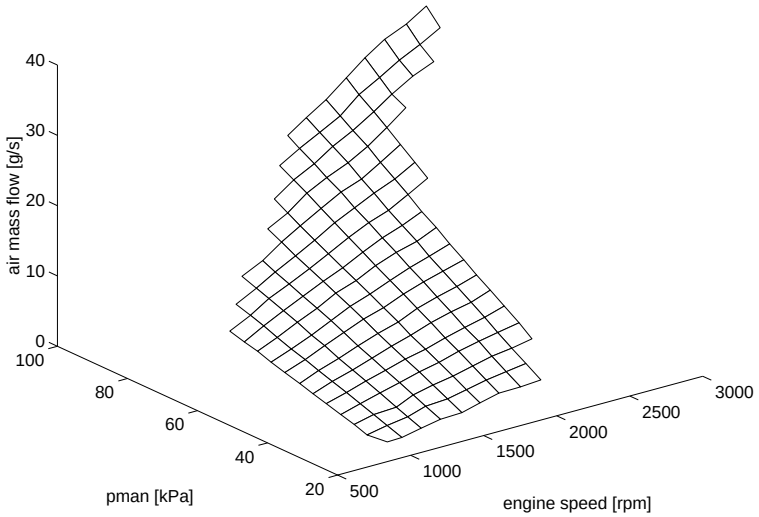

Figure 5.4: The air flow out from the manifold into the cylinders as a function of engine speed and manifold pressure.

When the engine operating point, defined by engine speed and manifold pressure, leaves the range where mapping data is available, it is not possible to do interpolation. Because the mapping range is chosen to match normal operating, this happens rarely, but when it happens, the model will produce no output data.

For the construction of the final model, also data from the test cycle were used. To incorporate these data in the model, a parametric model including four fitting parameters is introduced:

$$\hat{m}_{cyl} = b_0 \, interpolate(n, p_{man}) + b_1 n + b_2 p_{man} + b_3 \tag{5.6}$$

The parameters b_i were found by using the least-square method. The benefit with this approach, i.e. to use of interpolation in combination with a parametric model, is that it is possible to include both test-cycle data and mapping data when building the model. In addition, the parametric model provides for a straightforward way to adapt the model for process variations and individual-to-individual variations. Also the throttle model, described in the previous section, with its four parameters, has this feature.

5.2.3 Model Validation

The models (5.4) of m_{th} and (5.6) of m_{cyl} are validated during the FTP-75 test-cycle. Data were chosen from another test run, so the modeling data and the validation data were not the same. The upper plot of Figure 5.5 shows the

measured air flow m and the estimated air flow, for the two models respectively. Only one curve is seen, which means that the estimated air flow closely follows the measured. In the middle and lower plot, the difference between measured and estimated air flow are shown for both models respectively. It is again seen that both models manage to estimate the measured air flow well.

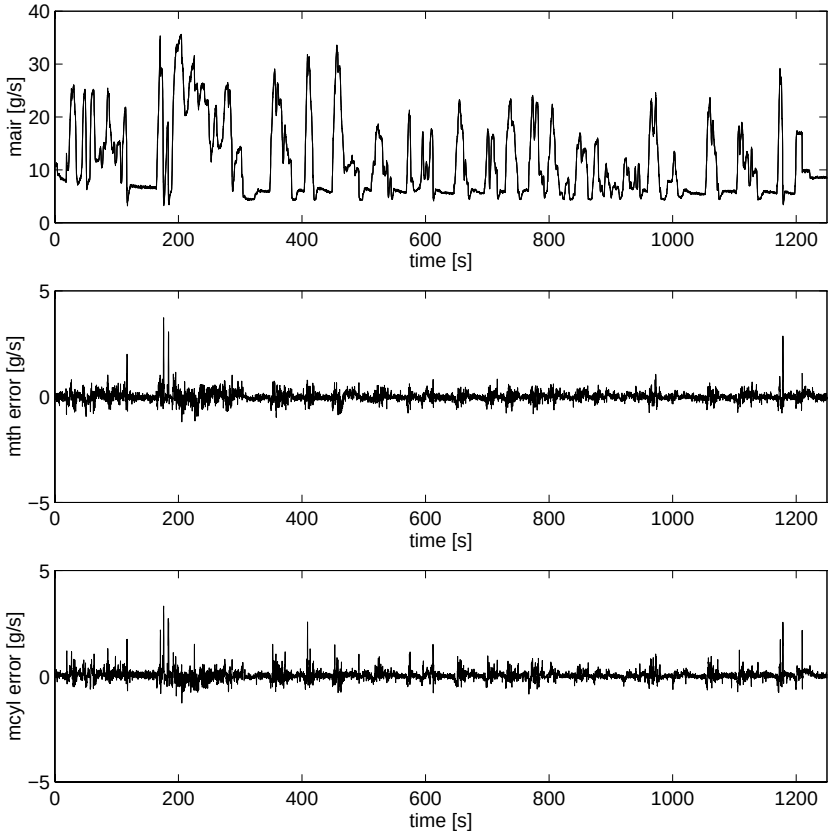

Figure 5.5: The upper plot shows measured and estimated air-mass flow. The other plots show the model error for m_{th} and m_{cyl} respectively.

5.3 Modeling Leaks

When a leak occurs, air will flow out of or into the air-intake system depending on the air pressure compared to ambient pressure. By using the measured air flow m, and the values $\hat{m}_{th}$ and $\hat{m}_{cyl}$ from the models (5.4) and (5.6) respectively, the leakage air-flow can be estimated as

$$\Delta m_{boostLeak} = m - \hat{m}_{th}$$

for boost leakage and

$$\Delta m_{manLeak} = \hat{m}_{th} - \hat{m}_{cyl}$$

for manifold leakage.

Figure 5.6 shows Δm_{boost} and Δm_{man} for a case where a 6.5 mm boost leak is present. In the lower plot it can be seen that Δm_{man} is almost zero, meaning that no leak air is added or lost in the manifold. However in the upper plot it is seen that measured air flow deviates from the estimate $\hat{m}_{th}$, which means that air is lost somewhere between the air-mass flow sensor and the throttle. In the lower plot, data are missing around time 200 s. The reason for this is that the interpolation involved in calculating $\hat{m}_{cyl}$ fails because the operating point of the engine leaves the range of the map shown in Figure 5.4.

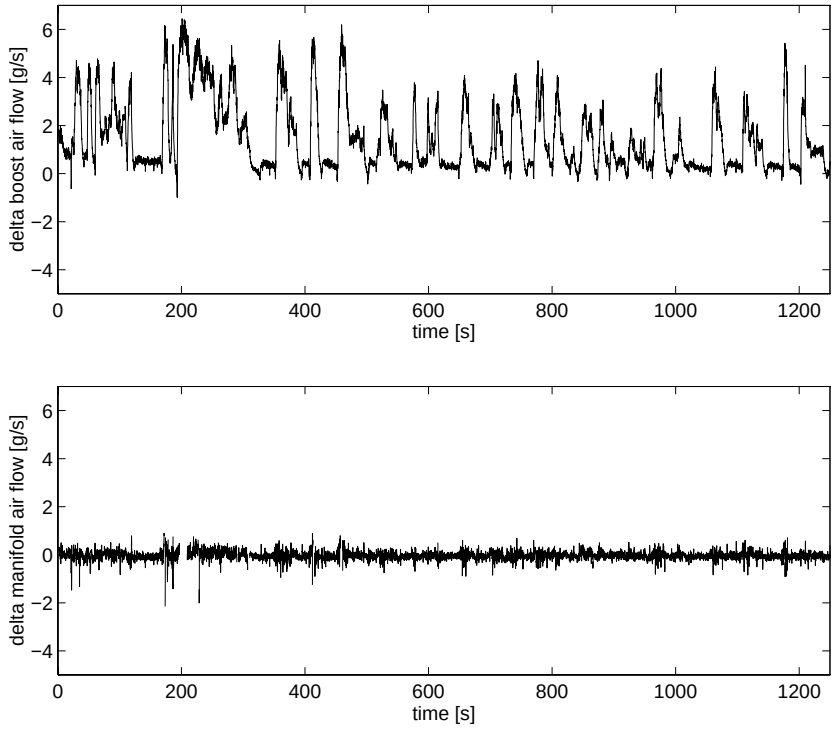

Figure 5.6: The upper plot shows Δm_{boost} and the lower plot Δm_{man} when a 6.5 mm boost leak is present.

Thus by looking at the level and also the variance of Δm_{boost} and Δm_{man}, it is possible to roughly detect when a leak is present. However to accurately estimate the size of the leak becomes difficult. To obtain high performance in terms of detecting leaks accurately a more sophisticated approach is needed; we need to model the air flow through the leaks.

5.3.1 Model of Boost Leaks

In the engine used in this work, the boost pressure is during normal operation always higher than ambient pressure. This means that the air flow through a boost leak will always be in the direction out from the air tube. This air flow is modeled as an air flow through a restriction, like the model for flow past the throttle, i.e. (5.2). The flow is dependent on the ambient pressure p_{amb} which is known because the engine is also equipped with a pressure sensor for measuring ambient pressure. The equation describing this air flow is

$$m_{boostLeak} = k_b h_b(p_b) = k_b \frac{p_b}{\sqrt{T}} \Psi(\frac{p_{amb}}{p_b}) \tag{5.7}$$

The parameter k_b is proportional to the leakage area and therefore denoted *equivalent area*.

The model for the whole air-intake system with a boost leak present is obtained by replacing Equation (5.1a) with

$$m = m_{th} + m_{boostLeak}$$

5.3.2 Model of Manifold Leaks

During most part of the operation of the engine, the manifold pressure is below ambient pressure. Therefore a manifold leak will mostly result in an air flow in the direction into the manifold. This flow is modeled in the same way as the model of flow through boost leaks, i.e.

$$m_{manLeak} = k_m h_m(p_m) = k_m \frac{p_{amb}}{\sqrt{T_{amb}}} \Psi(\frac{p_m}{p_{amb}}) \tag{5.8}$$

The model for the whole air-intake system with manifold leak present is obtained by replacing Equation (5.1b) with

$$m_{th} + m_{manLeak} = m_{cyl} \tag{5.9}$$

In the case the manifold pressure is higher than ambient pressure, which can occur because of the turbo-charger, the leak air-flow will be in the opposite direction. This means that the term $m_{manLeak}$ in (5.9) will change sign and p_{amb} and p_m in (5.8) are interchanged.

5.3.3 Validation of Leak Flow Models

For the validation of the leakage models, different leaks were applied to the engine and the FTP-75 test-cycle was used. First we investigate if the leakage model is able to correctly predict the leakage air-flow as a function of the pressure difference. Then the dependence on the leakage area is investigated.

Dependence on Pressure Difference

First "well behaved" leaks with known area, according to Section 5.1, were applied. The leaks ranged from 1 to 8 mm in diameter.

In Figure 5.7, a boost leak with 5 mm diameter, i.e. 19.6 mm^2, has been applied, and data collected during a test cycle have been used to calculate Δm_{boost} and Δm_{man}. In the upper plot, estimated air flow through the boost leak Δm_{boost} is plotted against p_{boost}. In the lower plot, estimated air flow through the manifold leak Δm_{man} is plotted against p_{man}. It is seen in the upper plot that for boost pressures close to ambient pressure (100 kPa), the estimated air flow through the leak is around zero. For higher boost pressures, the leak air-flow increases. The estimated air flow through the manifold leak is around zero for all manifold pressures.

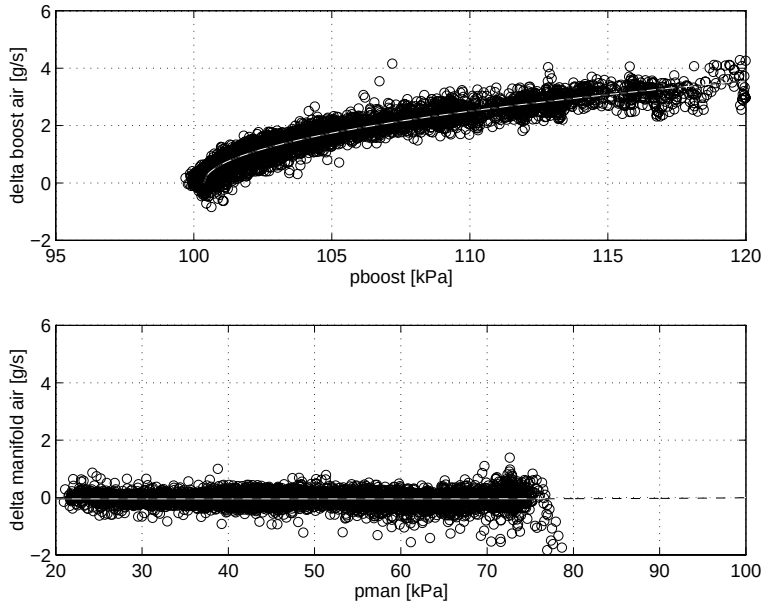

Figure 5.7: Estimated air flow through boost leak (upper plot) and manifold leak (lower plot) when a 5 mm (diameter) boost leak is present.

Correspondingly for a manifold leak with 5 mm diameter, i.e. 19.6 mm^2, Figure 5.8 shows similar data. This time, it is the estimated flow through the boost leak that is around zero and the estimated flow through the manifold leak that differs from zero. For the data collected in the test cycle, the manifold pressure is always less than ambient pressure. This results in a Δm_{man} which is always positive.

From Figures 5.7 and 5.8, it can be concluded that it is, from the estimations Δm_{boost} and Δm_{man}, possible to conclude if there is a leak and if the leak is

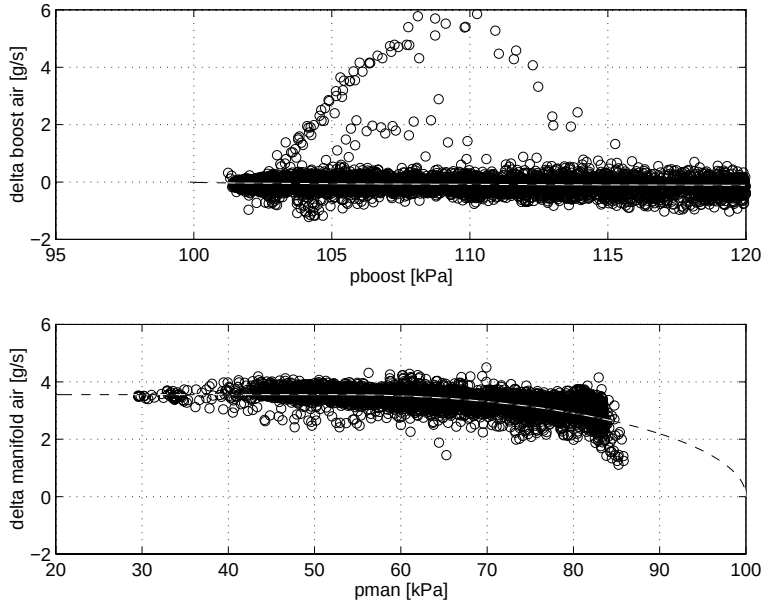

Figure 5.8: Estimated air flow through boost leak (upper plot) and manifold leak (lower plot) when a 5 mm (diameter) manifold leak is present.

before or after the throttle. Also included in Figures 5.7 and 5.8 are the outputs from the models (5.7) and (5.8) of the leak air-flow. These are represented by the dashed lines. For each case, the coefficients k_b and k_m have been obtained by using the least-square method to fit the curves to the data in the plots. Except for some outliers, which are very few compared to the total amount of data, it is seen that the estimated leak air-flows are described well by the models (5.7) and (5.8).

To validate this principle in the case of more realistic leaks, an experiment was performed in which the tube between the intercooler and the throttle was loosened at the throttle side. This had the effect that air leaked out from the system just before the throttle. In Figure 5.9 the estimated leak air-flows are again plotted against boost and manifold pressure respectively. It can be seen that also for this "realistic" leak, the model (5.7) is able to describe the leak air-flow well.

Dependence on Leakage Area

The coefficients k_b and k_m are, according to the leak flow models, proportional to the leakage area. This is validated in the following experiment. The k_b and k_m coefficients were obtained by fitting the leak flow models to measurement data for leaks with six different diameters: 1, 2, 3.5, 5, 6.5, and 8 mm. For the

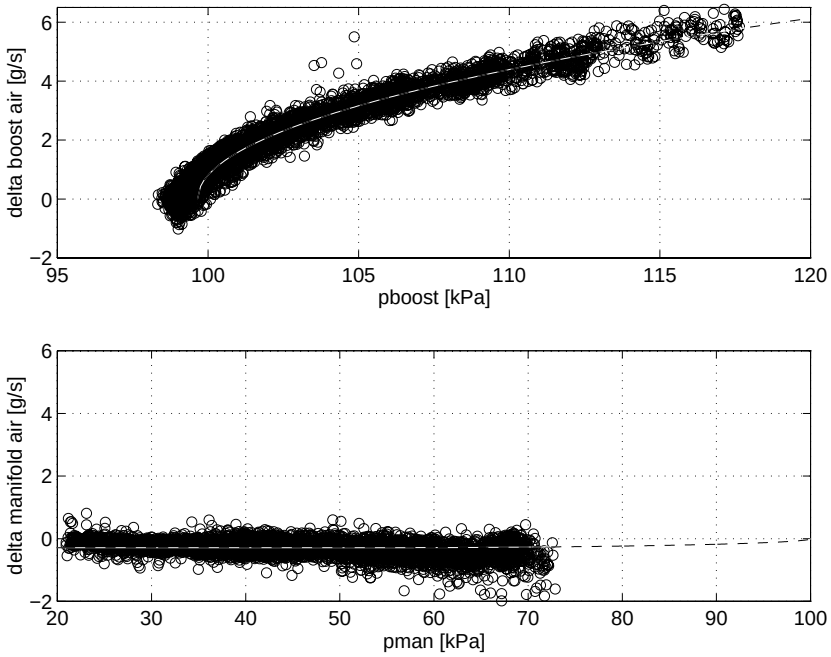

Figure 5.9: Estimated air flow through boost leak (upper plot) and manifold leak (lower plot) when a realistic boost leak is present.

manifold leak it was only possible to use the first five diameters, because the air-fuel mixture became too lean for the 8 mm hole.

The result of this study is shown in Figure 5.10 in which the estimated k_b and k_m coefficients are plotted against leakage area. The estimated k_b coefficient is plotted as solid lines and the estimated k_m coefficient is plotted as dashed lines. Both boost leaks and manifold leaks were studied. The experiments with boost leaks are marked with circles and the experiments with manifold leaks are marked with x-marks. It is seen in the figure that the k_b and k_m coefficient are close to linearly dependant on the leakage area. Also seen is that the coefficient, that should be zero for each leakage case, is close to zero for both boost and manifold leaks. The estimations of k_b for the case when a boost leak is present, and k_m for the case when a manifold leak is present, differs by a factor. One explanation is that because the bolts in these two cases, were mounted differently, the discharge coefficient were different even though the leakage area were equal.

For the "realistic" leak which were illustrated by Figure 5.9, the k_b coefficient is estimated to a value $k_b = 0.26$. In Figure 5.10 we can see that this corresponds to an equivalent area of 34 mm^2.

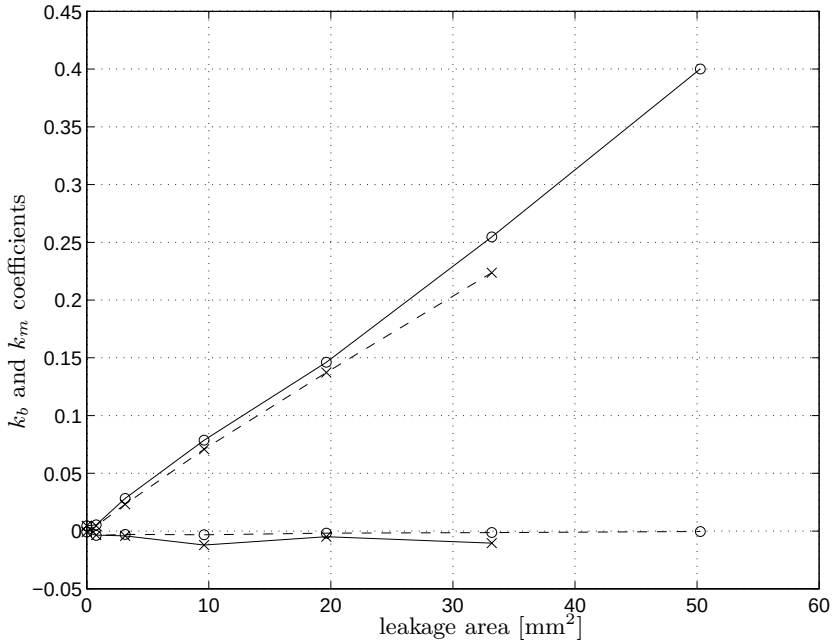

Figure 5.10: Estimated k_b coefficient (solid) and k_m coefficient (dashed), vs leakage area when boost leak is present (circles) and when manifold leak is present (x-marks).

5.4 Diagnosing Leaks

In the following sections, diagnosis of the air-intake system of the automotive engine, is discussed. First we will consider diagnosis of leakage only. Later in Section 5.6, the design of a diagnosis system capable of also diagnosing different kinds of sensor faults will be discussed. The discussion will be based on the framework and theory developed in the previous chapters. Especially we will use structured hypothesis tests which was described in Chapter 3. The objective is not to present a complete design but rather to give some examples that illustrates solutions for some typical cases.

Only single fault-modes are considered and for the diagnosis of leaks, we have three system fault-modes:

NF	No Fault
BL	Boost Leak
ML	Manifold Leak

Associated with these three fault modes, we have the models $\mathcal{M}_{NF}$, $\mathcal{M}_{BL}(k_b)$, and $\mathcal{M}_{ML}(k_m)$. This means that we have implicitly assumed two components: the boost pipe that will be indexed by b, and the manifold that will be indexed by m.

The model $\mathcal{M}_{NF}$ is obtained by using the fault-free model described in Section 5.2 in combination with

$$m_s = m \tag{5.10a}$$
$$p_{b,s} = p_b \tag{5.10b}$$
$$p_{m,s} = p_m \tag{5.10c}$$
$$\alpha_s = \alpha \tag{5.10d}$$
$$n_s = n \tag{5.10e}$$

where the index s denotes that for example m_s is the sensor signal in contrast to m which is the physical quantity. The identities (5.10) corresponds to the assumption that all sensors are fault-free. The resulting model $\mathcal{M}_{NF}$ can be written as

$$m = f(p_{b,s}, \alpha_s, p_{m,s}) \tag{5.11a}$$
$$f(p_{b,s}, \alpha_s, p_{m,s}) = g(p_{m,s}, n_s) \tag{5.11b}$$

where the function $g(p_{m,s}, n_s)$ describes the air-flow m_{cyl} in accordance with (5.6) and the function $f(p_{b,s}, \alpha_s, p_{m,s})$ describes the air-flow m_{th} in accordance with (5.2) and (5.5).

The model $\mathcal{M}_{BL}(k_b)$ is obtained by using the model described in Section 5.3.1 together with the identities (5.10). The scalar parameter k_b defines the equivalent area of the leakage and is in the model $\mathcal{M}_{BL}(k_b)$ constrained by $k_b \in \mathcal{D}^b_{BL} =]0, 0.5]$. This means that the model $\mathcal{M}_{BL}(k_b)$ can be written as

$$m - k_b h_b(p_b) = f(p_{b,s}, \alpha_s, p_{m,s}) \tag{5.12a}$$
$$f(p_{b,s}, \alpha_s, p_{m,s}) = g(p_{m,s}, n_s) \tag{5.12b}$$

where the function $h_b(p_b)$ describes the air-flow through the boost leakage and was defined in (5.7).

The model $\mathcal{M}_{ML}(k_m)$ is obtained in analogy with $\mathcal{M}_{BL}(k_b)$. The scalar parameter k_m is in the model $\mathcal{M}_{ML}(k_m)$ constrained by $k_m \in \mathcal{D}^m_{ML} =]0, 0.5]$.

From the above definitions of models, it is clear that the following relations between the fault-modes hold:

$$NF \preccurlyeq^* BL \tag{5.13a}$$
$$NF \preccurlyeq^* ML \tag{5.13b}$$

The knowledge of these relations will be used when discussing the construction of the hypothesis tests which is done next.

5.4.1 Hypothesis Tests

To develop the actual hypothesis tests, we first need to decide the set of hypotheses to test. With the relations (5.13) in mind, we know from Section 3.2.1 that the only possible sets M_k are $\{NF\}$, $\{NF, BL\}$, $\{NF, ML\}$, and $\{NF, BL, ML\}$.

Of these four possibilities, the first three are meaningful but we choose to use only two here:

$$M_{BL} = \{NF, BL\}$$
$$M_{ML} = \{NF, ML\}$$

These two sets means that there are two hypothesis tests and as seen, we have chosen to index the hypothesis tests with BL and ML. The two hypothesis tests δ_{BL} and δ_{ML} become

$$H^0_{BL} : F_p \in M_{BL} = \{NF, BL\} \qquad H^1_{BL} : F_p \in M^C_{BL} = \{ML\}$$
$$H^0_{ML} : F_p \in M_{ML} = \{NF, ML\} \qquad H^1_{ML} : F_p \in M^C_{ML} = \{BL\}$$

Next we will discuss the design of the test quantities. Only the prediction principle and the estimate principle will be discussed. In both cases we assume that the data x are all the measured sensor values and have been collected in a time window of length N.

Prediction Principle

As described in Section 4.2, the prediction principle is based on a comparison of signals and/or predictions of signals. It is straightforward to use this principle based on the models $\mathcal{M}_{BL}(k_b)$ and $\mathcal{M}_{ML}(k_m)$ described above.

Consider first the construction of the test quantity $T^{pp}_{BL}(x)$. (The index pp denotes "prediction principle" to distinguish this test quantity from the one constructed in the next section.) This test quantity should be a measure of the validity of the model (5.12). This can in a first step be achieved in accordance with the formulas (4.3) and (4.5) as follows:

$$\begin{aligned} T^{pp'}_{BL}(x) &= \min_{k_b \in \mathcal{D}^m} V_{BL}(k_b, x) = \\ &= \min_{k_b \in \mathcal{D}^m} \frac{1}{N} \sum_{t=1}^{N} \big(m_s - k_b h_b(p_b) - f(p_{b,s}, \alpha_s, p_{m,s})\big)^2 + \\ &\qquad + \frac{1}{N} \sum_{t=1}^{N} \big(f(p_{b,s}, \alpha_s, p_{m,s}) - g(p_{m,s}, n_s)\big)^2 \end{aligned} \tag{5.14}$$

To save space, the time-argument of all variables have been skipped. The expression (5.14) consists of two terms. Ideally, the first of these terms will always be zero for all possible fault modes. However, in reality the first term is non-zero and acts as an unknown disturbance in the test quantity $T^{pp'}_{BL}(x)$. Since the first term only acts as a disturbance, it can be skipped which results in the test quantity

$$T^{pp}_{BL}(x) = \frac{1}{N} \sum_{t=1}^{N} \big(f(p_{b,s}, \alpha_s, p_{m,s}) - g(p_{m,s}, n_s)\big)^2 \tag{5.15}$$

Similarly, the test quantity $T_{BL}(x)$ is constructed as

$$T_{ML}^{pp}(x) = \frac{1}{N}\sum_{t=1}^{N}\big(m_s - f(p_{b,s}, \alpha_s, p_{m,s})\big)^2 \tag{5.16}$$

Also here we have skipped the term that is close to zero all the time. The only drawback with this approach, to skip one of the terms, is when an unpredicted fault occurs, i.e. a fault not belonging to any of the fault modes BL or ML. Then it can happen that this fault is mistaken to belong to BL or ML.

In conclusion, the test quantity $T_{BL}^{pp}(x)$ has been constructed so that the fault modes BL and NF are decoupled, and $T_{ML}^{pp}(x)$ has been constructed so that the fault modes ML and NF are decoupled. This fulfills the requirements of the two hypothesis tests δ_{BL} and δ_{ML} specified above.

Estimate Principle

Using the estimate principle in accordance with Section 4.4, we base our test quantities on estimates of the equivalent areas k_b and k_m. First we discuss the construction of the test quantity $T_{ML}^{ep}(x)$. This test quantity is formed in accordance with the formula (4.26):

$$\begin{aligned}
T_{ML}^{ep'}(x) &= \|\hat{k}_b - 0\| = \hat{k}_b = \arg\min_{k_b} \min_{k_b\in\mathcal{D}^m, k_m\in\mathcal{D}^b} V_1([k_m, k_b], x) = \\
&= \arg\min_{k_b} \min_{k_b\in\mathcal{D}^m, k_m\in\mathcal{D}^b} \frac{1}{N}\sum_{t=1}^{N}\big(m_s - k_b h_b(p_b) - f(p_{b,s}, \alpha_s, p_{m,s})\big)^2 + \\
&\quad + \frac{1}{N}\sum_{t=1}^{N}\big(f(p_{b,s}, \alpha_s, p_{m,s}) - g(p_{m,s}, n_s) + k_m h_m(p_m)\big)^2 =^* \\
=^* \arg\min_{k_b\in\mathcal{D}^b} &\frac{1}{N}\sum_{t=1}^{N}\big(m_s - k_b h_b(p_b) - f(p_{b,s}, \alpha_s, p_{m,s})\big)^2 = \arg\min_{k_b\in\mathcal{D}^b} V_2(k_b, x)
\end{aligned}$$

Note that the measure $\|\cdot\|$ is here defined as the identity function. The function $V_1([k_m, k_b], x)$ is a model validity measure for the model $\mathcal{M}([k_m, k_b])$. It is here trivially derived in analogy with $T_{BL}(x)$ and $T_{ML}(x)$ (which are also model validity measures) from the previous section. The equality marked $=^*$ follows from the fact that the coefficient k_b is only present in one of the terms of $V_1([k_m, k_b], x)$.

The minimization of $V_2(k_b, x)$ is a linear regression problem which means that the least-square technique can be used. This results in an estimate

$$\hat{k}_b = \arg\min_{k_b\in\mathcal{D}_{BL}} V_2(k_b, x) = (\varphi_b^T\varphi_b)^{-1}\varphi_b^T Y_b \tag{5.17}$$

where

$$\varphi_b = \begin{bmatrix} h_b(p_{b,s}(t_1)) \\ \vdots \\ h_b(p_{b,s}(t_N)) \end{bmatrix} \qquad Y_b = \begin{bmatrix} f(m_s(t_1) - p_{b,s}(t_1), \alpha_s(t_1), p_{m,s}(t_1)) \\ \vdots \\ f(m_s(t_N) - p_{b,s}(t_N), \alpha_s(t_N), p_{m,s}(t_N)) \end{bmatrix}$$

The test quantity $T_{BL}^{ep'}$, i.e. the estimate $\hat{k}_m$, is formed in the same way with corresponding matrices φ_m and Y_m.

From Section 4.5.1, we know that we should use normalization to make the significance level of the hypothesis tests independent of the input signals. With normalization, the two test quantities $T_{BL}^{ep}(x)$ and $T_{ML}^{ep}(x)$ become

$$T_{BL}^{ep}(x) = \sqrt{\varphi_m^T \varphi_m} T_{BL}^{ep'}(x) = \sqrt{\varphi_m^T \varphi_m} \hat{k}_m \tag{5.18a}$$

$$T_{ML}^{ep}(x) = \sqrt{\varphi_b^T \varphi_b} T_{ML}^{ep'}(x) = \sqrt{\varphi_b^T \varphi_b} \hat{k}_b \tag{5.18b}$$

As in the previous section, test quantities $T_{BL}^{ep}(x)$ and $T_{ML}^{ep}(x)$ have been constructed and decoupling has been achieved in accordance with the specifications of the two hypothesis tests δ_{BL} and δ_{ML}.

5.4.2 A Comparison Between the Prediction Principle and the Estimate Principle

The diagnosis problem investigated here is in principle the same as the one investigated in Section 4.8.2. There we saw that the estimate principle gives the best possible test quantity. This means that the test quantities $T_{BL}^{ep}(x)$ and $T_{ML}^{ep}(x)$ given in (5.18) should be better than $T_{BL}^{pp}(x)$ and $T_{ML}^{pp}(x)$ given in (5.15) and (5.16) respectively.

The comparison of the performance of the two types of test quantities will be based on the principles discussed in Section 4.6.2. To compare $T_{BL}^{ep}(x)$ against $T_{BL}^{pp}(x)$, we will construct two hypothesis tests δ_{BL}^{ep} and δ_{BL}^{pp}. To compare $T_{ML}^{ep}(x)$ against $T_{ML}^{pp}(x)$, we will construct two hypothesis tests δ_{ML}^{ep} and δ_{ML}^{pp}.

To make the comparison, we need to obtain the power function for all four tests. In this situation, where there is no knowledge or assumptions about the model errors or the measurement errors, a good solution is to use the method based on measurements on the real process. In accordance with the procedure in Section 4.6.1, only a limited number of leakage areas are studied, i.e. corresponding to 0, 1, 2, and 3.5 mm diameter. To estimate the probability density function in this case is difficult because of the large amount of data that would be needed. Only 24 independent data sets were used for the analyses and therefore a simpler and less accurate approach has to be chosen.

Both boost leakage and manifold leakage were studied. The results of these studies are shown in Figures 5.11 to 5.14. Consider first manifold leakage and Figure 5.11. The x-axis represents the different leakage areas corresponding to 0, 1, 2, and 3.5 mm diameter. For each leakage area, the test quantities $T_{BL}^{ep}(x)$ and $T_{ML}^{ep}(x)$ were calculated for each of the 24 data sets. The values of $T_{BL}^{ep}(x)$ and $T_{ML}^{ep}(x)$ are indicated with "x" and "o" respectively. To make the plot more clear, all "x":s have been moved slightly to the right. For each leakage area, also the mean and the standard deviation are calculated and shown as horizontal bars. The middle bar is the mean and the upper and lower bars are two times the standard deviation.

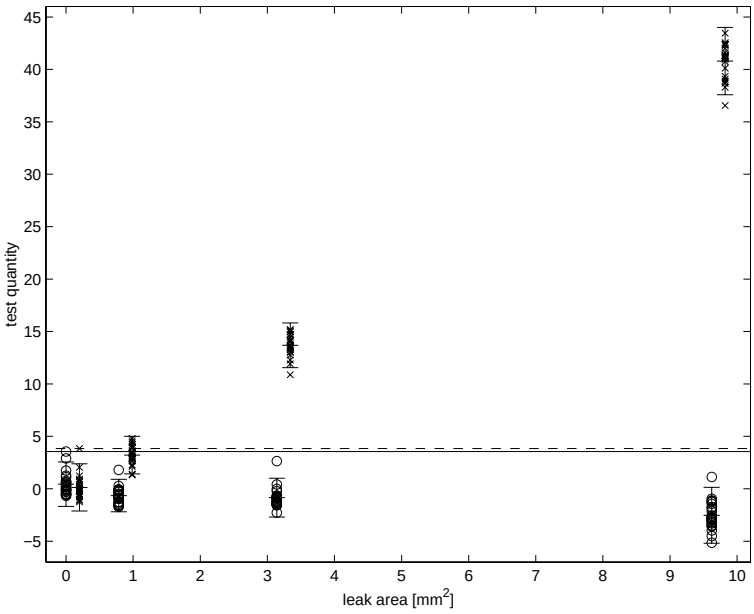

Figure 5.11: The test quantities $T_{BL}^{ep}(x)$ (x-marks) and $T_{ML}^{ep}(x)$ (circles), based on the estimate principle, for different manifold-leakage areas.

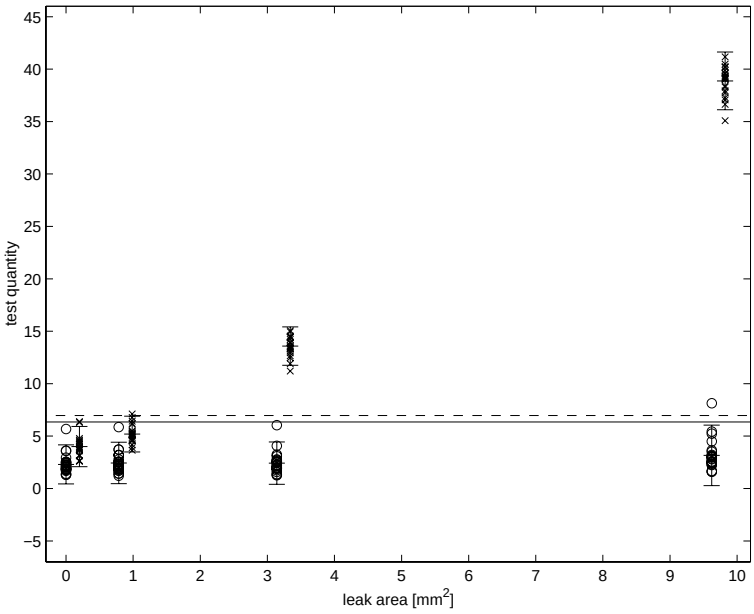

Figure 5.12: The test quantities $T_{BL}^{pp}(x)$ (x-marks) and $T_{ML}^{pp}(x)$ (circles), based on the prediction principle, for different manifold-leakage areas.

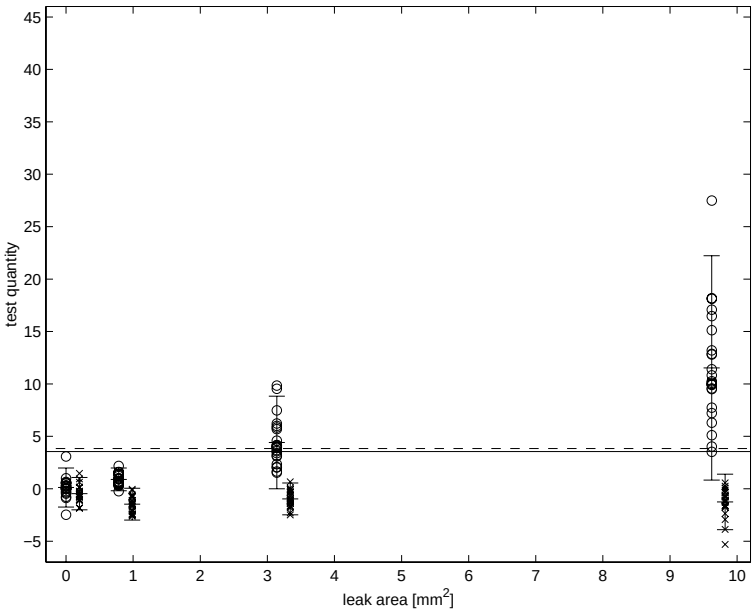

Figure 5.13: The test quantities $T_{BL}^{ep}(x)$ (x-marks) and $T_{ML}^{ep}(x)$ (circles), based on the estimate principle, for different boost-leakage areas.

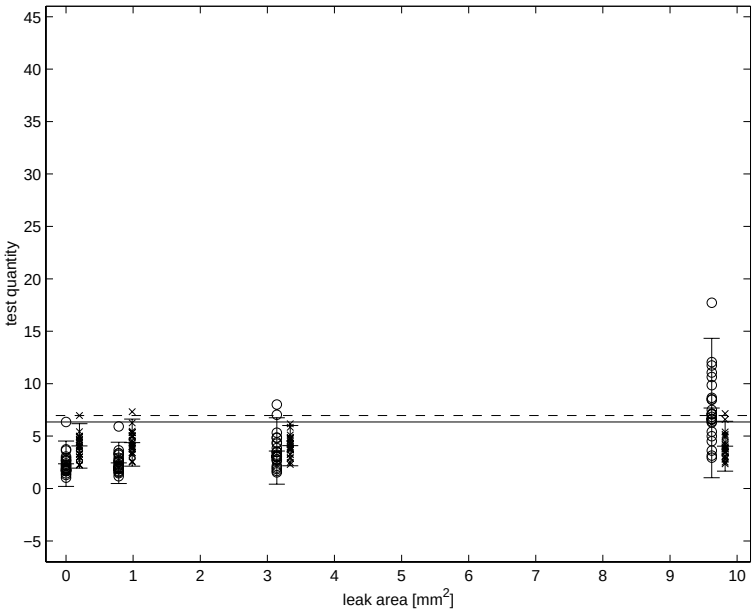

Figure 5.14: The test quantities $T_{BL}^{pp}(x)$ (x-marks) and $T_{ML}^{pp}(x)$ (circles), based on the prediction principle, for different boost-leakage areas.

According to Section 4.6.2, thresholds need to be chosen such that the significance level in two compared hypothesis tests becomes equal. Since we don't have the probability density function, this can not achieved. Instead, each threshold is chosen as the maximum value of the corresponding calculated test quantity in the fault-free case. Consider again Figure 5.11. The maximum values of the test quantities $T_{BL}^{ep}(x)$ and $T_{ML}^{ep}(x)$ for the fault-free case, i.e. leakage area 0, are marked by the dashed and solid lines respectively. Similarly, one can see how the thresholds for the test quantities $T_{BL}^{pp}(x)$ and $T_{ML}^{pp}(x)$ are chosen by studying Figure 5.14.

The power function is the probability to reject H_0, i.e. the probability that the test quantity is above the threshold. As was said above, we don't have the probability density function, which means that exact values of the power function can not be calculated. However, by studying Figure 5.11 and looking at the mean and standard deviation values, we can quite easily get a coarse estimate of the probability that the test quantity is above the threshold. For example, it is obvious that the power function $\beta_{BL}^{ep}([0\ k_m])$, corresponding to $T_{BL}^{ep}(x)$, will increase as the leakage area increases. Also, we can conclude that the power function for the leakage with an area of 3.1 mm^2 is large, which means that it should be no problem to detect a manifold leakage with this area. Further, the power function for the leakage with an area of 0.8 mm^2, is probably quite low, which means that it is hard to distinguish this leakage from the no-leakage case.

Now return to the comparison of test quantities. First compare Figure 5.11, showing the test quantities $T_{BL}^{ep}(x)$ and $T_{ML}^{ep}(x)$, and Figure 5.12, showing the test quantities $T_{BL}^{ep}(x)$ and $T_{ML}^{ep}(x)$. We see that the test quantity $T_{BL}^{ep}(x)$ is slightly more above the threshold than $T_{BL}^{pp}(x)$. This means that the power function $\beta_{BL}^{ep}([0\ k_m])$ is very likely to be larger than $\beta_{BL}^{pp}([0\ k_m])$. In other words, for the manifold leakage, the estimate principle is better than the prediction principle.

Next compare Figure 5.13 and Figure 5.14. From these figures it can be concluded that the power functions $\beta_{ML}^{ep}([k_b\ 0])$ and $\beta_{ML}^{pp}([k_b\ 0])$, along the "boost-leakage axis", are not as large as $\beta_{BL}^{ep}([0\ k_m])$ and $\beta_{BL}^{pp}([0\ k_m])$, along the "manifold-leakage axis". However, it is obvious that $\beta_{ML}^{ep}([k_b\ 0])$ is larger than $\beta_{ML}^{pp}([k_b\ 0])$. Consider for example the leakage area 3.1 mm^2. For this case $\beta_{ML}^{pp}([k_b\ 0])$ should be close to zero and $\beta_{ML}^{ep}([k_b\ 0])$ is probably larger than 0.5. Again we can conclude that the estimate principle is better than the prediction principle.

Discussion

Even though we have not been able to estimate density functions, we can from this study conclude that, of the two principles studied, the best principle for diagnosing leakage is the estimate principle. This is no surprise since we already in Section 4.8, in different similar situations, drew the same conclusion. However, in for example the theoretical study in Section 4.8.2, we used the assumption of independently and identically Gaussian distributed noise. This assumption do

not hold in the real case investigated in this section, but nevertheless it is obvious that the conclusion that the estimate principle is better than the prediction principle, still holds.

In production cars, a principle similar to the prediction principle is often used, e.g. see (*Air Leakage Detector for IC Engine*, 1994). A reason for this is that models of the leaks are not required (see the test quantities described by (5.15) and (5.16)). It is interesting to note that the technique developed here, i.e. to use models of the leaks and then estimate the leakage area, performs better than the solution common in production cars. This method is patent pending by SAAB Automobile. With this better solution, it is possible to make the legislative regulations harder, which means that all car manufacturer are forced to built diagnosis systems with better leakage detection performance. In the end this hopefully means that lower fleet emissions can be obtained.

5.5 Comparison of Different Fault Models for Leaks

So far we have modeled the leaks as deviations of constant parameters from their nominal values. Here we will extend the discussion and consider the following three different fault models from Section 2.1.4:

- The leakage area is assumed to be constant (as before).
- The leakage area is assumed to be changing slowly. That is, the leakage area is interpreted as a signal with low bandwidth.
- The leakage-area is assumed to be changing once and abruptly, i.e. the abrupt change model is assumed.

It can be argued that each of these fault models is good in some sense.

Although not further discussed here, it is also possible to assume that the leakage area is piecewise constant. Another possibility is for instance to use a combination of the low-bandwidth assumption together with the abrupt-change assumption, i.e. the leakage area is mainly of low bandwidth but contains abrupt jumps.

Next we will discuss the estimate and the prediction principle separately. In all cases, we assume that we can use all data generated up to the time-point the diagnosis is performed. This means that the time window is chosen to be growing (or infinite). Other choices, e.g. a sliding fixed-length time-window, are also possible.

5.5.1 Using the Estimate Principle

We will only discuss a test quantity based on the estimate $\hat{k}_b$. However, all results are applicable also for a test quantity based on the estimate $\hat{k}_m$.

For the constant model, the least square algorithm can be used, in accordance with (5.17). The estimate $\hat{k}_b$ will in this case be the average leakage area over

all time. However when a leakage occurs, it will take a long time before this average grows. A better choice is to weight recent data more. A common choice is to obtain $\hat{k}_b$, at the time t, as follows:

$$\hat{k}_b = \arg \min_{k_b \in \mathcal{D}_{BL}} \frac{1}{N} \sum_{k=0}^{t} \lambda^{t-k} \big(m_s(k) - k_b h_b(p_b(k)) - f(p_{b,s}(k), \alpha_s(k), p_{m,s}(k)) \big)^2 \tag{5.19}$$

Depending on the choice of λ, convergence time is traded against accuracy. In a recursive form, this is the RLS (Recursive Least Square) algorithm (Ljung, 1987). The test quantity $T_{ML}(x)$ can then be chosen as

$$T_{ML}(x) = \hat{k}_b \tag{5.20}$$

or possibly by also using some normalization.

Using the low-bandwidth model, i.e. the leakage area is assumed to be changing slowly, the parameter describing the leakage becomes a function of time, i.e. $\theta_b = k_b(t)$. An estimate of $k_b(t)$ can be obtained by using for example the RLS-algorithm in combination with (5.19). Since $\hat{\theta}_b = \hat{k}_b(t)$ is now a signal, it is not obvious how to form the test quantity, i.e. how to choose the measure $\|\cdot\|$ in (4.26). One solution is however to choose the most recent value of $\hat{k}_b(t)$ and in that case, the test quantity becomes equivalent to (5.20).

Using the abrupt-change model, we need to estimate both the change time t_{ch} and the leakage area k_b, i.e. $\theta_b = [t_{ch} \; k_b]$. However, in contrast to the approaches above, this is not a simple linear regression problem. The test quantity can then simply be chosen as the estimate $\hat{k}_b$, possibly normalized.

Experimental Results

The performance of an estimate with a weighting of recent data more, in accordance with (5.19), was validated by experiments. As was said above, this can correspond to the constant or the low-bandwidth model. The estimation of k_b and k_m are shown in Figures 5.15 and 5.16. Also for this experiment, the FTP-75 test-cycle was used. After approximately 500 seconds, a leak was applied suddenly. The most realistic fault model would therefore probably be the abrupt-change model.

The upper plot in both figures shows the k_b estimate as a function of time and the lower plot, the k_m estimate as a function of time. It is seen that the k_b estimate in both figures have discontinuities. The reason is that the on-line estimation of k_b is applied only when the boost pressure is higher than 102 kPa. This is because for boost pressures close to ambient pressure, the air flow through the boost leak is very small which means that the measurement data will contain no or very little information about the value of the k_b coefficient. If also these data were used, the k_b estimate would easily drift away from its real value. In other words, the exclusion of data corresponding to boost pressures lower than 102 kPa, is a primitive way of achieving robustness and should be seen as an alternative to normalization.

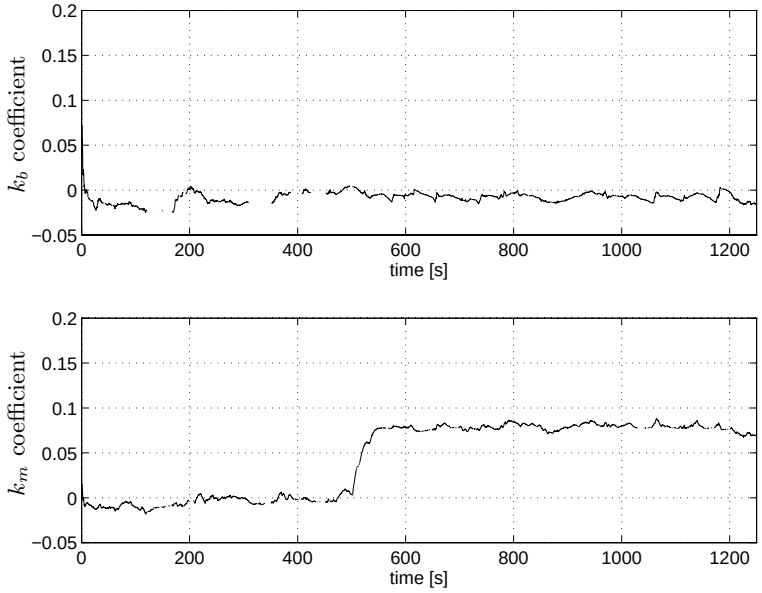

Figure 5.15: Estimation of the k_b coefficient (upper plot) and k_m coefficient (lower plot) when a 3.5 mm (diameter) manifold leak occurs at around $t = 500$ s.

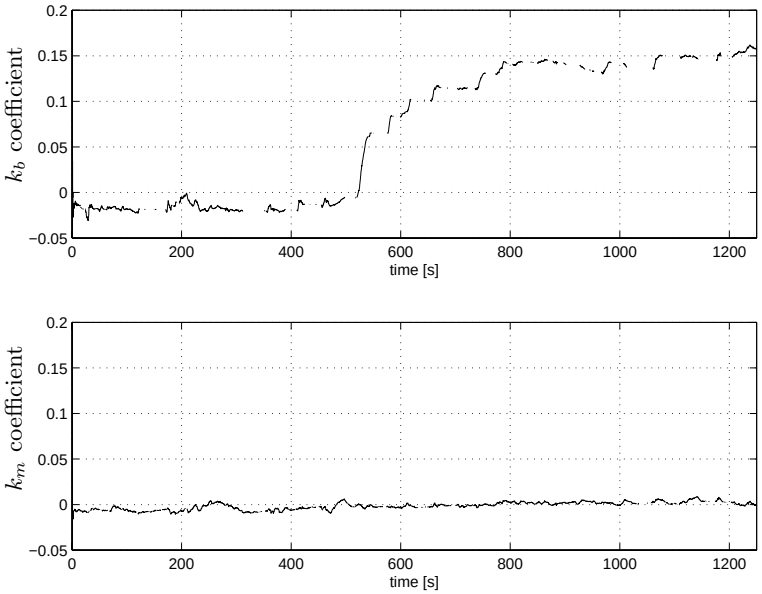

Figure 5.16: Estimation of the k_b coefficient (upper plot) and k_m coefficient (lower plot) when a 5 mm (diameter) boost leak occurs at around $t = 500$ s.

In Figure 5.15, the leak is a 3.5 mm manifold leak and it can be seen that the k_m estimate responds quickly when the leak occurs. Similarly in Figure 5.16, we see how the k_b estimate responds when a 5 mm boost leak occurs. In this case the estimate converges more slowly. The reason is, as said above, that the estimation is only active when the boost pressure is higher than 102 kPa.

From the clear responses shown in Figures 5.15 and 5.16, it is obvious that test quantities based on the estimates $\hat{k}_m$ and $\hat{k}_b$, will be quite good. Further, a diagnosis system using these test quantities is likely to have highly satisfactory performance.

5.5.2 Using the Prediction Principle

When discussing the prediction principle, we will assume that the test quantity is on a form similar to (5.14). That is, the test quantity is a model validity measure of the whole model and not only half the model as in (5.15). This means that the calculation of the test quantity must include a parameter estimation, as is seen in for example (5.14). We will discuss only the test quantity $T_{BL}(x)$, but the results are valid also for $T_{ML}(x)$.

As for the estimate principle, the use of a constant model without weighting recent data more, will result in bad performance. This is actually the case for the test quantity $T_{BL}^{pp'}(x)$ defined by (5.14). When a leakage occurs, it takes a long time before the estimate of k_b becomes good. This means that, from the moment the leakage occurs, to the moment k_b becomes good, the test quantity $T_{BL}^{pp'}(x)$ will become large and the decoupling of the fault mode BL will be bad. The underlying reason is of course that any leakage that *occurs*, in other words a leakage that is not present all the time, does not match the model assumption of constant leakage-size. As for the estimate principle, it is also possible to weight recent data more. This would result in that the estimation more quickly becomes good when a leakage occurs, which further implies improved decoupling of the fault mode BL.

If instead a low-bandwidth model is used and also the occurred leakage matches this fault model, then we can expect good results. This means that the test quantity can be written as

$$T_{BL}^{LP}(x) = \min_{k_b(t)\in\mathcal{LP}} \frac{1}{N}\sum_{t=1}^{N}\big(m_s(t) - f(p_{b,s}(t),\alpha_s(t),p_{m,s}(t)) - k_b(t)h_b(p_{b,s}(t))\big)^2 + \\ + \frac{1}{N}\sum_{t=1}^{N}\big(f(p_{b,s}(t),\alpha_s(t),p_{m,s}(t)) - g(p_{m,s}(t),n_s(t))\big)^2 \quad (5.21)$$

where $\mathcal{LP}$ is the set of low-bandwidth signals considered. To solve the optimization involved in calculating (5.21) can be quite difficult. However, by using the two-step approach from Section 4.2.1, the signal $k_b(t)$ can first be estimated by using the RLS-algorithm based on (5.19). This was done in the experiments reported below.

If the leakage occurs abruptly, the test quantity based on the low-bandwidth model will perform better than a test quantity based on the constant model. However, the performance will still not be perfect. The reason is again that the time-variant behavior of the leakage doesn't match the fault model. To handle the situation of abruptly changing leakage well, we need to use the abrupt-change model. By using similar ideas as in Example 4.2, we can construct such a test quantity as

$$T_{BL}^{ac}(x) = \min_{t_{ch},k_b} \frac{1}{t_{ch}-1} \sum_{t=1}^{t_{ch}-1} \big(m_s - f(p_{b,s},\alpha_s,p_{m,s})\big)^2 + \\ + \frac{1}{N-t_{ch}} \sum_{t=t_{ch}}^{N} \big(m_s - f(p_{b,s},\alpha_s,p_{m,s}) - k_b h_b(p_{b,s})\big)^2 + \\ + \frac{1}{N} \sum_{t=1}^{N} \big(f(p_{b,s},\alpha_s,p_{m,s}) - g(p_{m,s},n_s)\big)^2$$

Again the two-step approach can be used when calculating this test quantity. In the experiments reported below, the CUSUM algorithm (Basseville and Nikiforov, 1993) was first used to detect the change, i.e. to find t_{ch}.

Experimental Results

Diagnosis based on the low-bandwidth and the abrupt-change model were validated in experiments. Again the FTP-75 test cycle was used and in all experiments, the leakage occurs suddenly after around 500 seconds.

In Figures 5.17 and 5.18, the test quantities $T_{BL}^{LP}(x)$ and $T_{ML}^{LP}(x)$ are plotted as a function of time. In Figures 5.19 and 5.20, the test quantities $T_{BL}^{ac}(x)$ and $T_{ML}^{ac}(x)$ are plotted as a function of time.

Since all leaks occurs suddenly, the most accurate fault model should be the abrupt-change model. Therefore, the test quantities based on this model, i.e. $T_{BL}^{ac}(x)$ and $T_{ML}^{ac}(x)$, should perform better than the ones based on the low-bandwidth model. If this is the case we should expect to see some differences in the plots at least around the time the leakage occurs, i.e. around time $t = 500s$. By comparing the plots of $T_{ML}^{LP}(x)$ and $T_{ML}^{ac}(x)$ for the manifold leakage, we see that $T_{ML}^{LP}(x)$ has a small bump right after $t = 500$. Also by comparing the plots of $T_{BL}^{LP}(x)$ and $T_{BL}^{ac}(x)$ for the boost leakage, we see that also $T_{BL}^{LP}(x)$ has a small bump right after $t = 500$.

This means that the test quantities based on the abrupt-change model better manage to perform decoupling. Since the decoupling for $T_{BL}^{ac}(x)$ and $T_{ML}^{ac}(x)$ are better, we should be able to use lower thresholds and in that way obtain larger power functions. In other words, the test quantities $T_{BL}^{ac}(x)$ and $T_{ML}^{ac}(x)$, that are based on the fault model that best matches the real situation, are the best.

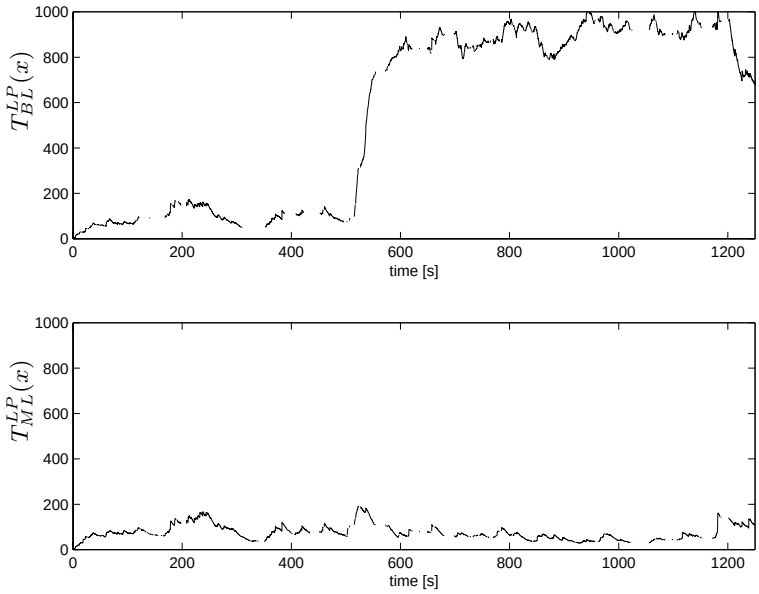

Figure 5.17: The test quantities $T_{BL}^{LP}(x)$ (upper plot) and $T_{ML}^{LP}(x)$ (lower plot), using the low-bandwidth model, when a 3.5 mm (diameter) manifold leak occurs at around $t = 500$ s.

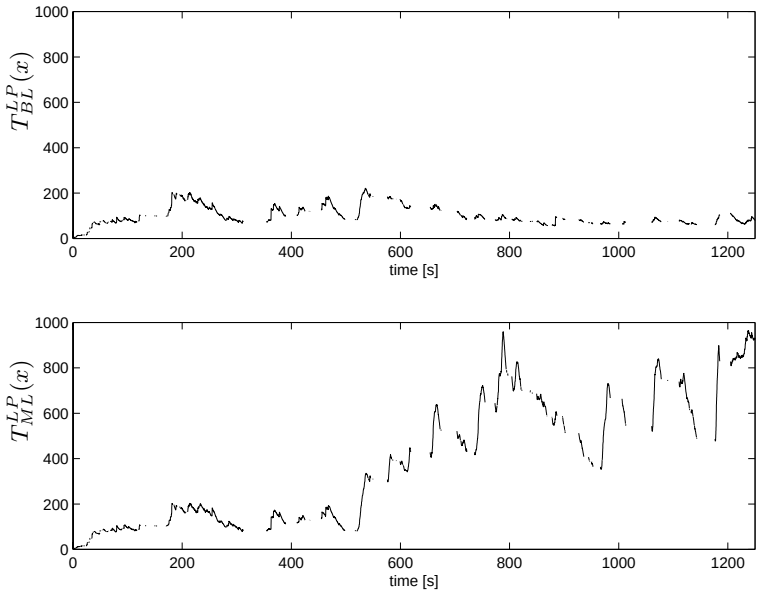

Figure 5.18: The test quantities $T_{BL}^{LP}(x)$ (upper plot) and $T_{ML}^{LP}(x)$ (lower plot), using the low-bandwidth model, when a 5 mm (diameter) boost leak occurs at around $t = 500$ s.

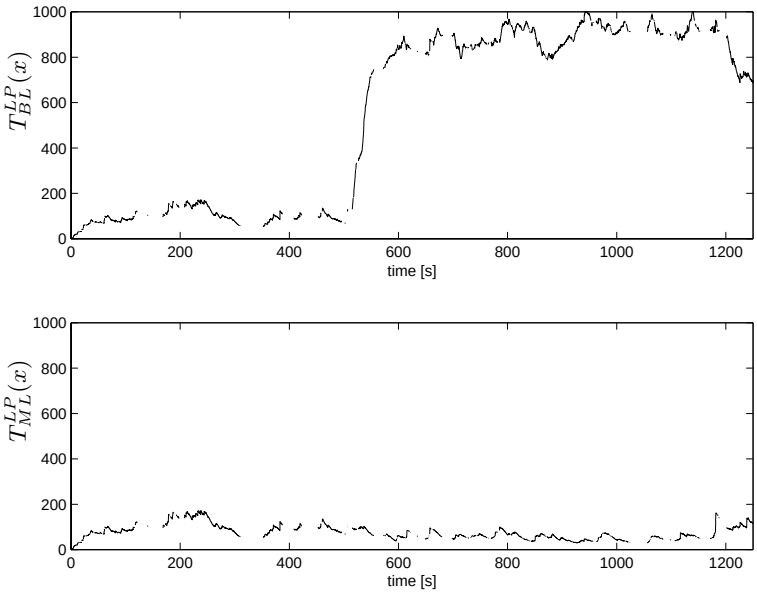

Figure 5.19: The test quantities $T_{BL}^{ac}(x)$ (upper plot) and $T_{ML}^{ac}(x)$ (lower plot), using the abrupt-change model, when a 3.5 mm (diameter) manifold leak occurs at around $t = 500$ s.

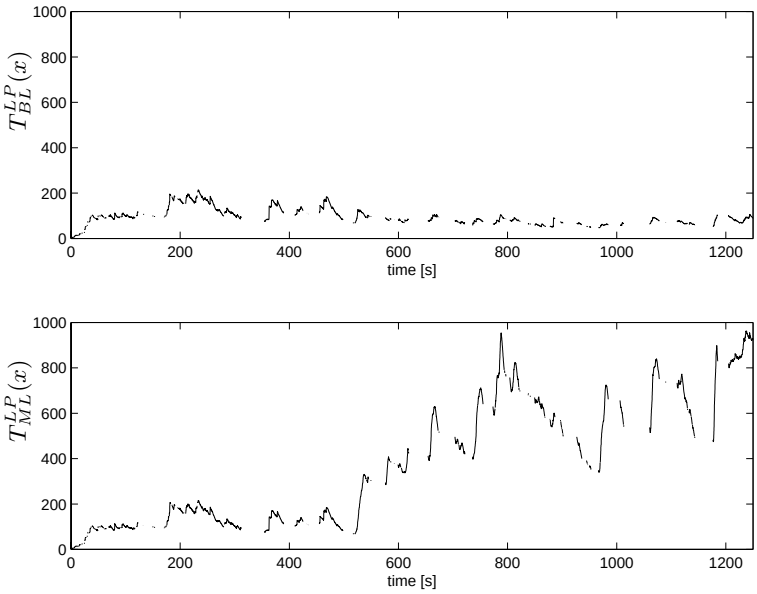

Figure 5.20: The test quantities $T_{BL}^{ac}(x)$ (upper plot) and $T_{ML}^{ac}(x)$ (lower plot), using the abrupt-change model, when a 5 mm (diameter) boost leak occurs at around $t = 500$ s.

5.6 Diagnosis of Both Leakage and Sensor Faults

This section presents the design of a diagnosis system capable of diagnosing both sensor faults and leakage. The constructed diagnosis system is then experimentally validated in Sections 5.7 and 5.8. Again we remind the reader that the objective is not to present a complete design but rather to illustrate principles.

index i	component name	component fault modes
b	boost pipe	NF^b, BL (Boost Leak)
m	manifold	NF^m, ML (Manifold Leak)
bs	boost pressure sensor	NF^{bs}, BB (Boost pressure sensor Bias), BAF (Boost pressure sensor Arbitrary Fault)
ms	manifold pressure sensor	NF^{ms}, MG (Manifold pressure sensor Gain-fault), MC (Manifold pressure sensor Cut-off)
ts	throttle sensor	NF^{ts}, TLF (Throttle sensor Linear Fault)
as	air mass-flow sensor	NF^{as}, ALC (Air mass-flow sensor Loose Contact)

Table 5.1: The components and component fault-modes considered.

5.6.1 Fault Modes Considered

The different components and corresponding component fault-modes that will be considered, are listed in Table 5.1. Further, the system fault-modes considered are listed in Table 5.2. In accordance with Section 2.2.1, the system fault-modes are written in bold-face letters to distinguish them from the component fault-modes. As seen, only single fault-modes are considered. Compared to the study in Section 5.4, six more fault modes have been included and all the new ones are related to sensor faults.

NF	No Fault
BL	Boost Leak
ML	Manifold Leak
BB	Boost Pressure Sensor Bias
BAF	Boost Pressure Sensor Arbitrary Fault
MG	Manifold Pressure Sensor Gain-Fault
MC	Manifold Pressure Sensor Cut-Off
TLF	Throttle Sensor Linear Fault
ALC	Air Mass-Flow Sensor Loose Contact

Table 5.2: The system fault-modes considered.

The definitions of each fault mode, in the form of models $\mathcal{M}_\gamma(\theta)$, will be given later in Section 5.6.3, where at the same time, the construction of the test quantities is described. There we will realize that the following relations between the (system) fault-modes hold:

$$\mathbf{NF} \preccurlyeq^* \mathbf{BL} \tag{5.22a}$$
$$\mathbf{NF} \preccurlyeq^* \mathbf{ML} \tag{5.22b}$$
$$\mathbf{NF} \preccurlyeq^* \mathbf{BB} \preccurlyeq \mathbf{BAF} \tag{5.22c}$$
$$\mathbf{NF} \preccurlyeq^* \mathbf{MG} \tag{5.22d}$$
$$\mathbf{NF} \preccurlyeq^* \mathbf{TLF} \tag{5.22e}$$
$$\mathbf{NF} \preccurlyeq^* \mathbf{ALC} \tag{5.22f}$$

Note that there is no relation involving the fault mode **MC**. The reason and consequences of this will become clear later.

5.6.2 Specifying the Hypothesis Tests

To develop the actual hypothesis tests, we first need to decide the set of hypotheses to test. We will use one hypothesis test for each fault mode. Thus the set of hypothesis tests becomes

$$H_k^0 : F_p \in M_k \tag{5.23a}$$
$$H_k^1 : F_p \in M_k^C \tag{5.23b}$$
$$k \in \{\mathbf{NF}, \mathbf{BL}, \mathbf{ML}, \mathbf{BB}, \mathbf{MG}, \mathbf{MC}, \mathbf{TLF}, \mathbf{ALC}, \mathbf{BAF}\}$$

Because of the relations (5.22), we know from Section 3.2.1 that the choice of sets M_k is not completely free. The choice to use one hypothesis test dedicated to each system fault-mode, together with a desire to decouple as few fault modes as possible in each test quantity, leads to the *unique* choice of sets M_k shown in Table 5.3.

k	M_k
NF	$\{\mathbf{NF}\}$
BL	$\{\mathbf{NF}, \mathbf{BL}\}$
ML	$\{\mathbf{NF}, \mathbf{ML}\}$
BB	$\{\mathbf{NF}, \mathbf{BB}\}$
BAF	$\{\mathbf{NF}, \mathbf{BB}, \mathbf{BAF}\}$
MG	$\{\mathbf{NF}, \mathbf{MG}\}$
MC	$\{\mathbf{MC}\}$
TLF	$\{\mathbf{NF}, \mathbf{TLF}\}$
ALC	$\{\mathbf{NF}, \mathbf{ALC}\}$

Table 5.3: The sets M_k for the nine hypothesis tests.

In the next section, the design of test quantities will be discussed. There, also all fault modes will be defined via models $\mathcal{M}_\gamma(\theta)$. All these definitions of

$\mathcal{M}_\gamma(\theta)$ will result in a fault-state vector θ as

$$\theta = [\theta_b \ \theta_m \ \theta_{bs} \ \theta_{ms} \ \theta_{ts} \ \theta_{as}] = [k_b, \ k_m, \ (b_{p_b}, \ c_2(t)), \ g_{p_m}, \ (g_\alpha, \ b_\alpha), \ c_1(t)]$$

where $c_1(t)$ and $c_2(t)$ are signals while the other parameters are scalar constants.

5.6.3 Fault Modeling and Design of Test Quantities

The test quantities will be designed using the prediction principle. Then we know, from Section 4.2, that the problem of designing the test quantities $T_k(x)$ consists of determining the model validity measure $V_k(\theta, \mathbf{x})$ and the set Θ_k^0. The test quantity $T_k(x)$ then becomes

$$T_k(x) = \min_{\theta \in \Theta_k^0} V_k(\theta, x) \tag{5.24}$$

Next, $V_k(\theta, \mathbf{x})$ and Θ_k^0 will be defined for all nine hypothesis tests corresponding to the sets M_k given in Table 5.3. Also the models $\mathcal{M}_\gamma(\theta)$ and the sets Θ_γ will be defined.

No Fault NF

The model $\mathcal{M}_{\mathbf{NF}}$, corresponding to the fault mode **NF**, has already been given in (5.11). The parameter space $\Theta_{\mathbf{NF}}$ is $\Theta_{\mathbf{NF}} = \{[0, 0, 0, \mathbf{0}, 1, 1, 0, \mathbf{1}]\}$, where bold-face numbers denote vectors. The set $M_{\mathbf{NF}}$ was defined as $M_{\mathbf{NF}} = \{\mathbf{NF}\}$. By remembering the expression for Θ_k^0 from (3.1), we realize that this means that the set $\Theta_{\mathbf{NF}}^0$ becomes $\Theta_{\mathbf{NF}}^0 = \Theta_{\mathbf{NF}}$.

Since $\Theta_{\mathbf{NF}}^0$ contains exactly one value of θ, the test quantity becomes, in accordance with (4.4), $T_{\mathbf{NF}}(\mathbf{x}) = V_{\mathbf{NF}}(\mathbf{x})$. The measure $V_{\mathbf{NF}}(\mathbf{x})$ is defined as

$$V_{\mathbf{NF}}(\mathbf{x}) = \frac{1}{N}\sum_{t=1}^{N}\big(m_s - f(p_{b,s}, \alpha_s, p_{m,s})\big)^2 + \\ + \frac{1}{N}\sum_{t=1}^{N}\big(f(p_{b,s}, \alpha_s, p_{m,s}) - g(p_{m,s}, n_s)\big)^2 \tag{5.25}$$

Note that, to simplify notation, we have dropped the time-argument of signals.

Using the measure (5.25) implies that if the present fault mode is **NF**, then the test quantity becomes small and for all other fault modes, the test quantity becomes large, or at least larger. This fulfills the specification of the hypothesis test $\delta_{\mathbf{NF}}$ given by (5.23).

Boost Leak BL

The model $\mathcal{M}_{\mathbf{BL}}(k_b)$ was given already by (5.12). The scalar parameter k_b defines the equivalent area of the leakage and is, as before, constrained by $k_b \in \mathcal{D}_{BL}^b =]0, 0.5]$. The measure $V_{\mathbf{BL}}(k_b, \mathbf{x})$ is

$$V_{\mathbf{BL}}(k_b, \mathbf{x}) = \frac{1}{N}\sum_{t=1}^{N}\big(m_s - f(p_{b,s}, \alpha_s, p_{m,s}) - k_b h_b(p_{b,s})\big)^2 + \\ + \frac{1}{N}\sum_{t=1}^{N}\big(f(p_{b,s}, \alpha_s, p_{m,s}) - g(p_{m,s}, n_s)\big)^2 \quad (5.26)$$

Compared to the measure used in (5.15), this expression contains two terms. The motivation for this here, is that we want the test quantity $T_{\mathbf{BL}}(x)$ to respond to as many of the other fault modes as possible. That is, in all cases the present fault mode does not belong to $M_{\mathbf{BL}} = \{\mathbf{NF}, \mathbf{BL}\}$, we want the null hypothesis $H_{\mathbf{BL}}^0$ to be rejected.

The parameter space $\mathcal{D}_{BL}^b$ also defines $\Theta_{\mathbf{BL}}$, in accordance with Section 2.2.1. The definition of the set $M_{\mathbf{BL}}$ implies that the set $\Theta_{\mathbf{BL}}^0$ becomes $\Theta_{\mathbf{BL}}^0 = \Theta_{\mathbf{NF}} \cup \Theta_{\mathbf{BL}}$.

Using the measure (5.26) implies that if the present fault mode belongs to $M_{\mathbf{BL}}$, then the test quantity becomes small and for all other fault modes, the test quantity becomes large. This fulfills the specification of the hypothesis test $\delta_{\mathbf{BL}}$ given by (5.23).

Manifold Leak ML

The model $\mathcal{M}_{\mathbf{ML}}(k_m)$ is obtained in analogy with $\mathcal{M}_{BL}(k_b)$. The scalar parameter k_m is constrained by $k_m \in \mathcal{D}_{ML}^m =]0, 0.5]$ and the measure $V_{ML}(k_m, \mathbf{x})$ is

$$V_{\mathbf{ML}}(k_m, \mathbf{x}) = \frac{1}{N}\sum_{t=1}^{N}\big(m_s - f(p_{b,s}, \alpha_s, p_{m,s})\big)^2 + \\ + \frac{1}{N}\sum_{t=1}^{N}\big(f(p_{b,s}, \alpha_s, p_{m,s}) - g(p_{m,s}, n_s) + k_m h_m(p_{m,s})\big)^2$$

The sets $\Theta_{\mathbf{ML}}$ and $\Theta_{\mathbf{ML}}^0$ follows accordingly.

Boost Pressure Sensor Bias BB

The model $\mathcal{M}_{\mathbf{BB}}(b_{p_b})$ corresponding to this fault mode is obtained by using the fault-free model (5.11) together with identities (5.10) but replacing (5.10b) with $p_{b,s} = p_b + b_{p_b}$. This means that the model $\mathcal{M}_{\mathbf{BB}}(b_{p_b})$ can be written as

$$m = f(p_{b,s} - b_{p_b}, \alpha_s, p_{m,s}) \\ f(p_{b,s} - b_{p_b}, \alpha_s, p_{m,s}) = g(p_{m,s}, n_s)$$

The scalar parameter b_{p_b} is constrained by $b_{p_b} \in [-30, 0[\ \cup \ ]0, 30]$ which means that the parameter $\theta_{bs} = [b_{p_b} \ c_2(t)]$ is constrained by $\theta_{bs} \in \mathcal{D}_{BB}^{bs} = [-30, 0[\ \cup \ ]0, 30] \times \{0\}^N$.

The measure $V_{\mathbf{BB}}(b_{p_b}, \mathbf{x})$ is

$$V_{BB}(b_{p_b}, \mathbf{x}) = \frac{1}{N} \sum_{t=1}^{N} \big(m_s - f(p_{b,s} - b_{p_b}, \alpha_s, p_{m,s})\big)^2 + \\ + \frac{1}{N} \sum_{t=1}^{N} \big(f(p_{b,s} - b_{p_b}, \alpha_s, p_{m,s}) - g(p_{m,s}, n_s))\big)^2$$

The sets $\Theta_{\mathbf{BB}}$ and $\Theta_{\mathbf{BB}}^0$ follows as before.

Boost Pressure Sensor Arbitrary Fault BAF

The model $\mathcal{M}_{\mathbf{BAF}}(c_2(t))$ corresponding to this fault mode is obtained by using the fault-free model (5.1) together with identities (5.10) but replacing (5.10b) with $p_{b,s} = p_b + c_2(t)$. The parameter $c_2(t)$ is now a signal taking arbitrary values. This means that the parameter space $\mathcal{D}_{BAF}^{bs}$ becomes $\mathcal{D}_{BAF}^{bs} = \{0\} \times (\mathbb{R}^N - \{0\}^N)$. Note that this definition of the model $\mathcal{M}_{\mathbf{BAF}}(c_2(t))$ explains the relation $\mathbf{BB} \preccurlyeq \mathbf{BAF}$ noted already in (5.22). That is, for each b_{p_b}, the signal $c_2(t)$ can always be chosen as $c_2(t) \equiv b_{p_b}$, which implies that $\mathcal{M}_{\mathbf{BB}}(b_{p_b}) = \mathcal{M}_{\mathbf{BAF}}(c_2(t))$

The measure $V_{\mathbf{BAF}}(c_2(t), \mathbf{x})$ is

$$V_{\mathbf{BAF}}(c_2(t), \mathbf{x}) = \frac{1}{N} \sum_{t=1}^{N} \big(m_s - f(p_{b,s} - c_2, \alpha_s, p_{m,s})\big)^2 + \\ + \frac{1}{N} \sum_{t=1}^{N} \big(f(p_{b,s} - c_2, \alpha_s, p_{m,s}) - g(p_{m,s}, n_s))\big)^2$$

The set $\Theta_{\mathbf{BAF}}$ follows as before. The set $\Theta_{\mathbf{BAF}}^0$ could be chosen via the expression (3.1) but an equivalent choice, which is computationally simpler, is $\Theta_{\mathbf{BAF}}^0 = \Theta_{\mathbf{NF}} \cup \Theta_{\mathbf{BAF}}$. This was implicitly assumed when designing $V_{\mathbf{BAF}}(c_2(t), \mathbf{x})$.

Manifold Pressure Sensor Gain-Fault MG

The model $\mathcal{M}_{\mathbf{MG}}(g_{p_m})$ corresponding to this fault mode is obtained by using the fault-free model (5.1) together with identities (5.10) but replacing (5.10c) with $p_{m,s} = g_{p_m} p_m$. The constraint on the scalar parameter g_{p_m} is $g_{p_m} \in \mathcal{D}_{MG}^{ms} = [0.5, 1[\ \cup \ ]1, 2]$.

The measure $V_{\mathbf{MG}}(g_{p_m}, \mathbf{x})$ is

$$V_{\mathbf{MG}}(g_{p_m}, \mathbf{x}) = \frac{1}{N}\sum_{t=1}^{N}\left(m_s - f(p_{b,s}, \alpha_s, p_{m,s}/g_{p_m})\right)^2 + \\ + \frac{1}{N}\sum_{t=1}^{N}\left(f(p_{b,s}, \alpha_s, p_{m,s}/g_{p_m}) - g(p_{m,s}/g_{p_m}, n_s)\right)^2$$

The sets $\Theta_{\mathbf{MG}}$ and $\Theta^0_{\mathbf{MG}}$ follows accordingly.

Manifold Pressure Sensor Cut-Off MC

This fault mode represents a cut-off in the electrical connection to the manifold pressure sensor. The model $\mathcal{M}_{\mathbf{MC}}$ corresponding to this fault mode is obtained by using the fault-free model (5.1) together with identities (5.10) but replacing (5.10c) with $p_{m,s} = g_{p_m} p_m$. The parameter g_{p_m} takes value 1 in the fault-free case and value 0 when there is a cut-off present. This means that for the model **MC**, $g_{p_m} \in \mathcal{D}^{ms}_{MC} = \{0\}$.

The definition of $\mathcal{D}^{ms}_{MC} = \{0\}$ means that the set $\Theta_{\mathbf{MC}}$ contains exactly one value. Remember that the set $M_{\mathbf{MC}}$ was defined as $M_{\mathbf{MC}} = \{\mathbf{MC}\}$. This implies that the set $\Theta^0_{\mathbf{MC}}$ becomes $\Theta^0_{\mathbf{MC}} = \Theta_{\mathbf{MC}}$ and thus, contains only one value. Therefore we have that $T_{\mathbf{MC}}(\mathbf{x}) = V_{\mathbf{MC}}(\mathbf{x})$. The measure $V_{\mathbf{MC}}(\mathbf{x})$ is

$$V_{\mathbf{MC}}(\mathbf{x}) = \frac{1}{N}\sum_{t=1}^{N} p_{b,s}^2$$

Note that, in spite of the simpleness of this expression, the test quantity $T_{\mathbf{MC}}(\mathbf{x})$ will become very large for all $\theta \notin \Theta_{\mathbf{MC}}$. The reason is that the manifold pressure never becomes zero. We can assume that this knowledge is implicitly included in the model of the air-intake system. This is also true for the fault mode **NF** which explains why, according to the relations (5.22), **NF** is *not* a submode (in the limit) of **MC**. Remember that this was the reason why the fault mode **NF** was not included in $M_{\mathbf{MC}}$.

Throttle Sensor Linear Fault TLF

The model $\mathcal{M}_{\mathbf{TLF}}([g_\alpha\ b_\alpha])$ corresponding to this fault mode is obtained by using the fault-free model (5.1) together with identities (5.10) but replacing (5.10d) with $\alpha_s = g_\alpha \alpha + b_\alpha$. The vector valued parameter $[g_\alpha\ b_\alpha]$ is constrained by $[g_\alpha\ b_\alpha] \in \mathcal{D}^{ts}_{TLF} = \mathbb{R}^2 - \{1, 0\}$ and the measure $V_{\mathbf{TLF}}(\theta_{ts}, \mathbf{x}) = V_{\mathbf{TLF}}([g_\alpha\ b_\alpha], \mathbf{x})$

is

$$V_{\mathbf{TLF}}([g_\alpha\ b_\alpha], \mathbf{x}) = \\ = \frac{1}{N}\sum_{t=1}^{N}\left(m_s - f(p_{b,s}, (\alpha_s - b_\alpha)/g_\alpha, p_{m,s})\right)^2 + \\ + \frac{1}{N}\sum_{t=1}^{N}\left(f(p_{b,s}, (\alpha_s - b_\alpha)/g_\alpha, p_{m,s}) - g(p_{m,s}, n_s))\right)^2$$

The sets $\Theta_{\mathbf{TLF}}$ and $\Theta^0_{\mathbf{TLF}}$ follows as before.

Air Mass-Flow Sensor Loose Contact ALC

The model $\mathcal{M}_{\mathbf{ALC}}(c_1(t))$ corresponding to this fault mode is obtained by using the fault free model (5.1) together with identities (5.10) but replacing (5.10a) with $m_s(t) = m(t)c_1(t)$. The parameter $\theta_{as} = c_1(t)$ is a stochastic process taking values such that $c_1(t) \in \{0, 1\}$. This means that the parameter space $\mathcal{D}^{as}_{ALC}$ becomes $\mathcal{D}^{as}_{ALC} = \{0, 1\}^N - \{0\}^N$ and the measure $V_{\mathbf{ALC}}(c_1(t), \mathbf{x})$ is

$$V_{\mathbf{ALC}}(c_1(t), \mathbf{x}) = \frac{1}{N}\sum_{t=1}^{N}\left(m_s - c_1 f(p_{b,s}, \alpha_s, p_{m,s})\right)^2 + \\ + \frac{1}{N}\sum_{t=1}^{N}\left(f(p_{b,s}, \alpha_s, p_{m,s}) - g(p_{m,s}, n_s))\right)^2$$

The sets $\Theta_{\mathbf{ALC}}$ and $\Theta^0_{\mathbf{ALC}}$ follows as before.

5.6.4 Decision Structure

With the test quantities defined in the previous section, the decision structure becomes as shown in Figure 5.21. There are a few interesting things with this decision structure, which will be discussed in this section.

By using the definition of S^1_k, i.e. (3.2), the fact that the set $M_{\mathbf{MC}}$ doesn't contain **NF** means that

$$S^1_{\mathbf{MC}} = M^C_{\mathbf{MC}} = \{\mathbf{NF}, \mathbf{BL}, \mathbf{ML}, \mathbf{BB}, \mathbf{BAF}, \mathbf{MG}, \mathbf{TLF}, \mathbf{ALC}\}$$

Remembering the relationship between the decision structure and the sets S^0_k and S^1_k, discussed in Section 3.4.2, this means that the row for $\delta_{\mathbf{MC}}$ must contain non-zero entries in all places except in the column for **MC**. We see in Figure 5.21 that this is really the case.

We noted in the previous section that the test quantity $T_{\mathbf{MC}}(x)$ will be large for all faults in all fault modes except **MC**. This means that the corresponding power function will be large for all fault modes except **MC**. According to the discussion in Section 4.7.2 and especially formula (4.43), the set $S^0_{\mathbf{MC}}$ becomes

$$S^0_{\mathbf{MC}} = \{\mathbf{MC}\}$$

	NF	**BL**	**ML**	**BB**	**BAF**	**MG**	**MC**	**TLF**	**ALC**
$\delta_{\mathbf{NF}}$	0	X	X	X	X	X	X	X	X
$\delta_{\mathbf{BL}}$	0	0	X	X	X	X	X	X	X
$\delta_{\mathbf{ML}}$	0	X	0	X	X	X	X	X	X
$\delta_{\mathbf{BB}}$	0	X	X	0	X	X	X	X	X
$\delta_{\mathbf{BAF}}$	0	X	X	0	0	X	X	X	X
$\delta_{\mathbf{MG}}$	0	X	X	X	X	0	X	X	X
$\delta_{\mathbf{MC}}$	1	1	1	1	1	1	0	1	1
$\delta_{\mathbf{TLF}}$	0	X	X	X	X	X	X	0	X
$\delta_{\mathbf{ALC}}$	0	X	X	X	X	X	X	X	0

Figure 5.21: The decision structure for the hypothesis tests using the test quantities defined in Section 5.6.3.

Again using the relationship between the decision structure and the sets S_k^0 and S_k^1, discussed in Section 3.4.2, this means that all entries in the row for $\delta_{\mathbf{MC}}$, except for **MC**, must be 1:s.

Next, study the entry 0 in the row for $\delta_{\mathbf{BAF}}$ and the column for **BB**. This entry follows from the definition of $M_{\mathbf{BAF}}$ as follows:

$$S_{\mathbf{BAF}}^1 = M_{\mathbf{BAF}}^C = \{\mathbf{BL}, \mathbf{ML}, \mathbf{MG}, \mathbf{MC}, \mathbf{TLF}, \mathbf{ALC}\}$$

That is, since $S_{\mathbf{BAF}}^1$ does not contain **NF**, **BB**, or **BAF**, there must be 0:s in the corresponding locations in the decision structure, including the column for **BB**.

We conclude this section by pointing out the fact that for all hypothesis tests, except $\delta_{\mathbf{MC}}$, the sets S_k^0 are $S_k^0 = \Omega$. Also, all sets S_k^1 are defined by $S_k^1 = M_k^C$.

5.6.5 The Minimization of $V_k(\mathbf{x})$

The procedure to compute (5.24), i.e. to minimize the measures $V_k(\mathbf{x})$, has not been addressed so far. In many cases the minimization procedure required is quite straightforward. However, for some of the test quantities defined above, the computational load of doing the actual minimization in (5.24) can be quite heavy, if not some special care is taken.

For the test quantity $T_{\mathbf{BAF}}(\mathbf{x})$, we want to perform minimization of $V_{\mathbf{BAF}}(c_2(t), \mathbf{x})$ with respect to a signal. This can be solved by using the two-step approach from Section 4.2.1. Instead of minimizing $V_{\mathbf{BAF}}(c_2(t), \mathbf{x})$ we choose to minimize the following function:

$$\bar{V}_{\mathbf{BAF}}(c_2(t), \mathbf{x}) = \frac{1}{N} \sum_{t=1}^{N} \big(m_s - f(p_{b,s} - c_2, \alpha_s, p_{m,s})\big)^2$$

This function is conveniently minimized by choosing

$$c_2(t) = p_{b,s} - f^{-1}(m_s(t), \alpha_s(t), p_{m,s}(t))$$

where $f^{-1}(m_s(t), \alpha_s(t), p_{m,s}(t))$ is the inverse of $f(p_{b,s}, \alpha_s, p_{m,s})$, with respect to $p_{b,s}$, and gives an estimate of $p_{b,s}$.

Also for the test quantity $T_{\mathbf{ALC}}(\mathbf{x})$, the minimization needs to be done with respect to a signal. First we realize that to minimize $V_{\mathbf{ALC}}(k_b, \mathbf{x})$ is equivalent to minimizing

$$V_{\mathbf{ALC}}(c_1(t), \mathbf{x}) = \frac{1}{N} \sum_{t=1}^{N} \big(m_s - c_1 f(p_{b,s}, \alpha_s, p_{m,s})\big)^2$$

When the engine is running, the air-mass flow m is always positive and above 4 g/s. This can for example be seen in Figure 5.4. This means that the function $V_{\mathbf{ALC}}(c_1(t), \mathbf{x})$ can be conveniently minimized by choosing

$$c_1(t) = \begin{cases} 0 & m_s(t) < \epsilon \\ 1 & m_s(t) \geq \epsilon \end{cases}$$

where ϵ is some constant between 0 and 4.

5.6.6 Discussion

The fault modeling in Section 5.6.3 above illustrates the fact that it can be useful to model faults in a number of different ways. For some fault modes, i.e. **BL**, **ML**, **BB**, **MG**, the fault is modeled as a change in a continuous scalar parameter. The fault modes **MC** and **TLF** are examples in which the fault is modeled as a change in a discrete and multidimensional parameter respectively. In contrast to this, a fault belonging to the fault mode **BAF** is modeled as an additive arbitrary signal. Then we have **ALC**, in which the fault is a signal, or a parameter, that jumps between two distinct values.

All these examples clearly show the large variety of fault models that can be used in conjunction with structured hypothesis tests. In fact, while in many papers, *only* constant parameters or *only* additive arbitrary signals are considered, it is shown here that almost any kind of fault models can be handled and this within the same framework and same diagnosis system.

5.7 Experimental Validation

The diagnosis system described in the previous section was implemented in Matlab and tested extensively with the experimental setup described in Section 5.1. The leakage faults were implemented in hardware, which was also described in Section 5.1. All other faults were emulated in software by applying appropriate changes to the sensor signals. For each fault mode, a number of different fault sizes were tested.

Good functionality was obtained for all kinds of faults but to limit the discussion, only four cases have been selected and these are shown in Tables 5.4 to 5.7. These four cases are not selected because they are representative but rather because they illustrates some interesting features of the diagnosis system.

In all these cases, the data length was $N = 1000$ which corresponds to 100 s. No special effort was made to find optimal threshold values J_k; they were all chosen to be $J_k = 0.4$.

5.7.1 Fault Mode NF

In Table 5.4, the present fault mode of the process was **NF**. Each row show the result of one individual hypothesis test δ_k. The value of the test quantity $T_k(\mathbf{x})$ for each hypothesis δ_k is shown in the second column. The threshold J_k is shown in the third column (as said above, all were chosen to the same value). The fourth column shows the diagnosis decision S_k of each hypothesis test. We remember from formula (3.3) that $S_k = S_k^1 = M_k^C$ if $T_k(\mathbf{x}) > J_k$, i.e. H_k^0 is rejected, and $S_k = S_k^0$ otherwise.

For the case shown in the table, only the null hypothesis $H_{\mathbf{MC}}^0$ is rejected. This result is the one expected because the set $M_{\mathbf{MC}}$ do not contain the fault mode **NF** while all other sets M_k do contain **NF**. Applying the intersection of the decision logic, i.e. (2.7), implies that the diagnosis statement contains 8 possible fault modes that can explain the behavior of the process. One of the fault modes is **NF** which means that we should *not* generate an alarm. As was said in Section 2.6.1, we can also use the refined diagnosis statement $\bar{S}$, which would imply that the output from the diagnosis system becomes **NF** only.

k	$T_k(\mathbf{x})$	J_k	M_k^C
NF	0.2074	0.4	Ω
BL	0.2063	0.4	Ω
ML	0.2075	0.4	Ω
BB	0.2043	0.4	Ω
MG	0.2027	0.4	Ω
MC	3608	0.4	ALC BAF BB BL MG ML NF TLF
TLF	0.2061	0.4	Ω
ALC	0.2074	0.4	Ω
BAF	0.1491	0.4	Ω
Diagnosis Statement:			ALC BAF BB BL MG ML NF TLF
NO ALARM			

Table 5.4: The hypothesis tests and the diagnosis statement for fault mode **NF** present.

5.7.2 Fault Mode TLF

In Table 5.5, the present fault mode of the process was **TLF**. Now all individual null hypothesis are rejected except $H_{\mathbf{TLF}}^0$. The diagnosis statement is the single fault mode **TLF**. That is, the diagnosis system managed to isolate the present fault mode **TLF**. Because the diagnosis statement does not contain **NF**, an alarm is generated.

k	$T_k(\mathbf{x})$	J_k	M_k^C
NF	250.8	0.4	ALC BAF BB BL MC MG ML TLF
BL	170.7	0.4	ALC BAF BB MC MG ML TLF
ML	230.2	0.4	ALC BAF BB BL MC MG TLF
BB	247	0.4	ALC BAF BL MC MG ML TLF
MG	175.6	0.4	ALC BAF BB BL MC ML TLF
MC	3608	0.4	ALC BAF BB BL MG ML NF TLF
TLF	0.2025	0.4	Ω
ALC	250.8	0.4	BAF BB BL MC MG ML TLF
BAF	273.7	0.4	ALC BL MC MG ML TLF
Diagnosis Statement: TLF			
ALARM			

Table 5.5: The hypothesis tests and the diagnosis statement for fault mode **TLF** present.

5.7.3 Fault Mode ML

In Table 5.6, the present fault mode of the process was **ML**. The actual fault was fairly small, which is reflected in the result that it could not be isolated. The diagnosis statement contains the fault modes **MG**, **ML**, and **TLF**. This should be interpreted as that in addition to the present fault mode **ML**, the fault modes **MG** and **TLF** can also explain the behavior of the process. Because the fault statement does not contain **NF**, an alarm is generated.

k	$T_k(\mathbf{x})$	J_k	M_k^C
NF	0.4921	0.4	ALC BAF BB BL MC MG ML TLF
BL	0.4985	0.4	ALC BAF BB MC MG ML TLF
ML	0.1881	0.4	Ω
BB	0.423	0.4	ALC BAF BL MC MG ML TLF
MG	0.328	0.4	Ω
MC	3742	0.4	ALC BAF BB BL MG ML NF TLF
TLF	0.3623	0.4	Ω
ALC	0.4921	0.4	BAF BB BL MC MG ML TLF
BAF	0.4642	0.4	ALC BL MC MG ML TLF
Diagnosis Statement: MG ML TLF			
ALARM			

Table 5.6: The hypothesis tests and the diagnosis statement for fault mode **ML** present.

5.7.4 Fault Mode BB

In Table 5.7 the present fault mode of the process was **BB**. The actual fault was not very small but in spite of this, it is obvious from the diagnosis statement that the present fault mode **BB** can not be isolated. This was very much expected since we have the relation

$$\mathbf{NF} \preccurlyeq^* \mathbf{BB} \preccurlyeq \mathbf{BAF}$$

and according to Theorem 2.1 and also Section 2.6.1, it is then impossible to isolate **BB** from **BAF**. In other words, the fault mode **BAF**, which represent an *arbitrary* boost-pressure sensor fault, is so general that it can also explain data generated from the process when fault mode **BB** is present.

When both **BB** and **BAF** can explain the data, as in this case, it is much more likely that the data has been generated by a process with fault mode **BB**. In agreement with the discussion in Section 2.6.1, we can use the refined diagnosis statement $\bar{S}$ which would imply that the only output from the diagnosis system would be **BB**.

k	$T_k(\mathbf{x})$	J_k	M_k^C
NF	1.958	0.4	ALC BAF BB BL MC MG ML TLF
BL	1.96	0.4	ALC BAF BB MC MG ML TLF
ML	1.96	0.4	ALC BAF BB BL MC MG TLF
BB	0.2043	0.4	Ω
MG	0.6725	0.4	ALC BAF BB BL MC ML TLF
MC	3608	0.4	ALC BAF BB BL MG ML NF TLF
TLF	0.419	0.4	ALC BAF BB BL MC MG ML
ALC	1.958	0.4	BAF BB BL MC MG ML TLF
BAF	0.1491	0.4	Ω
Diagnosis Statement: BAF BB			
ALARM			

Table 5.7: The hypothesis tests and the diagnosis statement for fault mode **BB** present.

5.8 On-Line Implementation

For implementation in on-board diagnosis systems in a vehicle, on-line performance is crucial. The experiments presented in Section 5.7, were based on data $\mathbf{x}$ collected during a quite long time. This may imply that it also takes quite a long time before a fault is detected. One way to obtain a faster response to faults, is to decrease the length of the time window. The consequences of this are discussed in this section.

One thing that becomes important is the fact that the absolute accuracy of the model is dependent on how the system is excited, which is something

that changes over time. The solution to this is, according to Section 4.5, to use normalization, or more exactly an adaptive threshold. The adaptive threshold used here, is chosen in accordance with the ideas presented in the end of Section 4.5.2. Consider first the following relation

$$\min_{\theta\in\Theta} V(\theta,x) = \min_{\gamma}\min_{\theta\in\Theta_\gamma} V(\theta,x) = \min_{k}\min_{\theta\in\Theta_k^0} V(\theta,x) = \min_k T_k(\mathbf{x}(t))$$

Here, $V(\theta,x)$ is a model validity measure for the model $\mathcal{M}(\theta)$ obtained in analogy with all measures $V_k(\theta,x)$ presented in Section 5.6.3. Then the adaptive threshold is chosen as

$$J_{adp}(t) = \min_k T_k(\mathbf{x}(t)) + c \tag{5.27}$$

The first term serves as a measure of the overall accuracy of the model at time t and the second term is a tuning parameter, here chosen as $c = 0.05$. The expression (5.27) should be compared to (4.36) which was shown to be based on similar ideas as the likelihood ratio.

The adaptive threshold (5.27) was used in all hypothesis tests except for $\delta_{\mathbf{MC}}$ which was based on a model, whose accuracy does not change over time.

5.8.1 Experimental Results

To illustrate the performance in an on-line implementation, the following experiment was setup. The fault mode of the process was **MG** and the size of the fault parameter was $g_{p_m} = 1.2$. The whole data set (from the FTP-75 test-cycle) spans over a time of 21 minutes. A non-overlapping window of length $N = 100$ was used which corresponds to a time-length of 10 s. This means that the original data set was divided in totally 125 smaller data sets.

Fault Mode	Number of Instances
NF	0
BL	0
ML	57
BB	2
MG	120
MC	0
TLF	1
ALC	0
BAF	0
unknown fault	1

Table 5.8: The number of instances of different fault modes in the diagnosis statement during the on-line experiment.

For all 125 data sets, the diagnosis system managed to detect a fault. The number of times each fault mode was contained in the diagnosis statement is

shown in Table 5.8. It is seen that except for the fact that **ML** was in the diagnosis statement 57 number of times, the performance was very good.

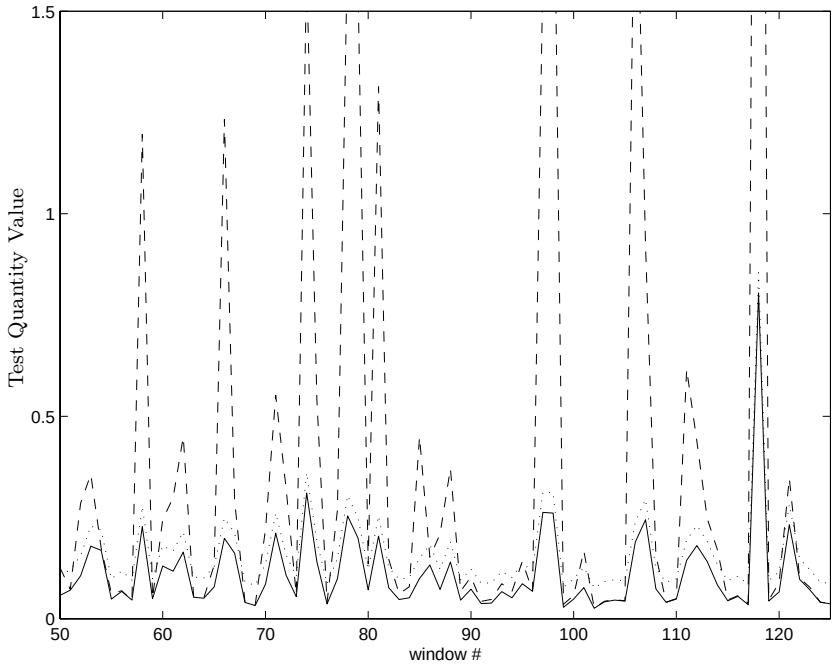

Figure 5.22: The test quantities $T_{\mathbf{ML}}(\mathbf{x}(t))$ (dashed) and $T_{\mathbf{MG}}(\mathbf{x}(t))$ (solid) together with the adaptive threshold $J_{adp}(t)$ (dotted).

To understand why **ML** is contained in the diagnosis statement so many times, Figure 5.22 has been included. The test quantities $T_{\mathbf{ML}}(\mathbf{x}(t))$ and $T_{\mathbf{MG}}(\mathbf{x}(t))$ are plotted together with the adaptive threshold $J_{adp}(t)$. Only data from time window #50 to #125 is shown. Ideally the test quantities $T_{\mathbf{MG}}(\mathbf{x}(t))$ should be below the threshold and $T_{\mathbf{ML}}(\mathbf{x}(t))$ should be above the threshold. This is the case most of the time but in some cases, both test quantities are below the threshold. These are the cases in which **ML** is contained in the diagnosis statement.

From the figure it is obvious that for some states of the process, the test quantity $T_{\mathbf{ML}}(\mathbf{x}(t))$ gets approximately the same value as $T_{\mathbf{MG}}(\mathbf{x}(t))$. This is due to a property of the air-intake system and not the diagnosis system. For example, during constant conditions, e.g. idle conditions, it is often possible to find a k_m in the model $\mathcal{M}_{\mathbf{ML}}(k_m)$ such that this model match the data also when the fault mode **MG** is present. We can because of this reason not expect that **ML** is, at all times, excluded from the diagnosis statement, no matter how the diagnosis system is designed.

5.9 Conclusions

This chapter has presented designs for two diagnosis systems for the air-intake system of an automotive engine. The whole design chain, including the modeling work, has been discussed. From this work, it is realized that a large part of the total work involved, when constructing a model-based diagnosis-system, may be to build the model including the fault models.

The first diagnosis system constructed, only focuses on diagnosis of leaks. The theoretical results from Section 4.8.2, regarding the optimality of the estimate principle, were validated in experiments on a real engine. Also investigated is how different types of fault models, with respect to the time-variant behavior of the leaks, affect the performance of the diagnosis system. It is concluded via experiments that, to choose a fault model with correct time-variant behavior, is important to maximize the diagnosis performance.

The method for leakage detection, often used in production cars, is the prediction principle, which in this case requires no leakage models. Therefore it is interesting to note that the method developed here, to use models of the leaks and then estimate the leakage area, performs better than the solution common in production cars.

The second diagnosis system constructed, is capable of diagnosing both leakage and a wide range of different types of sensor faults. Also in this case, the results were validated in experiments using data from a real engine. This application is an excellent example of the versatility of the method structured hypothesis tests. While in many papers, fault modeling using *only* constant parameters or *only* additive arbitrary signals are considered, it is shown here that almost any kind of fault models can be handled and this within the same framework and same diagnosis system. To the authors knowledge, a diagnosis system with this capacity, to diagnose such a large variety of different faults, can not be constructed using previous approaches to fault diagnosis.

This chapter has shown how a large part of the theory developed in earlier chapters, can be used in a real application. It has been shown that the theory has practical relevance for both design and analysis of diagnosis systems. The role of this automotive engine application has, during the work with this thesis, been more than only a validation of the techniques developed. In fact, this application inspired much of the development of the theoretical framework, since existing frameworks could not deal with many of the requirements.

Chapter 6

Evaluation and Automatic Design of Diagnosis Systems

When constructing a model-based diagnosis system, it is desirable that the solutions are the best possible or at least good. However, first we need to define what we mean by "good" and "best". This means that we need to develop performance measures and also a scheme for comparing different diagnosis systems. The topic of this chapter is to develop tools for this. These tools will also be used to develop a procedure for automatic design of diagnosis systems.

The performance measures and the comparison scheme developed are based on *decision theory* and is presented in Section 6.1 and 6.2. The performance measures become in most cases equal to for example probability of false alarm and probability of missed detection.

As said above, the second objective of this chapter is to find an automatic procedure for design of diagnosis systems. One motivation for this is to minimize the time-consuming engineering work, that is frequently needed for the design of diagnosis systems. Also it is desirable that we have a systematic, preferably automatic, procedure that gives diagnosis systems with as good performance as possible.

One area, in which it is highly desirable to have systematic and automatic procedures for diagnosis-system design, is the area of automotive engines. As was said in Chapter 5, environmentally based legislative regulations such as OBDII and EOBD specifies hard requirements on the performance of the diagnosis system. Automotive engines are rarely designed from scratch but often subject to small changes, e.g. for every new model year. Then usually also the diagnosis system needs to be changed. Since this may happen quite often and a car manufacturer typically has many different engine models in production, it is important for the car manufacturers that diagnosis systems can be reconstructed with minimal amount of work involved. Also, the diagnosis sys-

tems are often calibrated by personnel without extensive control background and it would therefore be beneficial to have an automatic procedure so that the diagnosis system could be calibrated with minimal human involvement.

For manufacturers of independent automotive diagnosis systems, to be used in independent repair-shops, the situation is even more critical. They need to design diagnosis systems for a large amount of different car brands and models. This makes it necessary to find procedures so that diagnosis systems can be constructed with very limited amount of engineering work.

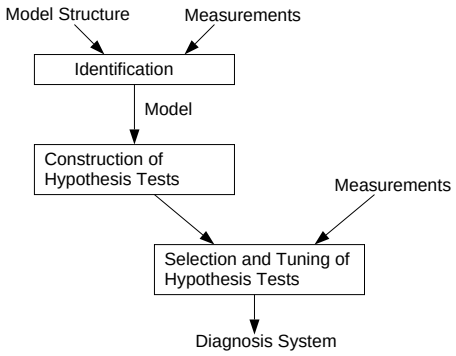

Figure 6.1: The process of constructing a diagnosis system.

The design process for construction of a diagnosis system is assumed to follow the flow-chart shown in Figure 6.1. The first part is to construct the model, in which at least it is possible to automatize parameter identification. The next part is the construction of hypothesis tests that we *possibly* want to include in the diagnosis system. Then the last step is to select hypothesis tests to be included, and also to tune the hypothesis tests, which should include at least tuning of thresholds.

This chapter deals with the step "selection and tuning of the hypothesis tests", for which an automatic procedure is presented in Section 6.3. The procedure is based on the performance measures and the comparison scheme developed in Section 6.1 and 6.2.

In Section 6.4, the construction of a diagnosis system for the air-intake system of a real automotive engine is approached using the automatic procedure developed. All steps in Figure 6.1 are discussed, i.e. modeling, construction of hypothesis tests, and the application of the automatic procedure for selection and tuning of hypothesis tests. The design is then experimentally evaluated in Section 6.4.6.

6.1 Evaluation of Diagnosis Systems

To evaluate a diagnosis system, we need some kind of measure of the performance. Here, this is done by first defining a loss function and then using the

risk function as a performance measure.

6.1.1 Defining a Loss Function

A loss function should reflect the "loss" for a given specific fault state of the plant and a specific decision made by the diagnosis system. The loss function is denoted $\mathcal{L}(\theta, S)$ and to define a loss function, we need to assign a value to each pair $\langle \theta, S \rangle$. For each θ, the set of all S can be divided into subsets which we will call *events*.

For the case $\theta \in \Theta_{NF}$, i.e. the fault free case, we define two events:

$$\begin{aligned} NA &= \{S;\ NF \in S\} && \text{No Alarm} \\ FA &= \{S;\ NF \notin S\} && \text{False Alarm} \end{aligned}$$

For the case $\theta \in \Theta_{F_i}$, i.e. fault mode F_i is present and $F_i \neq NF$, we define four events:

$$\begin{aligned} CI &= \{S;\ NF \notin S \wedge S = \{F_i\}\} && \text{Correct Isolation} \\ MD &= \{S;\ NF \in S\} && \text{Missed Detection} \\ ID &= \{S;\ NF \notin S \wedge F_i \notin S\} && \text{Incorrect Detection} \\ MI &= \{S;\ NF \notin S \wedge F_i \in S \wedge S \neq \{F_i\}\} && \text{Missed Isolation} \end{aligned}$$

The relation between these events are clarified in the tree-like structure in Figure 6.2. Each node defines an event as the intersection of the events corresponding to the particular node and all its parent nodes. A branch represents two disjunct events.

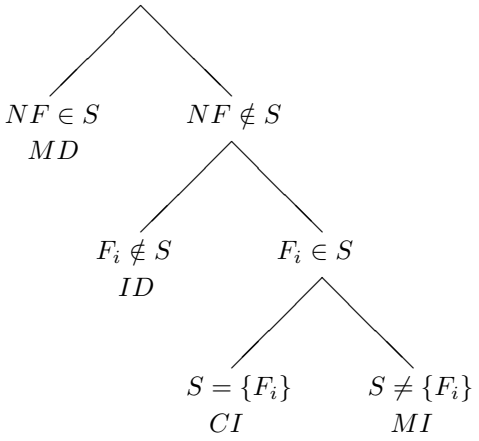

Figure 6.2: Relation between events.

It is obvious that NA and CI are the preferred events and should therefore correspond to $\mathcal{L}(\theta, S) = 0$. Also obvious is that the events FA, MD, ID, and MI should be "punished" by using a nonzero loss-function. With this in mind, the loss function $\mathcal{L}(\theta, S)$ can be defined as

$$\mathcal{L}(\theta, S) = \begin{cases} 0 & \text{if } NF \in S \quad \text{, i.e. } S \in NA \\ c_{FA}(\theta) & \text{if } NF \notin S \quad \text{, i.e. } S \in FA \end{cases} \qquad \theta \in \Theta_{NF}$$

and

$$\mathcal{L}(\theta, S) = \begin{cases} 0 & \text{if } S = \{F_i\} \quad \text{, i.e. } S \in CI \\ c_{MD}(\theta) & \text{if } NF \in S \quad \text{, i.e. } MD \\ c_{ID}(\theta) & \text{if } NF \notin S \wedge F_i \notin S \quad \text{, i.e. } ID \\ c_{MI}(\theta) & \text{if } NF \notin S \wedge F_i \in S \wedge S \neq \{F_i\} \quad \text{, i.e. } MI \end{cases} \qquad \theta \in \Theta_{F_i}$$

In general, the event MI is not as serious as MD and ID. This can be reflected in that $c_{MD}(\theta)$, $c_{ID}(\theta)$, and $c_{MI}(\theta)$ are selected such that $c_{MI}(\theta) < c_{ID}(\theta)$, and $c_{MI}(\theta) < c_{MD}(\theta)$.

We will classify faults into *insignificant faults* Θ_{insign} and *significant faults* Θ_{sign}. Insignificant faults are those faults that are "small" and we are not very interested in detecting. Significant faults are those faults that are "large" and that we really want to detect. It is reasonable to assume that if there is a **1**, in the column for F_i in the decision structure, then all faults belonging to fault mode F_i, are significant.

For insignificant faults, the events MD and MI are not very serious. This should be reflected in that $c_{MD}(\theta)$ and $c_{MI}(\theta)$ are chosen such that for $\theta \in \Theta_{insign}$, $c_{MD}(\theta)$ and $c_{MI}(\theta)$ are small or even zero. On the other hand, for significant faults, i.e. for $\theta \in \Theta_{sign}$, $c_{MD}(\theta)$ and $c_{MI}(\theta)$ should be large.

This reasoning about the choice of $c_{MD}(\theta)$, $c_{ID}(\theta)$, and $c_{MI}(\theta)$ can be summarized in a table:

	CI	MD	ID	MI
significant faults	0	$c_{MD}(\theta)$	$c_{ID}(\theta)$	$c_{MI}(\theta)$
insignificant faults	0	≈ 0	$c_{ID}(\theta)$	≈ 0

Examples of choices of the functions $c_{MD}(\theta)$, $c_{ID}(\theta)$, and $c_{MI}(\theta)$ are given in Figure 6.3, 6.4, and 6.5. For MD and MI, two examples are given, represented by the solid and dashed line. The exact choice of $c_{MD}(\theta)$, $c_{ID}(\theta)$, and $c_{MI}(\theta)$ depends on the specific application.

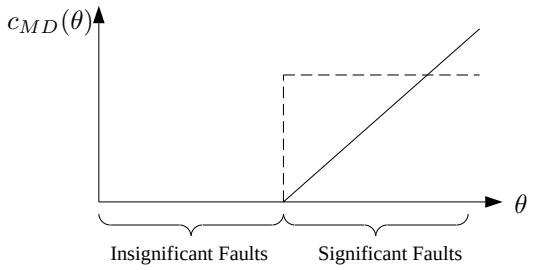

Figure 6.3: The function $c_{MD}(\theta)$.

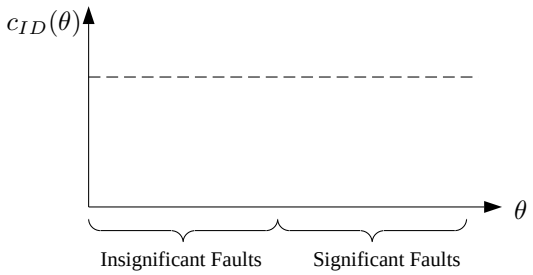

Figure 6.4: The function $c_{ID}(\theta)$.

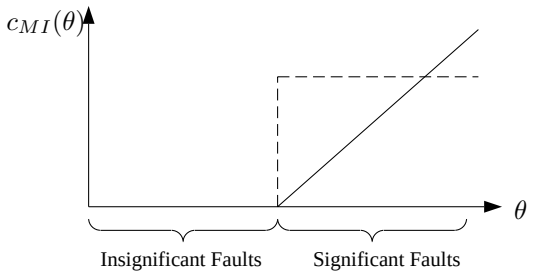

Figure 6.5: The function $c_{MI}(\theta)$.

6.1.2 Calculating the Risk Function

Recall the definition of risk function from Section 4.6. By using the loss function defined in the previous section, the risk function becomes

$$R(\theta, \delta(x)) = \begin{cases} c_{FA}(\theta)P(FA) & \text{if } \theta \in \Theta_{NF} \\ c_{MD}(\theta)P(MD) + c_{ID}(\theta)P(ID) + c_{MI}(\theta)P(MI) & \text{if } \theta \in \Theta_{F_i} \end{cases} \tag{6.1}$$

Note that the probabilities for the events MD, ID, and MI have been lumped together. It might be possible that it is interesting to study these probabilities individually. In the framework of loss and risk functions, this would correspond to a vector-valued loss and risk.

To calculate the risk function in the general case, it is obvious that we need to know the probabilities $P(FA)$, $P(MD)$, $P(ID)$, and $P(MI)$. The problem is that the probability density functions are multidimensional; the dimension equals the number of tests. In addition, the distributions can be complicated functions. This makes it hard to derive the probabilities analytically. Simulations is an alternative but since the probabilities of interest are related to the tails of the density functions, and they are multidimensional, an unrealistically large amount of data would be needed. In spite of the above stated problems, it is possible to calculate bounds of the risk function. The rest of Section 6.1 will be devoted to this issue, but as an alternative to (6.1), we will consider a somewhat simpler risk function.

A simpler risk function is obtained by defining the new event $MIM = MD \cup ID \cup MI$. Further we assume that for significant faults, $c_{MD}(\theta) = c_{ID}(\theta) = c_{MI}(\theta) \triangleq c_{MIM}(\theta)$ and for insignificant faults, $c_{MD}(\theta) = c_{MI}(\theta) = 0$. The dashed lines in Figure 6.3, 6.4, and 6.5 correspond to this assumption. With this assumption, the risk function becomes

$$R(\theta, \delta(x)) = \begin{cases} c_{FA}(\theta)P(FA) & \text{if } \theta \in \Theta_{NF} \\ c_{MIM}(\theta)P(MIM) & \text{if } \theta \in \Theta_{F_i} \text{ and } \theta \in \Theta_{sign} \\ c_{ID}(\theta)P(ID) & \text{if } \theta \in \Theta_{F_i} \text{ and } \theta \in \Theta_{insign} \end{cases} \tag{6.2}$$

The reason why this risk function is considered to be simpler than (6.1), is that the sums of probabilities, present in (6.1), have all been eliminated.

6.1.3 Expressing Events with Propositional Logic

This section explores how general events, e.g. FA, MD, and MIM, can be expressed by propositional logic formulas where the atoms are events for the individual hypothesis tests. For example, consider the event FA. With the set representation of the decisions S_k, this event can be written

$$FA = \{S;\ NF \notin S\} = \{S;\ NF \notin \bigcap_k S_k\} = \{S;\ \bigvee_k NF \notin S_k\}$$

To describe events, also a shorter form will be used, e.g. FA is written as

$$FA = \bigvee_k NF \notin S_k$$

We can further develop this expression by using the realistic assumption that $NF \in M_k$ for all k. This means that $NF \notin S_k$ is equivalent to $S_k = S_k^1$, and the event FA can be written

$$FA = \bigvee_k \{S_k = S_k^1\} \tag{6.3}$$

In general, the probability for an arbitrary event A can be expressed as

$$P(A) = P(\varphi)$$

where φ is a propositional logic expression in the proposition symbols $\{S_k = S_k^1\}$ and $\{S_k = S_k^0\}$.

In the next section we will assume that the events FA, MD, ID, MI, and MIM are expressed by a propositional logic expression in *minimal disjunctive normal form*. Before we give the definition of the minimal disjunctive normal form, consider the definition of *disjunctive normal form* (DNF):

Definition 6.1 (Disjunctive Normal Form) *If*

$$\varphi = \bigvee_i \bigwedge_j \varphi_{i,j}$$

where $\varphi_{i,j}$ is atomic or the negation of an atom, then φ is a disjunctive normal form *(DNF).*

The *minimal* disjunctive normal form is then defined as:

Definition 6.2 (Minimal Disjunctive Normal Form) *A DNF φ is minimal if there exist no other DNF φ' where φ' have a smaller total number of connectives $\wedge$ and $\vee$.*

To transform a DNF to a minimal DNF, the algorithm proposed by Quine-McCluskey (McCluskey, 1966) can be used. The principles of expressing events with propositional logic and minimal DNF's, is illustrated in the following example:

Example 6.1

Consider a diagnosis system with decision structure

	NF	F_1	F_2	F_3
δ_1	0	X	0	X
δ_2	0	1	X	0
δ_3	0	0	X	0

Expression (6.3) implies that the probability $P(FA)$ can be written

$$P(FA) = P(NF \notin S) = P(S_1 = S_1^1 \vee S_2 = S_2^1 \vee S_3 = S_3^1) \tag{6.4}$$

in which the event is described by a minimal DNF.

The probability of the event ID of F_3 can be written

$$P(ID) = P(NF \notin S \wedge F_3 \notin S) = P\big(\big(\bigvee_k NF \notin S_k\big) \wedge \big(\bigvee_k F_3 \notin S_k\big)\big) =$$
$$= P\big(\big(S_1 = S_1^1 \vee S_2 = S_2^1 \vee S_3 = S_3^1\big) \wedge \big(S_2 = S_2^1 \vee S_3 = S_3^1\big)\big) \tag{6.5}$$

This formula is not even a DNF but can be transformed to

$$P(ID) = P(S_2 = S_2^1 \vee S_3 = S_3^1) \tag{6.6}$$

which is a minimal DNF.

Thus we have shown two examples of how events can be expressed by propositional logic formulas and in particular, minimal DNF's. ■

6.1.4 Calculating Probability Bounds

This section gives two lemmas and two presumptions . Together, these can be used to calculate bounds of the probabilities $P(FA)$, $P(ID)$, and $P(MIM)$. However, we first need to introduce the terms *desired response* and *completely undesirable* event:

Definition 6.3 (Desired Response) *Let the* desired response *of test k to fault mode F_i be*

$$S_k^{des}(F_i) = \begin{cases} S_k^1 & \text{if } F_i \in S_k^1 \\ S_k^0 & \text{otherwise} \end{cases}$$

Definition 6.4 (Completely Undesirable Event) *An event A is* completely undesirable *if for any minimal DNF φ, describing A,*

$$\varphi = \bigvee_i \bigwedge_j \varphi_{i,j}$$

it holds that

$$\varphi_{i,j} = \{S_{k_j} \neq S_{k_j}^{des}\}$$

For example, both events described by (6.4) and (6.6) are completely undesirable.

The following two presumptions give bounds for general events that are completely undesirable. In all realistic cases, these presumptions are probably true or at least approximately true. Further, in Section 6.4.6, the validity of these bounds is confirmed using experimental data.

Presumption 6.1 *For a completely undesirable event A described by a minimal DNF φ,*

$$\varphi = \bigvee_i \bigwedge_j \varphi_{i,j}$$

it holds that

$$P(A) = P(\varphi) = P\big(\bigvee_i \bigwedge_j \varphi_{i,j}\big) \leq 1 - \prod_i \max_j \big(1 - P(\varphi_{i,j})\big) \tag{6.7}$$

Presumption 6.2 *For a completely undesirable event A described by a minimal DNF φ,*

$$\varphi = \bigvee_i \bigwedge_j \varphi_{i,j}$$

it holds that

$$P(A) = P(\varphi) = P\big(\bigvee_i \bigwedge_j \varphi_{i,j}\big) \geq \max_i \prod_j P(\varphi_{i,j}) \tag{6.8}$$

Motivation of Presumption 6.1 and 6.2

To motivate Presumption 6.1 and 6.2, we need the following lemma:

Lemma 6.1 *If a set of n events A_i can be ordered such that*

$$P(A_2 \mid A_1) \geq P(A_2) \tag{6.9}$$
$$P(A_3 \mid A_1 \cap A_2) \geq P(A_3) \tag{6.10}$$
$$\vdots \tag{6.11}$$
$$P(A_n \mid A_1 \cap A_2 \cap \cdots \cap A_{n-1}) \geq P(A_n) \tag{6.12}$$

then

$$P(\bigcap_{i=1}^{n} A_i) \geq \prod_{i=1}^{n} P(A_i) \tag{6.13}$$

Proof: By using the definition of conditional probability, the relation (6.9) can be rewritten as

$$\frac{P(A_2 \cap A_1)}{P(A_1)} \geq P(A_2)$$

which implies

$$P(A_2 \cap A_1) \geq P(A_1)P(A_2)$$

By continuing in this fashion with all relations (6.9) to (6.12), we arrive at the relation (6.13). ■

We first motivate the upper bound given by Presumption 6.1. First define the event A_i:

$$A_i = \{S;\ \bigwedge_j \varphi_{i,j}\}$$

and note that $P(A) = P(\bigcup_i A_i)$. Now assume that A_1^C has occurred. Since the event A is completely undesirable, the event A_1^C can be written

$$A_1^C = \{S;\ \bigvee \neg\varphi_{1,j}\} = \{S;\ \bigvee_j \{S_{k_j} = S_{k_j}^{des}\}\}$$

This implies that $S_k = S_k^{des}$ for some k, i.e. some tests responds according to the desired response. In the same way, A_2^C can also be interpreted as there is some tests (not necessarily the same as for A_1^C) responding according to the desired response. Now study the relation

$$P(A_2^C \mid A_1^C) \geq P(A_2^C)$$

This relation says that the event that some tests responds according to the desired response, given that some other tests responds according to the desired response, is *at least* as probable as the case when there are no *a priori* information given. It is reasonable to assume that this relation holds. It is also reasonable to assume that we can obtain a a whole set of relations that satisfies the requirements of Lemma 6.1. Then by using Lemma 6.1, we can conclude that

$$P(\bigcap_i A_i^C) \geq \prod_i P(A_i^C)$$

which is equivalent to

$$P(A) = P(\bigcup_i A_i) = 1 - P(\bigcap_i A_i^C) \leq 1 - \prod_i P(A_i^C) = 1 - \prod_i P(\bigvee_j \neg\varphi_{i,j})$$

From the fact that

$$P(\bigvee_j \neg\varphi_{i,j}) = P(\bigcup_j \{S;\ \neg\varphi_{i,j}\}) \geq \max_j P(\{S;\ \neg\varphi_{i,j}) = \max_j \big(1 - P(\varphi_{i,j})\big)$$

we get the upper bound given in Presumption 6.1.

To motivate the lower bound given in Presumption 6.2, we first note that

$$P(A) = P(\bigcup_i A_i) \geq \max_i P(A_i) \tag{6.14}$$

Now assume that the event described by $\varphi_{i,1}$ has occurred. This means that $S_k \neq S_k^{des}$ for some k, i.e. some test do not respond according to the desired

response. In the same way, the event described by $\varphi_{i,2}$ means that another test is not responding according to the desired response. By using the same reasoning as above, we can conclude that a reasonable assumption is

$$P(\varphi_{i,2} \mid \varphi_{i,1}) \geq P(\varphi_{i,2})$$

and further, again using Lemma 6.1, that

$$P(A_i) = P(\bigwedge_i \varphi_{i,j}) \geq \prod_i P(\varphi_{i,j})$$

This relation together with (6.14) gives the lower bound in Presumption 6.2.

Undesirability of ID, MIM, and FA

We will now prove that the events ID, MIM, and are completely undesirable. The reason why we want to prove this is that, if this is the case, Presumption 6.1 and 6.2 can be used to give probability bounds for ID, MIM, and FA.

We start with ID and MIM in the following lemma, which shows that if the decision structure contains no 1:s, then the events ID and MIM are completely undesirable.

Lemma 6.2 *If the decision structure contains no 1:s, and the column for NF only 0:s, then the events ID and MIM of F are completely undesirable.*

To prove Lemma 6.2, we first need the following lemma:

Lemma 6.3 *If φ is a minimal DNF, a is an atom, and $a \vee \varphi = \varphi \not\equiv \mathcal{T}$, then for one of the $\varphi_i = \bigwedge_j \varphi_{i,j}$, it must hold that $\varphi_i \equiv a$.*

Proof: Assume that the lemma does not hold. This means that φ can be written as

$$\varphi \equiv \beta_1 \vee \cdots \vee \beta_n \vee a \wedge \gamma_1 \vee \cdots \vee a \wedge \gamma_m$$

where β_i and γ_i are conjunctions not containing a. Because of the minimality of the DNF, it can be shown that it is possible to make the valuations $a = \mathcal{T}$ and $\beta_i = \gamma_i = \mathcal{F}$. This implies that $\varphi = \mathcal{F}$ and $a \vee \varphi = \mathcal{T}$. Thus a contradiction, which means that the Lemma must hold. ■

Now return to the proof of Lemma 6.2:

Proof: Assume that the event, ID or MIM, is described by a minimal DNF φ

$$\varphi = \bigvee_i \bigwedge_j \varphi_{i,j}$$

The proof consists of two parts corresponding to ID and MIM respectively.

ID of Fault Mode F

Consider first ID of fault mode F. Study the i':th conjunction of φ:

$$\varphi_{i'} = \bigwedge_j \varphi_{i',j}$$

The corresponding event must belong to ID. Since the decision structure contains no 1:s in the column for F, it must hold that $\forall k.F \in S_k^0$. We also know that ID means $F \notin S$. These two facts imply that the conjunction $\varphi_{i'}$ must contain a $\varphi_{i',j'}$ such that $\varphi_{i',j'} \equiv \{S_{k_{j'}} = S^1_{k_{j'}}\}$ and that $F \notin S^1_{k_{j'}}$. By using the assumption that the decision structure contains only 0:s in the column for NF, this means that $\varphi_{i',j'}$ alone must imply both $NF \notin S$ and $F \notin S$ and thus, the corresponding event belongs to ID. Therefore, $\varphi_{i',j'} \vee \varphi = \varphi$ and by applying Lemma 6.3, we can conclude that either

$$\varphi \equiv \cdots \vee \varphi_{i',j'} \vee \cdots \vee \varphi_{i'} \vee \cdots \tag{6.15}$$

or that

$$\varphi \equiv \cdots \vee \varphi_{i',j'} \vee \cdots \tag{6.16}$$

where the conjunction $\varphi_{i'}$ is not present in (6.16). Assume the φ corresponds to the first of these two expressions. Then it holds that

$$\varphi_{i',j'} \vee \varphi_{i'} = \varphi_{i',j'}$$

and thus, φ cannot be a minimal DNF. Therefore, φ must correspond to (6.16) and we can write

$$\varphi_{i'} \equiv \varphi_{i',1} \equiv \{S_{k_1} = S^1_{k_1}\}$$

Now since $F \notin S^1_{k_1}$, we know from Definition 6.3 that $S^{des}_{k_1} = S^0_{k_1}$. This further implies that

$$\varphi_{i'} \equiv \{S_{k_1} \neq S^{des}_{k_1}\}$$

which means that the event ID is completely undesirable. This ends the part of the proof for ID.

MIM of Fault Mode F

Now consider MIM and again an arbitrary chosen $\varphi_{i'}$. From the definition of MIM we know that each φ_i implies $F \notin S$ or that $\{F, F_c\} \subseteq S$ for some $F_c \neq F$.

Consider first a $\varphi_{i'}$ such that $F \notin S$. Then the reasoning for ID can be applied to $\varphi_{i'}$ and we can conclude that it must be the case that $\varphi_{i'} \equiv \{S_k = S^1_k\}$ and $\{S^{des}_k = S^0_k\}$.

Now consider a $\varphi_{i'}$ such that $\{F, F_c\} \subseteq S$ and assume that $\varphi_{i',1} \equiv \{S_{k_1} = S^1_{k_1}\}$. Then $\{F, F_c\} \subseteq S_{k_1}$ and $F_c \neq NF$. Since the decision structure contains no 1:s, which implies that $\forall k.F \in S^0_k$, we also know that

$$\bar{\varphi}_{i'} = \neg\varphi_{i',1} \wedge \varphi_{i',2} \wedge \dots$$

must imply $\{F, F_c\} \subseteq S_{k_1}$. This means that $\bar{\varphi}_{i'}$ belongs to MIM, which further implies that

$$\bar{\varphi}_{i'} \vee \varphi = \varphi \tag{6.17}$$

Also we have that

$$\bar{\varphi}_{i'} \vee \varphi_{i'} \equiv (\neg\varphi_{i',1} \wedge \varphi_{i',2} \wedge \cdots) \vee (\varphi_{i',1} \wedge \varphi_{i',2} \wedge \cdots) = \varphi_{i',2} \wedge \cdots = \varphi'_{i'} \tag{6.18}$$

Expression (6.17) and (6.18) together implies that

$$\varphi \equiv \cdots \vee \varphi_{i'} \vee = \bar{\varphi}_{i'} \vee \cdots \vee \varphi_{i'} \vee = \cdots \vee \varphi'_{i'} \vee$$

where $\varphi'_{i'}$ have fewer terms than $\varphi_{i'}$ and thus φ cannot be a minimal DNF. This contradiction gives that $\varphi_{i',1}$ and consequently all $\varphi_{i',j}$ must satisfy

$$\varphi_{i',j} \equiv \{S_{k_j} = S^0_{k_j}\}$$

Suppose now that $\varphi_{i',1} \equiv \{S_{k_1} = S^0_{k_1}\} = \{S_{k_1} = S^{des}_{k_1}\}$, i.e. $S^{des}_{k_1} = S^0_{k_1}$. This implies that $F \notin S^1_{k_1}$, and therefore $\neg\varphi_{i',1} = \{S_{k_1} = S^1_{k_1}\}$ alone must belong to ID and also MIM. This further implies

$$\neg\varphi_{i',1} \vee \varphi = \varphi$$

Using Lemma 6.3 implies that one of the conjunctions in φ is $\neg\varphi_{i',1}$. It holds that

$$\begin{aligned}
\varphi &\equiv \cdots \vee \neg\varphi_{i',1} \vee \cdots \vee \varphi_{i'} \vee \cdots \equiv \\
&\equiv \cdots \vee \neg\varphi_{i',1} \vee \cdots \vee (\varphi_{i',1} \wedge \varphi_{i',2} \wedge \cdots) \vee \cdots = \cdots \vee (\varphi_{i',2} \wedge \cdots) \vee \cdots \equiv \\
&\equiv \cdots \vee \varphi''_{i'} \vee \cdots
\end{aligned}$$

where $\varphi''_{i'}$ have fewer terms than $\varphi_{i'}$ and thus φ cannot be a minimal DNF. This contradiction gives that $\varphi_{i',1}$ and consequently all $\varphi_{i',j}$ must satisfy

$$\varphi_{i',j} \equiv \{S_{k_j} = S^0_{k_j}\} = \{S_{k_j} \neq S^{des}_{k_j}\}$$

In conclusion, for each conjunction φ_i of φ, it holds that either

$$\varphi_i \equiv \{S_k = S^1_k\} = \{S_k \neq S^{des}_k\}$$

or that

$$\varphi_i \equiv \bigwedge_j \{S_{k_j} = S^0_{k_j}\} = \{S_{k_j} \neq S^{des}_{k_j}\}$$

This means that MIM is completely undesirable. ∎

For the event ID, the proof of Lemma 6.2 is valid also for the less restrictive case that the decision structure contains no 1:s in the columns for F but still only 0:s in the column for NF.

The following lemma shows that the event FA is completely undesirable, which implies that Presumption 6.1 and 6.2 can be used.

Lemma 6.4 *If the decision structure contains no 1:s, and the column for NF only 0:s, then the event FA is completely undesirable.*

Proof: Define a new fault mode F_{new} that has a column in the decision structure which is identical with the column for NF. Then the event FA is equivalent to ID of fault mode F_{new}. Further, Lemma 6.2 implies that the event ID of F_{new} is completely undesirable and therefore also the event FA is completely undesirable. ∎

6.1.5 Some Bounds for $P(FA)$, $P(ID)$, and $P(MIM)$

The purpose of this section is to exemplify the use of Presumption 6.1 and 6.2, and at the same time derive some relations useful for selecting the significance level α_k of the individual tests. With the notation used here, the significance level becomes

$$\alpha_k = \sup_{\theta \in \Theta_k^0} P(\text{reject } H_k^0 \mid H_k^0 \text{ true}) = \sup_{\theta \in \Theta_k^0} P(S_k = S_k^1 \mid \theta)$$

where $\Theta_k^0 = \bigcup_{\gamma \in M_k} \Theta_\gamma$. We will assume that $\Theta_{NF} = \{\theta_0\}$ and that

$$\sup_{\theta \in \Theta_k^0} P(S_k = S_k^1 \mid \theta) = P(S_k = S_k^1 \mid \theta_0)$$

Thus, it holds that

$$\alpha_k = P(S_k = S_k^1 \mid \theta \in \Theta_{NF}) \tag{6.19}$$

In the following subsections, we will derive bounds for the probabilities $P(FA)$, $P(ID)$, and $P(MIM)$. In all cases we will assume that the decision structure contains no 1:s and the column for NF contains only 0:s.

Bounds for FA

Consider the event FA which can be described by the minimal DNF (6.3). In most expressions for probabilities below, we will assume that a specific θ is given, but to get a simple notation, this is not written out, i.e. $P(\dots|\theta)$ is written $P(\dots)$. Lemma 6.4 makes it possible to apply Presumption 6.1 and 6.2 to (6.3).

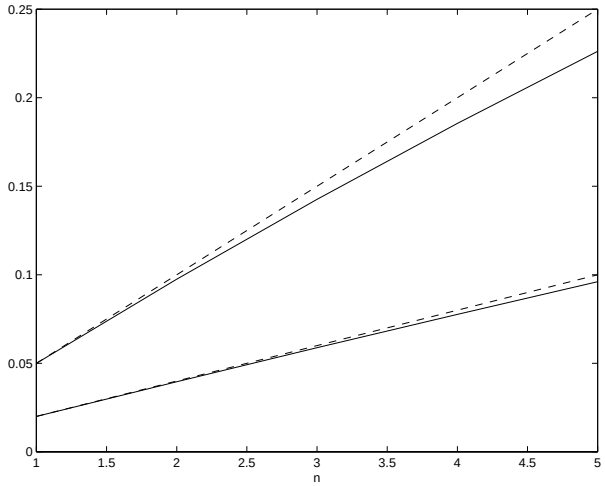

Figure 6.6: The functions $1-(1-\alpha)^n$ (solid) and $n\alpha$ (dashed) for $\alpha = 0.05$ and 0.02.

For the event FA, we know that $\theta \in \Theta_{NF}$ and by noting that $\alpha_k = P(S_k = S_k^1)$, the bounds become

$$\max_k \alpha_k = \max_k P(S_k = S_k^1) \leq P(FA) \leq$$
$$\leq 1 - \prod_k \big(1 - P(S_k = S_k^1)\big) = 1 - \prod_k \big(1 - \alpha_k\big)$$

Now assume that the significance of all tests are equal to α, i.e. $\forall k.\alpha_k = \alpha$. This implies that the bounds become

$$\alpha \leq P(FA) \leq 1 - (1-\alpha)^n$$

where n is the number of tests. In Figure 6.6, the functions $1-(1-\alpha)^n$ (solid) and $n\alpha$ (dashed) have been plotted as a function of n for $\alpha = 0.05$ and $\alpha = 0.02$. It is obvious that $1-(1-\alpha)^n < n\alpha$ and also that $1-(1-\alpha)^n \approx n\alpha$. This means that the simple expression $n\alpha$ is an upper level of $P(FA)$ and also an approximation of the upper level $1-(1-\alpha)^n$.

Bounds for ID

Now consider the event ID of fault mode F. The probability $P(ID)$ can be written

$$P(ID) = P(\big(\bigvee_k NF \notin S_k\big) \wedge \big(\bigvee_k F \notin S_k\big)) =$$
$$= P(\big(\bigvee_k \{S_k = S_k^1\}\big) \wedge \big(\bigvee_{k\in\mu} \{S_k = S_k^1\}\big)) = P(\bigvee_{k\in\mu} \{S_k = S_k^1\}) \quad (6.20)$$

where

$$\mu = \{k;\ F \notin S_k^1\}$$

That is, μ is the set of indices for tests δ_k with a 0 in the decision structure for F.

The rightmost expression of (6.20) is a minimal DNF which together with Lemma 6.2 implies that it is possible to use Presumption 6.1 and 6.2. With $\beta_k(\theta)$ denoting the power function of the k:th test, i.e. $\beta_k(\theta) = P(S_k = S_k^1 \mid \theta)$, the bounds become

$$\max_{k\in\mu} \beta_k(\theta) = \max_{k\in\mu} P(S_k = S_k^1) \leq P(ID) \leq$$
$$\leq 1 - \prod_{k\in\mu} \big(1 - P(S_k = S_k^1)\big) = 1 - \prod_{k\in\mu} \big(1 - \beta_k(\theta)\big)$$

Now it is reasonable to assume that for $\theta \in \Theta_F$ and $k \in \mu$, it holds that $\beta_k(\theta) = \beta_k(\theta')$ where $\theta' \in \Theta_{NF}$. This together with (6.19) means that $\beta_k(\theta) = \alpha_k$ and the bounds become

$$\max_{k\in\mu} \alpha_k \leq P(ID) \leq 1 - \prod_{k\in\mu} \big(1 - \alpha_k\big)$$

By assuming that $\forall k.\alpha_k = \alpha$, and again using the relationship $1-(1-\alpha)^n < n\alpha$, the bounds can be further simplified to

$$\alpha \leq P(ID) \leq 1 - (1-\alpha)^{n_\mu} < n_\mu \alpha$$

where n_μ denotes the number of elements in μ.

Bounds for MIM

Next consider the event MIM $(= MD \cup ID \cup MI)$ of fault mode F. By studying the proof of Lemma 6.2 it can be realized that the probability $P(MIM)$ can be expressed with a minimal DNF as

$$P(MIM) = P(\bigvee_{k\in\mu} \{S_k = S_k^1\} \bigvee_{i=1\ldots n_X} A_i) \tag{6.21}$$

where

$$\mu = \{k;\ F \notin S_k^1\}$$

and

$$A_i = \bigwedge_{k\in\psi_i} \{S_k = S_k^0\}$$

for some, typically small, number n_X and some sets ψ_i.

Since the formula in (6.21) is a minimal DNF, Lemma 6.2 implies that it is possible to use Presumption 6.1 and 6.2. The lower bound becomes

$$P(MIM) \geq \max\{\max_{k\in\mu} P(S_k = S_k^1), \max_{i\ldots n_X} \prod_{k\in\psi_i} P(S_k = S_k^0)\} =$$
$$= \max\{\max_{k\in\mu} \alpha_k, \max_{i\ldots n_X} \prod_{k\in\psi_i} (1 - \beta_k(\theta))\} \tag{6.22}$$

where we again have used the assumption $\beta_k(\theta) = \alpha_k$ for $k \in \mu$. The upper bound becomes

$$P(MIM) \leq 1 - \prod_{k\in\mu} \big(1 - P(S_k = S_k^1)\big) \prod_{i=1}^{n_X} \max_{k\in\psi_i}\{1 - P(S_k = S_k^0)\} =$$
$$= 1 - \prod_{k\in\mu} \big(1 - \alpha_k\big) \prod_{i=1}^{n_X} \max_{k\in\psi_i} \beta_k(\theta) \tag{6.23}$$

Now if, for each test δ_k, it holds that $\beta_k(\theta) \geq 1 - \alpha_k$, then an upper bound for (6.22) is

$$\max\{\max_{k\in\mu} \alpha_k, \max_{i\ldots n_X} \prod_{k\in\psi_i} \alpha_k\}$$

By again assuming $\forall k.\alpha_k = \alpha$, this expression becomes equal to

$$\max\{\alpha, \max_{i\ldots n_X} \alpha^{n_{\psi_i}}\} = \alpha \tag{6.24}$$

where n_{ψ_i} denotes the number of elements in ψ_i. Similarly, an upper bound for (6.23) becomes

$$1 - \prod_{k\in\mu} \big(1 - \alpha_k\big) \prod_{i=1}^{n_X} \max_{k\in\psi_i}(1 - \alpha_k)$$

and with the assumption $\forall k.\alpha_k = \alpha$, this expression becomes equal to

$$1 - (1 - \alpha)^{n_\mu}(1 - \alpha)^{n_X} = 1 - (1 - \alpha)^{n_\mu + n_X} \leq (n_\mu + n_X)\alpha \tag{6.25}$$

where n_μ denotes the number of elements in μ.

In conclusion, we have derived the bound (6.25), which is a upper bound to the upper bound (6.23) of $P(MIM)$, and (6.24), which is an upper bound to the lower bound (6.22) of $P(MIM)$. The usage of the bound (6.24) is that if we know that it is small, then the lower bound of $P(MIM)$ will be small.

Concluding Remarks

The bounds for $P(FA)$, $P(ID)$, and $P(MIM)$ derived in this section are summarized in Table 6.1. Although derived using some assumptions, the relations

Probability	Lower Bound	Upper Bound	Simple Upper Bound
$P(FA)$	α	$1-(1-\alpha)^n$	$n\alpha$
$P(ID)$	α	$1-(1-\alpha)^{n_\mu}$	$n_\mu\alpha$
$P(MIM)$	(6.22) $\leq \alpha$	$1-(1-\alpha)^{n_\mu+n_X}$	$(n_\mu+n_X)\alpha$

Table 6.1: Bounds for $P(FA)$, $P(ID)$, and $P(MIM)$ when $\forall k.\alpha_k = \alpha$. The bounds for MIM are obtained when $\beta_k(\theta) \geq 1-\alpha_k$.

in Table 6.1 and also the other relations derived in this section, are useful to be aware of when choosing the significant levels of the individual tests.

From above it is clear that the probabilities $P(FA)$, $P(ID)$, etc., can be estimated if we have the probabilities $P(S_k = S_k^1) = 1 - P(S_k = S_k^0)$. In principle, we are interested in the probability $P(S_k = S_k^1)$ for all different θ. That is, we need to estimate the power function $\beta_k(\theta) = P(S_k = S_k^1 \mid \theta)$ for all θ. As described in Section 4.6.1, the power function can be estimated directly or in some cases derived analytically by knowing the distribution of the measured data.

Assume that $\beta_k(\theta)$ is estimated directly by using measured data. In that case note that even though the amount of data needed to get accurate estimates can be large, it is still *much* less compared to if e.g. $P(FA)$ was going to be estimated directly.

6.1.6 Calculating Bounds of the Risk Function

The bounds (6.7) and (6.8) give upper and lower bounds of the probabilities $P(FA)$, $P(MIM)$, and $P(ID)$. These bounds can now be used to calculate the bounds of the risk function (6.2). The lower and upper bounds of the risk function $R(\theta,\delta)$ will be denoted $\underline{R}(\theta,\delta)$ and $\overline{R}(\theta,\delta)$ respectively. The derivation of bounds is exemplified in the following example:

Example 6.2

Consider the same diagnosis system as in Example 6.1. To calculate bounds of $R(\theta,\delta)$ in the case $\theta \in \Theta_{NF}$, we need bounds of $P(FA)$. Since (6.4) is a minimal DNF, Presumption 6.1 and 6.2 together with Lemma 6.4 give the bounds

$$\underline{R}(\theta,\delta) = c_{NF}(\theta)\Big[1 - P(S_1 = S_1^1)P(S_2 = S_2^1)P(S_3 = S_3^1)\Big]$$

and

$$\overline{R}(\theta,\delta) = c_{NF}(\theta)\max\{P(S_1 = S_1^1), P(S_2 = S_2^1)P(S_3 = S_3^1)\}$$

Next consider the case $\theta \in \Theta_{F_3} \cap \Theta_{insign}$, i.e. the fault belongs to fault mode F_3 *and* it is insignificant. To calculate the risk function (6.2), we need $P(ID)$ given by (6.6), which is a minimal DNF. Then Presumption 6.1 and 6.2, and

Lemma 6.2 give the bounds

$$\underline{R}(\theta,\delta) = c_{ID}(\theta)\Big[1 - P(S_2 = S_0^2)P(S_3 = S_0^3)\Big]$$

and

$$\overline{R}(\theta,\delta) = c_{NF}(\theta)\max\{P(S_2 = S_0^2), P(S_3 = S_0^3)\}$$

■

For each $\delta(x)$ and θ we get one lower and one upper bound. If a finite set of θ:s is considered, the values of the bounds for a certain $\delta(x)$ can be represented in a table:

θ_i	$\underline{R}(\theta,\delta)$	$\overline{R}(\theta,\delta)$
⋮	⋮	⋮

6.2 Finding the "Best" Diagnosis System

Given a set $\mathcal{C}$ of diagnosis systems, we will here discuss if we can find the "best" diagnosis system in $\mathcal{C}$, and in that case how to do it. The measure of performance is the risk function defined in Section 6.1.2 and we thus want to find the diagnosis system δ with minimal risk $R(\theta,\delta)$. The problem is that $R(\theta,\delta)$ for a given δ, is not a constant but a *function* of θ. Given two diagnosis systems δ_1 and δ_2, it can happen that $R(\theta,\delta_1) < R(\theta,\delta_2)$ for some values of θ while $R(\theta,\delta_1) > R(\theta,\delta_2)$ for some other values of θ. For example δ_1 performs better with respect to false alarm and δ_2 performs better with respect to missed detection. It is obvious that the original goal, of finding *the* best diagnosis system, must be modified and instead, we should try to find a, preferably small, *set* of *good* diagnosis systems.

The problem of a performance measure that is a *function* is not something unique for diagnosis systems. Actually, it is a common situation in general decision problems. In decision theory, several principles have therefore been developed to deal with this issue. In the next two sections, we discuss how these general principles can be applied to the problem of finding the "best" or at least good diagnosis systems.

6.2.1 Comparing Decision Rules (Diagnosis Systems)

Since this section discusses finding diagnosis systems from the standpoint of general decision theory, we will mainly refer to general *decision rules* instead of diagnosis systems.

To be able to compare different decision rules (here diagnosis systems), the relations *better* and *equivalent* are defined:

Definition 6.5 (Better) *A decision rule* δ_1 *is* better *than a decision rule* δ_2 *if*

$$\forall\theta.R(\theta,\delta_1) \leq R(\theta,\delta_1)$$

and

$$\exists\theta.R(\theta,\delta_1) < R(\theta,\delta_1)$$

A decision rule δ_1 *is* equivalent *to a decision rule* δ_2 *if*

$$\forall\theta.R(\theta,\delta_1) = R(\theta,\delta_1)$$

In the case where the risk is not available but instead, we have both an upper and a lower bound, the definition of *better* and *equivalent* must be modified:

Definition 6.6 (Better) *A decision rule* δ_1 *is* better *than a decision rule* δ_2 *if*

$$\forall\theta.\ \overline{R}(\theta,\delta_1) \leq \overline{R}(\theta,\delta_2)\ \wedge\ \underline{R}(\theta,\delta_1) \leq \underline{R}(\theta,\delta_2)$$

and

$$\exists\theta.\ \overline{R}(\theta,\delta_1) < \overline{R}(\theta,\delta_2)\ \vee\ \underline{R}(\theta,\delta_1) < \underline{R}(\theta,\delta_2)$$

A decision rule δ_1 *is* equivalent *to a decision rule* δ_2 *if*

$$\forall\theta.\overline{R}(\theta,\delta_1) = \overline{R}(\theta,\delta_2)$$

and

$$\forall\theta.\underline{R}(\theta,\delta_1) = \underline{R}(\theta,\delta_2)$$

The relations 6.5 and 6.6 define a partial order on the set of decision rules (or diagnosis systems). Corresponding to minimal elements of a partial order, decision theory uses the term *admissible*:

Definition 6.7 (Admissible Decision Rule) *A decision rule* δ *is* admissible *if there exists no better decision rule* δ'.

It is obvious that we need to consider only admissible decision rules (diagnosis systems) when trying to find good diagnosis systems. If $\mathcal{C}$ is the set of diagnosis systems considered, we use the notation $\mathcal{C}_{adm}$ for the set of admissible diagnosis systems in $\mathcal{C}$.

6.2.2 Choosing Diagnosis System

Even though the concept of admissibility reduces the set of diagnosis systems we need to consider to $\mathcal{C}_{adm}$, it is probable that the set $\mathcal{C}_{adm}$ is still to large. We need a principle to pick out one or possibly a few δ in $\mathcal{C}_{adm}$ that represents a good choice. We will here discuss three such principles: the Bayes' risk principle, the minimax principle, and the *approximate minimization* principle. The first two of these originates from classical decision theory and the third is presented in this work.

The Bayes' Risk Principle

Using the Bayes' risk principle, we assume that there is a prior distribution $\pi(\theta)$ on the parameter space Θ. Then we can evaluate the Bayes' risk:

$$r(\delta) = E\{R(\theta, \delta(X))\}$$

with expectation taken with respect to both θ and X (X is the data). Then the *Bayes' risk principle* is to choose the diagnosis system with lowest Bayes' risk. The problem with this principle is that a prior $\pi(\theta)$ is rarely available, i.e. we seldom know the probability of different faults. However, an alternative is to see $\pi(\theta)$ as a design parameter.

The Minimax Principle

Consider the quantity

$$\sup_{\theta \in \Theta} R(\theta, \delta) \tag{6.26}$$

which represents the worst thing that can happen if δ is used. The *minimax principle* is to choose the diagnosis system which minimizes (6.26). The problem with this principle is that, even though it is the worst case, only one θ-value for each δ is used.

Figure 6.7 illustrates the problem. With the minimax principle, the diagnosis system δ_2 with the right risk function would be preferred to a diagnosis system δ_1 with the left risk function. However, in most cases the δ_1 would be a much better choice since its performance approximately equals the performance of δ_2 for small θ-values, and for all other θ-values, δ_1 outperforms δ_2. It can also be the case that the prior $\pi(\theta)$ for small θ-values is very small and then the minimax principle would be even worse.

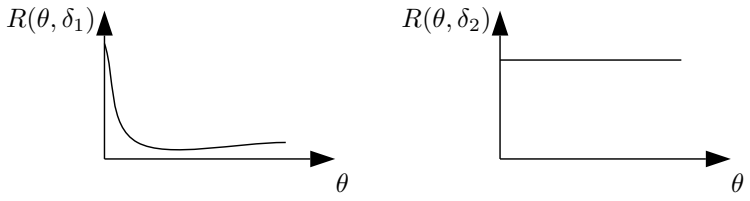

Figure 6.7: The problem with the minimax principle.

The Approximate Minimization Principle

As described above, there are arguments to not use the well-known Bayes' or minimax principles. There is a need for a principle that do not require a prior

but in some way consider more than one θ-value. To meet these requirements, we first define the function

$$R_{min}(\theta) = \min_{\delta} R(\theta, \delta)$$

which for each θ represents the best performance of any diagnosis system. Then we define a scalar measure of a risk function:

$$\|R(\theta, \delta)\| = \sup_{\theta} \big(R(\theta, \delta) - R_{min}(\theta)\big)$$

The *approximate minimization principle* is then to choose the diagnosis system which minimizes $\|R(\theta, \delta)\|$. It can be the case that not one single diagnosis system minimizes $\|R(\theta, \delta)\|$, but rather a whole set, which we will denote $\mathcal{C}_{\approx min}$. The result can be seen as that $R(\theta, \delta)$ is "almost" minimized for each θ-value, thereby the name "approximate minimization".

With this principle, the functions c_{FA}, c_{MD}, etc. defined in (6.1.1) works as weighting functions that can be used to emphasize different θ-values.

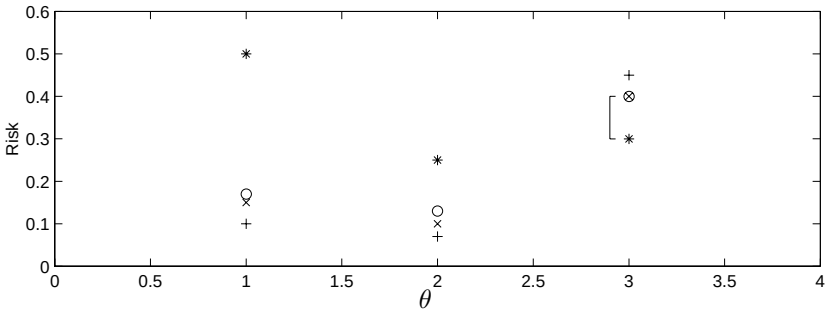

Figure 6.8: Illustration of the Approximate Minimization principle.

Example 6.3

Consider Figure 6.8. The parameter set is $\Theta = \{1, 2, 3\}$. Four decision rules are considered and their risk functions have been plotted, marked with $*$, $\circ$, $\times$, and $+$ respectively. The value of $R_{min}(\theta)$ becomes [0.1 0.07 0.3]. The measure $\|R(\theta, \delta)\|$ becomes

	$\|R(\theta, \delta)\|$
$*$	0.4
$\circ$	0.1
$\times$	0.1
$+$	0.15

and thus the measure is minimized by $\circ$ and $\times$. The size of this minimized measure is shown in the figure as a vertical bar. ■

6.3 A Procedure for Automatic Design of Diagnosis Systems

With the theory presented in the previous sections, it is quite straightforward to formulate a procedure for systematic and automatic design of diagnosis systems. The procedure presented here has been developed from procedures in (Nyberg and Nielsen, 1997*a*) and (Nyberg, 1998).

Consider the set of all diagnosis systems $\mathcal{D}$. Then by using the loss function defined in Section 6.1.1, we want to search in $\mathcal{D}$ for admissible diagnosis systems and then apply the principle of approximate minimization. The problem is that $\mathcal{D}$ is too large. A solution is to first restrict $\mathcal{D}$ to a set $\mathcal{C} \subset \mathcal{D}$, which hopefully contains most of the good diagnosis systems. Thus, the first step in the procedure is to find a good initial set $\mathcal{C}$.

6.3.1 Generating a Good Initial Set $\mathcal{C}$ of Diagnosis Systems

As input to the procedure, we use a set of hypothesis tests $\mathcal{T}$, called *test candidates*, and a set of measurement data $\mathcal{M} = \langle \mathcal{M}_1, \ldots \mathcal{M}_n \rangle$. By using the measurement data $\mathcal{M}$ and computing the test quantities, we can estimate the correlation between them By restricting the set $\mathcal{T}$ so that it does not contain highly correlated test quantities, the size of $\mathcal{T}$ can be reduced. This is desirable to save computational load in later steps of the procedure.

Also by using measurement data, the tests can be tuned for good performance. For each test, this should include at least tuning of the threshold so that a desired significance level is obtained. Also possible to include is a "tuning" of the sets S_k^1 and S_k^0, i.e. to add or remove some fault modes. Note that an equivalent way of describing this is that the decision structure is modified, e.g. some 0:s are changed to X:s.

The selection of threshold and sets S_k^1 and S_k^0 in each test, largely affects the performance of the tests and also the diagnosis system. To analyze this, we can use the principles that were discussed in Section 4.7. There it was concluded that we need the power function $\beta_k(\theta)$ which can be estimated from the measurement data $\mathcal{M}$ in accordance with Section 4.6.1.

Optimal thresholds values are difficult to obtain but we can use the heuristic to choose a certain level of significance α and then select the thresholds of each test δ_k such that $\alpha_k = \alpha$. This has the advantage that the probability of the events FA, ID, and MIM can be quite easily expressed in α as shown in Section 6.1.5 and especially in Table 6.1.

The set S_k^1 can be tuned by using the formula (4.42). That is, fault modes for which the power function is not small, should be added to S_k^1. This means that, if the incidence structure was used to determine a first choice of S_k^1, many fault modes may not have been included in S_k^1. However, when measurement data are used, model errors become important, and some of the fault modes, that were not originally included in S_k^1, must now be added. It is possible to also use a similar "tuning" of the sets S_k^0 by using the formula (4.43).

Note that it is also possible to include two or more different tunings of a single hypothesis test, in the set $\mathcal{T}$. For example, assume that the original set S_3^1 does not contain a fault mode F_1, but the power function derived from measurement data shows that F_1 should be added to S_3^1. Then, it is possible to include two tests δ_3 and $\delta_{3'}$, corresponding to different choices of S_3^1, in the set $\mathcal{T}$.

After that each hypothesis test has been tuned, and possibly several variations of some tests have been included in $\mathcal{T}$, the set $\mathcal{C}$ is obtained as all possible nonempty subsets of $\mathcal{T}$, i.e.

$$\mathcal{C} = 2^{\mathcal{T}} - \emptyset$$

6.3.2 Summary of the procedure

The whole procedure for systematic and automatic diagnosis system design can be summarized as follows.

Input: The input is $\langle \mathcal{T}, \mathcal{M} \rangle$, where $\mathcal{T}$ is a set of hypothesis test candidates and $\mathcal{M}$ a set of measurement data.

Step 1: Generate a good set $\mathcal{C}$ of diagnosis systems:

1. Start with a set of test candidates $\mathcal{T}$.
2. Use measurement data $\mathcal{M}$ to estimate correlation and reduce $\mathcal{T}$ such that it does not contain highly correlated test quantities.
3. For each test candidates in $\mathcal{T}$, use measurement data $\mathcal{M}$, and estimate $\beta_k(\theta)$ for different thresholds and for different θ_i:s corresponding the measurements $\mathcal{M}_i$.
4. Use the estimated $\beta_k(\theta)$ to tune each test, which includes tuning of thresholds and possibly also the sets S_k^1 and S_k^0.
5. Let $\mathcal{C}$ be all possible nonempty subsets of $\mathcal{T}$.

Step 2: Calculate $\underline{R}(\theta_i, \delta)$ and $\overline{R}(\theta_i, \delta)$ for all $\delta \in \mathcal{C}$:

1. For each $\delta \in \mathcal{C}$, derive propositional logic expressions for the events FA, ID, and MIM, for the different fault modes.
2. For each $\delta \in \mathcal{C}$, transform the propositional logic expressions to minimal DNF's.
3. For each $\delta \in \mathcal{C}$, use the minimal DNF's, Presumption 6.1 and 6.2, and the estimate of $\beta_k(\theta)$ to calculate probability bounds for the cases $\theta_1, \ldots, \theta_n$.
4. For each $\delta \in \mathcal{C}$, use the probability bounds to calculate $\underline{R}(\theta_i, \delta)$ and $\overline{R}(\theta_i, \delta)$.

Step 3: Pick out the admissible set $\mathcal{C}_{adm} \subseteq \mathcal{C}$.

Step 4: Apply approximate minimization to get $\mathcal{C}_{\approx min} \subseteq \mathcal{C}_{adm}$.

Output: The output is $\mathcal{C}_{\approx min}$.

An alternative to the above procedure is to switch the order of steps 3 and 4. It can be realized that this gives the same result, i.e.

$$\{\mathcal{C}_{adm}\}_{\approx min} = \{\mathcal{C}_{\approx min}\}_{adm}$$

The reason for switching steps 3 and 4 is that the operation of extracting a set of admissible decision rules is computationally more heavy than the operation of approximate minimization.

As said in Section 6.2.2, the output $\mathcal{C}_{\approx min}$ can be more than one diagnosis system. If this is the case, the diagnosis system containing the least number of hypothesis tests should be chosen for implementation. This is to minimize the diagnosis system complexity and computational load.

6.3.3 Discussion

Design of diagnosis systems is an optimization problem. The optimization problem addressed by the procedure described above, is to optimize the risk with respect to thresholds (and possibly other parameters of the individual tests) and selection of individual tests to be included. In the solution of the procedure, the optimization problem is divided into two subproblems: first the thresholds are fixed for each test and then, hypothesis tests to be included are selected. Because this "two-stage approach" is used, global optimum is not guaranteed. However, if sufficient computer power is available, it is possible to try several thresholds for each test. This could be done by increasing the size of $\mathcal{T}$ such that each test is included more than once, but with different thresholds. This makes it possible to get closer to a globally optimal solution. Another reason for non-optimality is that minimizing the bounds may not necessarily result in that the actual risks are minimized.

One potential problem with using the procedure is application specific requirements of low probabilities of false alarm, missed detection, etc. Because of this, the thresholds must be chosen such that the probabilities of $P(S_k = S_0^1)$ becomes highly dependent on the tail of the density functions of the test quantities. In this area, the probability estimates become unreliable which further implies that the bounds of the risk function becomes unreliable. The output from the procedure might be far from optimal.

To overcome this problem there are at least three possible solutions. One is to use longer measurement sequences. However, practical limitations can make this difficult. Another solution is to estimate parametric models of the probability density functions, e.g. see (Gustavsson and Palmqvist, 1997). The third solution is to accept a higher rate of undesirable events, i.e. false alarm, missed detection, etc., and then take care of these undesirable events by adding some after-treatment of the output from the diagnosis system. For example, the time, for which the original diagnosis system signals alarm, can be summed up and the alarm can be suppressed until the time-sum reaches a threshold.

The whole procedure is automatic, i.e. when input data are provided, all steps can be performed without any human involvement. This means that the

only thing left to automatize is the construction of the test quantities in the hypothesis tests. A general solution to this is a topic of future research but for limited classes of diagnosis problems, e.g. for the case of linear systems with fault modeled as additive signals, solutions are already available.

The procedure has been implemented as a Matlab command. The part of the procedure, requiring most computing power, is to derive the minimal DNF's of the events FA, ID, etc. However, in many cases it might be possible to "pre-calculate" minimal DNF expressions for the events of interest.

6.4 Application to an Automotive Engine

When constructing a model-based diagnosis system for automotive engines, it is desirable to strive for an optimum performance and at the same time minimize the amount of engineering work required. Automotive engines are rarely designed from scratch but often subject to small changes, e.g. for every new model year. Then usually also the diagnosis system needs to be changed. Since this may happen quite often and a car manufacturer typically has many different engine models in production, it is important for the car manufacturers that diagnosis systems can be reconstructed with minimal amount of work involved.

For manufacturers of independent diagnosis systems, to be used in independent repair-shops, the situation is even more critical. They need to design diagnosis systems for a large amount of different car brands and models. This makes it necessary to find procedures such that diagnosis systems can be constructed with very limited amount of work.

Thus, in the automotive area, there is a large need for a systematic and automatic procedure like the one presented in the previous section. In this section, the procedure is applied to the construction of the diagnosis system for the air-intake system. The resulting diagnosis system is then experimentally evaluated in Section 6.4.6.

6.4.1 Experimental Setup

The engine is a 2.3 liter 4 cylinder SAAB production engine mounted in a test bench together with a Schenk "DYNAS NT 85" AC dynamometer. Note that this is not the same engine as the one used in Chapter 5. The measured variables are the same as the ones used for engine control. A schematic picture of the whole engine is shown in Figure 6.9.

The part of the engine, that is considered to be the air-intake system, is everything to the left of the dashed line in Figure 6.9. When studying the air intake system, also the engine speed must be taken into account because it affects the amount of air that is drawn into the engine.

6.4.2 Model Construction

As we noted in Chapter 5, the automotive engine is a non-linear plant and it has been indicated in several works by different authors, that for the purpose of

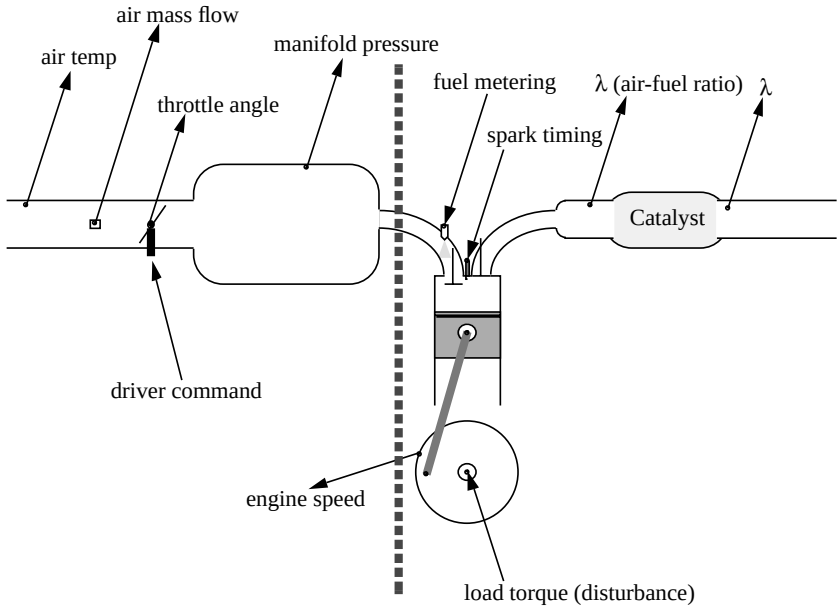

Figure 6.9: A basic automotive engine.

diagnosis, a linear model is not sufficient. As for the applications in Chapter 5, there is no need for extremely fast fault detection, and therefore a so called *mean value model* (Hendricks, 1990) is chosen. This means that no within cycle variations are covered by the model. The model is continuous and has one state which is the manifold pressure. The air dynamics is derived from the ideal gas law.

The process inputs are the throttle angle α (which is assumed to be unknown), and the engine speed n. The outputs are the throttle angle sensor α_s, the air-mass flow sensor m_s and the manifold pressure sensor p_s. The equations describing the fault-free model can be written as

$$\dot{p} = \frac{RT_{man}}{V_{man}}(m_{th} - m_{ac}) \tag{6.27a}$$

$$m_{th} = f(p, \alpha) \tag{6.27b}$$

$$m_{ac} = g(p, n) \tag{6.27c}$$

where p is the manifold pressure, R the gas constant, T_{man} the manifold air temperature, V_{man} the manifold volume, m_{th} the air-mass flow past the throttle, m_{ac} the air-mass flow out from the manifold into the cylinders, α the throttle angle, and n the engine speed.

The model consists of a physical part, (6.27a), and a black box part, the functions (6.27b) and (6.27c). Even if variations in ambient pressure and temperature do affect the system, they are here assumed to be constant. The static

functions $f(p, \alpha)$ and $g(p, n)$, are represented by polynomials. The identification of the functions f and g, and the constant V_{man}, is described in (Nyberg and Nielsen, 1997*b*).

6.4.3 Fault Modes Considered

The components that are to be diagnosed are the throttle angle sensor, the air-mass flow meter, and the manifold pressure sensor. Four system fault modes are considered:

NF	No Fault
M	air-Mass sensor fault
A	throttle-Angle sensor fault
P	manifold-Pressure sensor fault

As seen only single-fault modes are considered. In all cases the faults are modeled as arbitrary signals added to the physical quantities, i.e.

$$m_s(t) = m(t) + f_M(t) \tag{6.28a}$$
$$\alpha_s(t) = \alpha(t) + f_A(t) \tag{6.28b}$$
$$p_s(t) = p(t) + f_P(t) \tag{6.28c}$$

where the index s represents measured sensor signals. For fault mode NF, all functions $f_M(t)$, $f_A(t)$, and $f_P(t)$ are zero and for each of the other fault modes, one of the three functions are nonzero.

All this means that the fault state parameter θ at a particular time t_0 is

$$\theta = [f_M(t)\ f_A(t)\ f_P(t)] \qquad t \leq t_0$$

That is, θ is a vector of three *functions*. The definition of the parameter spaces Θ, Θ_{NF}, Θ_M, Θ_A, and Θ_P follows naturally.

6.4.4 Construction of the Hypothesis Test Candidates

The inputs to the diagnosis system, and therefore also the individual tests, are m_s, α_s, p_s, and n. Because the faults are modeled as additive arbitrary signals, the test quantity in each of the hypothesis tests becomes a residual generator.

The model of the air-intake system is non-linear. Because of the scarcity of design methods for residual generators for non-linear systems, we have to rely mostly on ad-hoc design. To not introduce unnecessary constraints, the design of residuals is not restricted to one method. Instead a combination of static relationships, non-linear diagnostic observers, and parity equations is used to construct 12 residuals of the type where an output is compared to an estimate of the output, or two estimates of the same output are compared. The

computational form of these 12 residuals are

$$\begin{aligned}
r_1 &= m_s - \hat{m}_1(\alpha_s, p_s) \\
r_2 &= m_s - \hat{m}_2(n, p_s) \\
r_3 &= p_s - \hat{p}_1(\alpha_s, n, p_s) \\
r_4 &= m_s - \hat{m}_3(\alpha_s, n, m_s) \\
r_5 &= p_s - \hat{p}_2(a_s, m_s, n) \\
r_6 &= \alpha_s - \hat{a}_1(u, \alpha_s, m_s, p_s) \\
r_7 &= m_s - \hat{m}_4(\alpha_s, n, p_s) \\
r_8 &= r_2 - r_1 = \hat{m}_1(\alpha_s, p_s) - \hat{m}_2(n, p_s) \\
r_9 &= r_4 - r_2 = \hat{m}_2(n, p_s) - \hat{m}_3(\alpha_s, n, m_s) \\
r_{10} &= r_4 - r_1 = \hat{m}_1(\alpha_s, p_s) - \hat{m}_3(\alpha_s, n, m_s) \\
r_{11} &= r_3 - r_5 = \hat{p}_2(\alpha_s, m_s, n) - \hat{p}_1(\alpha_s, n, p_s) \\
r_{12} &= \alpha_s - \hat{\alpha}_2(m_s, p_s)
\end{aligned}$$

where $\hat{m}_i$, $\hat{\alpha}_i$, and $\hat{p}_i$ are different estimates of the output signals. The details on how these estimates are formed can be found in Appendix 6.A.

Each of the 12 residuals is used to form a hypothesis test and thus we have a set $\mathcal{T}$ of 12 hypothesis test candidates. Different test quantities (i.e. the residual generators) are sensitive to different faults. This can be seen by studying the equations of the residuals and is summarized in Table 6.2, which contains the incidence structure for the 12 test quantities. As can be seen, there are some X:s in the incidence structure. The reason for this was explained in Example 3.2.

From the incidence structure, the decision structure is derived by replacing 1:s by X:s. Because of how the fault models (6.28) are constructed, the decision structure will contain only 0:s and X:s.

	NF	M	A	P
r_1	0	1	1	X
r_2	0	1	0	X
r_3	0	0	1	1
r_4	0	1	1	0
r_5	0	1	1	1
r_6	0	1	1	1
r_7	0	1	1	X
r_8	0	1	0	1
r_9	0	1	1	1
r_{10}	0	1	1	1
r_{11}	0	1	1	1
r_{12}	0	1	1	1

Table 6.2: The incidence structure of the test quantities for the 12 hypothesis test.

6.4.5 Applying the Procedure for Automatic Design

Following is a description of how all steps in the procedure, listed in Section 6.3.2, is applied to the design of a diagnosis system for the air-intake system.

Input

The input to the procedure is the set of 12 test candidates $\mathcal{T}$ defined in the previous section and a set of measurement data $\mathcal{M}$. The measurement data $\mathcal{M}$ were collected from the real engine during a one minute fault-free test cycle, see (Nyberg and Nielsen, 1997*b*). All faults were added to fault-free measurements and constant bias faults were chosen. The fault sizes were $\pm 2\%$, $\pm 4\%$, and $\pm 6\%$ for the α-fault, $\pm 2.5\%$, $\pm 5\%$, and $\pm 7.5\%$ for the m-fault, and $\pm 2\%$, $\pm 4\%$, and $\pm 6\%$ for the p-fault. For each sensor, the two smallest fault sizes (negative or positive) are considered to be insignificant faults and rest of the four fault sizes are considered to be significant faults. In addition there were one fault-free measurement. This means that measurements have been collected for 19 points in the infinitely large parameter space Θ.

Step 1: Generation of the set $\mathcal{C}$

The measurement data set in $\mathcal{M}$ corresponding to $\theta \in \Theta_{NF}$, i.e. fault free measurements, is used to calculate correlation between the test quantities of the tests $\mathcal{T}$. From studying the correlation coefficients, it is concluded that test quantities 1 and 7 are highly correlated, $C(r_1, r_7) = 0.99$, and also test quantities 5 and 11, $C(r_5, r_{11}) = 0.99$. Therefore, test quantities 7 and 11 are omitted from $\mathcal{T}$. This means that we are left with a $\mathcal{T}'$ containing 10 test candidates.

The power function $\beta_k(\theta)$ is estimated, using the measurement data $\mathcal{M}$, for thresholds in the range 0 to 20. With its help, $P(S_k \neq S_k^{des};\ |\ \theta_i)$ is plotted in Figure 6.10. The fact that test quantities 1 and 7, and 5 and 11, are highly correlated is seen in these plots because the plots for the corresponding pairs are very similar.

By using the power function $\beta_k(\theta)$ for $\theta \in \Theta_{NF}$, the threshold J_k for each test δ_k is chosen such that the significance level becomes $\alpha_k = 0.05$. Table 6.3 shows the threshold levels J_k for all tests in $\mathcal{T}$. The sets S_k^1 and S_k^0 need not to be modified because formulas (4.42) and (4.43) are fulfilled.

Now when thresholds and sets S_k^1 and S_k^0 have been fixed, let $\mathcal{C}$ be all possible nonempty subsets of $\mathcal{T}'$. The size of $\mathcal{C}$ is $2^{10} - 1 = 1023$.

Step 2: Calculation of $\underline{R}(\theta_i, \delta)$ and $\overline{R}(\theta_i, \delta)$

The power functions $\beta_k(\theta)$ estimated in the previous step can now be used to obtain estimates of the probabilities $P(S_k = S_k^1\ |\theta)$ for all θ_i, $i = 1, \ldots, 19$ and all tests in $\mathcal{T}$. This means that $19 \cdot 10 = 190$ probabilities are estimated. These are used to estimate the bounds $\underline{R}(\theta_i, \delta)$ and $\overline{R}(\theta_i, \delta)$. In total, there are $1023 \cdot 19 \cdot 2 = 38874$ bounds.

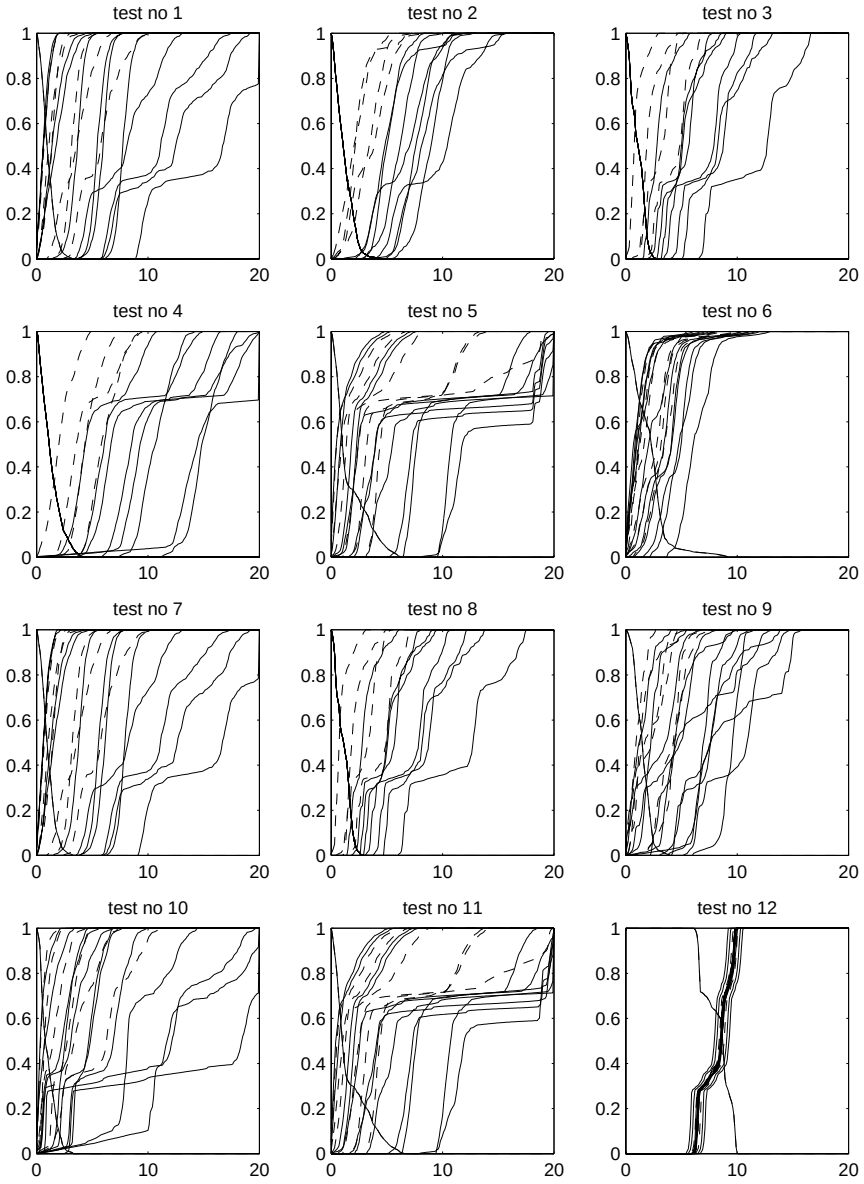

Figure 6.10: The probability $P(S_k \neq S_k^{des})$ for each test as a function of the threshold. The lines for significant faults are solid and for insignificant faults dashed. Also the lines for fault mode NF are solid.

δ_k	J_k
δ_1	2.15
δ_2	2.55
δ_3	2.05
δ_4	3.15
δ_5	4.85
δ_6	4.25
δ_7	2.15
δ_8	2.15
δ_9	2.55
δ_{10}	2.25
δ_{11}	5.05
δ_{12}	9.85

Table 6.3: The thresholds for all tests in $\mathcal{T}$.

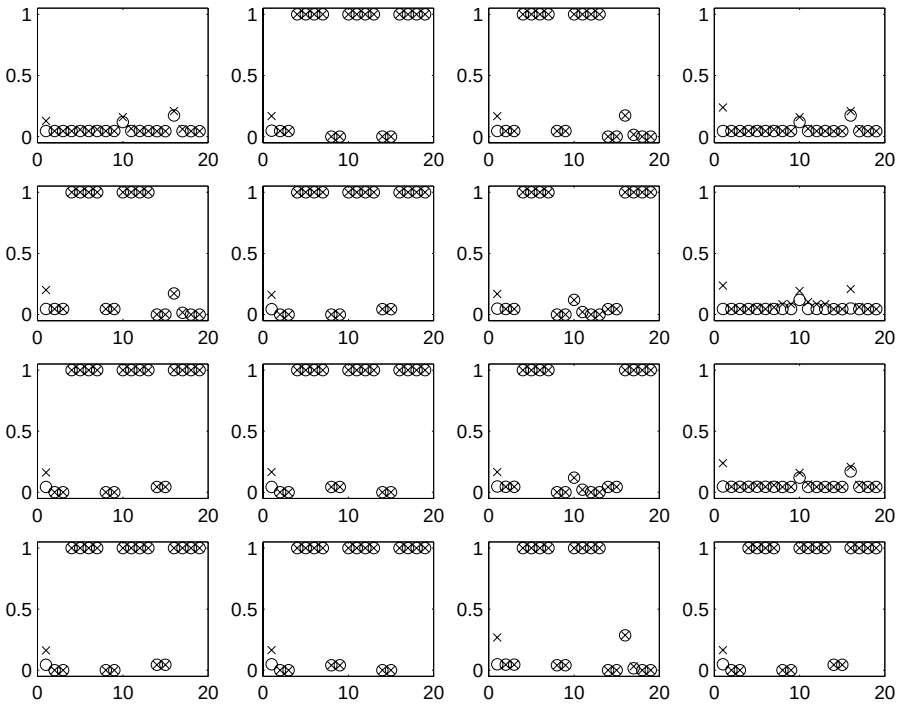

Figure 6.11: The risk bounds for 16 different diagnosis systems.

In Figure 6.11, these bounds for 16 different diagnosis systems $\delta \in \mathcal{C}$ have been plotted. The x-axis in each diagnosis system shows the index i of θ_i and the y-axis shows the value of $\underline{R}(\theta_i, \delta)$ and $\overline{R}(\theta_i, \delta)$. The x-marks represent $\overline{R}(\theta_i, \delta)$ and the circles represent $\underline{R}(\theta_i, \delta)$. By visual inspection, it is seen that the diagnosis system represented by the top left plot, is the best of these 16.

Step 3&4: Finding the Admissible Set and Approximate Minimization

The admissible set $\mathcal{C}_{adm}$ contains 15 diagnosis systems. After applying approximate minimization, there is only 1 diagnosis systems left in the set $\mathcal{C}_{\approx min}$. We will denote this diagnosis system with δ^{best}. The decision structure for δ^{best} is

	NF	M	A	P
δ_2	0	0	X	X
δ_3	0	X	0	X
δ_4	0	X	X	0

The risk bounds for δ^{best} are plotted in the top left plot of Figure 6.11.

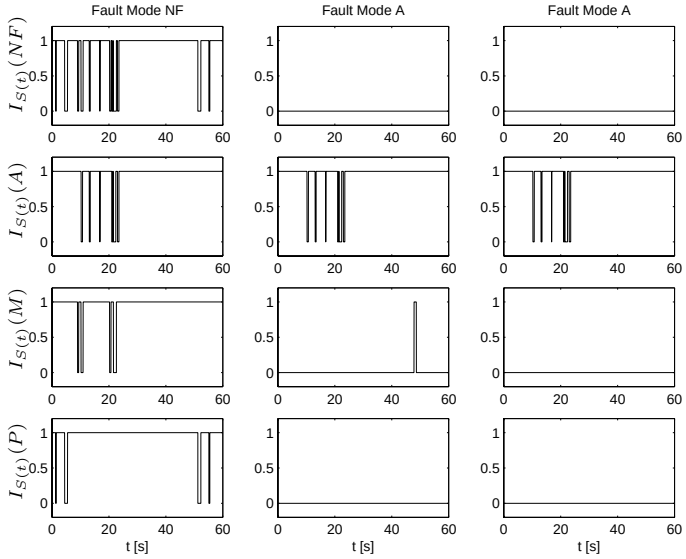

Figure 6.12: Confirmation of the diagnosis system δ^{best} for the cases, NF, insignificant A, and significant A.

6.4.6 Confirmation of the Design

To confirm the design, the single diagnosis system, that was the output from the procedure, is tested using the 19 fault cases that was used for the design. Of these 19 cases, the result of 6 cases are shown in Figure 6.12 and 6.13.

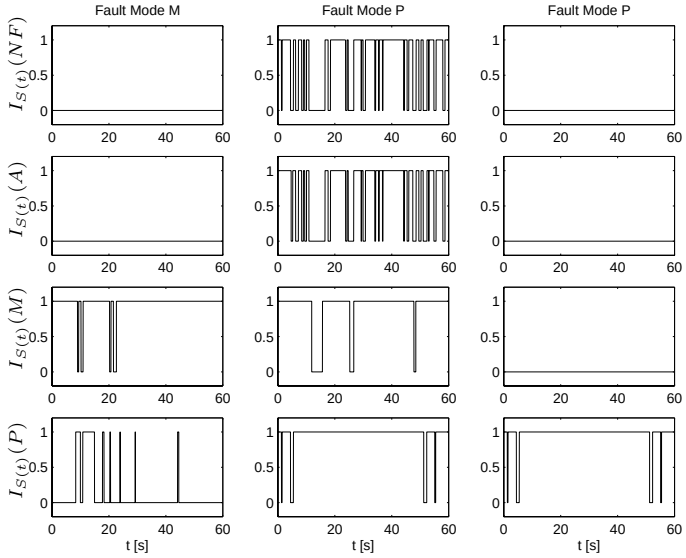

Figure 6.13: Confirmation of the diagnosis system δ^{best} for the cases significant M, insignificant P, and significant P.

Each column of plots represents one test case and the present fault mode is indicated on the top. Each row of plots represents the *indicator functions* $I_{S(t)}(F_i)$ for each of the fault modes (indicated to the left). The indicator function $I_{S(t)}(F_i)$ has the value 1 if $F_i \in S(t)$ and 0 if $F_i \notin S(t)$. For example consider the leftmost column of Figure 6.12. In this case, the present fault mode is NF which means that the event of interest is FA and to prevent that FA occurs, NF should belong to S all the time. As seen in the upper left plot, this is however not the case. On several occasions, the indicator function goes to zero which means that $NF \notin S$, i.e. the event FA occurs.

Consider next the second column of Figure 6.12. The present fault mode is A and the fault is insignificant. This means that the event of interest is ID and to prevent ID, the indicator function for A (the second row) must be one. Also here, there are some occasions where this indicator function goes to zero, which means that ID occurs.

For the third column of Figure 6.12, the present fault mode is also A but this time, the fault is significant. Thus, the event of interest is MIM and to prevent MIM, the indicator function for A must be one and all other indicator functions must be zero. As seen in the plots, this is true almost all the time.

It is clear that the automatic procedure successfully manage to construct a diagnosis system for the air-intake system of the engine. The performance is not perfect but we should remember that the fault sizes that are considered to be significant faults are comparably small for this application. If better performance, in terms of fewer false alarms etc., is required, then the smallest

faults in Θ_{sign} must be moved to Θ_{insign}. Then the threshold levels can be increased, with maintained low probability of MIM, and the probability of FA and ID will get lower.

The result of all 19 confirmation tests is summarized in Table 6.4. The first column shows the index i of the measurement. The second column shows the fault mode present during each measurement. Depending on what fault mode is present and if it is significant or not, the risk function is proportional to the probability of the event listed in the third column (compare with (6.2)). Then in columns four to six, the pre-calculated bounds of the risk function $R(\theta_i, \delta)$ is compared to the actual relative frequency of the corresponding events. It is seen that, although the bounds are derived using certain assumptions, they manage to surround the actual value in almost all cases. Only for the 5:th test case, the actual value is outside the range specified by the bounds. Also for this case, the bound is still pretty good although not perfect. The reason for this may be that the actual value is only the relative frequency and not the expectation, or that the assumptions made to derive the bounds do not hold.

i	Fault Mode Present	Event	$\underline{R}(\theta_i, \delta)$	actual frequency	$\overline{R}(\theta_i, \delta)$
1	NF	FA	0.0454	0.112	0.129
2	A	ID	0.0454	0.0454	0.0454
3	A	ID	0.0454	0.0454	0.0454
4	A	MIM	0.0454	0.0454	0.0454
5	A	MIM	0.0454	0.0573	0.0567
6	A	MIM	0.0454	0.0454	0.0454
7	A	MIM	0.0454	0.0573	0.0587
8	M	ID	0.0451	0.0451	0.0451
9	M	ID	0.0451	0.0451	0.0451
10	M	MIM	0.12	0.151	0.159
11	M	MIM	0.0451	0.0495	0.0651
12	M	MIM	0.0451	0.0451	0.0451
13	M	MIM	0.0451	0.0451	0.0451
14	P	ID	0.0441	0.0441	0.0441
15	P	ID	0.0441	0.0441	0.0441
16	P	MIM	0.172	0.191	0.208
17	P	MIM	0.0441	0.0464	0.0609
18	P	MIM	0.0441	0.0441	0.0441
19	P	MIM	0.0441	0.0441	0.0441

Table 6.4: The actual frequency and the risk bounds.

6.5 Conclusions

It is highly desirable to systematize and automate the process of designing diagnosis systems. The reason is that in many applications, high diagnosis performance is required and at the same time, the time-consuming engineering work of designing diagnosis systems must be minimized. In this chapter model-based diagnosis based on structured hypothesis tests was considered, and for this kind of diagnosis systems, a systematic and automatic design procedure has been proposed.

Concepts from decision theory are used to define a performance measure, which reflects the probability of e.g. false alarm and missed detection. These kinds of probabilities are usually hard to obtain, since they typically require knowledge and analysis of multidimensional density functions. However, this problem is solved here by using measurement data to estimate one-dimensional density functions and then using relations developed, to derive the probability of e.g. false alarm.

The automatic procedure tries to optimize the performance measure by selecting the optimal set of hypothesis tests to be included, and also by tuning each hypothesis test with respect to thresholds and sets S_k^1 and S_k^0. The procedure is successfully applied to the problem of designing a diagnosis system for the air-intake system of an automotive engine. The complete design chain has been discussed, including model construction, design of test quantities, and selection and tuning of the hypothesis tests. The resulting diagnosis system is then experimentally validated.

Appendix

6.A Estimation of Engine Variables

Below, we shortly presents how the estimates of the engine variables p, m, and α are formed. The estimation principles relies on the model (6.27) which was developed in (Nyberg and Nielsen, 1997*b*).

Estimates of Manifold Pressure p

The two different estimates of the manifold pressure p are based on observers of p, and are formed as:

$$\begin{aligned}\dot{\hat{p}} &= \frac{RT_{man}}{V_{man}}\big(f(\hat{p},\alpha_s) - g(\hat{p},n) + K_1(p_s - \hat{p})\big)\\ \hat{p}_1(\alpha_s, n, p_s) &= p\end{aligned}$$

$$\begin{aligned}\dot{\hat{p}} &= \frac{RT_{man}}{V_{man}}\Big(f(\hat{p},\alpha_s) - g(\hat{p},n) + K_2\big(m_s - f(\hat{p},\alpha_s)\big)\Big)\\ \hat{p}_2(\alpha_s, m_s, n) &= p\end{aligned}$$

Estimates of Air-Mass Flow m

For the estimates of the air-mass flow m, we can use both static and dynamic relationships in the model (6.27). In forming $m_2(n, p_s)$ we assume that an estimate of $\dot{p}$ is available. The four different estimates of m are:

$$\hat{m}_1(\alpha_s, p_s) = f(p, \alpha_s)$$

$$\hat{m}_2(n, p_s) = g(p_s, n) - \frac{V_{man}}{RT_{man}}\dot{\hat{p}}$$

$$\begin{aligned}\dot{\hat{p}} &= \frac{RT_{man}}{V_{man}}\Big(f(\hat{p},\alpha_s) - g(\hat{p},n) + K_3\big(m_s - f(\hat{p},\alpha_s)\big)\Big)\\ \hat{m}_3(\alpha_s, m_s, n) &= f(\hat{p}, \alpha_s)\end{aligned}$$

$$\begin{aligned}\dot{\hat{p}} &= \frac{RT_{man}}{V_{man}}\big(f(\hat{p},\alpha_s) - g(\hat{p},n) + K_4(p_s - \hat{p})\big)\\ \hat{m}_4(\alpha_s, n, p_s) &= f(\hat{p}, \alpha_s)\end{aligned}$$

Estimates of Throttle Angle α

The first estimate of the throttle angle α utilizes the fact that the throttle is controlled by a DC-servo and that we know the input $u(t)$ to the DC-servo. Also, we have an model available of the DC-servo, and this model have two states: the angular velocity ω and the throttle angle α. The load disturbance originating from the air-flow past the throttle must also be taken into account. This air flow is modeled by a static function $h(p_s, m_s, \alpha_s)$. More information on the DC-servo model can be found in (Nyberg and Nielsen, 1997*b*).

The estimate $\hat{\alpha}_1$ is formed by using an observer of the DC-servo states:

$$\begin{aligned}\dot{\hat{\omega}} &= a\hat{\omega} + b\big(u(t) - h(p_s, m_s, \alpha_s)\big) + k_1(\alpha_s - \hat{\alpha}) \\ \dot{\hat{\alpha}} &= \hat{\omega} + k_2(\alpha_s - \hat{\alpha}) \\ \hat{\alpha}_1(u, \alpha_s, m_s, p_s) &= \alpha\end{aligned}$$

The second estimate of α is derived by first assuming that α is a state with dynamics $\dot{\alpha} = 0$. Then the estimate $\hat{\alpha}_2$ is formed by using an observer for the state α:

$$\begin{aligned}\dot{\hat{\alpha}} &= K(m_s - f(p_s, \hat{\alpha})) \\ \hat{\alpha}_2(m_s, p_s) &= \alpha\end{aligned}$$

Chapter 7

Linear Residual Generation

Residual generation was shortly mentioned in Section 4.2.2 as a special case of the prediction principle. When talking about residual generation, we assume that all faults are modeled as signals $f(t)$ and a setup with a residual generator can therefore be illustrated as in Figure 7.1. The residual generator filters the known signals and generates a test quantity which is seen as a signal $r(t)$, the *residual*. The residual should be "small" (ideally 0) in the fault-free case and "large" when a fault is acting on the system.

In Figure 7.1, we have also assumed that all, if any, disturbances are modeled as signals denoted $d(t)$. We remember from Section 4.2 and 4.5 that the test quantity, here the residual, should be made insensitive to disturbances. That is, when generating the residual $r(t)$, disturbances should be decoupled.

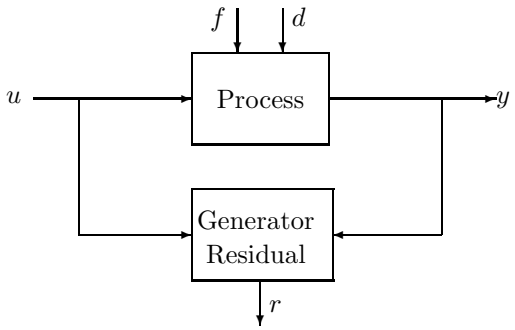

Figure 7.1: A residual generator.

This chapter is a study of how to design *linear* residual generators for *linear* systems with no model uncertainties. Most of the discussion will be focused on decoupling. Further, only perfect decoupling of the disturbances is considered, and the issue of *approximate decoupling* associated with e.g. robust diagnosis (see Section 4.5) is not considered here.

The limitation to linear models is quite hard since few real systems are modeled well by linear models. As was said in Chapter 1, this limitation is also much harder in diagnosis compared to closed-loop control. The reason is that the feedback, used in closed-loop control, tends to be forgiving against model errors. Diagnosis should be compared to open-loop control since no feedback is involved. All model errors propagates through the diagnosis system and degrades the diagnosis performance.

In Section 7.1, we will more exactly formulate the problem of linear residual generation. We will see that the actual problem is to design *polynomial parity functions*. Most of this chapter contains discussions around two design methods for polynomial parity functions (or equivalently linear residual generators): the novel *minimal polynomial approach* and the well-known Chow-Willsky scheme. In Section 7.2, the *minimal polynomial approach* is presented and the notion of a basis for all polynomial parity functions is introduced. Then in Section 7.3, it is proved that a basis of degree less or equal to the order of the system, always exists. The Chow-Willsky scheme is explained in Section 7.4, and the relation between the minimal polynomial approach and the Chow-Willsky scheme is investigated in Section 7.5. Finally Section 7.6 contains a design example.

Many concepts and terms from linear systems theory will be used. The most important ones are summarized in Appendix 7.B.

7.1 Problem Formulation

As was said in Section 4.2, to be able to perform isolation, not only the disturbances but also some faults need to be decoupled. It is convenient to distinguish between *monitored* and *non-monitored* faults. Monitored faults are the fault signals that we want the residual to be sensitive to. Non-monitored faults are the fault signals that we want the residual to be not sensitive to, i.e. the faults that we want to decouple.

A formal definition of a residual is as follows:

Definition 7.1 (Residual) *A residual is a scalar signal that for all known inputs $u(t)$ and all disturbances $d(t)$ (including non-monitored faults), should be zero, i.e. $r(t) \equiv 0$, in the fault-free case, and should be non-zero, i.e. $r(t) \not\equiv 0$, when monitored faults are present.*

We also define a residual generator formally:

Definition 7.2 (Residual Generator) *A residual generator is a system that takes process input and output signals as inputs and generates a residual.*

The residual generator can be a static system if it is based on *static redundancy*, or a dynamic system if it is based on *temporal redundancy*. Note that from the residual generator point of view, there is no difference between disturbances and non-monitored faults. Therefore, everywhere the word disturbance is used in this chapter, it also includes non-monitored faults.

Residual generator design in general includes a large amount of model building. However, here we consider the model to be given. Also, although many different types of considerations are important when designing residual generators, we will here solely study the decoupling of disturbances. Altogether, the problem studied in this chapter will be called the *decoupling problem* and can be phrased as follows:

Decoupling Problem 1 *Given a model, the decoupling problem is to design a residual generator so that the residual becomes insensitive to the known input u and the disturbance d (including non-monitored faults) and sensitive to monitored faults f, i.e.*

(a) For all $u(t)$ and $d(t)$, it should hold that $f(t) \equiv 0$ implies $r(t) \equiv 0$.

(b) For all $u(t)$ and $d(t)$, it should hold that $f(t) \not\equiv 0$ implies $r(t) \not\equiv 0$.

The restriction to limit the discussion to the above decoupling problem may seem to be hard; other important issues, not covered by the decoupling problem, are for example response time and sensitivity to disturbances. However, it should be noted that this restriction is made in most diagnosis literature.

7.1.1 The Linear Decoupling Problem

From now on, the discussion will be restricted even more, namely to linear systems. Then the model given is linear and represented either by transfer functions or in state-space form. The transfer function representation is

$$y = G(\sigma)u + H(\sigma)d + L(\sigma)f \tag{7.1}$$

where y is the measured output with dimension m, u is the known input with dimension k_u, d is the disturbance with dimension k_d, f is the fault with dimension k_f, and $G(\sigma)$, $H(\sigma)$, and $L(\sigma)$ are transfer-matrices of suitable dimensions. Note again that we will always assume that d includes the non-monitored faults and f does only contain monitored faults. The operator σ represents the differentiation operator p (or s) in the continuous case and the time-shift operator q (or z) in the discrete case.

The state-space form representation is

$$\begin{aligned} \sigma x(t) &= Ax(t) + B_u u(t) + B_d d(t) + B_f f(t) && \text{(7.2a)} \\ y(t) &= Cx(t) + D_u u(t) + D_d d(t) + D_f f(t) && \text{(7.2b)} \end{aligned}$$

and, unless especially mentioned, no assumptions about controllability or observability are made.

A general linear residual generator is a linear filter and can be written

$$r = Q(\sigma) \begin{bmatrix} y \\ u \end{bmatrix} \tag{7.3}$$

i.e. $Q(\sigma)$ is a transfer matrix with dimension $1 \times (m + k_u)$. We will define the *order* of a linear residual generator to be its McMillan degree, i.e. the number of states in a minimal realization.

A number of design methods for designing linear residual generators, have been proposed in literature, see for example (Patton and Kangethe, 1989) (Wünnenberg, 1990; White and Speyer, 1987; Massoumnia, Verghese and Willsky, 1989; Nikoukhah, 1994; Chow and Willsky, 1984; Nyberg and Nielsen, 1997*c*). All these methods are methods to design the transfer matrix $Q(\sigma)$. Note that this includes for example the case when the residual generator is based on observers formulated in state space.

If expression (7.3) is developed, we see that a linear residual generator can also be represented as

$$r = Q(\sigma) \begin{bmatrix} y \\ u \end{bmatrix} = c^{-1}(\sigma) F(\sigma) \begin{bmatrix} y \\ u \end{bmatrix} = \tag{7.4}$$

$$= \frac{A_1(\sigma)y_1 + \ldots + A_m(\sigma)y_m + B_1(\sigma)u_1 + \ldots + B_k(\sigma)u_{k_u}}{c(\sigma)} \tag{7.5}$$

where $F(\sigma)$ is a polynomial row-vector and $A_i(\sigma)$, $B_j(\sigma)$, and $c(\sigma)$, are scalar polynomials in σ. Note that the order of the residual generator is equal to the degree of the polynomial $c(\sigma)$.

According to the Decoupling Problem, the objective is to create a signal that is affected by monitored faults but not by any other signals. This is equivalent to finding a filter $Q(\sigma)$ which fulfills the following two requirements:

- The transfer functions from known inputs u and disturbances d, to the residual must be zero.
- The transfer functions from monitored faults f to the residual must be non-zero.

These two requirements introduce a constraint on the numerator polynomial of (7.5) only, i.e. $F(s)$ or equivalently $A_i(\sigma)$ and $B_j(\sigma)$. The only constraints on the denominator polynomial $c(\sigma)$ is that the residual generator must be realizable and asymptotically stable. The first of these constraints means that it must have a degree greater or equal to the row-degree of $F(\sigma)$, i.e. the largest degree of the numerator polynomials $A_i(\sigma)$ and $B_j(\sigma)$. That is, the minimal order of the residual generator is determined by the row-degree of $F(\sigma)$. The second constraint means, for example in the continuous case, that $c(\sigma)$ must have all its zeros placed in the left half plane.

It is obvious that $c(\sigma)$, or equivalently the poles of the residual generator, can be chosen almost arbitrarily. This statement is valid for a large class of residual generator design methods, including diagnostic observer design, e.g. eigenstructure (Patton and Kangethe, 1989) or the unknown input observer (Wünnenberg, 1990), in which poles also are placed arbitrarily. Although we *can* choose $c(\sigma)$ arbitrarily, is often suitable to choose it so that a low-pass filtering effect is achieved.

It is clear that the numerator of (7.5) is of great importance for residual generation. In fact, when the Decoupling Problem is considered, the numerator is the only thing we need to care about. This numerator will be called a *polynomial parity function*. In accordance with (Chow and Willsky, 1984), we also define the *order* of the polynomial parity function as the highest degree α of σ^α, which is present in the parity function, i.e. the row-degree of the polynomial vector $F(s)$.

The *linear* decoupling problem can now be expressed as follows:

Linear Decoupling Problem 1 *Given a linear model, (7.1) or (7.2), the linear decoupling problem is to design a polynomial parity function, or equivalently a polynomial vector $F(s)$, so that*

(a) For all $u(t)$ and $d(t)$, it should hold that $f(t) \equiv 0$ implies $F(\sigma)\begin{pmatrix} y \\ u \end{pmatrix} \equiv 0$.

(b) For all $u(t)$ and $d(t)$, it should hold that $f(t) \not\equiv 0$ implies $F(\sigma)\begin{pmatrix} y \\ u \end{pmatrix} \not\equiv 0$.

Although the following result may have been realized at this point, it is here expressed as a theorem to emphasize its importance.

Theorem 7.1 *When linear models and linear residual generators are considered, the Decoupling Problem is equivalent to the Linear Decoupling Problem.*

Proof: Assuming a scalar polynomial $c(\sigma)$ that is non-zero, it holds that

$$F(\sigma)\begin{pmatrix} y \\ u \end{pmatrix} \equiv 0 \tag{7.6}$$

if and only if

$$c^{-1}(\sigma)F(\sigma)\begin{pmatrix} y \\ u \end{pmatrix} \equiv 0 \tag{7.7}$$

and this proves the theorem. ■

There are indications that this theorem can be generalized to also the case when robust residual generation is considered (Frisk, 1998).

We will in this chapter discuss two algorithms for design of polynomial parity functions: the new *minimal polynomial basis approach* and the well-known *Chow-Willsky scheme*. There will be a focus on the following three questions:

- Does the method find all possible polynomial parity functions?
- Does the method explicitly find polynomial parity functions of minimal order?
- Does the solution represent a minimal parameterization, of all polynomial parity functions, or is it over parameterized?

The reason for the interest in the minimal order property of the polynomial parity function is primarily that we want to depend on the model as little as possible. A low order usually implies that only a small part of the model is utilized. Since all parts of the model has errors, this further means that few model errors will affect the residual. The residual will then become small when no faults are present.

It is obvious that if we can find a design algorithm, for which the answer is "yes" to all these questions, then we have also found a design algorithm for *residual generators* that can find all possible residual generators, explicitly the ones of minimal order, and with a minimal parameterization. In addition to the above three questions, we will also discuss numerical properties of the algorithms. All of these questions are quite natural but in spite of this, they have not gained very much attention before in the literature.

7.1.2 Parity Functions

Before the discussion of the algorithms, we will try to bring some clarity to the terms *parity function*, *parity equation* etc., that are frequently encountered in the diagnosis literature.

In the seventies, research about using *analytical redundancy* for fault detection and diagnosis was intensified. One main area of interest was fault detection for aircrafts and especially their control and navigation systems. In a work within this field, Potter and Suman (1977) defined *parity equation* and *parity function* (and also *parity space* and *parity vector*). This was originally a concept for utilizing analytical redundancy in the form of linear *direct redundancy*. In 1984 the concept was generalized by Chow and Willsky (1984) to include also dynamic systems, i.e. to utilize *temporal redundancy*. However only discrete time parity equations were considered.

Since then, a number of different usages of the term *parity function* and *parity equation* have occured in the literature. However, no other usages of the term *parity equations*, than in accordance with the definitions made by Potter and Suman (1977) and later extended by Chow and Willsky (1984), have been widely accepted in the research community.

To clarify the meaning here, we use the terms *polynomial parity equation* and *polynomial parity functions*, which are the type of parity equations/functions defined in (Chow and Willsky, 1984).

The definition of polynomial parity functions becomes:

Definition 7.3 (Polynomial Parity Function) *A polynomial parity function is a function $h(u(t), y(t))$ that can be written as*

$$h(u, y) = A(\sigma)y + B(\sigma)u$$

where $A(\sigma)$ and $B(\sigma)$ are polynomial vectors in σ. The value of the function is zero if no faults are present.

A *polynomial parity equation* is then basically a polynomial parity function set to zero, i.e. $h(u, y) = 0$.

Some researchers, e.g. (Höfling, 1993), have worried about that polynomial parity functions are not possible to implement directly or a least give bad performance. However, in these cases they forget to add the poles represented by $c(s)$ in expression (7.5).

Remark: Parity equations that are not polynomial are often mentioned in the literature, e.g. ARMA parity equation (Gertler, 1991), dynamic parity relations (Gertler and Monajemy, 1995). In accordance with standard mathematical notion, these should be called *rational parity equations*. A *rational parity function* is then identical with a linear residual generator.

Note that parity equations/functions are in this view not a design method; it is solely an equation/function with specific properties.

Example 7.1

Consider the discrete linear system

$$y(t) = \frac{B(q)}{A(q)}u(t) + f(t)$$

where u is the input, y the output and f the fault. If the fault is omitted, this relationship can be rewritten as

$$A(q)y(t) = B(q)u(t)$$

This is an example of one polynomial parity equation that can be formed, and it will be satisfied as long as the fault is zero. From the polynomial parity equation, we can derive the parity function

$$h(t) = A(q)y(t) - B(q)u(t)$$

It is obvious that this polynomial parity function will respond to the fault f. If this expression is multiplied with an appropriate backward time-shift q^{-n}, the resulting parity function can therefore serve as a residual generator. ■

In the following sections, we will discuss the two methods for designing polynomial parity functions: the *minimal polynomial basis approach* and the well-known *Chow-Willsky scheme*. These methods are explicitly focused on polynomial parity functions but in principle, all linear residual generator design methods are methods, at least implicitly, for design of polynomial parity functions.

7.2 The Minimal Polynomial Basis Approach

This section introduces the *minimal polynomial basis approach* to the design of polynomial parity functions. With this approach, it is shown that the Decoupling Problem is transformed into finding a minimal basis for a null-space of a polynomial matrix. This is a standard problem in established linear systems theory, which means that numerically efficient computational tools are generally

available. It is shown that the minimal polynomial basis approach can find all possible residual generators, explicitly those of minimal order, and the solution has a minimal parameterization. All derivations are performed in the continuous case but the corresponding results for the time-discrete case can be obtained by substituting s by z and *improper* by *non-causal*. To simplify notation, the term *parity function* will from now on be used instead of *polynomial parity function*. Several concepts from linear systems theory, especially polynomial matrices, will be used. A short description of some key terms and concepts are given in Appendix 7.B.

7.2.1 Basic Idea

By utilizing the model description (7.1), a parity function can be expressed as

$$F(s)\begin{bmatrix} y \\ u \end{bmatrix} = F(s)\begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix}\begin{bmatrix} u \\ d \end{bmatrix} + F(s)\begin{bmatrix} L(s) \\ 0 \end{bmatrix} f$$

It is obvious that to fulfill condition (a) of the Linear Decoupling Problem, it must hold that

$$F(s)\begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} = 0$$

This condition is fulfilled if and only if $F(s)$ belongs to the left null-space of

$$M(s) = \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} \tag{7.8}$$

The left null-space of the matrix $M(s)$ will be denoted $\mathcal{N}_L(M(s))$.

The polynomial vector $F(s)$ needs to fulfill two requirements: belong to the left null-space of $M(s)$ and also have good fault sensitivity properties. If, in a first step of the design, *all* $F(s)$ that fulfill the first requirement are found, then a single $F(s)$ with good fault sensitivity properties can be selected. Thus, in a first step of the design of the parity function $F(s)[y^T\ u^T]^T$, we need not consider f or $L(s)$. The problem is then to find *all* polynomial vectors $F(s) \in \mathcal{N}_L(M(s))$. Of special interest are the parity functions of minimal order, i.e. the polynomial vectors $F(s)$ of minimal row degree.

Thus we want to find all $F(s) \in \mathcal{N}_L(M(s))$ and explicitly those of minimal order. This can be done by finding a *minimal polynomial basis* for the rational vector-space $\mathcal{N}_L(M(s))$. Procedures for doing this will be described in Section 7.2.2 and 7.2.3. Let the basis be formed by the rows of a matrix denoted $N_M(s)$. By inspection of (7.8), it can be realized that the dimension of $\mathcal{N}_L(M(s))$ (i.e. the number of rows of $N_M(s)$) is

$$\begin{aligned} \text{Dim } \mathcal{N}_L(M(s)) = m + k_u - \text{Rank}\, M(s) = m + k_u - (k_u \text{Rank}\, H(s)) = \\ = m - \text{Rank}\, H(s) =^* m - k_d \end{aligned} \tag{7.9}$$

where m is the number of outputs, i.e. the dimension of $y(t)$, and k_d is the number of disturbances, i.e. the dimension of $d(t)$. The last equality, marked $=^*$, holds only if rank $H(s) = k_d$, but this should be the normal case.

Forming a Parity Function

The second and final design-step is to use the polynomial basis $N_M(s)$ to form the parity function. For this, consider the following theorem:

Theorem 7.2 ((Kailath, 1980), Irreducible Basis) *If the rows of $N(s)$ is an irreducible polynomial basis for a space $\mathcal{F}$, then all polynomial row vectors $f(s) \in \mathcal{F}$ can be written $f(s) = \phi(s)N(s)$ where $\phi(s)$ is a polynomial row vector.*

The proof is given in Appendix 7.B.

The minimal polynomial basis $N_M(s)$ is irreducible (see Theorem 7.14 Appendix 7.B) and then, according to Theorem 7.2, all decoupling *polynomial* vectors $F(s)$ can be parameterized as

$$F(s) = \phi(s)N_M(s) \tag{7.10}$$

where $\phi(s)$ is a polynomial vector of suitable dimension. The parameterization vector $\phi(s)$ can for example be used to shape the fault-to-residual response or simply to select one row in $N_M(s)$. Since $N_M(s)$ is a basis, the parameterization vector $\phi(s)$ have minimal number of elements, i.e. a minimal parameterization.

One of the rows of $N_M(s)$ corresponds to a parity function of minimal order. The reason for this can be explained as follows. Consider a basis $N_M(s)$ with three rows and the row-degrees are d_1, d_2, and d_3 respectively. Since $N_M(s)$ is a minimal polynomial basis, we know that $d_1 + d_2 + d_3$ is minimal (see Theorem 7.14 Appendix 7.B). Now assume that the minimal order of any parity function is d_{min} and that $d_{min} < d_i$ for all d_i. Then by using a minimal order parity function, we can obtain a new basis with less order. Thus $N_M(s)$ can not be a minimal basis, which shows that one of the rows of $N_M(s)$ must correspond to a parity function of minimal order.

7.2.2 Methods to find a Minimal Polynomial Basis to $\mathcal{N}_L(M(s))$

The problem of finding a minimal polynomial basis to the left null-space of the rational matrix $M(s)$ can be solved by a transformation to a problem of finding a minimal polynomial basis to the left null space of a polynomial matrix. This transformation can be done in several different ways. In this section, three possibilities are demonstrated, where the first is used if the model is given on the transfer function form (7.1), the second if the model is given in the state-space form (7.2), and the third if the model contains no disturbances. A description on how to compute a basis for the null-space of a polynomial matrix, will be given in Section 7.2.3.

The motivation for this transformation to a polynomial problem, is that there exists well established theory (Kailath, 1980) regarding polynomial matrices. In addition, the generally available Polynomial Toolbox (Henrion, Kraffer, Kwakernaak, M.Sebek and Strijbos, 1997) for MATLAB contains an extensive set of tools for numerical handling of polynomial matrices. We will see that the results in this and the next section, give us a a *computationally simple*, *efficient*,

and *numerically stable* method, to find a *polynomial* basis for the left null-space of $M(s)$.

Frequency Domain Solution

One way of transforming the rational problem to a polynomial problem is to perform a right MFD on $M(s)$, i.e.

$$M(s) = \widetilde{M}_1(s)\widetilde{D}^{-1}(s) \tag{7.11}$$

One simple example is

$$M(s) = \widetilde{M}_1(s)d^{-1}(s)$$

where $d(s)$ is the least common multiple of all denominators. By finding a polynomial basis for the left null-space of the *polynomial* matrix $\widetilde{M}_1(s)$, a basis is found also for the left null-space of $M(s)$. No solutions are missed because $\widetilde{D}(s)$ (e.g. $d(s)$) is of full normal rank. Thus the problem of finding a minimal polynomial basis to $\mathcal{N}_L(M(s))$ has been transformed into finding a minimal polynomial basis to $\mathcal{N}_L(\widetilde{M}_1(s))$.

State-Space Solution

Assume that the system is described the state-space form (7.2). To be able to obtain a basis that is irreducible, will need to require that the state x is controllable from only u and d. If this requirement is not fulfilled, the system must be transformed to a realization

$$\begin{aligned}\begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} &= \begin{bmatrix} A_x & A_{12} \\ 0 & A_z \end{bmatrix}\begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} B_{u,x} \\ 0 \end{bmatrix} u + \\ &\quad \begin{bmatrix} B_{d,x} \\ 0 \end{bmatrix} d + \begin{bmatrix} B_{f,x} \\ B_{f,z} \end{bmatrix} f \qquad (7.12a)\\ y &= [C_x C_z]\begin{bmatrix} x \\ z \end{bmatrix} + D_u u + D_d d + D_f f \qquad (7.12b)\end{aligned}$$

where the state x is controllable from $\begin{bmatrix} u^T & d^T \end{bmatrix}^T$ and the state z is controllable from the fault f. It is assured from Kalman's decomposition theorem that such a realization always exists. Finally it is assumed that the state z is asymptotically stable, which is the same as saying that the whole system is stabilizable. The notations A, B_u, B_d, B_f, and C will still be used and with the same meaning as before, e.g. $C = [C_x\ C_z]$ and $B_u = [B_{u,x}^T\ 0]^T$. To denote the dimension of the states x and z, we will use n_x and n_z respectively. Also we use n to denote the dimension of the total state, i.e. $n = n_x + n_z$.

To find the left null-space to $M(s)$ it is convienient to use the *system matrix* in state-space form (Rosenbrock, 1970). The system matrix has been used before in the context of fault diagnosis, see e.g. (Nikoukhah, 1994; Magni and

Mouyon, 1994). Denote the system matrix $M_x(s)$, describing the system with disturbances as inputs:

$$M_x(s) = \begin{bmatrix} C_x & D_d \\ -sI + A_x & B_{d,x} \end{bmatrix}$$

Define the matrix P_x as

$$P_x = \begin{bmatrix} I & -D_{u,x} \\ 0 & -B_{u,x} \end{bmatrix}$$

Then the following theorem gives a direct method on how to find a minimal polynomial basis to $\mathcal{N}_L(M(s))$ via the system matrix.

Theorem 7.3 *If the pair $\{A_x, [B_{u,x}\ B_{d,x}]\}$ is controllable and the rows of the polynomial matrix $V(s)$ is a minimal polynomial basis for $\mathcal{N}_L(M_x(s))$, then $W(s) = V(s)P_x$ is a minimal polynomial basis for $\mathcal{N}_L(M(s))$.*

Before this theorem can be proven, a lemma is needed:

Lemma 7.1 *Let $M(s)$ be the system matrix of any realization (not necessarily controllable from $\left[u^T\ d^T\right]^T$), i.e.*

$$M_s(s) = \begin{bmatrix} C & D_d \\ -(sI - A) & B_d \end{bmatrix}$$

Then it holds that

$$\mathrm{Dim}\,\mathcal{N}_L(M(s)) = \mathrm{Dim}\,\mathcal{N}_L(M_s(s))$$

The proof of this lemma is placed in Appendix 7.A.

Now, return to the proof of Theorem 7.3:

Proof: In the fault free case, i.e. $f = 0$, consider the following relation between the matrices $M(s)$ and $M_x(s)$:

$$\begin{aligned} P_x \begin{pmatrix} y \\ u \end{pmatrix} = P_x M(s) \begin{pmatrix} u \\ d \end{pmatrix} &= \begin{bmatrix} C_x(sI - A_x)^{-1}B_{u,x} & C_x(sI - A_x)^{-1}B_{d,x} + D_d \\ -B_{u,x} & 0 \end{bmatrix} \begin{pmatrix} u \\ d \end{pmatrix} = \\ &= \begin{bmatrix} C_x & D_d \\ -(sI - A_x) & B_{d,x} \end{bmatrix} \begin{bmatrix} (sI - A_x)^{-1}B_{u,x} & (sI - A_x)^{-1}B_{d,x} \\ 0 & I_{k_d} \end{bmatrix} \begin{pmatrix} u \\ d \end{pmatrix} = \\ &= M_x(s) \begin{pmatrix} x \\ d \end{pmatrix} \end{aligned}$$

If $V(s)M_x(s) = 0$, then since the signals $u(t)$ and $d(t)$ can be chosen arbitrarily, $P_x M(s)$ must also be 0. This implies that $W(s)M(s) = V(s)P_x M(s) = 0$, i.e. $W(s) \in \mathcal{N}_L(M(s))$. It is also immediate that if $V(s)$ is polynomial, $W(s) = V(s)P_x$ is also polynomial.

From Lemma 7.1, we have that $\mathrm{Dim}\,\mathcal{N}_L(M_x(s)) = \mathrm{Dim}\,\mathcal{N}_L(M(s))$. Then since both V and $W(s)$ has the same number of rows, the rows of $W(s)$ must span the whole null-space $\mathcal{N}_L(M(s))$, i.e. $W(s)$ must be a basis for $\mathcal{N}_L(M(s))$.

It is clear that the following relation must hold:

$$V(s)[P_x \; M_x(s)] = V(s) \begin{bmatrix} I & -D_u & C_x & D_d \\ 0 & -B_{u,x} & -(sI - A_x) & B_{d,x} \end{bmatrix} = [W(s) \; 0] \quad (7.13)$$

Consider the matrix $[P_x \; M_x(s)]$. Since the state x is controllable from u and d, the PBH test (see Appendix 7.B) implies that the lower part of this matrix has full rank for all s, i.e. it is irreducible. Now assume that $W(s)$ is not irreducible, i.e. there is a s_0 such that $W(s_0)$ does not have full row-rank. This means that there exists a $\gamma \neq 0$ such that $\gamma V(s_0)[P_x \; M_x(s_0)] = \gamma[W(s_0) \; 0] = 0$. Since $[P_x \; M_x(s_0)]$ has full row-rank it must hold that $\gamma V(s_0) = 0$. Therefore, $V(s)$ cannot be irreducible but this contradicts with the fact that $V(s)$ is a minimal polynomial basis. This contradiction implies that $W(s)$ must be irreducible.

The matrix $W(s)$ is now proven to be a polynomial, irreducible basis for $\mathcal{N}_L(M(s))$. According to Theorem 7.14, the only thing left to prove is that the basis $W(s)$ is row-reduced. Partition $V(s) = [V_1(s) \; V_2(s)]$ according to the partition of $M_x(s)$. Let

$$\begin{aligned} V_1(s) &= S_1(s)V_{1,hr} + q_1(s) \\ V_2(s) &= S_2(s)V_{2,hr} + q_2(s) \end{aligned}$$

The matrices $S_i(s)$ is diagonal matrices with diagonal elements $s^{k_{ij}}$ where k_{ij} is the row-degrees of $V_i(s)$. The constant matrices $V_{i,hr}$ is the highest-row-degree coefficient matrix and $q_i(s)$ is the rest polynomial. Since $V(s) \in \mathcal{N}_L(M_x(s))$, it holds that $V_1(s)C_x = V_2(s)(sI - A_x)$, i.e.

$$\begin{aligned} S_1(s)V_{1,hr}C_x + q_1(s)C_x &= S_2(s)V_{2,hr}(sI - A_x) + q_2(s)(sI - A_x) \\ &= sS_2(s)V_{2,hr} + \tilde{q}_2(s) \end{aligned}$$

By identifying the highest order terms on each side it is immediate that $sS_2(s) = S_1(s)$, i.e. each row in $V_2(s)$ has lower degree than the corresponding row in $V_1(s)C_x$. It also holds that the row-degrees in $V_1(s)C_x$ has less or equal row-degrees than $V_1(s)$ since C_x is a constant matrix. Thus, each row-degree in $V_2(s)$ has less degree than the corresponding row in $V_1(s)$ and therefore $V_{hr} = [V_{1,hr} \; 0]$. Since $V(s)$ is a minimal polynomial basis, it is row reduced. That is, the highest-row-degree coefficient matrix for $V(s)$ has full row rank. Since $V_{hr} = [V_{1,hr} \; 0]$, it follows that $V_{1,hr}$ has full row rank.

From the definition of P_x it follows that

$$[W_1(s) \; W_2(s)] = [V_1(s) \quad (-V_1(s)D_u - V_2(s)B_{u,x})]$$

From the degree discussion above it follows that the highest-row-degree coefficient matrix of $W(s)$ looks like $W_{hr} = [V_{1,hr} \; \star]$, which obviously has full row-rank, i.e. $W(s)$ is row reduced.

Thus we have shown that $W(s)$ is an irreducible basis and row reduced, which implies that it is a minimal polynomial basis. ∎

The next result tells us what happens when the realization considered is not controllable from $\begin{bmatrix} u^T & d^T \end{bmatrix}^T$. For this consider a system matrix

$$M_s(s) = \begin{bmatrix} C & D_d \\ -(sI - A) & B_d \end{bmatrix}$$

and the pair $\{A, [B_u \ B_d]\}$ is not necessarily controllable.

Theorem 7.4 *If the rows of the polynomial matrix $V(s)$ is a polynomial basis for $\mathcal{N}_L(M_s(s))$, then $W(s) = V(s)P$ is a polynomial basis for $\mathcal{N}_L(M(s))$.*

Proof: The first part of the proof of Theorem 7.3 is valid also for this theorem. ■

Note that compared to Theorem 7.3, we have in Theorem 7.4 relaxed the requirements of controllability and the minimality of the basis $V(s)$. The result is that $W(s)$ becomes here only a basis and not a minimal basis. Theorem 7.4 is only of theoretical interest in the context of parity function design but will be used for the detectability analysis presented in the next chapter.

The following examples illustrates Theorem 7.4. Also, it shows that the condition that $\{A, [B_u \ B_d]\}$ must be controllable, is really necessary when constructing a minimal polynomial basis for $\mathcal{N}_L(M(s))$.

Example 7.2

The system has one disturbance and two outputs:

$$A = \begin{bmatrix} -2 & -3 \\ 0 & -1 \end{bmatrix} \quad B_u = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad B_d = \begin{bmatrix} -2 \\ 0 \end{bmatrix} \quad B_f = \begin{bmatrix} -6 \\ -6 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 4 \\ 2 & 4 \end{bmatrix} \quad D_u = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad D_d = \begin{bmatrix} 6 \\ 5 \end{bmatrix} \quad D_f = \begin{bmatrix} -2 \\ 0 \end{bmatrix}$$

It is clear that the second state is not controllable from $\begin{bmatrix} u^T & d^T \end{bmatrix}^T$. By setting up $M_s(s)$ and finding a minimal polynomial basis $V(s)$ for $\mathcal{N}_L(M_s(s))$, we form the basis $N_M(s)$ as

$$\begin{aligned} N_M(s) &= V(s)P = \\ &= \begin{bmatrix} -0.833s^2 - 1.83s - 1 & s^2 + 2.67s + 1.67 & -1.167s - 1.167 \end{bmatrix} = \\ &= (s+1)\begin{bmatrix} -0.833s - 1 & s + 1.67 & -1.167 \end{bmatrix} \end{aligned}$$

The basis $N_M(s)$ is not irreducible since it looses rank for $s = -1$. ■

In conclusion, as in the previous subsection, the problem of finding a minimal polynomial basis to $\mathcal{N}_L(M(s))$ has been transformed into finding a minimal polynomial basis to a polynomial matrix, in this case the system matrix $M_x(s)$.

No Disturbance Case

If there are no disturbances, i.e. $H(s) = 0$, the matrix $M(s)$ gets a simpler structure

$$M_{nd}(s) = \begin{bmatrix} G(s) \\ I \end{bmatrix} \tag{7.14}$$

A minimal polynomial basis for the left null-space of $M_{nd}(s)$ is particularly simple due to the special structure and a minimal basis is then given directly by the following theorem:

Theorem 7.5 ((Kailath, 1980)) *, If $G(s)$ is a proper transfer matrix and $\bar{D}_G(s)$, $\bar{N}_G(s)$ form an irreducible left MFD, i.e. $\bar{N}_G(s)$ and $\bar{D}_G(s)$ are left co-prime and $G(s) = \bar{D}_G^{-1}(s)\bar{N}_G(s)$. Then,*

$$N_M(s) = [\bar{D}_G(s) \ -\bar{N}_G(s)] \tag{7.15}$$

forms a minimal basis for the left null-space of the matrix

$$M(s) = \begin{bmatrix} G(s) \\ I \end{bmatrix}$$

Here, the *dimension* of the null-space is m, i.e. the number of measurements, and the *order* of the minimal basis is given by the following theorem:

Theorem 7.6 *The set of observability indices of a transfer function $G(s)$ is equal to the set of row-degrees of $\bar{D}_G(s)$ in any row-reduced irreducible left MFD $G(s) = \bar{D}_G^{-1}(s)\bar{N}_G(s)$.*

A proof of the dual problem, controllability indices, can be found in (Chen, 1984) (p. 284).

Thus, a minimal polynomial basis for matrix $M_{nd}(s)$ is given by a left MFD of $G(s)$ and the order of the basis is the sum of the observability indices of $G(s)$.

The result (7.15) implies that finding the left null-space of the rational transfer matrix (7.8), in the general case with disturbances included, can be reduced to finding the left null-space of the rational matrix

$$\widetilde{M}_2(s) = \bar{D}_G(s)H(s) \tag{7.16}$$

By performing a right MFD on $H(s)$, e.g. $H(s) = \bar{N}_H(s)d_H^{-1}(s)$, the problem becomes to find a basis for the left null-space of the polynomial matrix $\bar{D}_G(s)\bar{N}_H(s)$. In other words, this is an alternative to the use of the matrix $\widetilde{M}_1(s)$ in (7.11). This view closely connects with the so called frequency domain methods, which are further examined in Section 7.2.4.

7.2.3 Finding a Minimal Polynomial Basis for the nullspace of a General Polynomial Matrix

For the general case, including disturbances, the only remaining problem is how to find a minimal polynomial basis to a polynomial matrix. This is a well-known problem in the general literature on linear systems and a number of different algorithms exist. In this section, two algorithms will be presented. The first is based on the *Hermite form* (Kailath, 1980) and a second algorithm is based on the *polynomial echelon form* (Kailath, 1980). Both methods are implemented in the Polynomial Toolbox (Henrion et al., 1997) for MATLAB. Again we remind the reader of Appendix 7.B in which many of the terms used in this section are explained.

The two algorithms have very different numerical properties. Although the algorithm based on Hermite form is easy to understand, it has poor numerical properties. It is included here mostly to gain some basic understanding of the problem. However the algorithm based on polynomial echelon form is both fast and numerically stable and should therefore be the preferred choice for design.

The Hermite Form Algorithm

Any polynomial matrix can be transformed into column Hermite form by elementary row operations. Assume $M(s)$ is a $p \times q$ matrix. Then there exists a $p \times p$, unimodular matrix $U(s) = [U_1^T(s)\ U_2^T(s)]^T$ such that

$$\begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} M(s) = \begin{bmatrix} R(s) \\ 0 \end{bmatrix}$$

where $R(s)$ is a $(p-r) \times q$ matrix and r is the normal rank of $M(s)$. The, non-unique, matrix $U(s)$ can be found e.g. as described in Theorem 6.3-2 in (Kailath, 1980). The last r rows in $U(s)$, i.e. $U_2(s)$, thus spans the left nullspace of $M(s)$. The matrix $U_2(s)$ is irreducible because $U(s)$ is unimodular. $U_2(s)$ is however not necessarily row-reduced, i.e. $U_2(s)$ is not necessarily a minimal basis. However, $U_2(s)$ can be made row-reduced by elementary row operations. This is best illustrated with an example that shows the main idea and also illustrates how the minimality property is connected with the row-reduced property.

Example 7.3

Consider the polynomial matrix $M(s)$ with rank $r = 2$

$$M(s) = \begin{bmatrix} 1 & 0 & -s \\ 0 & s^3 + 2s^2 + s & s^3 + 2s^2 + s \\ s & s^3 + 2s^2 + s & s^3 + s^2 + s \\ s^2 & 0 & -s^3 \end{bmatrix}$$

The column Hermite form of $M(s)$ is

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -s & -1 & 1 & 0 \\ -s^2 & 0 & 0 & 1 \end{bmatrix} M(s) = \begin{bmatrix} 1 & 0 & -s \\ 0 & s+2s^2+s^3 & s+2s^2+s^3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Here, the last two rows of $U(s)$ form a basis for the left null-space of $M(s)$ and is denoted $F(s)$.

$$F(s) = \begin{bmatrix} -s & -1 & 1 & 0 \\ -s^2 & 0 & 0 & 1 \end{bmatrix}$$

The matrix $F(s)$ is obviously irreducible, it is however not row-reduced because the highest-row-degree coefficient matrix F_{hr} is

$$F_{hr} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

and not of full rank. However, by multiplication from the left with a suitably chosen unimodular matrix, $F(s)$ can be made row-reduced. General algorithms to find the unimodular matrix making $F(s)$ row-reduced is available, e.g. (Callier, 1985). In the example above,

$$\begin{bmatrix} -1 & 0 \\ -s & 1 \end{bmatrix} F(s) = \begin{bmatrix} s & 1 & -1 & 0 \\ 0 & s & -s & 1 \end{bmatrix} = F_{\min}(s)$$

The matrix $F_{\min}(s)$ is both irreducible and row-reduced, and accordingly to Theorem 7.14 (in Appendix 7.B), it is a minimal basis for the left null-space. ■

The Polynomial Echelon Form Algorithm

The polynomial echelon form method is described in (Kailath, 1980; Kung, Kailath and Morf, 1977). Below follows a very brief description of the algorithm to illustrate the algorithm usage and computational complexity. The concepts presented here are also needed later in both this and the next chapter.

Consider the polynomial equation

$$F(s)M(s) = 0 \tag{7.17}$$

Assume that the polynomial basis $F(s)$ is in canonical *polynomial echelon form.* This assumption is not restrictive because of the following theorem

Theorem 7.7 ((Kailath, 1980), Section 6.7.2[1]) *For each space of rational vectors, there exists a minimal polynomial basis in (canonical) polynomial echelon form.*

[1] This theorem is not stated as a theorem in (Kailath, 1980), but the fact is contained in the text.

Proof: The theorem follows from the fact that all full row rank polynomial matrices can be transformed to polynomial echelon form by elementary row operations, i.e. by multiplication from the left with a unimodular matrix. ■

The left hand side of (7.17) can be rewritten as

$$F(s)M(s) = (F_0 + F_1 s + \dots F_\nu s^\nu)M(s) = [F_0 \dots F_\nu] \begin{bmatrix} M(s) \\ sM(s) \\ \vdots \\ s^\nu M(s) \end{bmatrix} =$$

$$= \widetilde{F}\mathcal{M}(s) = \widetilde{F}\widetilde{\mathcal{M}}\Psi_{k_u+k_d}(s)$$

which also defines $\mathcal{M}(s)$ and the *coefficient matrices* $\widetilde{F}$ and $\widetilde{\mathcal{M}}$. (The matrix $\widetilde{\mathcal{M}}$ is also known as the generalized resultant matrix of $M(s)$.) Note that the integer ν is usually not known *a priori*.

By examining the rows of $\mathcal{M}(s)$, from top to bottom, the rows can be classified as *independent rows* or *dependent rows*. A row is dependent if it can be written as a linear combination of previous rows, using only constant coefficients. The procedure to search for dependent rows in this way will be refered to as the *row-search* algorithm. Independent and dependent rows can equally well be determined from the coefficient matrix $\widetilde{\mathcal{M}}$. Note that the order, here top-to-bottom, is important. The order bottom-to-top would result in another set of dependent rows.

Since $F(s)$ is in polynomial echelon form, the rows of $\widetilde{F}$ must define a set of primary dependent rows in $\mathcal{M}(s)$. Also from the fact that $F(s)$ is in polynomial echelon form, we know that of all sets of primary dependent rows, the set defined by $\widetilde{F}$ must be of minimal order. That is, there is no other set of primary dependent rows, containing the same number of rows and with lower row-degrees.

Each set of primary dependent rows spans a subspace of $\mathcal{N}_l(M(s))$. Therefore, since $F(s)$ spans the whole left null-space of $\mathcal{M}(s)$, the set of primary dependent rows defined by $\widetilde{F}$, must be of largest possible size.

With these statements in mind, we know that the matrix $\widetilde{F}$, and also $F(s)$, can be found by searching, from top to bottom, in $\widetilde{\mathcal{M}}$ for the largest uppermost set of primary dependent rows. We summarize this result in the following theorem:

Theorem 7.8 ((Kailath, 1980), Section 6.7.2[1]) *Let $\widetilde{\mathcal{M}}$ be the coefficient matrix of $\mathcal{M}$. Let $\{w_1 \dots w_p\}$ be a set, of largest possible size, with primary dependent rows, in order top-to-bottom, of $\widetilde{\mathcal{M}}$. Then if the rows of $\widetilde{\mathcal{F}}$ define these dependencies, the matrix $F(s)$ is in quasi-canonical polynomial echelon form. The matrix $F(s)$ is also a row-reduced, but not necessarily irreducible, polynomial basis for $\mathcal{N}_L(M(s))$.*

Furthermore, if $\{w_1 \dots w_p\}$ is the uppermost set (i.e. the first encountered when searching top-to-bottom), of largest possible size, with primary dependent rows of $\widetilde{\mathcal{M}}$, then the matrix $F(s)$ is a minimal *polynomial basis for $\mathcal{N}_L(M(s))$.*

Proof: It follows trivially that $F(s)$ is in quasi-canonical polynomial echelon form. A matrix in quasi-canonical polynomial echelon form is always row-reduced and does always have full rank. Further, it trivially holds that $F(s)M(s) = 0$.

According to Theorem 7.7, there exist a minimal polynomial basis $F_{min}(s)$ in polynomial echelon form. Assume that the dimension of this basis is q. Since the basis $F_{min}(s)$ is in polynomial echelon form, its rows define a set of primary dependent rows of $\widetilde{\mathcal{M}}$. This set of primary dependent rows is of size q. Thus any set, of largest possible size, with primary dependent rows must have q elements. Therefore, the basis $F(s)$ has also dimension q which shows that it is a polynomial basis for $\mathcal{N}_L(M(s))$.

If $\{w_1 \dots w_p\}$ is the uppermost set, this means that the corresponding polynomial basis will have the same order as a minimal order basis $F_{min}(s)$ and thus is a minimal polynomial basis. ∎

In general, a search for the largest and uppermost set of primary dependent rows does *not* result in a unique basis, and thereby the name *quasi*-canonical polynomial echelon form. However if the dependencies are described in a specific way, the basis will be in *canonical* polynomial echelon form and thus unique.

When performing the search for primary dependent rows, it is important to know when to stop. That is, we need to know what the largest possible size, of a set of primary dependent rows, is. There are two possibilities. The first is that we know the rank of $M(s)$. Then the largest set of primary dependent rows will contain $p - \operatorname{rank} M(s)$ rows. The other possibility is to use a known upper limit of ν, when constructing the matrix $\mathcal{M}(s)$. Note that this is equivalent to that we know an upper limit of the maximum row-degree of a minimal basis. According to (Henrion et al., 1997), there is such an upper limit, i.e. $\nu \leq (p-1)\deg M(s)$, where $\deg M(s)$ denotes the maximum row (and column) degree of $M(s)$. We will see in Section 7.3, that in the special case of a minimal basis for the left nullspace of the matrix (7.8), an upper limit of ν is actually n_x, i.e. the dimension of the state controllable from $\begin{bmatrix} u^T & d^T \end{bmatrix}^T$.

Next follows an example to illustrate the calculation procedure.

Example 7.4

Consider the matrix

$$M(s) = \begin{bmatrix} s^4 + 2s^3 - 5s - 4 & 2s^3 + 2s^2 - 2s - 8 \\ -s^4 + 7s^3 + 7s^2 + 14s + 6 & -2s^4 - 5s^3 + s^2 + 3s \\ -2s^3 - s^2 - 17s - 9 & 2s^4 + 3s^3 - s^2 - s - 2 \\ 2s^4 + 3s^3 - s^2 - 9s - 4 & 0 \\ 0 & 2s^4 + 3s^3 - s^2 - 9s - 4 \end{bmatrix}$$

which has rank 2. Without no special reason, we will try to use the polynomial

echelon form algorithm with $\nu = 2$. Then the coefficient matrix $\widetilde{\mathcal{M}}$ becomes

$$\widetilde{\mathcal{M}} = \left[\begin{array}{rrrrrrrrrrrrrr}
-4 & -8 & -5 & -2 & 0 & 2 & 2 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\
6 & 0 & 14 & 3 & 7 & 1 & 7 & -5 & -1 & -2 & 0 & 0 & 0 & 0 \\
-9 & -2 & -17 & -1 & -1 & -1 & -2 & 3 & 0 & 2 & 0 & 0 & 0 & 0 \\
-4 & 0 & -9 & 0 & -1 & 0 & 3 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\
0 & -4 & 0 & -9 & 0 & -1 & 0 & 3 & 0 & 2 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & -4 & -8 & -5 & -2 & 0 & 2 & 2 & 2 & 1 & 0 & 0 & 0 \\
0 & 0 & 6 & 0 & 14 & 3 & 7 & 1 & 7 & -5 & -1 & -2 & 0 & 0 \\
0 & 0 & -9 & -2 & -17 & -1 & -1 & -1 & -2 & 3 & 0 & 2 & 0 & 0 \\
0 & 0 & -4 & 0 & -9 & 0 & -1 & 0 & 3 & 0 & 2 & 0 & 0 & 0 \\
0 & 0 & 0 & -4 & 0 & -9 & 0 & -1 & 0 & 3 & 0 & 2 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & -4 & -8 & -5 & -2 & 0 & 2 & 2 & 2 & 1 & 0 \\
0 & 0 & 0 & 0 & 6 & 0 & 14 & 3 & 7 & 1 & 7 & -5 & -1 & -2 \\
0 & 0 & 0 & 0 & -9 & -2 & -17 & -1 & -1 & -1 & -2 & 3 & 0 & 2 \\
0 & 0 & 0 & 0 & -4 & 0 & -9 & 0 & -1 & 0 & 3 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & -4 & 0 & -9 & 0 & -1 & 0 & 3 & 0 & 2
\end{array}\right]$$

By searching from the top to the bottom, we find that row 8, 9, 13, 14 and 15 are dependent. Of these, row 8, 9 and 15 is the largest set of primary dependent row with least order. The number of rows in this set is 3 which corresponds to the dimension of the null-space which means that we do not have to consider any other dependent rows. The dependencies in these three primary dependent rows can be described by

$$\widetilde{F} = \left[\begin{array}{rrrrr|rrrrr|rrrrr}
0 & 1 & 2 & -3 & -1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 1 & 2 & -2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
-3 & -5 & -6 & 9 & 9 & -10 & -1 & 0 & 0 & -1 & 1 & 1 & 0 & 0 & 1
\end{array}\right]$$

The corresponding polynomial matrix $F(s)$ in polynomial echelon form is

$$F(s) = \begin{bmatrix}
s & s+1 & s+2 & -3 & -1 \\
-2s-1 & 0 & 0 & s+1 & 2 \\
s^2-10s-3 & s^2-s-5 & -6 & 9 & s^2-s+9
\end{bmatrix}$$

which is also a minimal polynomial basis for the left null-space of $M(s)$. ■

Numerical Considerations

The two algorithms presented in this section have very different numerical properties. Although the algorithm based on Hermite form is easy to understand, no (to the author's knowledge) numerically stable algorithm exists. Simulations have shown that the algorithm to make the basis row-reduced, proposed in (Callier, 1985) and implemented in (Henrion et al., 1997), is numerically unstable.

On the other hand, the algorithm based on the polynomial echelon form is both fast and numerically stable. The critical step in the algorithm is the

search for primary dependent rows in the matrix $\widetilde{\mathcal{M}}$. The search for dependent rows can be performed by using a numerically stable projection algorithm described in (Chen, 1984), p. 546. First transform $\widetilde{\mathcal{M}}$ to lower triangular form by multiplication from the right with a matrix L. The matrix L is obtained by a series of numerically stable Householder transformations ((G.H. Golub, 1996), Chapter 5). Now the matrix that defines the dependent rows, is easily obtained by solving for A in the equation

$$A\widetilde{\mathcal{M}}L = 0$$

Since $\widetilde{\mathcal{M}}L$ is lower triangular, A can be obtained by straightforward, numerically stable substitutions (Chen, 1984). This algorithm is implemented in the Polynomial Toolbox, (Henrion et al., 1997).

7.2.4 Relation to Frequency Domain Approaches

A number of design methods described in literature are called *frequency domain methods* where the residual generators are designed with the help of different transfer matrix factorization techniques. This section discusses the relation between the minimal polynomial basis approach and these frequency domain methods. Examples of frequency domain methods are (Frank and Ding, 1994*a*) for the general case with disturbances and (Ding and Frank, 1990; Viswanadham, Taylor and Luce, 1987) in the non-disturbance case. These methods can be summarized as methods where the residual generator is parameterized as

$$\begin{aligned} r &= R(s)[\tilde{D}(s) \ -\tilde{N}(s)] \begin{pmatrix} y \\ u \end{pmatrix} \\ &= R(s)(\tilde{D}(s)y - \tilde{N}(s)u) \end{aligned} \tag{7.18}$$

where $\tilde{D}(s)$ and $\tilde{N}(s)$ form a left co-prime factorization of $G(s)$ over $\mathcal{RH}_\infty$, i.e. the space of stable real-rational transfer matrices. Note the close relationship with Equation (7.15) where the factorization is performed over polynomial matrices instead of over $\mathcal{RH}_\infty$.

Inserting (7.1) into Equation (7.18) and as before assuming $f = 0$, gives

$$r = R(s)\tilde{D}(s)H(s)d$$

Therefore to achieve disturbance decoupling, the parameterization transfer matrix $R(s)$, must be belong to the left null-space of $\tilde{D}(s)H(s)$, i.e.

$$R(s)\tilde{D}(s)H(s) = 0$$

Here, note the close connection with $\widetilde{M}_2(s)$ in (7.16). This solution however does not generally generate a residual generator of minimal order. In (Ding and Frank, 1990) and (Frank and Ding, 1994*a*), the co-prime factorization is performed via a minimal state-space realization of the complete system, including the disturbances as in equation (7.2). This results in $\tilde{D}(s)$ and $\tilde{N}(s)$ of

McMillan degree n that, in the general case, is larger than the lowest possible McMillan degree of a disturbance decoupling residual generator. Thus, to find a residual generator of minimal order or a basis of minimal order that spans all residual generators $Q(s) = R(s)[\tilde{D}(s) \ -\tilde{N}(s)]$, extra care is required since "excess" states need to be canceled. Note that the polynomial basis approach on the other hand, has no need for cancelations and is in this sense more elegant.

7.3 Maximum Row-Degree of the Basis

This section shows that a minimal polynomial basis for the left null-space of the matrix (7.8), has a maximum row-degree (or column-degree) less or equal to n_x, i.e. the dimension of the state controllable from $\begin{bmatrix} u^T & d^T \end{bmatrix}^T$. This is the result of Corollary 7.1, which is a direct consequence of Theorem 7.9, and both are presented below.

Related problems have been investigated in (Chow and Willsky, 1984) and (Gertler, Fang and Luo, 1990). In (Chow and Willsky, 1984), it was shown that, in the no-disturbance case, there exist a parity function of order $\leq n$. In (Gertler et al., 1990), it was shown that for a restricted class of disturbances, there exist a parity function of order $\leq n$. However the result of Corollary 7.1 is much stronger since it includes *arbitrary* disturbances and shows that there exist a *basis* in which the maximum row-degree is $\leq n_x$.

The result of Corollary 7.1 are important for at least three reasons:

- The parity functions obtained directly from the minimal basis, are in one sense the only ones needed. All other are filtered versions (i.e. linear combinations) of these parity functions. With this argument, Corollary 7.1 shows that we do not need to consider parity functions of order greater than n_x.

- When calculating a basis for the left-null space of $M(s)$ using the polynomial echelon form algorithm, the maximum row-degree of the basis is needed as an input to the algorithm, i.e. ν. To keep the computational load down it is important to have a ν as small as possible. Without the result of Corollary 7.1, we are forced to used the bound $\nu \leq (p-1)\deg M(s)$ (Henrion et al., 1997). Consider finding a basis for $\mathcal{N}_L(M_x(s))$. Then ν is chosen as

  $$\nu \leq (p-1)\deg M_x(s) = n_x + m - 1 \geq n_x$$

  This means that the bound n_x is tighter than $\nu \leq (p-1)\deg M(s)$. As will be seen in the upcoming sections, the number ν is, of the same reason, important also for the Chow-Willsky scheme.

- For the detectability analysis presented in Chapter 8, it is important to know ν. We will see that ν is needed explicitly in detectability conditions based on the Chow-Willsky scheme and implicitly in some other detectability conditions.

Next, Theorem 7.9 is presented:

Theorem 7.9 *A matrix whose rows form a minimal polynomial basis for* $\mathcal{N}_L(M_s(s))$ *has row-degrees* $\leq n$.

Note that the theorem does not require the pair $\{A_x, [B_{u,x}\ B_{d,x}]\}$ to be controllable. Instead $M_s(s)$ can be based on any realization, but the most interesting is of course to use it with $M_x(s)$ which represents a minimal realization. Before Theorem 7.9 can be proven, we need two lemmas.

Lemma 7.2 *Let* $P(s)$ *be a matrix with maximum row-degree 1. Then the maximum row-degree of a minimal polynomial basis for* $\mathcal{N}_L(P(s))$ *is less or equal to* $\operatorname{Rank} P(s)$.

Proof: The matrix $P(s)$ is a matrix pencil, i.e. $P(s) = sE + F$. By constant elementary row and column operations, $P(s)$ can be transformed to to Kronecker canonical form (Kailath, 1980). This means that there exists nonsingular matrices U and V such that

$$\bar{P}(s) = UP(s)V = \text{block diag } \{L_{\mu_1}, \ldots, L_{\mu_\alpha}, \widetilde{L}_{\nu_1}, \ldots, \widetilde{L}_{\nu_\beta}, sJ - I, sI - K\}$$

where $\widetilde{L}_{\nu_i}$ is a $(\nu_i + 1) \times \nu_i$ matrix of the form

$$\begin{bmatrix} s & & \\ -1 & \ddots & \\ & \ddots & s \\ & & -1 \end{bmatrix}$$

This matrix has rank ν_i, so it is obvious that $\nu_i \leq \operatorname{Rank} P(s)$. All other matrices, i.e. L_{μ_i}, $sJ-I$, and $sI-K$, have full row-rank. Therefore, a minimal polynomial basis for the left null-space of $\bar{P}(s)$, i.e. $\mathcal{N}_L(\bar{P}(s))$, is

$$N(s) = \begin{bmatrix} 0 & \ldots & 1 & s & \ldots & s^{\nu_1} & 0 & & & \ldots & & & 0 \\ & & & & & & \ddots & & & & & & \\ 0 & & & \ldots & & & 0 & 1 & \ldots & s^{\nu_\beta} & 0 & \ldots & 0 \end{bmatrix}$$

A basis for $\mathcal{N}_L(P(s))$ is then $N(s)U$. The matrix $N(s)$ is irreducible, row-reduced, and has maximum row-degree $\operatorname{Rank} P(s)$. Multiplication from the right with U doesn't change these facts and thus, $N(s)U$ is also irreducible, row-reduced. This means that $N(s)U$ is minimal polynomial basis with maximum row-degree $\operatorname{Rank} P(s)$. ∎

Lemma 7.3 *It holds that*

$$\operatorname{Rank} M_s(s) = \operatorname{Rank} \begin{bmatrix} D_d \\ B_d \end{bmatrix} + \operatorname{Rank} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix}$$

where the columns of N_{DB} *form a basis for the left null-space of* $[D_d^T\ B_d^T]^T$.

Proof: Without loss of generality, we can assume that $N_{DB}^T N_{DB} = I$. Since N_{DB} and $[D_d^T \ B_d^T]^T$ together span the whole space $\mathbb{R}^{m+n}$, there is a $Y(s)$ and an $X(s)$ such that

$$\begin{bmatrix} -C \\ sI - A \end{bmatrix} = N_{DB} Y(s) + \begin{bmatrix} D_d \\ B_d \end{bmatrix} X(s)$$

where

$$Y(s) = N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix}$$

Further we have that

$$\begin{aligned} \operatorname{Rank} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix} &= \operatorname{Rank} \begin{bmatrix} -C^T & sI - A^T \end{bmatrix} N_{DB} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix} \leq \\ &\leq \operatorname{Rank} N_{DB} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix} \leq \operatorname{Rank} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix} \end{aligned}$$

which means that it must hold that

$$\operatorname{Rank} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix} = \operatorname{Rank} N_{DB} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix} \tag{7.19}$$

Then

$$\begin{aligned} \operatorname{Rank} M_s(s) =& \operatorname{Rank} \begin{bmatrix} -C & D_d \\ sI - A & B_d \end{bmatrix} = \operatorname{Rank} \begin{bmatrix} N_{DB} Y(s) + \begin{bmatrix} D_d \\ B_d \end{bmatrix} X(s), \begin{bmatrix} D_d \\ B_d \end{bmatrix} \end{bmatrix} = \\ =& \operatorname{Rank} \begin{bmatrix} N_{DB} Y(s), \begin{bmatrix} D_d \\ B_d \end{bmatrix} \end{bmatrix} = \operatorname{Rank} N_{DB} Y(s) + \operatorname{Rank} \begin{bmatrix} D_d \\ B_d \end{bmatrix} = \\ =& \operatorname{Rank} N_{DB} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix} + \operatorname{Rank} \begin{bmatrix} D_d \\ B_d \end{bmatrix} = \\ =& \operatorname{Rank} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix} + \operatorname{Rank} \begin{bmatrix} D_d \\ B_d \end{bmatrix} \end{aligned}$$

where (7.19) has been used in the last step. ∎

Now return to the proof of Theorem 7.9.

Proof: Consider the matrix

$$M_s(s) = \begin{bmatrix} -C & D_d \\ sI - A & B_d \end{bmatrix}$$

and let the columns of N_{DB} be a basis for the left null-space of $[D_d^T \ B_d^T]^T$. Then we have that

$$N_{DB}^T M_s(s) = \begin{bmatrix} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix}, 0 \end{bmatrix} \tag{7.20}$$

The left part of the matrix (7.20) has rank $\leq n$. From Lemma 7.2 we know that a minimal polynomial basis for (7.20) has row degrees less or equal to n. Let the rows of a matrix $Q(s)$ form such a basis.

The basis N_{DB} has $m+n-\operatorname{Rank}[D_d^T\ B_d^T]$ columns. The left null-space of $N_{DB}^T M_s(s)$ has therefore the dimension

$$d = m + n - \operatorname{Rank}\begin{bmatrix} D_d \\ B_d \end{bmatrix} - \operatorname{Rank} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix}$$

This must also be the rank of $Q(s)$.

Now study the matrix $Q(s)N_{DB}^T$. Since $Q(s)$ is irreducible and N_{DB}^T has full row-rank, also the matrix $Q(s)N_{DB}^T$ must be irreducible. Since $Q(s)$ is row-reduced, it can be written $Q(s) = S(s)D_{hr} + L(s)$, where D_{hr} has full row-rank. Multiplication from the right with N_{DB}^T, which is also full row-rank, results in $D_{hr}N_{DB}^T$ which has also full row-rank. This implies that the matrix $Q(s)N_{DB}^T$ is row-reduced.

It must hold that $\operatorname{Rank} Q(s)N_{DB}^T = \operatorname{Rank} Q(s)$. By using Lemma 7.3, we know that the rank of $Q(s)N_{DB}^T$ is

$$\begin{aligned}\operatorname{Rank} Q(s)N_{DB}^T = \operatorname{Rank} Q(s) &= m + n - \operatorname{Rank}\begin{bmatrix} D_d \\ B_d \end{bmatrix} - \operatorname{Rank} N_{DB}^T \begin{bmatrix} -C \\ sI - A \end{bmatrix} = \\ &= m + n - \operatorname{Rank} M_s(s)\end{aligned}$$

All this implies that $Q(s)N_{DB}^T$ is a minimal polynomial basis for $\mathcal{N}_L(M_s(s))$. Further the row-degrees of $Q(s)N_{DB}^T$ is $\leq n$. Then since all minimal polynomial bases have the same set of row-degrees, it holds that all minimal polynomial bases of $Q(s)N_{DB}^T$ have row-degrees $\leq n$. ■

From Theorem 7.9, we now get the following result:

Corollary 7.1 *A matrix whose rows form a minimal polynomial basis for $\mathcal{N}_L(M(s))$ has row-degrees $\leq n_x$.*

Proof: According to Theorem 7.3, $W(s) = V(s)P_x$ is a minimal polynomial basis for $\mathcal{N}_L(M(s))$ if $V(s)$ is a minimal polynomial basis for $\mathcal{N}_L(M_x(s))$. Since we know from Theorem 7.9 that the maximum row-degree of $V(s)$ is n_x, then also the maximum row-degree of $W(s)$ is n_x. ■

7.4 The Chow-Willsky Scheme

The most well-known method for direct construction of polynomial parity functions was presented in (Chow and Willsky, 1984). This method is usually referred to as the Chow-Willsky scheme. In (Chow and Willsky, 1984), it was formulated for discrete systems but before that, similar ideas had been developed by Mironovskii (1980), who considered both discrete and continuous systems.

Based on the method in (Chow and Willsky, 1984), a number of extensions have been proposed. One important extension, provided by Frank (1990), includes also decoupling of disturbances and non-monitored faults into the design. Among other extensions is for example the handling of the case when perfect decoupling is not possible (Lou, Willsky and Verghese, 1986).

The Chow-Willsky scheme and its extensions have been extensively used in the literature, probably because of its simplicity compared to many other residual generator design methods. However, the Chow-Willsky scheme can for high order systems be numerically unstable, as will be explained in Section 7.5.2, and care should therefore be taken when practical residual generator design is considered.

In this section we will see that the original formulation of the Chow-Willsky scheme (and also its extensions) have several disadvantages. First, it is not able to generate all parity functions for some linear system. Second, the solution does not give a parity function of minimal order. However, by a stepwise improvement we will in this section show how the Chow-Willsky scheme can be modified so that these disadvantages dissappear. In Section 7.5.1, the Chow-Willsky scheme will be even further modified so that it generates a minimal polynomial basis in similarity with the minimal polynomial basis approach. Related results, valid for some special cases and showing a relation between parity functions and a polynomial-like method, were noted in (Massoumnia and Velde, 1988).

7.4.1 The Chow-Willsky Scheme Version I: the Original Solution

The following description of the Chow-Willsky scheme mainly follows (Frank, 1990), except for that the description here is formulated for the continuous case. However by replacing s by z (or the time-shift operator q) all formulas are valid also for the discrete case. The Chow-Willsky scheme assumes that the system model is given in the state-space form:

$$sx = Ax + B_u u + B_d d + B_f f \tag{7.21a}$$
$$y = Cx + D_u u + D_d d + D_f f \tag{7.21b}$$

Now by substituting (7.21a) into (7.21b), we can obtain sy as

$$\begin{aligned} sy &= Csx + D_u su + D_d sd + D_f sf = \\ &= CAx + CB_u u + D_u su + CB_d d + D_d sd + CB_f f + D_f sf \end{aligned}$$

By continuing in this fashion for $s^2y \ \ldots \ s^\rho y$, the following equation can be obtained:

$$Y(t) = Rx(t) + QU(t) + HV(t) + PF(t) \tag{7.22}$$

where Q is a lower triangular Toeplitz matrix describing the propagation of the input u through the system. Similarly, H and P describes the propagation

of the disturbance d and the fault f respectively. Written out, the matrices in (7.22) are

$$Y(t) = \begin{bmatrix} y(t) \\ sy(t) \\ \vdots \\ s^\rho y(t) \end{bmatrix} \qquad R = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^\rho \end{bmatrix}$$

$$Q = \begin{bmatrix} D_u & 0 & 0 & \dots \\ CB_u & D_u & 0 & \dots \\ \vdots & & \ddots & \\ CA^{\rho-1}B_u & \dots & CB_u & D_u \end{bmatrix} \qquad U(t) = \begin{bmatrix} u(t) \\ su(t) \\ \vdots \\ s^\rho u(t) \end{bmatrix}$$

$$H = \begin{bmatrix} D_d & 0 & 0 & \dots \\ CB_d & D_d & 0 & \dots \\ \vdots & & \ddots & \\ CA^{\rho-1}B_u & \dots & CB_d & D_d \end{bmatrix} \qquad V(t) = \begin{bmatrix} d(t) \\ sd(t) \\ \vdots \\ s^\rho d(t) \end{bmatrix}$$

$$P = \begin{bmatrix} D_f & 0 & 0 & \dots \\ CB_f & D_f & 0 & \dots \\ \vdots & & \ddots & \\ CA^{\rho-1}B_f & \dots & CB_f & D_f \end{bmatrix} \qquad F(t) = \begin{bmatrix} f(t) \\ sf(t) \\ \vdots \\ s^\rho f(t) \end{bmatrix}$$

The size of Y is $(\rho+1)m \times 1$, R is $(\rho+1)m \times n$, Q is $(\rho+1)m \times (\rho+1)k_u$, U is $(\rho+1)k_u \times 1$, H is $(\rho+1)m \times (\rho+1)k_d$, V is $(\rho+1)k_d \times 1$, P is $(\rho+1)m \times (\rho+1)k_f$, and F is $(\rho+1)k_f \times 1$. The constant ρ determines the maximum possible order of the parity function. The choice of ρ is discussed in Section 7.4.3.

Now, with a column vector w of length $(\rho+1)m$, a function $h(y,u)$ can be formed as

$$h(y,u) = w^T(Y - QU) \tag{7.23}$$

For later use, note that this function can also be written as

$$h(y,u) = w\begin{bmatrix} \Psi_m(s) & -Q\Psi_{k_u}(s) \end{bmatrix}\begin{bmatrix} y \\ u \end{bmatrix} \tag{7.24}$$

where

$$\Psi_m(s) = \begin{bmatrix} I_m \\ sI_m \\ \vdots \\ s^\rho I_m \end{bmatrix} \qquad \Psi_{k_u}(s) = \begin{bmatrix} I_{k_u} \\ sI_{k_u} \\ \vdots \\ s^\rho I_{k_u} \end{bmatrix}$$

Equation (7.22) implies that the following equality will hold:

$$h(y,u) = w^T(Rx + HV + PF) \tag{7.25}$$

If $h(y,u)$ is going to be a parity function, it must hold that it is zero in the fault free case and the disturbances must be decoupled. This is fulfilled if w satisfies

$$w^T [R\, H] = 0 \tag{7.26}$$

In other words, if w belongs to the left null-space of $[R\ H]$. For use in fault detection, it is also required that the parity function is non-zero in the case of faults. This is assured by letting

$$w^T P \neq 0 \tag{7.27}$$

In conclusion, using the Chow-Willsky scheme, a parity function is constructed by first setting up all the matrices in (7.22) and then finding a w such that (7.26) and (7.27) are fulfilled.

7.4.2 The Original Chow-Willsky Scheme is Not Universal

Following is an example showing that the Chow-Willsky scheme is not universal, i.e. there are cases in which it can not generate all possible parity functions. This happens when the system has dynamics controllable *only* from the fault.

Example 7.5

Consider a system described by the transfer functions

$$y_1 = \frac{1}{s-1}u + \frac{1}{s+1}f \qquad y_2 = \frac{1}{s-1}u + \frac{s+3}{s+1}f$$

and the realization

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} f \\ y &= \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} f \end{aligned}$$

Also consider the function

$$h = (1 - s + s^2)y_1 - s^2 y_2 + u \tag{7.28}$$

If y_1 and y_2 in (7.28) are substituted with their transfer functions we get

$$\begin{aligned} h = \frac{1}{s-1}\big((1-s+s^2) - s^2 + (s-1)\big)u + \\ + \frac{1}{s+1}\big((1-s+s^2) - s^2(s+3)\big)f = \frac{-s^3-2s^2-s+1}{s+1}f \end{aligned}$$

We see that h is zero in the fault free case and becomes non-zero when the fault occurs. Therefore the function (7.28) is, according to Definition 7.3, a parity function. With the matrices used in Equation (7.22), the parity function (7.28) can be written as

$$h = [1\ \ 0\ \ -1\ \ 0\ \ 1\ \ -1]\,(Y - QU) = w^T(Y - QU)$$

in which w is uniquely defined. With the realization above, the matrix R is

$$R = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & -1 \\ 1 & -2 \\ 1 & 1 \\ 1 & 2 \end{bmatrix}$$

The first column of R is orthogonal to w but not the second. This means that the parity function (7.28) can not be obtained from the Chow-Willsky scheme. ■

The problem in the previous example is the second column of R. This column originates from x_2, which is controllable only from the fault f. The problem is solved if we can relax the requirement that w must be orthogonal to the second column of R. This is the topic of the next section.

7.4.3 Chow-Willsky Scheme Version II: a Universal Solution

To make the Chow-Willsky scheme universal, we need to require that the realization is controllable from $[u^T \ d^T]^T$. If this requirement is not fulfilled, the system must be transformed to the realization (7.12). We can compare this with the state-space solution in the minimal polynomial basis approach, where we also had to require that the realization is controllable from $[u^T \ d^T]^T$.

Now assume that the realization is on the form (7.12). Then the matrix R can be partitioned into $R = [R_x \ R_z]$ where

$$R_x = \begin{bmatrix} C_x \\ C_x A_x \\ \vdots \\ C_x A_x^\rho \end{bmatrix}$$

This means that equation (7.25) can be written

$$h(y,u) = w^T(R_x x + R_z z + HV + PF)$$

As with the minimal polynomial approach, we need only to consider the fault free case, i.e. z can be assumed to be zero. Then a sufficient and necessary condition to make this expression a parity function is that w must satisfy

$$w^T[R_x \ H] = 0 \tag{7.29}$$

The first column R in Example 7.5 corresponds to R_x and the second column to R_z. Thus, if we had used the condition (7.29), the parity function (7.28) could have been generated by the Chow-Willsky scheme.

Replacing condition (7.26) with (7.29) results in a modified Chow-Willsky scheme which in this work is referred to as the Chow-Willsky scheme, version II. This version of the Chow-Willsky scheme is universal in the sense that it can generate all parity function up to order ρ. This fact is shown in the following theorem:

Theorem 7.10 *Consider the matrix $M(s)$ in (7.8). For each vector $F(s) \in \mathcal{N}_L(M(s))$ and with a row-degree $\leq \rho$, there is a vector w such that $F(s) = w^T [\Psi_m(s) \ -Q\Psi_{k_u}(s)]$ and $w^T[R_x \ H] = 0$.*

Proof: If $F(s) \in \mathcal{N}_L(M(s))$, then we know that for all inputs u and disturbances d, and in the fault free case, it holds that

$$h = F(s) \begin{bmatrix} G(s) & H(s) \\ I_{k_u} & 0 \end{bmatrix} \begin{bmatrix} u \\ d \end{bmatrix} = [F_1(s) \ F_2(s)] \begin{bmatrix} y \\ u \end{bmatrix} = \tag{7.30}$$

$$= [\widetilde{F}_1 \Psi_m(s) \ \widetilde{F}_2 \Psi_{k_u}(s)] \begin{bmatrix} y \\ u \end{bmatrix} = \widetilde{F} \begin{bmatrix} Y \\ U \end{bmatrix} = 0 \tag{7.31}$$

where $\widetilde{F}_i$ is the coefficient matrix of $F_i(s)$. By using (7.22), (7.31) can be rewritten as

$$\begin{bmatrix} \widetilde{F}_1 \ \widetilde{F}_2 \end{bmatrix} \begin{bmatrix} Y \\ U \end{bmatrix} = \begin{bmatrix} \widetilde{F}_1 \ \widetilde{F}_2 \end{bmatrix} \begin{bmatrix} R_x x + QU + HV \\ U \end{bmatrix} =$$
$$= \widetilde{F}_1(R_x x + QU + HV) + \widetilde{F}_2 U =$$
$$= \widetilde{F}_1 R_x x + \widetilde{F}_1 HV + (\widetilde{F}_1 Q + \widetilde{F}_2)U = 0$$

Since x is controllable from inputs and disturbances, this equation must hold for all x, all U, and all V, which implies $\widetilde{F}_1 R_x = 0$, $\widetilde{F}_1 H = 0$, and $\widetilde{F}_1 Q + \widetilde{F}_2 = 0$.

Now choose w as $w^T = \widetilde{F}_1$, which is clearly a possible choice since we know that $\widetilde{F}_1 [R_x \ H] = 0$. This together with the fact $\widetilde{F}_2 = -\widetilde{F}_1 Q = -w^T Q$, implies that

$$w^T[\Psi_m(s) \ -Q\Psi_{k_u}(s)] = [\widetilde{F}_1 \Psi_m(s) \ -\widetilde{F}_1 Q \Psi_{k_u}(s)] =$$
$$= [\widetilde{F}_1 \Psi_m(s) \ \widetilde{F}_2 \Psi_{k_u}(s)] = F(s)$$

which proves the theorem. ∎

The Chow-Willsky scheme, version II, implies that all possible parity equations are parameterized as follows. Let $N_{R_x H}$ denote a matrix of dimension $\eta \times (\rho + 1)$, and let its rows form a basis for the η-dimensional left null-space of the matrix $[R_x \ H]$. Then all parity functions up to order ρ can be obtained by in (7.24) selecting w as $w^T = \gamma N_{R_x H}$, where γ is an arbitrary row vector of dimension η. Thus, a complete parameterization of all decoupling row-vectors $F(s)$ of maximum row-degree ρ (i.e. all parity functions up to order ρ), is

$$F(s) = \gamma N_{R_x H} [\Psi_m(s) \ -Q\Psi_{k_u}(s)] \tag{7.32}$$

This expression should be compared to (7.10) which was also complete parameterization of all decoupling row vectors $F(s)$. The difference is that the parameter γ in (7.32) is constant while the parameter $\phi(s)$ in (7.10) is polynomial. Also, (7.10) covers arbitrary row-degree while (7.32) can only handle row-degrees up to ρ.

In Section 7.3, we argued that only parity functions up to order n_x need to be found. The reason is that any other parity function, of arbitrary order, is a filtered version of a parity function of an order less or equal to n_x. All this is a consequence of Corollary 7.1. If this reasoning is applied to the Chow-Willsky scheme version II, we see that it is sufficient to chose $\rho = n_x$. In other words, $\rho = n_x$ is sufficient to generate a basis for the left null-space of $M(s)$ in (7.8).

As was said in the end of Section 7.2.1, the minimal polynomial basis approach implies that parity functions, and therefore also residual generators, of minimal order are explicitly found. This is not the case with the Chow-Willsky scheme (version I or version II). The reason is that the only requirement of the vector w, or alternatively the basis $N_{R_x H}$, is that $w^T[R_x \ H] = 0$. This means that in general, the parity function will be of order ρ. However, we can place further constraints on the vector w such that minimal order parity functions are obtained and this is done next.

7.4.4 Chow-Willsky Scheme Version III: a Minimal Solution

From a numerical perspective, the preferred algorithm for finding the null space to a general constant matrix is often the SVD (Singular Value Decomposition). As was said above, this does not in general imply that the parity functions get minimal order. However, a minimal solution is obtained if w (or $N_{R_x H}$) is instead found with the *row-search* algorithm, shortly described in Section 7.2.3. If we search from top-to-bottom in $[R_x \ H]$ for dependent rows, the matrix describing these dependencies is then a basis for the left null-space of $[R_x \ H]$. Since the search is from the top to the bottom, we realize from the structure of (7.24) that a minimal order parity function is obtained. To explicitly use this procedure for finding w (or $N_{R_x H}$) will here be called the Chow-Willsky scheme version III. Note that the minimal order parity function can also be found by using the Chow-Willsky scheme version II with $\rho = 0$ and then incrementally trying larger and larger values of ρ.

In our stepwise improvement of the Chow-Willsky scheme, we have now arrived in an algorithm which can generate a matrix $F_{CW}(s)$ as

$$F_{CW}(s) = N_{R_x H} \left[\Psi_m(s) \ \ - Q\Psi_{k_u}(s)\right] \tag{7.33}$$

This matrix $F_{CW}(s)$ will span the left null-space of $[R_x \ H]$ and it has a certain minimality property. However it is still not a basis since it in general have more than $m - k_d$ rows, which was the dimension of $\mathcal{N}_L(M(s))$ according to (7.9).

7.5 Connection Between the Minimal Polynomial Basis Approach and the Chow-Willsky Scheme

Even though many pieces of the relation between the minimal polynomial basis approach and the Chow-Willsky scheme have already been discussed in the previous section, there are some pieces left. Here we will investigate more thoroughly the properties of the matrix $F_{CW}(s)$ defined in the previous section. The result of this investigation is that the Chow-Willsky scheme can in fact be modified even further so that the matrix $F_{CW}(s)$ becomes a minimal polynomial basis for $\mathcal{N}_L(M(s))$.

We start by considering the equation

$$F(s)\begin{bmatrix} I_{k_u} & 0 \\ G(s) & H(s) \end{bmatrix} = F(s)M'(s) = 0 \tag{7.34}$$

where $F(s)$ is here a minimal polynomial basis for the left null-space of $M'(s)$. Note that we have switched the lower and upper part of this matrix, compared to $M(s)$ in (7.8). This will lead to simplifications later during the investigation.

Next we realize from Section 7.2.3 that solving (7.34) is equivalent to solving

$$[F_0\ F_1 \dots F_\nu]\begin{bmatrix} I_{k_u} & 0 \\ G(s) & H(s) \\ sI_{k_u} & 0 \\ sG(s) & sH(s) \\ \vdots & \vdots \\ s^\nu I_{k_u} & 0 \\ s^\nu G(s) & s^\nu H(s) \end{bmatrix} = 0 \tag{7.35}$$

where again ν is not known *a priori*. The goal now, is to show that the minimal polynomial basis $F(s)$ can in fact be obtained by searching for the largest and uppermost set of primary dependent rows in $[R_x\ H]$. For this we will use three lemmas.

Lemma 7.4 *For any vector or matrix $\widetilde{F} = [F_0\ F_1 \dots F_\nu]$, it holds that equation (7.35) is fulfilled if and only if*

$$[F_0\ F_1 \dots F_\nu]\left[\begin{array}{ccc} 0 & [I_{k_u}\ 0 \dots 0] & 0 \\ R_0 & Q_0 & H_0 \\ \hline 0 & [0\ I_{k_u}\ \dots 0] & 0 \\ R_1 & Q_1 & H_1 \\ \hline \vdots & \vdots & \vdots \\ 0 & [0 \dots 0\ I_{k_u}] & 0 \\ R_\nu & Q_\nu & H_\nu \end{array}\right] = 0 \tag{7.36}$$

Proof: Let us first study the rows of (7.35) containing $G(s)$ and $H(s)$. The transfer matrix $[G(s)\ H(s)]$ can be written

$$[G(s)\ H(s)] = C(sI - A)^{-1}[B_u\ B_d] + [D_u\ D_d] = \\ = \sum_{i=1}^{\infty} CA^{i-1}[B_u\ B_d]s^{-i} + [D_u\ D_d] \quad (7.37)$$

where $\{A, [B_u\ B_d], C, [D_u\ D_d]\}$ is any controllable realization of the transfer function $[G(s)\ H(s)]$. Define

$$X(s) = \sum_{i=1}^{\infty} s^{-i} A^{i-1}[B_u\ B_d] \quad (7.38)$$

Now note that

$$s^j \sum_{i=1}^{\infty} s^{-i} A^{i-1} = \sum_{i=1}^{j} s^{j-i} A^{i-1} + \sum_{i=j+1}^{\infty} s^{j-i} A^{i-1} = \\ = \sum_{i=0}^{j-1} s^{j-i-1} A^i + \sum_{i=1}^{\infty} s^{-i} A^{i-1+j} = \sum_{i=0}^{j-1} s^i A^{j-1-i} + A^j \sum_{i=1}^{\infty} s^{-i} A^{i-1} \quad (7.39)$$

By using both (7.37), (7.38) and (7.39) we can derive the following relation:

$$s^j [G(s)\ H(s)] = CA^j X(s) + C \sum_{i=0}^{j-1} s^i A^{j-1-i}[B_u\ B_d] + [D_u\ D_d]s^j \quad (7.40)$$

This formula implies that we can write

$$\begin{bmatrix} I_m \\ sI_m \\ \vdots \\ s^\nu I_m \end{bmatrix} [G(s)\ H(s)] = \\ = \begin{bmatrix} C \\ \vdots \\ CA^\nu \end{bmatrix} X(s) + \begin{bmatrix} [D_u\ D_d] \\ C[B_u\ B_d] + [D_u\ D_d]s \\ \vdots \\ CA^{\nu-1}[B_u\ B_d] + CA^{\nu-2}[B_u\ B_d]s + \cdots + [D_u\ D_d]s^\nu \end{bmatrix} = \\ = RX(s) + [Q\ H] \begin{bmatrix} \Psi_{k_u}(s) & 0 \\ 0 & \Psi_{k_d}(s) \end{bmatrix} = [R\ Q\ H] \begin{bmatrix} X(s) \\ \begin{bmatrix} \Psi_{k_u}(s) & 0 \\ 0 & \Psi_{k_d}(s) \end{bmatrix} \end{bmatrix} \quad (7.41)$$

Now Equation (7.35) can be rewritten

$$[F_0\ F_1 \dots F_\nu] \begin{bmatrix} 0 & [I_{k_u}\ 0 \dots 0] & 0 \\ R_0 & Q_0 & H_0 \\ 0 & [0\ I_{k_u}\ \dots 0] & 0 \\ R_1 & Q_1 & H_1 \\ \vdots & \vdots & \vdots \\ 0 & [0 \dots 0\ I_{k_u}] & 0 \\ R_\nu & Q_\nu & H_\nu \end{bmatrix} \begin{bmatrix} X(s) \\ \begin{bmatrix} \Psi_{k_u}(s) & 0 \\ 0 & \Psi_{k_d}(s) \end{bmatrix} \end{bmatrix} = 0 \qquad (7.42)$$

where R_i, Q_i, and H_i denotes the i:th block of m rows in each matrix R, Q, and H respectively. By studying the definitions of $X(s)$, $\Psi_{k_u}(s)$, and $\Psi_{k_d}(s)$, it can be realized that the coefficient matrix for the rightmost matrix in (7.42) becomes

$$\begin{bmatrix} \dots A^2[B_u\ B_d] & A[B_u\ B_d] & [B_u\ B_d] & 0 & \dots & 0 \\ & & & I_{k_u+k_d} & & \\ & & & & \ddots & \\ & & & & & I_{k_u+k_d} \end{bmatrix} \qquad (7.43)$$

Note that this matrix has an infinite number of columns. This means that the coefficient matrix for the right matrix of (7.35) becomes

$$\begin{bmatrix} 0 & [I_{k_u}\ 0 \dots 0] & 0 \\ R_0 & Q_0 & H_0 \\ 0 & [0\ I_{k_u}\ \dots 0] & 0 \\ R_1 & Q_1 & H_1 \\ \vdots & \vdots & \vdots \\ 0 & [0 \dots 0\ I_{k_u}] & 0 \\ R_\nu & Q_\nu & H_\nu \end{bmatrix} \begin{bmatrix} \dots A^2[B_u\ B_d] & A[B_u\ B_d] & [B_u\ B_d] & 0 & \dots & 0 \\ & & & I_{k_u+k_d} & & \\ & & & & \ddots & \\ & & & & & I_{k_u+k_d} \end{bmatrix} \qquad (7.44)$$

Since the realization is controllable, the matrix $[A^{n-1}[B_u\ B_d] \dots [B_u\ B_d]]$ has full row-rank and therefore also the matrix (7.43). This means that (7.42) implies (7.36). The converse follows trivially, and since (7.42) is equivalent to (7.35), the lemma is proven. ∎

From Section 7.2.3 and Theorem 7.8, we know that a minimal polynomial basis for the left null-space of the matrix $M'(s)$ in (7.34) can be obtained by searching for the largest and uppermost set of primary dependent rows in the right matrix of (7.35) (or equivalently in the coefficient matrix (7.44)). Lemma 7.4 implies that we can equally well perform the search for primary dependent rows in the

matrix

$$\left[\begin{array}{ccc} 0 & [I_{k_u}\ 0 \dots 0] & 0 \\ R_0 & Q_0 & H_0 \\ \hline 0 & [0\ I_{k_u}\ \dots 0] & 0 \\ R_1 & Q_1 & H_1 \\ \hline \vdots & \vdots & \vdots \\ 0 & [0 \dots 0\ I_{k_u}] & 0 \\ R_\nu & Q_\nu & H_\nu \end{array}\right] \tag{7.45}$$

It will be shown that this row search can be simplified even further and for this we first need the following lemma.

Lemma 7.5 *There exists a vector* $\bar{t} = [t_0 \dots t_l] \neq 0$ *and*

$$[t_0 \dots t_l] \begin{bmatrix} R_0 & H_0 \\ \vdots & \vdots \\ R_l & H_l \end{bmatrix} = 0 \tag{7.46}$$

if and only if there exists a vector $\bar{t}' = [v_0\ t_0 \dots v_l\ t_l] \neq 0$ *and*

$$[v_0\ t_0\ v_1\ t_1\ \dots\ v_l\ t_l] \left[\begin{array}{ccc} 0 & [I_{k_u}\ 0 \dots 0] & 0 \\ R_0 & Q_0 & H_0 \\ 0 & [0\ I_{k_u}\ \dots 0] & 0 \\ R_1 & Q_1 & H_1 \\ \vdots & \vdots & \vdots \\ 0 & [0 \dots 0\ I_{k_u}] & 0 \\ R_l & Q_l & H_l \end{array}\right] = 0 \tag{7.47}$$

where

$$v_i = -[t_i \dots t_l] \begin{bmatrix} D \\ CB_u \\ \vdots \\ CA^{l-i-1}B \end{bmatrix}$$

Proof: The *only-if* part of the proof is realized by inspection of the definition of v_i and the equation (7.47).

For the *if* part, assume the specific case $l = 2$, and study the matrix (7.45), which becomes

$$\begin{bmatrix} 0 & I_{k_u} & 0 & 0 & 0 & 0 & 0 \\ C & D_u & 0 & 0 & D_d & 0 & 0 \\ 0 & 0 & I_{k_u} & 0 & 0 & 0 & 0 \\ CA & CB_u & D_u & 0 & CB_d & D_d & 0 \\ 0 & 0 & 0 & I_{k_u} & 0 & 0 & 0 \\ CA^2 & CAB_u & CB_u & D_u & CAB_d & CB_d & D_d \end{bmatrix} \tag{7.48}$$

From this example it is obvious that the elements t_i can not be zero. This is enough to prove that (7.46) holds and that $\bar{t} \neq 0$. ■

The next lemma is the last needed to prove Theorem 7.11, which will tell us how to find a minimal polynomial basis with the Chow-Willsky scheme.

Lemma 7.6 *There is a one-to-one correspondence between the dependent rows, in order top-to-bottom, of the matrix (7.45), and the dependent rows of the matrix* $[R\ H]$*. That is, the row for the* k*:th output in the* l*:th block of* $[R\ H]$ *is a dependent row if and only if the row for the* k*:th output in the* l*:th block of (7.45) is a dependent row.*

Proof: Consider a dependent row, in order top-to-bottom, in $[R\ H]$ and assume it is in the $l+1$:th block of rows. Then let the vector $[t_0 \dots t_l]$ describe this dependency. Then from Lemma 7.5, we know that (7.47) is fulfilled. This further means that the the corresponding row in the matrix (7.45) must also be a dependent row.

For the converse, assume the specific case $l = 2$, and study the matrix (7.45) which become (7.48). It is seen that it generally must hold that all dependent rows in the matrix (7.45) must occur in the rows starting with CA^i. By again using Lemma 7.5, it is seen that a dependent row in the matrix (7.45) directly implies that the corresponding row in $[R_x\ H]$ also must be dependent. ■

Note that the primary dependent rows are a subset of the dependent rows. Therefore, Lemma 7.6 shows that the search for primary dependent rows in the right matrix of (7.35) can be performed by a row-search in the much simpler matrix $[R\ H]$, which can be recognized from the Chow-Willsky scheme.

7.5.1 Chow-Willsky Scheme Version IV: a Polynomial Basis Solution

All results reached so far are summarized in the following theorem:

Theorem 7.11 *Let* W *define the largest and uppermost set of primary dependent rows in* $[R_x\ H]$*. Then* $F_{CW}(s) = W[\Psi_m(s) \quad -Q\Psi_{k_u}(s)]$ *is a minimal polynomial basis for the left null-space of*

$$M(s) = \begin{bmatrix} G(s) & H(s) \\ I_{k_u} & 0 \end{bmatrix}$$

Proof: Let W define the largest and uppermost set of primary dependent rows in $[R_x\ H]$. Then according to Lemma 7.6, this uniquely identifies the largest and uppermost set of primary dependent rows in also (7.45). From Theorem 7.8 and Lemma 7.4, we realize that this gives a minimal polynomial basis for

$$\begin{bmatrix} I_{k_u} & 0 \\ G(s) & H(s) \end{bmatrix}$$

From Lemma 7.5, we see that each row-vector $f(s)$ in the polynomial basis, can be written as

$$f(s) = [v_0\ t_0\ \ldots\ v_l\ t_l] \begin{bmatrix} I_{k_u} & 0 \\ 0 & I_m \\ sI_{k_u} & 0 \\ 0 & sI_m \\ \vdots & \vdots \\ s^l I_{k_u} & 0 \\ 0 & s^l I_m \end{bmatrix} = [t_0 \ldots t_l] \begin{bmatrix} -Q \begin{bmatrix} I_{k_u} \\ sI_{k_u} \\ \vdots \\ s^l I_{k_u} \end{bmatrix} & \begin{bmatrix} I_m \\ sI_m \\ \vdots \\ s^l I_m \end{bmatrix} \end{bmatrix}$$

Note that the second equality follows from the definition of v_i in Lemma 7.5. Then a basis for $\mathcal{N}_L(M(s))$ is trivially $F_{CW}(s) = W[\Psi_m(s) \quad -Q\Psi_{k_u}(s)]$. ∎

From Theorem 7.11, we realize that an alternative to searching for primary dependent rows in $\tilde{\mathcal{M}}$, a minimal polynomial basis can be obtained by searching for primary dependent rows in the matrix $[R_x\ H]$. This means that we now know how to use the Chow-Willsky scheme to generate a minimal polynomial basis for $\mathcal{N}_L(M(s))$. This final modification of the Chow-Willsky scheme becomes version IV.

The next theorem answers the question of what happens when the primary dependent rows are searched in the matrix $[R\ H]$ instead of $[R_x\ H]$. This result is of minor importance here but will be used to derive a detectability criterion in Chapter 8. However, note that to use $[R\ H]$ instead of $[R_x\ H]$ has exactly the same effect as to use a realization not controllable from $[u^T\ d^T]^T$, in the state-space solution of the minimal polynomial basis approach. The following Theorem 7.12 should be compared with Theorem 7.4.

Theorem 7.12 *Let W define the largest and uppermost set of primary dependent rows of $[R\ H]$. Then $F(s) = W[\Psi_m(s) \quad -Q\Psi_{k_u}(s)]$ is a polynomial basis (not necessarily irreducible) for the left null-space of*

$$M(s) = \begin{bmatrix} G(s) & H(s) \\ I_{k_u} & 0 \end{bmatrix}$$

Before this theorem can be proven, we need a lemma:

Lemma 7.7 *Consider the matrix*

$$[R_x\ H] = \begin{bmatrix} C_x & D_d & & \\ C_x A_x & C_x B_{d,x} & D_d & \\ \vdots & \vdots & & \ddots \\ C_x A_x^\rho & C_x A_x^{\rho-1} B_{d,x} & \ldots & D_d \end{bmatrix} \tag{7.49}$$

If the i:th row in the last block of this matrix is dependent then the i:th row of

the last block in the following matrix is also dependent:

$$[R\ H] = \begin{bmatrix} C & D_d & & \\ CA & CB_d & D_d & \\ \vdots & \vdots & & \ddots \\ CA^{\rho+n_z} & CA^{\rho+n_z-1}B_d & \dots & D_d \end{bmatrix} \tag{7.50}$$

Proof: First realize that $C_x A_x^i B_{d,x} = CA^i B_d$ for all $i \geq 0$. Then the fact that the i:th row in the last block of the matrix (7.49) is dependent, means that there is a vector $\bar{t} = [t_1 \dots t_{\rho+1}]$, where $t_{\rho+1} = [t_{\rho+1,1}, \dots t_{\rho+1,i-1}, 1, 0, \dots 0]$, such that

$$\begin{aligned} t_1 C_x + t_2 C_x A_x + \cdots + t_{\rho+1} C_x A_x^\rho &= 0 \\ t_1 D_d + t_2 CB_d + \cdots + t_{\rho+1} CA^{\rho-1} B_d &= 0 \\ &\vdots \\ t_\rho D_d + t_{\rho+1} CB_d &= 0 \\ t_{\rho+1} D_d &= 0 \end{aligned} \tag{7.51}$$

Study the equations containing D_d. All terms in these equations, except $t_i D_d$, have a B_d multiplied from the right. This means that the rows of B_d must span all $t_i D$, $i = 1, \dots \rho + 1$. Therefore there exists a matrix D_B so that

$$t_i D_d = t_i D_B B_d$$

for $i = 1, \dots \rho + 1$. Equations (7.51) can now be rewritten as

$$\begin{aligned} t_1 C_x + t_2 C_x A_x + \cdots + t_{\rho+1} C_x A_x^\rho &= 0 \\ t_1 D_B B_d + t_2 CB_d + \cdots + t_{\rho+1} CA^{\rho-1} B_d &= 0 \\ &\vdots \\ t_\rho D_B B_d + t_{\rho+1} CB_d &= 0 \\ t_{\rho+1} D_B B_d &= 0 \end{aligned} \tag{7.52}$$

Let the rows of a matrix N_x be a basis for the left null-space of $B_{d,x}$ and define $N = [N_x\ 0]$ and $M = [0\ I]$. Then an equivalent description of Equations (7.52) is that there exists f_i:s and g_i:s so that

$$\begin{aligned} t_1 C + t_2 CA + \cdots + t_{\rho+1} CA^\rho + g_0 M &= 0 \\ t_1 D_B + t_2 C + \cdots + t_{\rho+1} CA^{\rho-1} + f_1 N + g_1 M &= 0 \\ &\vdots \\ t_\rho D_B + t_{\rho+1} C + f_\rho N + g_\rho M &= 0 \\ t_{\rho+1} D_B + f_{\rho+1} N + g_{\rho+1} M &= 0 \end{aligned} \tag{7.53}$$

By multiplying the first equation with A from the right, different number of times, we can obtain the equations:

$$\begin{aligned} t_1CA^{n_z} + t_2CA^{n_z+1} + \cdots + t_{\rho+1}CA^{\rho+n_z} + g_0MA^{n_z} &= 0 \\ &\vdots \\ t_1CA + t_2CA^2 + \cdots + t_{\rho+1}CA^{\rho+1} + g_0MA &= 0 \end{aligned} \tag{7.54}$$

Next, note that

$$MA^i = [0\ I] \begin{bmatrix} A_x & A_{12} \\ 0 & A_z \end{bmatrix}^i = [0\ A_z^i] = A_zM$$

By using this expression and putting together the equations (7.53) and (7.54), we arrive at

$$\begin{aligned} t_1CA^{n_z} + t_2CA^{n_z+1} + \cdots + t_{\rho+1}CA^{\rho+n_z} + g_0A_z^{n_z}M &= 0 \\ &\vdots \\ t_1CA + t_2CA^2 + \cdots + t_{\rho+1}CA^{\rho+1} + g_0A_zM &= 0 \\ t_1C + t_2CA + \cdots + t_{\rho+1}CA^{\rho} + g_0M &= 0 \\ t_1D_B + t_2C + \cdots + t_{\rho+1}CA^{\rho-1} + f_1N + g_1M &= 0 \\ &\vdots \\ t_\rho D_B + t_{\rho+1}C + f_\rho N + g_\rho M &= 0 \\ t_{\rho+1}D_B + f_{\rho+1}N + g_{\rho+1}M &= 0 \end{aligned} \tag{7.55}$$

Now denote these equations with $\Phi_{-n_z}, \ldots, \Phi_{\rho+1}$, from top to bottom. Also define all Φ_i, $i > \rho + 1$, as a notation for the equation

$$0 = 0$$

Let the coefficients $a_{n_z-1}\ \ldots\ a_0$ be the coefficients in the characteristic polynomial. Then according to Cayley-Hamilton theorem,

$$A_z^{n_z} = a_{n_z-1}A^{n_z-1} + \cdots + a_1A^1 + a_0I$$

A new set of equations can be obtained as

$$\begin{bmatrix} 1 & -a_{n_z-1} & \ldots & -a_0 \end{bmatrix} \begin{bmatrix} \Phi_{-n_z} \\ \vdots \\ \Phi_0 \end{bmatrix}$$

$$\vdots \tag{7.56}$$

$$\begin{bmatrix} 1 & -a_{n_z-1} & \ldots & -a_0 \end{bmatrix} \begin{bmatrix} \Phi_{\rho+1} \\ \vdots \\ \Phi_{\rho+1+n_z} \end{bmatrix}$$

Introduce the notation

$$t'_i = [1 \ -a_{n_z-1} \ \dots \ -a_0] \begin{bmatrix} t_i \\ \vdots \\ t_{i+n_z} \end{bmatrix}$$

$$f'_i = [1 \ -a_{n_z-1} \ \dots \ -a_0] \begin{bmatrix} f_i \\ \vdots \\ f_{i+n_z} \end{bmatrix} \qquad g'_i = [1 \ -a_{n_z-1} \cdots -a_0] \begin{bmatrix} g_i \\ \vdots \\ g_{i+n_z} \end{bmatrix}$$

and let $t_i = 0$ and $f_i = 0$ for $i < 1$ and $i > \rho + 1$. Further let $g_i = g_0 A_z^{-i}$ for $i \leq 0$ and $g_i = 0$ for $i > \rho + 1$. Note these definitions imply that $t'_{\rho+1} = t_{\rho+1}$, $f'_{\rho+1} = f_{\rho+1}$, and $g'_{\rho+1} = g_{\rho+1}$.

Now the equations (7.56) can be written as

$$\begin{aligned}
t'_{-n_z+1}C + t'_{-n_z+2}CA + \cdots + t_{\rho+1}CA^{\rho+n_z} + g'_{-n_z}M &= 0 \\
t'_{-n_z+1}D_B + t'_{-n_z+2}C + \cdots + t_{\rho+1}CA^{\rho+n_z-1} + f'_{-n_z+1}N + g'_{-n_z+1}M &= 0 \\
&\vdots \\
t'_1 D_B + t'_2 C + \cdots + t_{\rho+1}CA^{\rho-1} + f'_1 N + g'_1 M &= 0 \\
&\vdots \\
t'_\rho D_B + t_{\rho+1}C + f'_\rho N + g'_\rho M &= 0 \\
t_{\rho+1}D_B + f_{\rho+1}N + g_{\rho+1}M &= 0
\end{aligned} \tag{7.57}$$

Note that

$$g'_{-n_z} = [1 \ -a_{n_z-1} \ \dots \ -a_0] \begin{bmatrix} g_{-n_z} \\ g_{-n_z+1} \\ \vdots \\ g_0 \end{bmatrix} = [1 \ -a_{n_z-1} \ \dots \ -a_0] \begin{bmatrix} g_0 A_z^{n_z} \\ g_0 A_z^{n_z-1} \\ \vdots \\ g_0 \end{bmatrix} =$$
$$= g_0(A_z^{n_z} - a_{n_z-1}A_z^{n_z-1} \cdots - a_0 I) = 0$$

Finally multiply all but the first of the equations (7.57) with B_d from the right. This will result in the equations

$$\begin{aligned}
t'_{-n_z+1}C + t'_{-n_z+2}CA + \cdots + t_{\rho+1}CA^{\rho+n_z} &= 0 \\
t'_{-n_z+1}D + t'_{-n_z+2}CB + \cdots + t_{\rho+1}CA^{\rho+n_z-1}B &= 0 \\
&\vdots \\
t'_\rho D + t_{\rho+1}CB &= 0 \\
t_{\rho+1}D &= 0
\end{aligned}$$

Note that the vector $t_{\rho+1}$ is the same here as in (7.51). This result is equivalent to that the i:th row of the last block of the matrix (7.50) is dependent, which ends the proof. ■

Now return to the proof of Theorem 7.12:

Proof: Introduce the notation $[R\ H]_{\rho=n}$, meaning that the matrix $[R\ H]$ is defined by using $\rho = n$. Lemma 7.7 says that if the i:th row in some block of $[R_x\ H]_{\rho=n_x}$ is dependent, then the i:th row in some block of $[R\ H]_{\rho=n}$ is also dependent. This means that a set of primary dependent rows of $[R\ H]_{\rho=n}$, of largest possible size, consists of the same number of rows as a set of primary dependent rows of $[R_x\ H]_{\rho=n}$, of largest possible size.

Assume now that W defines a set of primary dependent rows of $[R\ H]_{\rho=n}$, of largest possible size. The matrix $[R\ H]$ can also be written $[R_x\ R_z\ H]$. This means that the row indices, defining the set of primary dependent rows in $[R_x\ R_z\ H]$, also define a set of primary dependent rows in $[R_x\ H]$. It is important to note that there is no guarantee that this set is the uppermost.

Now Lemma 7.6 implies that we also have found a set of primary dependent rows of (7.45), of largest possible size. Note that neither this set is the uppermost. Then by using the same reasoning as in the proof of Theorem 7.11, we can conclude that $F(s) = W[\Psi_m(s) \quad -Q\Psi_{k_u}(s)]$ is a polynomial basis for the left null-space of $M(s)$. However, this time we used a set of not uppermost primary dependent rows, which according to Theorem 7.8 means that the basis will not be irreducible.

■

7.5.2 Numerical Properties of the Chow-Willsky Scheme

We have now shown that algebraically, the Chow-Willsky scheme version IV, is equivalent to the minimal polynomial basis approach. However, from a numerical perspective, the Chow-Willsky scheme is not as good as the minimal polynomial basis approach. The reason is that, for anything but small ρ, the matrix $[R_x\ H]$ will have high powers of A. It is likely that this results in that $[R_x\ H]$ becomes ill-conditioned. Thus to find the left null-space of $[R_x\ H]$ can imply severe numerical problems. The minimal polynomial basis approach does not have these problems of high power of A or any other term. This difference is highlighted in (Frisk, 1998), where both the Chow-Willsky scheme and the minimal polynomial basis approach are applied to the problem of designing polynomial parity functions for a turbo-jet aircraft-engine. The Chow-Willsky scheme fails because of numerical problems, while the minimal polynomial basis approach, manage to generate a basis for all parity functions.

7.6 Design Example

This model, taken from (Maciejowski, 1989), represents a linearized model of vertical-plane dynamics of an aircraft. The inputs and outputs of the model are

	Inputs		Outputs
u_1:	spoiler angle [tenth of a degree]	y_1:	relative altitude [m]
u_2:	forward acceleration [ms^{-2}]	y_2:	forward speed [ms^{-1}]
u_3:	elevator angle [degrees]	y_3:	Pitch angle [degrees]

The model has state-space matrices:

$$A = \begin{bmatrix} 0 & 0 & 1.132 & 0 & -1 \\ 0 & -0.0538 & -0.1712 & 0 & 0.0705 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0.0485 & 0 & -0.8556 & -1.013 \\ 0 & -0.2909 & 0 & 1.0532 & -0.6859 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -0.12 & 1 & 0 \\ 0 & 0 & 0 \\ 4.419 & 0 & -1.665 \\ 1.575 & 0 & -0.0732 \end{bmatrix}$$

$$C = [I_3 \ 0] \qquad D = 0_{3\times 3}$$

Suppose the faults of interest are three sensor-faults (denoted f_1, f_2, and f_3), and two actuator-faults (denoted f_4 and f_5). Also, assume that the faults are modeled with additive fault models. In addition, there is an additive disturbance d acting on the third actuator, i.e. the elevator angle actuator.

The total model, including faults and the disturbance, then becomes:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = G(s) \left(\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} f_4 \\ f_5 \\ d \end{bmatrix} \right) + \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

where $G(s) = C(sI - A)^{-1}B + D$.

7.6.1 Decoupling of the Disturbance in the Elevator Angle Actuator

The first design example is intended to illustrate the design procedure and also illustrate how available design freedom can be utilized. The goal is to design a residual generator $Q_1(s)$ that decouples the disturbance d in the elevator angle actuator. Then, matrix $H(s)$ from (7.1) corresponds to all signals that are to be decoupled, i.e. considered disturbances. In this case, $H(s)$ becomes the third column in $G(s)$. Matrix $L(s)$ corresponds to the faults and therefore $L(s)$ becomes $[I_3 \ g_1(s) \ g_2(s)]$, where $g_i(s)$ denotes the i:th column of $G(s)$. Further, the matrix B_d in (7.2) becomes equal to the third column of B. Note also that the realization $\{A, B, C, D\}$ is controllable, i.e. the state x is controllable from u.

Minimal Polynomial Basis Solution

Since the model is given in state-space form and $\{A, [B \ B_d]\}$ is controllable, Theorem 7.3 is used to extract $N_M(s)$. According to formula (7.9), the dimension of the null-space $\mathcal{N}_L(M(s))$ is 2, i.e. there exists exactly two linearly independent parity functions that decouples d.

Calculations using the Polynomial Toolbox (Henrion et al., 1997) give the basis

$$N_M(s) = \begin{bmatrix} 0.0705s & s+0.0538 & 0.091394 & 0.12 & -1 & 0 \\ 22.7459s^2+14.5884s & -6.6653 & s^2-0.93678s-16.5141 & 31.4058 & 0 & 0 \end{bmatrix} \tag{7.58}$$

The command used is `xab`[2] and this gives the basis in canonical polynomial echelon form, i.e. the basis (7.58) is actually unique. The row-degrees of the basis is 1 and 2, i.e. it is a basis of order 3. From this it is clear that the filter of least degree, which decouples d, is a first order filter corresponding to the first row in the basis. To select the first row then corresponds to setting ϕ in (7.10) to $\phi = [1\ 0]$.

Chow-Willsky Solution

We use the Chow-Willsky Scheme version III, i.e. the left null-space of $[R\ H]$ $(= [R_x\ H]$ in this case) is calculated using the row-search procedure. From Section 7.3 and 7.4.3, we know that a good choice of ρ is $\rho = n$. The row-search procedure is implemented in the command `rwsearch` in the Polynomial Toolbox (Henrion et al., 1997). Using this command together with the expression for $F_{CW}(s)$, given in (7.33), results in

$$F_{CW}(s) = \begin{bmatrix} 0.0705s & s+0.0538 & 0.0914 \\ 0.0705s^2-0.00379s & s^2-0.00289 & 0.0914s-0.00492 \\ 22.7s^2+14.6s & -6.67 & s^2-0.937s-16.5 \\ 0.0705s^3-2.08s^2-1.33s & s^3+0.609 & 0.0807s+1.51 \\ 22.7s^3+35.9s^2+14.1s & -5.89 & s^3-17.4s-14.9 \\ 0.0705s^4-2.08s^3-3.17s^2-1.22s & s^4+0.505 & 1.59s+1.28 \\ 22.7s^4+35.9s^3+410s^2+254s & -116 & s^4-31.2s-287 \\ 0.0705s^5-2.08s^4-3.17s^3-37.3s^2-23.2s & s^5+10.5 & 2.76s+26.1 \\ 22.7s^5+35.9s^4+410s^3+963s^2+463s & -201 & s^5-316s-504 \end{bmatrix}$$

$$\begin{bmatrix} 0.12 & -1 & 0 \\ 0.12s-0.00646 & -s+0.0538 & 0 \\ 31.4 & 0 & 0 \\ 0.12s^2-0.00646s-2.87 & -s^2+0.0538s-0.00289 & 0 \\ 31.4s+30.2 & -6.67 & 0 \\ 0.12s^3-0.00646s^2-2.87s-2.61 & -s^3+0.0538s^2-0.00289s+0.609 & 0 \\ 31.4s^2+30.2s+547 & -6.67s-5.89 & 0 \\ 0.12s^4-0.00646s^3-2.87s^2-2.61s-49.8 & -s^4+0.0538s^3-0.00289s^2+0.609s+0.505 & 0 \\ 31.4s^3+30.2s^2+547s+992 & -6.67s^2-5.89s-116 & 0 \end{bmatrix}$$

The command `rwsearch` gives its answer in canonical echelon form which means that the result is unique.

[2] The command `xab` (in version 1.6) is actually not perfectly suited for this case since it uses an unnecessarily large ν.

Now compare $F_{CW}(s)$ with $N_M(s)$ in (7.58). We see that the first and third row of $F_{CW}(s)$ equals the rows of the basis $N_M(s)$, but this is no coincidence. Theorem 7.11 tells us that the uppermost and largest set of primary dependent rows in $[R_x \; H]$ gives a minimal polynomial basis for $\mathcal{N}_L(M(s))$. This was also the idea of the version IV of the Chow-Willsky scheme. Theorem 7.11 together with the uniqueness (because of canonical echelon form) of both $N_M(s)$ and $F_{CW}(s)$, implies that the first and third row of $F_{CW}(s)$ must equal $N_M(s)$.

Forming the Residual Generator

Now we want to use the parity function obtained from the first row of $N_M(s)$ (or equivalently $F_{CW}(s)$) to construct a residual generator. From Section 7.2.1 we know that the minimality property of the basis implies that this parity function is of minimal order. A residual generator can be formed by using the expression (7.5). Since the parity function is of order 1, the scalar polynomial $c(s)$ must have a degree ≥ 1. Let $c(s)$ be $c(s) = 1 + s$ which results in the following filter (residual generator)

$$Q_1(s) = \frac{1}{1+s}\begin{bmatrix}0.0705s & s+0.0538 & 0.091394 & 0.12 & -1 & 0\end{bmatrix} \tag{7.59}$$

Now we know that this residual generator is of minimal order. Also, because of the choice $c(s) = 1 + s$, it is able to detect faults with energy in frequency ranges up to 1 rad/s.

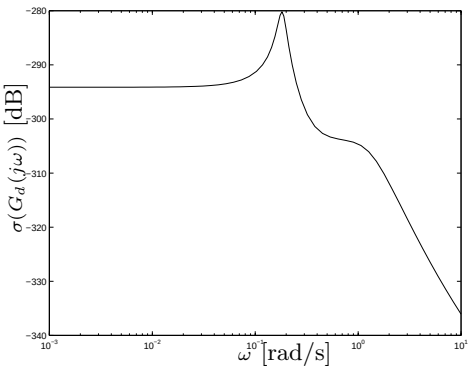

Figure 7.2: Singular value of the transfer function from u and d to r.

Figure 7.2 shows the singular value (maximum gain in any direction) for

$$G_{rud}(s) = Q_1(s)\begin{bmatrix}G(s) & H(s)\\ I & 0\end{bmatrix}$$

This plot should theoretically be exactly 0, but because of finite word length in MATLAB it doesn't become exactly 0. The plot shows that the control signals and the decoupled fault has no significant influence on the residual. Figure 7.3

shows how the monitored faults influence the residual which clearly shows that fault influence is significantly larger than influence from the decoupled fault and control signals plotted in Figure 7.2. The leftmost plot in Figure 7.3 also shows that DC-gain from fault f_1 to the residual is 0. Therefore, fault f_1 is difficult to detect since the effect in the residual of a constant fault f_1 disappears. This effect is more studied in the next chapter.

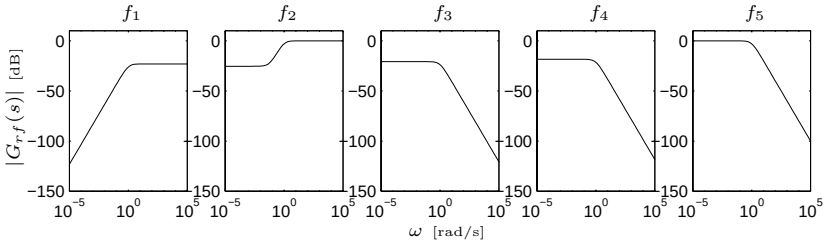

Figure 7.3: Magnitude bode plots for the monitored faults to the residual.

7.7 Conclusions

The topic of this chapter has been design of linear residual generators, which is a special case of the prediction principle. First the relation between linear residual generators and polynomial parity functions was cleared out, and it is concluded that the linear decoupling problem is equivalent to designing polynomial parity functions.

A new method, the *minimal polynomial basis approach* has been developed. The focus has been on four issues, namely that the method (1) is able to generate *all* possible residual generators, (2) explicitly gives the solutions with minimal McMillan degree, (3) results in a minimal parameterization of the solutions, i.e. all residual generators, and (4) has good numerical properties.

In the minimal polynomial basis approach, the residual generator design problem is formulated with standard notions from linear algebra and linear systems theory such as polynomial bases for rational vector spaces, and it is shown that the design problem can be seen as the problem of finding *polynomial* matrices in the left null-space of a rational matrix $M(s)$. Within this framework, the completeness of solution, i.e. issue (1) above, and minimality, i.e. issues (2) and (3), are naturally handled by the concept of *minimal polynomial bases*.

Finding a minimal polynomial basis for a null-space is a well-known problem and there exists computationally simple, efficient, and numerically stable algorithms, to generate the bases. That is, issue (4) is satisfied. In addition, generally available implementations of these algorithms exists.

The order of linear residual generators is investigated and it is concluded that to generate a basis, for all polynomial parity functions or residual generators, it is sufficient to consider orders up to the system order. This result is new since

previous related results only deal with the *existence* of residual generators and also only for some restricted cases.

The question of minimality and completeness of solution is not obvious for other design methods. The well known Chow-Willsky scheme is investigated and it is concluded that in its original version, none of the four issues above are satisfied. However, a modification of the Chow-Willsky scheme is presented and this new version is algebraically equivalent to the minimal polynomial basis approach. This means that the first three, of the issues above, are satisfied. However, it is concluded that numerically, this modified version of the Chow-Willsky scheme is still *not* as good as the minimal polynomial basis approach.

Appendix

7.A Proof of Lemma 7.1

Lemma 7.1 *Let $M(s)$ be the system matrix of any realization (not necessarily controllable from $\left[u^T\, d^T\right]^T$), i.e.*

$$M_s(s) = \begin{bmatrix} C & D_d \\ -(sI - A) & B_d \end{bmatrix}$$

Then it holds that

$$\operatorname{Dim}\mathcal{N}_L(M(s)) = \operatorname{Dim}\mathcal{N}_L(M_s(s))$$

Proof: Consider the realization

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} &= \begin{bmatrix} A_x & A_{12} \\ 0 & A_z \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} B_{u,x} \\ B_{u,z} \end{bmatrix} u + \\ &\quad \begin{bmatrix} B_{d,x} \\ 0 \end{bmatrix} d + \begin{bmatrix} B_{f,x} \\ B_{f,z} \end{bmatrix} f \qquad (7.60a) \\ y &= [C_x C_z] \begin{bmatrix} x \\ z \end{bmatrix} + D_u u + D_d d + D_f f \qquad (7.60b) \end{aligned}$$

where it is assumed that x is controllable from d. Note that this is *not* the same type of realization as (7.12). Then form the matrix $M_{xd}(s)$ as

$$M_{xd}(s) = \begin{bmatrix} C_x & D_d \\ -sI + A_x & B_{d,x} \end{bmatrix}$$

Let n_x be the number of controllable states, i.e. the dimension of x in (7.60).

We will first show that

$$\operatorname{Dim}\mathcal{N}_L(M(s)) = \operatorname{Dim}\mathcal{N}_L(M_{xd}(s)) \qquad (7.61)$$

The dimension of the null-space $\mathcal{N}_L(M_{xd}(s))$ is $m + n_x - \operatorname{Rank} M_{xd}(s)$. The dimension of the null-space $\mathcal{N}_L(M(s))$ is $m + k_u - \operatorname{Rank} M(s)$. Further, it holds that $\operatorname{Rank} M(s) = \operatorname{Rank} H(s) + k_u$. All this means that to show (7.61), it is sufficient to show that

$$\operatorname{Rank} M_{xd}(s) = \operatorname{Rank} H(s) + n_x \qquad (7.62)$$

By using the generalized Bezout identity, it is easy to derive (see (Kailath, 1980), Section 6.4.2) that the following matrices have the same Smith form:

$$\begin{bmatrix} -sI + A_c & B_{d,c} \\ C_c & D_d \end{bmatrix} \overset{S}{\sim} \begin{bmatrix} I_{n_x} & 0 \\ 0 & C_c\Psi(s) + D_d D_H(s) \end{bmatrix} \qquad (7.63)$$

where $\{A_c, B_{d,c}, C_c\}$ is a controller-form realization of $\{A_x, B_{d,x}, C_x\}$ and $\{\Psi(s), D_H(s)\}$ is a specific right MFD of $(sI - A_x)^{-1}B_{d,x} =$

$(sI - A_c)^{-1}B_{d,c}$ (see (Kailath, 1980) for a definition of $\Psi(s)$ and $D_H(s)$). By defining $N_H(s) = C_c\Psi(s) + D_dD_H(s)$, we see that

$$H(s) = C_c\Psi(s)D_H^{-1}(s) + D_d = (C_c\Psi(s) + D_dD_H(s))D_H^{-1}(s) = N_H(s)D_H^{-1}(s)$$

That is, $\{N_H(s), D_H(s)\}$ is a right MFD for $H(s)$. Further, since $M_{xd}(s)$ represents a controllable realization, it has the same Smith form as the controller-form realization, which together with (7.63) means that

$$M_{xd}(s) = \begin{bmatrix} -sI + A_x & B_{d,x} \\ C_x & D_d \end{bmatrix} \overset{S}{\sim} \begin{bmatrix} I_{n_x} & 0 \\ 0 & N_H(s) \end{bmatrix}$$

This further means that

$$\operatorname{Rank} M_{xd}(s) = \operatorname{Rank} N_H(s) + n_x = \operatorname{Rank} H(s) + n_x$$

and thus, (7.62) and (7.61) have been shown.

Let T represent the similarity transformation relating the realization in $M_s(s)$ with the realization (7.60). Then we have that

$$\begin{aligned} \operatorname{Rank} M_s(s) &= \operatorname{Rank} \begin{bmatrix} T^{-1} & 0 \\ 0 & I_m \end{bmatrix} M_s(s) \begin{bmatrix} T & 0 \\ 0 & I_{k_u} \end{bmatrix} = \\ &= \operatorname{Rank} \begin{bmatrix} -sI + A_x & A_{12} & B_{d,x} \\ 0 & -sI + A_z & 0 \\ C_x & C_z & D_d \end{bmatrix} = \operatorname{Rank} M_{xd}(s) + n_z \end{aligned} \quad (7.64)$$

where n_z is the dimension of the state z in (7.60). The last equality holds since the submatrix

$$\begin{bmatrix} A_{12} \\ -sI + A_z \\ C_z \end{bmatrix}$$

has rank n_z and all columns are independent of the other parts of the matrix. The relation (7.64) implies that

$$\begin{aligned} \operatorname{Dim} \mathcal{N}_L(M_{xd}(s)) &= n_x + m - \operatorname{Rank} M_{xd}(s) = \\ &= n_x + n_z + m - \operatorname{Rank} M_s(s) = \operatorname{Dim} \mathcal{N}_L(M_s(s)) \end{aligned}$$

This result together with (7.61) shows the lemma. ∎

7.B Linear Systems Theory

This appendix is included to serve as a compilation of definitions, theorems, and basic properties of linear systems, polynomial matrices, and polynomial bases used in this thesis. Sources describing these matters in detail are e.g. (Forney, 1975; Kailath, 1980; Chen, 1984) for control oriented views, and (Lancaster and Tismenetsky, 1985) for a purely mathematical view.

Definition 7.4 (Dependent Row) *Consider a matrix A. A dependent row, in order top-to-bottom, is a row that is a linear combination of previous rows (i.e. the rows above).*

Definition 7.5 (Primary Dependent Rows) *Let A be a matrix organized in equally sized blocks A_i as follows:*

$$A = \begin{bmatrix} A_0 \\ A_1 \\ \vdots \\ A_\nu \end{bmatrix}$$

Further let each dependent row be associated with a row index α_i telling the placement within its block. Then a set of dependent rows are primary *dependent rows if*

$$\alpha_i \neq \alpha_j, \qquad i \neq j$$

Example 7.6

Consider

$$A = \left[\begin{array}{ccc} 1 & 0 & 1 \\ 0 & 1 & 1 \\ \hline 1 & 1 & 2 \\ 0 & 0 & 1 \\ \hline 2 & 2 & 4 \\ 0 & 2 & 2 \end{array}\right]$$

The dependent rows are row 3, 5, and 6. Of these, row 3 and 6 are primary dependent. Row 5 is not primary dependent since it has the same block location as row 3 which is also dependent. ■

Theorem 7.13 (PBH Rank Test (Kailath, 1980) p. 136) *A pair $\{A, B\}$ will be controllable if and only if the matrix*

$$[sI - A \quad B] \quad \textit{has rank } n \textit{ for all } s$$

7.B.1 Properties of Polynomial Matrices

To avoid unnecessary misunderstandings: a polynomial matrix, which in some literature is called matrix polynomials (Lancaster and Tismenetsky, 1985), is any matrix $F(s)$ where the individual elements are scalar polynomials in s. Here, the coefficients in the polynomials will always be real.

Definition 7.6 (Normal Rank) *The (normal) rank of a polynomial matrix $F(s)$ is the largest rank $F(s)$ has for any $s \in \mathbb{C}$.*

Sometimes the word normal is omitted, when the text only says rank it is always meant normal rank.

Definition 7.7 (Row-reduced Matrix) *Consider a polynomial $p \times q$ matrix $F(s)$ with row-degrees μ_i. It is always possible to write*

$$F(s) = S(s)D_{hr} + L(s)$$

where

$S(s) = diag\{s^{\mu_i}, i = 1, \ldots, p\}$

$D_{hr} =$ *the highest-row-degree coefficient matrix*

$L(s) =$ *the remaining term with row-degrees strictly less than those of F(s)*

A full row rank matrix $F(s)$ is said to be row-reduced if its highest-row-degree coefficient matrix D_{hr} has full row rank.

Definition 7.8 (Irreducible and Unimodular Matrices) *A polynomial matrix $F(s)$ is said to be irreducible if it has full rank for all finite s. If $F(s)$ is irreducible and square it is said to be unimodular. A unimodular matrix has a unimodular inverse.*

7.B.2 Properties of Polynomial Bases

Definition 7.9 (Degree of a Polynomial Vector) *The degree of a polynomial vector is the highest degree of all the entries of the vector. If the vector is a row-vector, it is called row-degree.*

The order of a polynomial basis is defined in (Kailath, 1980) as

Definition 7.10 (Order of a polynomial basis) *Let the rows of $F(s)$ form a basis for a vector space $\mathcal{F}$. Let μ_i be the row-degrees of $F(s)$. The order of $F(s)$ is defined as $\sum \mu_i$.*

A *minimal polynomial basis* for $\mathcal{F}$ is then any basis that minimizes this order.

Theorem 7.14 (Minimal Polynomial Bases (Kailath, 1980)) *Consider a full row (normal) rank polynomial matrix $F(s)$. Then the following statements are equivalent*

- *The rows of $F(s)$ form a minimal basis for the rational vector space they generate.*
- *$F(s)$ is row-reduced and irreducible.*
- *$F(s)$ has minimal order.*

Theorem 7.2 (Irreducible Basis) *If the rows of $N(s)$ is an irreducible polynomial basis for a space $\mathcal{F}$, then all polynomial row vectors $f(s) \in \mathcal{F}$ can be written $f(s) = \phi(s)N(s)$ where $\phi(s)$ is a polynomial row vector.*

Proof: Since $N(s)$ is a basis, all $f(s) \in \mathcal{F}$ can be written $f(s)g(s) = \phi(s)N(s)$. For each root α of $g(s)$ it holds that

$$f(\alpha)g(\alpha) = \phi(\alpha)N(\alpha) = 0$$

Since $N(s)$ is irreducible, it has full row rank for all s and in particular $s = \alpha$. This implies that $\phi(\alpha) = 0$, i.e. all roots of $g(s)$ are also roots of $\phi(s)$. Thus $\phi(s)$ can be factorized as $\phi(s) = g(s)\bar{\phi}(s)$ and

$$f(s)g(s) = g(s)\bar{\phi}(s)N(s)$$

This implies

$$f(s) = \bar{\phi}(s)N(s)$$

■

To illustrate the concept of rational vector-spaces and polynomial bases, the following example has been included.

Example 7.7

Let the rows of the matrix $F(s)$ be a basis for the rational vector-space $\mathcal{F}$.

$$F(s) = \begin{bmatrix} s & 0 & 1 \\ 1 & 1 & 0 \\ 0 & -s & 2 \end{bmatrix}$$

It is clear that $F(s)$ is a basis since $\det(F(s)) = s \neq 0$, i.e. the matrix has full rank and therefore, the rows are linearly independent. Any polynomial vector of dimension 3 will of course belong to $\mathcal{F}$. Consider for example the vector

$$b_1(s) = \begin{bmatrix} s & 0 & 0 \end{bmatrix} \in \mathcal{F}$$

This vector can be written as a linear combination of the columns as follows:

$$b_1(s) = \begin{bmatrix} 2 & -s & -1 \end{bmatrix} \begin{bmatrix} s & 0 & 1 \\ 1 & 1 & 0 \\ 0 & -s & 2 \end{bmatrix} = x(s)F(s)$$

Here, $x(s)$ happens to be a polynomial vector. However, in general rational vectors are needed. Consider for example the vector

$$b_2(s) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{2}{s} & -1 & -\frac{1}{s} \end{bmatrix} \begin{bmatrix} s & 0 & 1 \\ 1 & 1 & 0 \\ 0 & -s & 2 \end{bmatrix} = x(s)F(s)$$

In this case, $x(s)$ is rational and there exists no *polynomial* $x(s)$ such that $b_2(s) = x(s)F(s)$.

If the polynomial basis is irreducible, then according to Theorem 7.2, only polynomial $x(s)$:s are needed. An irreducible basis for the same vector-space $\mathcal{F}$ is for example

$$F'(s) = \begin{bmatrix} 1 & 0 & s \\ 0 & 1 & s \\ 0 & 0 & 1 \end{bmatrix}$$

Now $b_2(s)$ can be written

$$b_2(s) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -s \end{bmatrix} \begin{bmatrix} 1 & 0 & s \\ 0 & 1 & s \\ 0 & 0 & 1 \end{bmatrix} = x(s)F(s)$$

■

Chapter 8

Criterions for Fault Detectability in Linear Systems

The topic of this chapter is fault detectability, or more exactly, if it is possible to construct a residual generator that is sensitive to a certain fault modeled as a signal. As in the previous chapter, only linear systems will be considered. Detectability of faults that are modeled as constant signals are explicitly investigated. Such detectability is usually called *strong fault detectability*,

Criterions for both fault detectability and strong fault detectability are derived. A few of these are already known results, but most of the criterions, especially those for strong fault detectability, are new.

We will see that the analyses becomes quite simple. This is due to the notion of bases developed in the previous chapter. For simplicity reasons, we assume that only one fault affects the system, i.e. f is scalar.

In Section 8.1, we will study how the general definitions of fault detectability from Chapter 2, are specialized when only linear systems are considered. Then the criterions for fault detectability and strong fault detectability are derived in Sections 8.2 and 8.3 respectively. Finally Sections 8.4 and 8.5 contain discussions and examples.

8.1 Fault Detectability and Strong Fault Detectability

Recall the definition of uniform partial detectability in a diagnosis system, i.e. Definition 2.23. Uniform partial detectability was defined via uniform partial isolability, i.e. Definition 2.19. Combining these two definitions we get:

> A fault mode F is uniformly and partially detectable in a diagnosis system δ if for all initial conditions, for all inputs, and for all modeled

disturbances, it holds that

$$\exists \theta \in \Theta_F . F \in S \wedge NF \notin S$$

and

$$\exists \theta \in \Theta_{NF} . NF \in S$$

Now assume that we have a diagnosis system based on a single hypothesis test (in this case a residual generator) and that the fault mode F is modeled by a fault signal $f(t)$. For F to be detectable in this diagnosis system, the above requirements imply that

$$\begin{aligned} &\forall u(t), d(t),\ \exists f(t) \neq 0\ .\ S = S^1 = \{F, \dots\} \text{ and } NF \notin S \\ &\forall u(t), d(t)\ .\ f(t) = 0\ \rightarrow\ S = S^0 = \{NF, \dots\} \end{aligned}$$

Note that $\exists f(t) \neq 0$ in the above expression means that there exists a fault signal modeled by $f(t)$ belonging to a *specific* fault mode (and not that there exists a signal belonging to some arbitrary fault mode). By assuming ideal condition, these requirements can be formulated as

$$\begin{aligned} &\forall u(t), d(t),\ \exists f(t) \neq 0\ .\ r(t) \neq 0 \\ &\forall u(t), d(t)\ .\ f(t) = 0\ \rightarrow\ r(t) = 0 \end{aligned}$$

For linear system, we can phrase this in terms of transfer functions which leads to the following definition of fault detectability in a residual generator:

Definition 8.1 (Fault Detectability in a Residual Generator) *A fault f is detectable in a residual generator if the transfer function from the fault to the residual is nonzero, i.e. $G_{rf}(\sigma) \neq 0$, and the transfer functions from the known input u and the disturbance d to the residual are zero, i.e. $G_{ru}(\sigma) = 0$ and $G_{rd}(\sigma) = 0$.*

As in the previous chapter, the operator σ represents the differentiation operator p (or s) in the continuous case and the time-shift operator q (or z) in the discrete case.

Next consider uniform complete fault detectability, i.e. Definition 2.23 and 2.16. It is clear that using a linear residual generator, uniform complete fault detectability can only be achieved if $G_{rf}(\sigma) = C \neq 0$. This is a very strong requirement and we will instead focus on uniform complete fault detectability of constant faults, i.e. $f(t) \equiv c$. Then for a diagnosis system based on a single residual generator, we have the following requirements:

$$\begin{aligned} &\forall u(t), d(t),\ \forall f(t) \equiv c \neq 0.\ r(t) \neq 0 \\ &\forall u(t), d(t)\ .\ f(t) \equiv 0\ \rightarrow\ r(t) = 0 \end{aligned}$$

For linear system, we can phrase this in terms of transfer functions which leads to the following definition of *strong* fault detectability in a residual generator:

Definition 8.2 (Strong Fault Detectability in a Residual Generator)
A fault f is strongly detectable in a residual generator if the transfer function from the fault to the residual $G_{rf}(\sigma)$ has a nonzero DC-gain, e.g. $G_{rf}(0) \neq 0$ in the continuous case, and the transfer functions from the known input u and the disturbance d to the residual are zero, i.e. $G_{ru}(\sigma) = 0$ and $G_{rd}(\sigma) = 0$.

Faults that are detectable but not strongly detectable will be called *weakly detectable* faults.

The importance of strong detectability is illustrated the following example.

Example 8.1

Consider a DC-servo which can be modeled as

$$y_1 = \frac{1}{s(1+s)}u + f_1 \tag{8.1}$$

$$y_2 = \frac{1}{1+s}u + f_2 \tag{8.2}$$

$$\tag{8.3}$$

where y_1 is the output from an angle sensor and y_2 is the output from a tachometer (i.e. an angular velocity sensor). There are two possible sensor faults modeled by the fault signals f_1 and f_2.

Consider two residual generators:

$$r_1 = \frac{s(s+1)y_1 - u}{(s+4)^2} = \frac{s(s+1)f_1}{(s+4)^2}$$

$$r_2 = \frac{16(s+1)y_2 - u}{(s+4)^2} = \frac{16(s+1)f_2}{(s+4)^2}$$

The residual r_1 will only be sensitive to f_1 and r_2 will only be sensitive to f_2. It is obvious that $G_{r_1 f_1}(0) = 0$, which means that the fault f_1 is weakly detectable in the residual generator generating r_1.

Their response to two step faults are plotted in Figure 8.1. The two residuals $r_1(t)$ and $r_2(t)$ has fundamentally different behavior since $r_1(t)$ only reflects *changes* on the fault signal and $r_2(t)$ has approximately the same shape as the fault signal. In a real case, where noise and model uncertainties are present, it is significantly more difficult to use $r_1(t)$ than $r_2(t)$. ■

In accordance with Definition 2.23 and 2.22, we can also define fault detectability and strong fault detectability as properties of the system.

Definition 8.3 (Fault Detectability) *A fault f is detectable in a system if there exists a residual generator such that the transfer function from the fault to the residual is nonzero, i.e. $G_{rf}(\sigma) \neq 0$, and the transfer functions from the known input u and the disturbance d to the residual are zero, i.e. $G_{ru}(\sigma) = 0$ and $G_{rd}(\sigma) = 0$.*

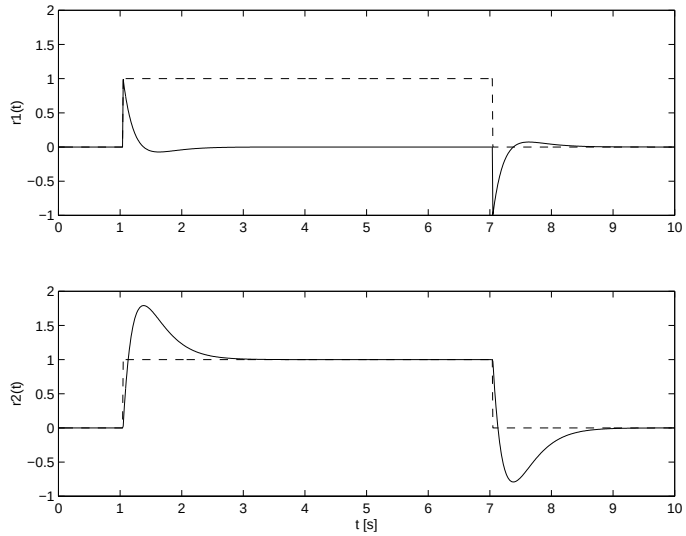

Figure 8.1: A weakly detectable fault (upper plot) and a strongly detectable fault (lower plot). The fault signal is the dashed line and the residual is the solid line.

Definition 8.4 (Strong Fault Detectability) *A fault f is strongly detectable in a system if there exists an asymptotically stable residual generator such that the transfer function from the fault to the residual $G_{rf}(\sigma)$ has a nonzero DC-gain, e.g. $G_{rf}(0) \neq 0$ in the continuous case, and the transfer functions from the known input u and the disturbance d to the residual are zero, i.e. $G_{ru}(\sigma) = 0$ and $G_{rd}(\sigma) = 0$.*

By excluding diagnosis systems in which the fault is not detectable, from the definition of diagnosis systems, fault detectability as a system property is also referred to as the existence of a diagnosis system, see for example (Mironovskii, 1980) and (Frank and Ding, 1994*b*).

The above two definitions of fault detectability and strong fault detectability as system properties, will from now on be our primary interest. The question is:

> Given a model of the system, is a particular fault f strongly detectable, only weakly detectable, or not detectable at all?

As in the previous chapter, we assume that the model is given either in the transfer function form

$$y = G(\sigma)u + H(\sigma)d + L(\sigma)f \tag{8.4}$$

or in the state-space form

$$\begin{aligned} \sigma x(t) &= Ax(t) + B_u u(t) + B_d d(t) + B_f f(t) && \text{(8.5a)} \\ y(t) &= Cx(t) + D_u u(t) + D_d d(t) + D_f f(t) && \text{(8.5b)} \end{aligned}$$

In particular cases, it can be quite simple to show that a fault is for example only weakly detectable. This is illustrated in the following example.

Example 8.2

Consider the same system as in Example 8.1. There we saw that the fault f_1 was only weakly detectable in the residual generator generating r_1. The question now is if the fault f_1 is strongly detectable (using Definition 8.4). That is, does there exist any residual generator in which fault f_1 becomes strongly detectable.

According to the expression (7.5), a general linear residual generator can be written as

$$r = \frac{A_1(s)y_1 + A_2(s)y_2 + B(s)u}{c(s)}$$

Since f_2 must be decoupled, it is considered to be a disturbance, and the term $A_2(s)$ must therefore be 0. Thus a general expression for a residual generator is

$$r = \frac{A_1(s)y_1 + B(s)u}{c(s)}$$

In the fault free case, the residual is zero, and therefore it must hold that

$$A_1(s)y_1 + B(s)u = 0 \tag{8.6}$$

If the expression for y_1 in the fault-free DC-servo model (8.1), i.e. $f = 0$, is substituted into Equation (8.6), we get

$$A_1(s)\frac{1}{s(1+s)}u + B(s)u = 0$$

This equation must hold for all u which implies that the following equation must be satisfied:

$$A_1(s) = s(1+s)B(s)$$

This in turn means that the polynomial $A_1(s)$ must contain the factor s. The transfer function from the fault f_1 to the residual becomes

$$G_{rf_1} = \frac{A(s)}{c(s)}$$

If the residual generator is asymptotically stable, i.e. the polynomial $c(s)$ has all its poles in the left half plane, the transfer function G_{rf_1} will have a zero in the origin. Thus for the angle sensor fault, modeled as in (8.1), it is impossible to find a residual in which the fault becomes strongly detectable. ■

It is clear that in some cases, like the one in the example, we are forced to use a residual generator in which the fault is weakly detectable. Even though a fully satisfactory solution can not be obtained unless we reconstruct the system, weakly detectable faults can more easily be detected by filtering the residual with a filter that acts approximately like an integrator. This was demonstrated in for example (Frisk, Nyberg and Nielsen, 1997).

In Example 8.2, we manage to quite simply prove that f_1 is not strongly detectable. However, in general cases, this can be much more difficult. Therefore it would be useful to have criterions for both fault detectability and strong fault detectability. Such criterions are developed in the next two sections. For simplicity reasons, we will, as we did in Chapter 7, only discuss the continuous case. However, the corresponding results for the discrete case can be derived in a similar manner. Throughout this chapter, we will assume that the fault signal $f(t)$ is a scalar signal. This makes most sense since we are interesting in checking detectability with respect to one particular fault. We will use the notation $\text{Im}\, A(s)$ to denote the *column image* (also called the *column range*) of a matrix $A(s)$.

8.2 Detectability Criteria

In this section, a four general detectability criterions are presented. The two first criterions assume that the system is given on the transfer function form (8.4) and the next two criterions assume that the system is given on the transfer function form (8.5). Also included is a necessary criterion based on the dimensions of the system.

8.2.1 The Intuitive Approach

The first criterion assumes that the system is given on the transfer function form (8.4). The reasoning follows intuitively from the the basic results of Section 7.2.1.

Theorem 8.1 *A fault f is detectable in a system if and only if*

$$\text{Im} \begin{bmatrix} L(s) \\ 0 \end{bmatrix} \not\subseteq \text{Im} \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} \tag{8.7}$$

Proof: The criterion of the theorem is equivalent to that there exists a rational $Q(s)$ such that

$$Q(s) \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} = 0 \tag{8.8}$$

and

$$Q(s) \begin{bmatrix} L(s \\ 0 \end{bmatrix} \neq 0 \tag{8.9}$$

If there exists a $Q(s)$ that fulfills (8.8) and (8.9), then $r = Q(s)[y^T \ u^T]^T$ is a residual for which $G_{ru}(s) = G_{rd}(s) = 0$ and $G_{rf}(s) \neq 0$. This means that fault f is detectable.

If $r = Q(s)[y^T \ u^T]^T$ is a residual in which fault f is detectable, i.e. $G_{uf}(s) = G_{uf}(s) = 0$ and $G_{rf}(s) \not\equiv 0$, then (8.8) and (8.9) will be fulfilled. ∎

The easiest way to check condition (8.7) is probably by studying the rank as follows: a fault is detectable if and only if

$$\text{Rank} \begin{bmatrix} G(s) & H(s) & L(s) \\ I & 0 & 0 \end{bmatrix} > \text{Rank} \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix}$$

It is obvious that this rank-condition is equivalent to (8.7). The normal rank of a polynomial matrix can be calculated quite easily by using the formula

$$\text{Rank}\, A(s) = \max_s \text{Rank}\, A(s) \tag{8.10}$$

obtained from the definition of normal rank (see Appendix 7.B). Note that the rank on the left-hand side of (8.10) refers to the normal rank of a polynomial matrix while the rank on the right-hand side refers to the rank of a constant matrix. We can substitute different random numbers for s and thus obtaining a set of constant matrices. The normal rank is then the maximum rank of these constant matrices. This procedure is implemented in the polynomial toolbox (Henrion et al., 1997).

A second alternative to check condition (8.7) is to calculate a basis for $\mathcal{N}_L(M)$, i.e. the left null-space of $M(s)$. As in the previous chapter we let the rows of a matrix $N_M(s)$ form a basis for $\mathcal{N}_L(M)$. Then we have that a fault is detectable if and only if

$$N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix} \neq 0 \tag{8.11}$$

However, to calculate a basis for the null-space requires more involved algorithms than a rank test, as was seen in Section 7.2.3.

8.2.2 The "Frequency Domain" Approach

Here we will present a somewhat simpler, but closely related, alternative to Theorem 8.1. Again we assume that the system is given on the transfer function form (8.4).

Theorem 8.2 *A fault f is detectable in a system if and only if*

$$\text{Im}\, L(s) \not\subseteq \text{Im}\, H(s) \tag{8.12}$$

Proof: We will first show that it holds that

$$\text{Im} \begin{bmatrix} L(s) \\ 0 \end{bmatrix} \subseteq \text{Im} \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} \tag{8.13}$$

if and only if

$$\operatorname{Im} L(s) \subseteq \operatorname{Im} H(s)$$

If (8.13) holds, then there exists a rational matrix $[X_1^T(s)\ X_2^T(s)]^T$ such that $0 = IX_1(s)$ and $L(s) = G(s)X_1(s) + H(s)X_2(s) = H(s)X_2(s)$, which means that $\operatorname{Im} L(s) \subseteq \operatorname{Im} H(s)$. This proofs the only-if part and the if-part is easier.

This means that we have shown the equivalence between the condition (8.7) in Theorem 8.1 and (8.12), which ends the proof. ■

This criterion was given in (Ding and Frank, 1990) and as seen, it is much simpler than Theorem 8.1, since it does not include $G(s)$. Also here, the check can be performed by doing a rank test or to calculate a basis for the null space. Particularly simple is the rank test which becomes:

$$\operatorname{Rank}[H(s)\ L(s)] > \operatorname{Rank} L(s) \tag{8.14}$$

8.2.3 Using the System Matrix

The criterion presented here is based on the results from Section 7.2.2, about the minimal polynomial basis approach using the state-space representation. It is assumed that the system is given on the state-space form (8.5).

Theorem 8.3 *A fault f is detectable in a system if and only if*

$$\operatorname{Im} \begin{bmatrix} B_f \\ D_f \end{bmatrix} \not\subseteq \operatorname{Im} \begin{bmatrix} A - sI & B_d \\ C & D_d \end{bmatrix} \tag{8.15}$$

Proof: We will first show that it holds that

$$\operatorname{Im} \begin{bmatrix} D_f \\ B_f \end{bmatrix} \subseteq \operatorname{Im} \begin{bmatrix} C & D_d \\ A - sI & B_d \end{bmatrix} \tag{8.16}$$

if and only if

$$\operatorname{Im} \begin{bmatrix} L(s) \\ 0 \end{bmatrix} \subseteq \operatorname{Im} \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} \tag{8.17}$$

Let the row vectors of $V(s)$ be a basis for $\mathcal{N}_L(M_s(s))$ and form $W(s) = V(s)P$, where P is, as before, defined as

$$P = \begin{bmatrix} I_m & -D_u \\ 0 & -B_u \end{bmatrix}$$

According to Theorem 7.4, the rows of $W(s)$ are a basis for $\mathcal{N}_L(M(s))$.

Now consider the relation

$$\begin{aligned} W(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix} =& V(s)P \begin{bmatrix} L(s) \\ 0 \end{bmatrix} = V(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix} = V(s) \begin{bmatrix} C(sI - A)^{-1}B_f + D_f \\ 0 \end{bmatrix} = \\ =& [V_1(s) \quad V_1(s)C(sI - A)^{-1}] \begin{bmatrix} D_f \\ B_f \end{bmatrix} = V(s) \begin{bmatrix} D_f \\ B_f \end{bmatrix} \end{aligned} \tag{8.18}$$

The last equality follows from the fact that $V(s)M_s(s) = 0$. The relation (8.18) implies that

$$W(s)\begin{bmatrix} L(s) \\ 0 \end{bmatrix} = 0 \Longleftrightarrow V(s)\begin{bmatrix} D_f \\ B_f \end{bmatrix} = 0$$

Since $W(s)$ and $V(s)$ are bases for $\mathcal{N}_L(M(s))$ and $\mathcal{N}_L(M_s(s))$ respectively, this statement is equivalent to that (8.16) holds if and only if (8.17) holds. This is further equivalent to the condition (8.15) in the theorem. ∎

Similar conditions for fault detectability were noted in for example (Magni and Mouyon, 1994). Note that, in contrast to design of polynomial parity functions using the minimal polynomial basis approach, we do not need to care about controllability from u and d when checking detectability.

8.2.4 Using the Chow-Willsky Scheme

The criterions given here are based on the results from the study of the Chow-Willsky scheme, performed in Sections 7.4 and 7.5. Again we assume that the system is given on the state-space form (8.5).

Theorem 8.4 *A fault f is detectable in a system if and only if*

$$\operatorname{Im} P_{\rho=n} \nsubseteq \operatorname{Im} [R\ H]_{\rho=n} \tag{8.19}$$

Proof: We will first show that it holds that

$$\operatorname{Im} P_{\rho=n} \subseteq \operatorname{Im} [R\ H]_{\rho=n} \tag{8.20}$$

if and only if

$$\operatorname{Im}\begin{bmatrix} L(s) \\ 0 \end{bmatrix} \subseteq \operatorname{Im}\begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} \tag{8.21}$$

Let the rows of a matrix W define the largest and uppermost set of primary dependent rows in $[R\,H]_{\rho=n}$. Then according to Theorem 7.11, $F(s) = W[\Psi_m(s) \ -Q\Psi_{k_u}(s)]$ becomes a polynomial basis for $\mathcal{N}_L\{M(s)\}$.

Define $X(s)$ as follows:

$$X(s) = \sum_{i=1}^{\infty} s^{-i} A^{i-1} B_f$$

Then by using the same reasoning as in the formulas (7.39), (7.40), and (7.41), we can conclude that

$$\Psi_m(s)L(s) = RX(s) + P\Psi_1(s)$$

Now assume that (8.20) holds. This implies the following:

$$F(s)\begin{bmatrix} L(s) \\ 0 \end{bmatrix} = W[\Psi_m(s) \ -Q\Psi_{k_u}(s)]\begin{bmatrix} L(s) \\ 0 \end{bmatrix} = W\Psi_m(s)L(s) = \\ = W(RX(s) + P\Psi_1(s)) = 0 \tag{8.22}$$

The last equality holds since $W[R\,H] = 0$ which, according to (8.20), also implies that $WP = 0$. Since $F(s)$ is a polynomial basis for $\mathcal{N}_L\{M(s)\}$, equation (8.22) is equivalent to (8.11) which is further equivalent to (8.21), and thus the *only-if* part of the proof have been shown.

For the *if* part, assume that w_1 is an arbitrary row-vector such that $w_1[R\,H] = 0$. Pick other w_i:s such that $W = [w_1^T w_2^T \ldots]^T$ defines a set of primary dependent rows in $[R\ H]$. This implies that $W[R\ H] = 0$ and according to Theorem 7.12, $F(s) = W[\Psi_m(s) \quad -Q\Psi_{k_u}(s)]$ becomes a polynomial basis (not necessarily irreducible) for $\mathcal{N}_L\{M(s)\}$. Assume that (8.21) holds. Then we know that

$$0 = F(s)\begin{bmatrix} L(s) \\ 0 \end{bmatrix} = W[\Psi_m(s) \quad -Q\Psi_{k_u}(s)]\begin{bmatrix} L(s) \\ 0 \end{bmatrix} = W\Psi_m(s)L(s) = \\ = W(RX(s) + P\Psi_1(s)) = WP\Psi_1(s) \quad (8.23)$$

This implies that $WP = 0$ and thus $w_1 P = 0$ which proves the *if* part.

This means that we have shown the equivalence between the condition (8.7) in Theorem 8.1 and (8.19), which ends the proof. ∎

Note that also in this case, we do not need to care about controllability from u and d when checking detectability. This is in contrast to *design* of parity functions using the Chow-Willsky scheme, for which we showed in Section 7.4.3 that in order to find all parity functions, we have to care about controllability from u and d.

8.2.5 Necessary Condition Based on Dimensions

The following criterion is trivial and stated in several works, e.g. (Gertler, 1998), but nevertheless very useful since it uses only the dimensions of the system.

Theorem 8.5 *Assume that $H(s)$ has full column rank. Then a fault f is detectable in the system only if*

$$m > k_d \quad (8.24)$$

where m is the number of outputs and k_d is the number of linearly independent disturbances.

Proof: Theorem 8.2 and expression (8.14) implies that if a fault is detectable, then it must hold that

$$m \geq \text{Rank}\,[H(s)\ L(s)] > \text{Rank}\,H(s) = k_d$$

which gives the condition (8.24). An alternative proof is to use the formula (7.9) which imply that the condition (8.24) must hold. ∎

In other words, the condition (8.24) is a necessary condition for fault detectability. For most systems this simply means that there must be more outputs than disturbances if we are going to be able to detect any fault modeled by the signal $f(t)$.

8.3 Strong Detectability Criteria

It is well know that faults often become weakly detectable when the system contains an integration. For instance, this was the case in Example 8.2. However, faults can be weakly detectable also if the system do *not* contain an integration. This is demonstrated in the following example.

Example 8.3

Consider a system described by the following transfer functions:

$$G(s) = \begin{bmatrix} \frac{2}{s+1} \\ \frac{1}{s+1} \end{bmatrix} \quad H(s) = \begin{bmatrix} \frac{1}{s+2} \\ \frac{1}{s+2} \end{bmatrix} \quad L(s) = \begin{bmatrix} \frac{s+1}{s+3} \\ \frac{1}{s+3} \end{bmatrix}$$

Note that no part of the system contains an integration. An MFD of the matrix $M(s)$ is

$$M(s) = \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 1 & 1 \\ s+1 & 0 \end{bmatrix} \begin{bmatrix} (s+1)^{-1} & 0 \\ 0 & (s+2)^{-1} \end{bmatrix} = N(s)D^{-1}(s)$$

An irreducible basis for the left null-space of $N(s)$ is $F(s) = [s+1 \;\; -s-1 \;\; -1]^T$. Using the corresponding parity function in a residual generator means that the transfer function from the fault to the residual becomes

$$G_{rf}(s) = c^{-1}(s)F(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix}$$

To check strong fault detectability, we evaluate $G_{rf}(0)$:

$$\Big(c^{-1}(s)[s+1 \;\; -s-1 \;\; -1] \begin{bmatrix} L(s) \\ 0 \end{bmatrix}\Big)|_{s=0} = c^{-1}(0)\big(\frac{(s+1)^2}{s+3} - \frac{s+1}{s+3}\big)|_{s=0} =$$
$$= c^{-1}(0)\big(\frac{1}{3} - \frac{1}{3}\big) = 0$$

Thus, the fault is not strongly detectable in the residual generator. Later in this section we will see that since $F(s)$ is an irreducible basis, it actually holds that there exists no residual generators in which the fault is strongly detectable. The fault is therefore not strongly detectable in the sense of Definition 8.4. ■

Thus, no poles in the origin, is not a sufficient condition for strong detectability. It is neither a necessary condition which is shown in the following example:

Example 8.4

Consider the following system:

$$y_1 = \frac{1}{s}u + f_1$$
$$y_2 = \frac{1}{s(s+1)}u + f_2$$

Consider next the residual generator

$$r = \frac{(s+1)y_2 - y_1}{s+2}$$

The transfer functions from the faults to the residual become

$$G_{rf_1}(s) = \frac{-1}{s+2}$$
$$G_{rf_2}(s) = \frac{s+1}{s+2}$$

which shows that both faults are strongly detectable in spite of that the system has a pole in the origin. ■

The previous two examples show that the problem of checking strong fault detectability is more involved than only checking the existence of poles in the origin. Below we will investigate how the four criterions given in Section 8.2, can be modified to become general criterions for strong fault detectability.

We first note that when checking strong detectability, it is not possible to use conditions similar to (8.7), (8.12), or (8.15), without computing a basis for the null-space. We saw in Section 8.2 that checking strong fault detectability can be associated with calculating a basis $\mathcal{N}_L\{M(s)\}$. Similarly, we will see in this section that checking strong fault detectability is associated with evaluating the expression $\mathcal{N}_L\{M(s)\}|_{s=0}$. The reason why (8.7), (8.12), or (8.15), can not be used is that in general

$$\mathcal{N}_L\{M(s)\}|_{s=0} \neq \mathcal{N}_L\{M(0)\}$$

This will be illustrated in Example 8.5, included in the next section below.

8.3.1 The Intuitive Approach

The criterion corresponding to Theorem 8.1 becomes:

Theorem 8.6 *A fault f is strongly detectable in a system if and only if*

$$\left(N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix}\right)|_{s=0} \neq 0 \tag{8.25}$$

where the rows of $N_M(s)$ is an irreducible polynomial basis for $\mathcal{N}_L\{M(s)\}$.

Proof: From Section 7.1 and 7.2.1, we recall that all residual generators r can be parameterized as

$$r = c^{-1}(s)\phi(s)N_M(s) \begin{bmatrix} y \\ u \end{bmatrix}$$

where $c(s)$ is a scalar polynomial with its roots in the left half-plane and $\phi(s)$ is a polynomial vector. The fact that a fault is not strongly detectable can be expressed as

$$\forall c(s), \phi(s) \; . \; \big(c^{-1}(s)\phi(s)N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix} \big)|_{s=0} = 0$$

Since we know that $c(0) \neq 0$, this is equivalent to

$$\forall \phi(s) \; . \; \big(\phi(s)N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix} \big)|_{s=0} = 0$$

which is further equivalent to

$$\big(N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix} \big)|_{s=0} = 0$$

The negation of this condition is then equivalent to the condition (8.25) in the theorem. ∎

Note that when using this Theorem 8.6, it is important to first evaluate the vector

$$N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix}$$

i.e. carry out all multiplications and cancelations, and afterwards substitute s with 0.

As was said above, when checking strong fault detectability, it is important that we calculate the left null-space of $M(s)$ and not $M(0)$. The following example illustrates this.

Example 8.5

Consider a system described by the following transfer functions:

$$G(s) = \begin{bmatrix} \frac{1}{s+1} \\ \frac{s}{s+2} \end{bmatrix} \quad H(s) = \begin{bmatrix} \frac{1}{s+1} \\ \frac{s}{s+2} \end{bmatrix} \quad L(s) = \begin{bmatrix} \frac{s+1}{s+3} \\ \frac{1}{s+3} \end{bmatrix}$$

Then a right MFD of the matrix $M(s)$ is

$$M(s) = \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} = \begin{bmatrix} 1 & s \\ 1 & s \\ s+1 & 0 \end{bmatrix} \begin{bmatrix} s+1 & 0 \\ 0 & s+2 \end{bmatrix}^{-1}$$

A minimal polynomial basis for the left null-space of $M(s)$ is $[1 \; -1 \; 0]$. By using Theorem 8.6, the check for strong fault detectability becomes

$$[1 \quad -1 \; 0] \begin{bmatrix} L(s) \\ 0 \end{bmatrix} |_{s=0} = -\frac{s}{s+3} \; |_{s=0} = 0$$

and the fault is therefore not strongly detectable.

Now we will show that it is not sufficient to consider the left null-space of $M(0)$. A minimal polynomial basis for $\mathcal{N}_L(M(0))$ is

$$\begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}$$

The check for strong fault detectability would be

$$\begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} L(s) \\ 0 \end{bmatrix} |_{s=0} = \begin{bmatrix} -\frac{s}{s+3} \\ \frac{1}{s+3} \end{bmatrix} |_{s=0} = \begin{bmatrix} 0 \\ \frac{1}{3} \end{bmatrix} \neq 0$$

which wrongly indicates that the fault is strongly detectable. This means that

$$N_M(0) \begin{bmatrix} L(0) \\ 0 \end{bmatrix} \neq 0$$

is *not* a condition for strong fault detectability. ■

8.3.2 The "Frequency Domain" Approach

We have concluded that a basis for the null-space must be calculated to check strong fault detectability. However, even if we do so, the "frequency domain" approach, from Section 8.2.2, will not work. This is shown by the following example:

Example 8.6

Consider a system described by the following transfer functions:

$$G(s) = \begin{bmatrix} \frac{1}{s} \\ \frac{1}{s} \end{bmatrix} \quad H(s) = \begin{bmatrix} \frac{1}{s} \\ 1 \end{bmatrix} \quad L(s) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Then an MFD of the matrix $M(s)$ is

$$M(s) = \begin{bmatrix} G(s) & H(s) \\ I & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & s \\ s & 0 \end{bmatrix} \begin{bmatrix} s^{-1} & 0 \\ 0 & s^{-1} \end{bmatrix} = N(s)D^{-1}(s)$$

An irreducible basis for the left null-space of $N(s)$ is $[s^2 \ -s \ -s+1]^T$. By using Theorem 8.6, the check for strong fault detectability becomes

$$[s^2 \ -s \ -s+1] \begin{bmatrix} L(s) \\ 0 \end{bmatrix} |_{s=0} = -s \, |_{s=0} = 0$$

and the fault is therefore not strongly detectable.

Now the question is if we can use a condition for strong fault detectability based on (8.12), if we actually calculate a basis for the null-space, i.e.

$\mathcal{N}_L\{H(s)\}L(s)|_{s=0} \neq 0$. Therefore, we calculate a basis for the null-space $\mathcal{N}_L(H(s))$, which becomes $[s \ -1]$. Then we have that

$$\mathcal{N}_L\{H(s)\}L(s)|_{s=0} = [s \ -1] \begin{bmatrix} 0 \\ -1 \end{bmatrix} |_{s=0} = 1 \neq 0$$

which wrongly indicates that the fault is strongly detectable. This means that $\mathcal{N}_L\{H(s)\}L(s)|_{s=0} \neq 0$ is *not* a condition for strong detectability. ■

8.3.3 Using the System Matrix

The criterion for strong fault detectability, corresponding to Theorem 8.3, becomes as follows:

Theorem 8.7 *A fault f is strongly detectable in a system if and only if*

$$N_{M_s}(0) \begin{bmatrix} D_f \\ B_f \end{bmatrix} \neq 0$$

where the rows of $N_{M_s}(s)$ is a basis for the left null-space of the matrix

$$M_s(s) = \begin{bmatrix} C & D_d \\ -sI + A & B_d \end{bmatrix}$$

To prove this theorem, we first need two lemmas.

Lemma 8.1 *Let $A(s)$ be a rational matrix and assume $A(0)$ exists. Let $B(s)$ be a rational matrix.*

a) If $B(0)$ exists, then

$$\Big(A(s)B(s)\Big)|_{s=0} = A(0)\Big(B(s)\Big)|_{s=0}$$

b) If $A(s)$ is square, $A(0)$ has full rank, and $\Big(A(s)B(s)\Big)|_{s=0}$ exists, then also $B(0)$ exists.

Proof: To prove (a), write $A(s)$ and $B(s)$ as follows:

$$\begin{aligned} A(s) &= A(0) + sA_1(s) \\ B(s) &= B(0) + sB_1(s) \end{aligned}$$

Since both $A(s)$ and $B(s)$ exists, the last terms must go to zero as s goes to zero. Now study the relation

$$A(s)B(s) = A(0)B(0) + sA_1(s)B(0) + sB_1(s)A(0) + s^2A_1(s)B_1(s)$$

All terms, except $A(0)B(0)$, on the right hand side will become zero as $s \to 0$ which proves the (a)-part of the lemma.

To prove (b), we will use an indirect proof. Assume $B(0)$ does not exist. This means that some column $b_i(0)$ in $B(0)$ does not exist which further implies

$$\|b_i(s)\| \longrightarrow \infty \quad \text{as} \quad s \longrightarrow 0$$

Let $\underline{\sigma}(A(s))$ denote the smallest singular value of $A(s)$. Since $A(0)$ has full rank, there exists a constant C such that $\underline{\sigma}(A(s)) \geq C > 0$ for small s. This implies that for small s it holds that

$$0 < C\|b_i(s)\| \leq \underline{\sigma}(A(s))\|b_i(s)\| \leq \|A(s)b_i(s)\|$$

Now let $s \to 0$ which implies that $\|A(s)b_i(s)\| \to \infty$. The matrix $\Big(A(s)B(s)\Big)|_{s=0}$ can therefore not exist. ■

Lemma 8.2 *A fault f is strongly detectable in a system if and only if*

$$\big(N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix}\big)|_{s=0} \neq 0 \tag{8.26}$$

where the rows of $N_M(s)$ is a polynomial basis for $\mathcal{N}_L\{M(s)\}$ and $N_M(0)$ has full row-rank.

Proof: The basis $N_M(s)$ can be written

$$N_M(s) = R(s)N_M^{irr}(s)$$

where $R(s)$ is a greatest common divisor with full rank and $N_M^{irr}(s)$ is an irreducible basis. Since $N_M(0)$ has full row-rank, $R(0)$ must have full rank.

Now study

$$I = \big(R(s)R^{-1}(s)\big)|_{s=0} = R(0)\big(R^{-1}(s)\big)|_{s=0} = R(0)R^{-1}(0)$$

where we have used Lemma 8.1 in the second equality. This means that $R^{-1}(0)$ must exist and have full rank.

The condition (8.25) for strong fault detectability can be written as

$$0 = \Big(N_M^{irr}(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix}\Big)|_{s=0} = \Big(R^{-1}(s)N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix}\Big)|_{s=0} = \\ = R^{-1}(0)\Big(N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix}\Big)|_{s=0}$$

where the last equality follows from Lemma 8.1. Since $R^{-1}(0)$ has full rank, this condition is equivalent to

$$\big(N_M(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix}\big)|_{s=0} = 0$$

The negation of this condition is then equivalent to (8.26), which proves the lemma. ■

Now return to the proof of Theorem 8.7:

Proof: Let the row vectors of $V(s)$ be a minimal polynomial basis for $\mathcal{N}_L(M_s(s))$ and form $W(s) = V(s)P$, where P is, as before,

$$P = \begin{bmatrix} I_m & -D_u \\ 0 & -B_u \end{bmatrix}$$

According to Theorem 7.4, the rows of $W(s)$ form a polynomial basis for $\mathcal{N}_L(M(s))$. Now note that

$$\begin{aligned}[W(s)\ 0] =& V(s)[P\ M_s(s)] = \begin{bmatrix} I & -D_u & C & D_d \\ 0 & -B_u & A-sI & B_d \end{bmatrix} = \\ =& V(s) \begin{bmatrix} I & -D_u & C_x & C_z & D_d \\ 0 & -B_{u,x} & A_x - sI & A_{12} & B_{d,x} \\ 0 & 0 & 0 & A_z - sI & 0 \end{bmatrix}\end{aligned}$$

In the last equality, we have used the assumption of a realization on the form (7.12). The controllability of the state x from u and d implies, via the PBH test, that the middle block of rows in the matrix $[P\ M_s(s)]$, has full row-rank. Also, A_z has full row-rank because of the assumption that the state z is asymptotically stable. Therefore, the matrix $[P\ M_s(s)]$ has full row-rank for $s = 0$. Since $V(s)$ is irreducible, it has also full row-rank for $s = 0$. This implies that $W(0)$ has full row-rank.

Now consider the relation

$$\begin{aligned}W(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix} =& V(s)P \begin{bmatrix} L(s) \\ 0 \end{bmatrix} = V(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix} = V(s) \begin{bmatrix} C(sI-A)^{-1}B_f + D_f \\ 0 \end{bmatrix} = \\ =& [V_1(s) \quad V_1(s)C(sI-A)^{-1}] \begin{bmatrix} D_f \\ B_f \end{bmatrix} = V(s) \begin{bmatrix} D_f \\ B_f \end{bmatrix} \end{aligned} \tag{8.27}$$

The last equality follows from the fact that $V(s)M_s(s) = 0$. The relation (8.27) implies that

$$\left(W(s) \begin{bmatrix} L(s) \\ 0 \end{bmatrix}\right)|_{s=0} = 0 \Longleftrightarrow \left(V(s) \begin{bmatrix} D_f \\ B_f \end{bmatrix}\right)|_{s=0} = 0$$

This equality together with the fact that $W(0)$ has full row-rank, implies that we can apply Lemma 8.2, which proves the theorem. ∎

8.3.4 Using the Chow-Willsky Scheme

The criterion for strong fault detectability, corresponding to Theorem 8.4, becomes as follows:

Theorem 8.8 *A fault is strongly detectable if and only if*

$$(N_{RH} P\mu)_{\rho=n} \neq 0 \tag{8.28}$$

where N_{RH} is a basis for the left null space of $[R\,H]$ and $\mu = [1\,0 \ldots 0]^T$.

To prove this theorem we first need the following lemma.

Lemma 8.3 *Assume the rows of the matrix W define the largest and uppermost set of primary dependent rows in $[R\ H]_{\rho=n}$. Then $F(s) = W[\Psi_m(s)\ -Q\Psi_{k_u}(s)]$ is a polynomial basis for $\mathcal{N}(M(s))$ and $F(0) = W[\Psi_m(0)\ -Q\Psi_{k_u}(0)]$ has full row rank.*

Proof: From Theorem 7.12, it is clear that $F(s)$ is a polynomial basis. To prove that $F(0)$ has full row rank, we first partition the matrix W as

$$W = \begin{bmatrix} W_{11} & 0 \\ W_{21} & W_{22} \end{bmatrix}$$

where W_{11} has m columns and the first row of W_{22} is *not* zero. Let k denote the number of rows in W_{11}. Then we note that the first k rows of $F(s)$ can be written as $[W_{11}\ \ -W_{11}D_u]$ and has full row-rank for *all* s, i.e. the first k rows of $F(0)$ has full row rank. This means that if $F(0)$ has not full row rank, there must exist a row-vector $\phi = [\phi_1 \dots \phi_p\ 1\ 0 \dots 0]$ where $p \geq k$ and $\phi F(0) = 0$. This further implies that

$$\begin{aligned} &\phi W[\Psi_m(0)\ \ -Q\Psi_{k_u}(0)\ \ R\ \ H] = \\ &\quad = \phi W \begin{bmatrix} I & -D_u & C & D_d & & \\ 0 & -CB_u & CA & CB_d & D_d & \\ \vdots & \vdots & \vdots & \vdots & & \ddots \\ 0 & -CA^{n-1}B_u & CA^n & CA^{n-1}B_d & \dots & CB_d\ \ D_d \end{bmatrix} = 0 \end{aligned}$$

Since the first block column contains the identity matrix I, it must hold that

$$\begin{aligned} &\phi \begin{bmatrix} 0 & 0 \\ 0 & W_{22} \end{bmatrix} [\Psi_m(0)\ \ -Q\Psi_{k_u}(0)\ \ R\ \ H] = \\ &\quad = \phi' W_{22} \begin{bmatrix} -CB_u & CA & CB_d & D_d & & \\ \vdots & \vdots & \vdots & & \ddots & \\ -CA^{n-1}B_u & CA^n & CA^{n-1}B_d & \dots & CB_d & D_d \end{bmatrix} = 0 \end{aligned} \quad (8.29)$$

Next it can be realized that

$$\begin{aligned} &\begin{bmatrix} -CB_u & CA & CB_d & D_d & & \\ \vdots & \vdots & \vdots & & \ddots & \\ -CA^{n-1}B_u & CA^n & CA^{n-1}B_d & \dots & CB_d & D_d \end{bmatrix} = \\ &\quad = \begin{bmatrix} C & D_d & & & \\ \vdots & \vdots & \ddots & & \\ CA^{n-1} & CA^{n-2} & \dots & CB_d & D_d \end{bmatrix} \begin{bmatrix} -B_u & A & B_d & 0 \\ 0 & & 0 & I_{nk_d} \end{bmatrix} = \\ &= \begin{bmatrix} C & D_d & & & \\ \vdots & \vdots & \ddots & & \\ CA^{n-1} & CA^{n-2} & \dots & CB_d & D_d \end{bmatrix} \begin{bmatrix} -B_{u,x} & A_x & A_{12} & B_{d,x} & 0 \\ 0 & 0 & A_z & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{nk_d} \end{bmatrix} \end{aligned}$$

Since the pair $\{A_x, [B_{u,x}\ B_{d,x}]\}$ is controllable, it follows, via the PBH-test, that the uppermost block of rows in the rightmost matrix, has full row-rank. Further, the fact that z is asymptotically stable implies that A_z has full rank, and therefore, the whole rightmost matrix has full row-rank. This means that (8.29) implies that

$$\phi' W_{22} \begin{bmatrix} C & D_d & & & \\ \vdots & \vdots & \ddots & & \\ CA^{n-1} & CA^{n-2} & \dots & CB_d & D_d \end{bmatrix} = \phi' W_{22}[R'\ H'] = 0$$

This means that

$$\begin{aligned} &\phi' W_{22}[R'\ H'] = \\ &= [\phi_{k+1} \dots \phi_p\ 1] \begin{bmatrix} w_{k+1,m+1} & \dots w_{k+1,\mu_1} & 0 & & \dots & 0 \\ \vdots & & & & & \\ w_{p+1,m+1} & \dots & & w_{p+1,\mu_p} & 0 \ \dots & 0 \end{bmatrix} [R'\ H'] = \\ &\qquad = [\bar{w}_{m+1}\ \dots\ \bar{w}_{\mu_p - 1}\ w_{p+1,\mu_p}\ 0 \dots 0][R'\ H'] = \bar{w}[R'\ H'] \end{aligned}$$

The row vector $\bar{w}$ defines a dependent row of $[R'\ H']$ or equivalently of $[R\ H]$. By comparing $\bar{w}$ and the row vector $[w_{p+1,\mathbf{1}}\ \dots\ w_{p+1,\mu_p}\ 0\ \dots\ 0]$, it can be concluded that the dependent row defined by $\bar{w}$ is actually above the dependent row defined by the row vector $[w_{p+1,1}\ \dots\ w_{p+1,\mu_p}\ 0\ \dots\ 0]$ in W. This means that W can not define the uppermost set of primary dependent rows of $[R\ H]$. This contradiction means that $F(0) = W[\Psi_m(0)\ -Q\Psi_{k_u}(0)]$ must have full row rank. ∎

Now return to the proof of Theorem 8.8:

Proof: We will start with the *only-if* part of the proof and an indirect proof is used. Therefore assume that

$$(N_{RH} P \mu)_{\rho=n} = 0 \tag{8.30}$$

Let the rows of a matrix W define the largest and uppermost set of primary dependent rows in $[R\ H]_{\rho=n}$. Then according to Lemma 8.3, $F(s) = W[\Psi_m(s)\ - Q\Psi_{k_u}(s)]$ becomes a polynomial basis for $\mathcal{N}_L\{M(s)\}$ and $F(0)$ has full row rank.

Define $X(s)$ as follows:

$$X(s) = \sum_{i=1}^{\infty} s^{-i} A^{i-1} B_f$$

Then by using the same reasoning as in the formulas (7.39), (7.40), and (7.41), we can conclude that

$$\Psi_m(s) L(s) = RX(s) + P\Psi_1(s)$$

Now assume that (8.30) holds. This implies the following:

$$\begin{aligned}\big(F(s)\begin{bmatrix}L(s)\\0\end{bmatrix}\big)|_{s=0} &= \big(W[\Psi_m(s) \quad -Q\Psi_{k_u}(s)]\begin{bmatrix}L(s)\\0\end{bmatrix}\big)_{s=0} = \\ &= \big(W\Psi_m(s)L(s)\big)_{s=0} = \big(W(RX(s)+P\Psi_1(s))\big)_{s=0} =^* \\ &=^* WP\Psi_1(0) = WP\mu = 0 \end{aligned} \tag{8.31}$$

The equality marked $=^*$ holds since $W[R\ H] = 0$ and the last equality holds because of (8.30). Since $F(s)$ is a polynomial basis for $\mathcal{N}_L\{M(s)\}$ and $F(0)$ has full row rank, Lemma 8.2 implies that the fault is not strongly detectable. Thus the *only-if* part of the proof has been shown.

Also for the *if* part, an indirect proof will be used. Therefore we assume that the fault is *not* strongly detectable and want to prove that (8.30) holds. Assume that w_1 is an arbitrary row-vector in N_{RH} which means that $w_1[R\ H] = 0$. Pick other w_i:s so that $W = [w_1^T w_2^T \dots]^T$ defines a set of primary dependent rows in $[R\ H]$. This implies that $W[R\ H] = 0$ and according to Theorem 7.12, $F(s) = W[\Psi_m(s) \quad -Q\Psi_{k_u}(s)]$ becomes a polynomial basis (not necessarily irreducible) for $\mathcal{N}_L\{M(s)\}$. Then we know that for some polynomial matrix $\phi(s)$, it holds that $F(s) = \phi(s)N_M(s)$, where $N_M(s)$ is a minimal polynomial basis for $\mathcal{N}_L(M(s))$. Theorem 8.6 together with the assumption that the fault is not strongly detectable implies that

$$\Big(F(s)\begin{bmatrix}L(s)\\0\end{bmatrix}\Big)_{s=0} = \Big(\phi(s)N_M(s)\begin{bmatrix}L(s)\\0\end{bmatrix}\Big)_{s=0} =^* \phi(0)\Big(N_M(s)\begin{bmatrix}L(s)\\0\end{bmatrix}\Big)_{s=0} = 0$$

where the equality marked $=^*$ holds because of Lemma 8.1. Also we have that

$$\begin{aligned}\big(F(s)\begin{bmatrix}L(s)\\0\end{bmatrix}\big)_{s=0} =& \big(W[\Psi_m(s) \quad -Q\Psi_{k_u}(s)]\begin{bmatrix}L(s)\\0\end{bmatrix}\big)_{s=0} = \big(W\Psi_m(s)L(s)\big)_{s=0} = \\ =& \big(W(RX(s)+P\Psi_1(s))\big)_{s=0} = WP\Psi_1(0) = \\ =& WP\mu = 0\end{aligned}$$

This implies that $w_1 P\mu = 0$ which proves the *if* part. ∎

Note that only constant matrices are involved in Theorem 8.8 which implies that the condition (8.28) can also be written

$$\operatorname{Im} P\mu \not\subseteq \operatorname{Im}[R\ H]$$

8.4 Discussions and Comparisons

In the previous two section, we have given a number of different criterions for fault detectability and strong fault detectability. When faced with a real problem, we want to know what criterion that is the most suitable.

If the system model is given on transfer function form and we want to check fault detectability, then the easiest approach is probably the "frequency domain

approach", i.e. the criterion given by Theorem 8.2. The reason is that, compared to the "intuitive approach", we do not need to care about the transfer function $G(s)$. To use this criterion, the rank test described in Section 8.2.1, is probably the preferred method.

If the system model is given on state-space form and we want to check fault detectability, it is probably the criterion based on the system matrix, i.e. Theorem 8.3, that is the best choice. The reason for this is that in Section 7.5.2, we noted that the Chow-Willsky scheme is more numerically sensitive than the minimal polynomial basis approach. However note that the criterion based on the Chow-Willsky scheme, i.e. Theorem 8.4, uses only constant matrices, in contrast to the criterion based on the system matrix. This might in some cases be an advantage since we do not need special algorithms that can handle polynomial matrices. No matter what the preferred criterion is, in both cases, the actual test is probably most easily performed by the rank test.

If the system model is given on transfer function form and we want to check strong fault detectability, there is only one alternative. Since the "frequency domain approach" doesn't work we have to use the "intuitive approach", i.e. Theorem 8.6.

Finally, if the system model is given on state-space form and we want to check strong fault detectability, the criterion based on the system matrix, i.e. Theorem 8.7, is probably the best choice. The reason is again the numerical considerations from Section 7.5.2. However an advantage with the criterion based on the Chow-Willsky scheme, i.e. Theorem 8.8, is that only constant matrices are needed and also that the rank test is possible to perform. That is, we do not need to calculate a null-space.

All the criterions for models given on state-space form, have been formulated without the need to care about controllability from u and d. This is in contrast to the design of polynomial parity functions for which we saw in Chapter 7 that for both the minimal polynomial basis approach and the Chow-Willsky scheme, controllability from u and d was important to be able to find all parity functions.

If we want to, it is however possible to check fault detectability and strong fault detectability using a minimal state-space representation in which the state is controllable from u and d. This means that we are neglecting the states that are controllable from only the fault. For example for the Chow-Willsky scheme, the criterion for fault detectability becomes

Theorem 8.9 *A fault f is detectable in a system if and only if*

$$\operatorname{Im} P_{\rho=n_x} \not\subseteq \operatorname{Im} [R_x \; H]_{\rho=n_x}$$

and the criterion for strong fault detectability becomes

Theorem 8.10 *A fault f is strongly detectable in a system if and only if*

$$\operatorname{Im} (P\mu - R_z A_z^{-1} B_{f,z})_{\rho=n_x} \not\subseteq \operatorname{Im} [R_x \; H]_{\rho=n_x}$$

The proofs of both these theorems can be found in (Nyberg, 1997).

8.5 Examples

In an inverted pendulum example in (Chen and Patton, 1994), an observer based residual generator was used. It was shown that no residual generator with this specific structure could strongly detect a fault in sensor 1. It was posed as an open question if any residual generator, in which this fault is strongly detectable, exists and in that case how to find it. In the following example, this problem is re-investigated by means of the theorems from this section.

Example 8.7

The system description, from (Chen and Patton, 1994), represents a continuous model of an inverted pendulum. It has one input and three outputs:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -1.93 & -1.99 & 0.009 \\ 0 & 36.9 & 6.26 & -0.174 \end{bmatrix} \qquad D = 0_{3\times 1}$$

$$B = [0 \;\; 0 \;\; -0.3205 \;\; -1.009]^T \qquad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

The faults considered are sensor faults. There are no disturbances and also, there are no states controllable only from faults.

To check both fault detectability and strong fault detectability, we set up the matrix $M_s(s)$ and calculate a basis $N_{M_s}(s)$ for the left null-space of $M_s(s)$. Then we calculate

$$N_{M_s}(s)\begin{bmatrix} D_f \\ B_f \end{bmatrix} =$$

$$= \begin{bmatrix} s & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & -0.009s + 1.93 & s + 1.99 & 0 & -0.009 & 1 & 0 \\ 0 & s^2 + 0.174s - 36.9 & -6.26 & 0 & s + 0.174 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} =$$

$$= \begin{bmatrix} s & 0 & -1 \\ 0 & -0.009s + 1.93 & s + 1.99 \\ 0 & s^2 + 0.174s - 36.9 & -6.26 \end{bmatrix} \neq 0 \quad (8.32)$$

Now using Theorem 8.3, we can conclude that all three sensor faults are detectable. To check strong detectability, we substitute s with 0. Then the first column in (8.32) becomes zero and the other non-zero. By using Theorem 8.7 we then conclude that the second and third sensor faults are strongly detectable, i.e. for each of these faults, a residual generator can be found for which the

fault is strongly detectable. Also concluded is that the first sensor fault is only weakly detectable. Thus, the answer to the open question, posed in (Chen and Patton, 1994), is that it is not possible to construct a residual generator in which the fault in sensor 1 is strongly detectable. ■

Example 8.8

Consider again the design example given in Section 7.6. In Figure 7.3 it is seen that the transfer function from f_1 to the residual r has zero DC-gain. This can be validated by using Theorem 8.1 and the basis $N_M(s)$ from (7.58):

$$\left(N_M(s)\begin{bmatrix}L(s)\\0\end{bmatrix}\right)|_{s=0} = \begin{bmatrix}0 & 0.0538 & 0.091394 & 0.12 & -1 & 0\\ 0 & -6.6653 & -16.5141 & 31.4058 & 0 & 0\end{bmatrix}\begin{bmatrix}1\\0\\0\\0\\0\\0\end{bmatrix} = 0$$

Thus, the fault in sensor 1 is not strongly detectable. ■

8.6 Conclusions

In this chapter, criterions for fault detectability and strong fault detectability, seen as system properties, have been derived. A few of these were known earlier but most of them are new. In particular, to the authors knowledge, general condition for strong fault detectability has not been presented elsewhere.

Criterions for models given both on transfer function form and state-space form are considered. All the proofs, for the different criterions, become quite simple thanks to the notion of bases for linear residual generators, introduced in the previous chapter.

For the case of strong fault detectability, it is shown that the existence of integrations in the system, can not be used, neither as a necessary nor sufficient condition.

Bibliography

Air Leakage Detector for IC Engine (1994), *Patent RD 368014* .

Basseville, M. (1997), 'Information criteria for residual generation and fault detection and isolation', *Automatica* **33**(5), 783–803.

Basseville, M. and Nikiforov, I. (1993), *Detection of Abrupt Changes*, PTR Prentice-Hall, Inc.

Berger, J. O. (1985), *Statistical Decision Theory and Bayesian Analysis*, Springer.

Bøgh, S. (1995), 'Multiple hypothesis-testing approach to fdi for the industrial actuator benchmark', *Control Engineering Practice* **3**(12), 1763–1768.

Bøgh, S. (1997), Fault Tolerant Control Systems - a Development Method and Real-Life Case Study, PhD thesis, Aalborg University.

California's OBD-II Regulation (1993), *(section 1968.1, Title 13, California Code of Regulations), Resolution 93-40, July 9* pp. 220.7 – 220.12(h).

Callier, F. (1985), 'On polynomial matrix spectral factorization by symmetric factor extraction', *IEEE Trans. Automatic Control* **30**(5), 453–464.

Casella, G. and Berger, R. (1990), *Statistical Inference*, Duxbury Press.

Chen, C.-T. (1984), *Linear System Theory and Design*, Holt, Rinehart and Winston, New York.

Chen, J. and Patton, R. (1994), A re-examination of fault detectability and isolability in linear dynamic systems, Fault Detection, Supervision and Safety for Technical Processes, IFAC, Espoo, Finland, pp. 567–573.

Chen, J. and Patton, R. J. (1999), *RobusT Model-Based Fault Diagnosis for Dynamic Systems*, Kluwer Academic Publishers.

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

Clark, R. (1979), The dedicated observer approach to instrument fault detection, Proc. of the 15th CDC, pp. 237–241.

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

Ding, X. and Frank, P. (1991), Frequency domain approach and threshold selector for robust model-based fault detection and isolation, IFAC Fault Detection, Supervision and Safety for Technical Processes, Baden-Baden, Germany, pp. 271–276.

Forney, G. (1975), 'Minimal bases of rational vector spaces, with applications to multivariable linear systems', *SIAM J. Control* **13**(3), 493–520.

Frank, P. (1990), 'Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - a survey and some new results', *Automatica* **26**(3), 459–474.

Frank, P. (1993), Advances in observer-based fault diagnosis, Proc. TOOLDIAG'93, CERT, Toulouse, France, pp. 817–836.

Frank, P. and Ding, X. (1994*a*), 'Frequency domain approach to optimally robust residual generation and evaluation for model-based fault diagnosis', *Automatica* **30**(5), 789–804.

Frank, P. and Ding, X. (1994*b*), 'Frequency domain approach to optimally robust residual generation and evaluation for model-based fault diagnosis', *Automatica* **30**(5), 789–804.

Frisk, E. (1998), Residual Generation for Fault Diagnosis: Nominal and Robust Design, Licentiate thesis LIU-TEK-LIC-1998:74, Linkping University.

Frisk, E. and Nielsen, L. (1999), Robust residual generation for diagnosis including a reference model for residual behavior, IFAC.

Frisk, E., Nyberg, M. and Nielsen, L. (1997), FDI with adaptive residual generation applied to a DC-servo, Fault Detection, Supervision and Safety for Technical Processes, IFAC, Hull, United Kingdom.

Gertler, J. (1991), Analytical redundancy methods in fault detection and isolation; survey and synthesis, IFAC Fault Detection, Supervision and Safety for Technical Processes, Baden-Baden, Germany, pp. 9–21.

Gertler, J. (1998), *Fault Detection and Diagnosis in Engineering Systems*, Marcel Dekker.

Gertler, J., Costin, M., Fang, X., Hira, R., Kowalalczuk, Z., Kunwer, M. and Monajemy, R. (1995), 'Model based diagnosis for automotive engines - algoritm development and testing on a production vehicle', *IEEE Trans. on Control Systems Technology* **3**(1), 61–69.

Gertler, J., Costin, M., Fang, X., Hira, R., Kowalczuk, Z. and Luo, Q. (1991), Model-based on-board fault detection and diagnosis for automotive engines, IFAC Fault Detection, Supervision and Safety for Technical Processes, Baden-Baden, Germany, pp. 503–508.

Gertler, J., Fang, X. and Luo, Q. (1990), 'Detection and diagnosis of plant failures: the orthogonal parity equation approach', *Control and Dynamic Systems* **37**, 159–216.

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

Gertler, J. and Singer, D. (1990), 'A new structureal framework for parity equation-based failure detecation and isolation', *Automatica* **26**(2), 381–388.

G.H. Golub, C. v. L. (1996), *Matrix Computations*, third edition edn, John Hopkins.

Grainger, R., Holst, J., Isaksson, A. and Ninnes, B. (1995), 'A parametric statistical approach to fdi for the industrial actuator benchmark', *Control Engineering Practice* **3**(12), 1757–1762.

Gustavsson, F. and Palmqvist, J. (1997), Change detection design for low false alarm rates, IFAC Fault Detection, Supervision and Safety for Technical Processes, Hull, England, pp. 1021–1026.

Hendricks, E. (1990), 'Mean value modelling of spark ignition engines', *SAE–Technical Paper Series* (900616).

Henrion, D., Kraffer, F., Kwakernaak, H., M.Sebek, S. P. and Strijbos, R. (1997), The Polynomial Toolbox for Matlab, URL: http://www.math.utwente.nl/polbox/.

Heywood, J. B. (1992), *Internal Combustion Engine Fundamentals*, McGraw-Hill series in mechanical engineering, McGraw-Hill.

Höfling, T. (1993), Detection of parameter variations by continuous-time parity equations, IFAC World Congress, Sydney, Australia, pp. 513–518.

Höfling, T. and Isermann, R. (1996), 'Fault detection based on adaptive parity equations and single-parameter tracking', *Control Eng. Practice* **4**(10), 1361–1369.

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

Kailath, T. (1980), *Linear Systems*, Prentice-Hall.

Krishnaswami, V., Luh, G. and Rizzoni, G. (1994), Fault detection in IC engines using nonlinear parity equations, Proceedings of the American Control Conference, Baltimore, Maryland, pp. 2001–2005.

Kung, S., Kailath, T. and Morf, M. (1977), Fast and stable algorithms for minimal design problems, Int. Symp. on Multivariable Technological Systems, IFAC, pp. 97–104.

Lancaster, P. and Tismenetsky, M. (1985), *The theory of matrices*, 2nd edn, Academic Press.

Larsson, M. (1997), On Modeling and Diagnosis of Discrete Event Dynamic Systems, Licentiate thesis LIU-TEK-LIC-1997:49, Linkping University.

Lehmann, E. L. (1986), *Testing Statistical Hypotheses*, second edn, Springer Verlag.

Ljung, L. (1987), *System Identification: Theory for the User*, Prentice Hall.

Lou, X., Willsky, A. and Verghese, G. (1986), 'Optimally robust redundancy relations for failure detection in uncertain systems', *Automatica* **22**(3), 333–344.

Luenberger, D. (1989), *Linear and Nonlinear Programming*, Addison Wesley.

Maciejowski, J. (1989), *Multivariable Feedback Design*, Addison-Wesley.

Magni, J. and Mouyon, P. (1994), 'On residual generation by observer and parity space approaches', *IEEE Trans. on Automatic Control* **39**(2), 441–447.

Massoumnia, M. and Velde, W. (1988), 'Generating parity relations for detecting and identifying control system component failures', *Journal of Guidance, Control, and Dynamics* **11**(1), 60–65.

Massoumnia, M., Verghese, G. and Willsky, A. (1989), 'Failure detection and identification', *IEEE Trans. on Automatic Control* **AC-34**(3), 316–321.

McCluskey, E. (1966), 'Minimization of boolean functions', *Bell System Technical Journal* **35**(6), 1417–1444.

Mironovskii, L. (1980), 'Functional diagnosis of linear dynamic systems', *Automation and Remote Control* pp. 1198–1205.

Nikoukhah, R. (1994), 'Innovations generation in the presence of unknown inputs: Application to robust failure detection', *Automatica* **30**(12), 1851–1867.

Nyberg, M. (1997), *Model Based Diagnosis with Application to Automotive Engines*, Licentiate Thesis, Linköping University, Sweden.

Nyberg, M. (1998), SI-engine air-intake system diagnosis by automatic FDI-design, IFAC Workshop Advances in Automotive Control, Columbus, Ohio, pp. 225–230.

Nyberg, M. and Frisk, E. (1999), A minimal polynomial basis solution to residual generation for fault diagnosis in linear systems, IFAC, Beijing, China.

Nyberg, M. and Nielsen, L. (1997*a*), Design of a complete FDI system based on a performance index with application to an automotive engine, Proc. IFAC Fault Detection, Supervision and Safety for Technical Processes, Hull, United Kingdom, pp. 812–817.

Nyberg, M. and Nielsen, L. (1997*b*), 'Model based diagnosis for the air intake system of the SI-engine', *SAE Paper* (970209).

Nyberg, M. and Nielsen, L. (1997*c*), Parity functions as universal residual generators and tool for fault detectability analysis, IEEE Conf. on Decision and Control, pp. 4483–4489.

Patton, R. (1994), Robust model-based fault diagnosis:the state of the art, IFAC Fault Detection, Supervision and Safety for Technical Processes, Espoo, Finland, pp. 1–24.

Patton, R., Frank, P. and Clark, R., eds (1989), *Fault diagnosis in Dynamic systems*, Systems and Control Engineering, Prentice Hall.

Patton, R. and Kangethe, S. (1989), *Robust Fault Diagnosis using Eigenstructure Assignment of Observers*, in Patton et al. (1989), chapter 4.

P.H.Garthwaite, I.T. Jolliffe, B. J. (1995), *Statistical Interference*, Prentice Hall.

Potter, J. and Suman, M. (1977), 'Threshold redundancy management with arrays of skewed instruments', *Integrity Electron. Flight Contr. Syst.* pp. 15–11 to 15–25.

Reiter, R. (1987), 'A theory of diagnosis from first principles', *Artificial Intelligence* **32**(1), 57–95.

Riggins, R. and Rizzoni, G. (1990), The distinction between a special class of multiplicative events and additive events: Theory and application to automotive failure diagnosis, American Control Conf., San Diego, California, pp. 2906–2911.

Rosenbrock, H. (1970), *State-Space and Multivariable Theory*, Wiley, New York.

Sandewall, E. (1991), *Tillämpad Logik*, Department of Computer and Information Science, Linköping University, Sweden.

Taylor, C. F. (1994), *The Internal Combustion Engine in Theory and Practice*, second edn, The M.I.T. Press.

Viswanadham, N., Taylor, J. and Luce, E. (1987), 'A frequency-domain approach to failure detection and isolation with application to GE-21 turbine engine control systems', *Control - Theory and advanced technology* **3**(1), 45–72.

White, J. and Speyer, J. (1987), 'Detection filter design: Spectral theory and algorithms', *IEEE Trans. Automatic Control* **AC-32**(7), 593–603.

Wünnenberg, J. (1990), Observer-Based Fault Detection in Dynamic Systems, PhD thesis, University of Duisburg.

Index