UC Berkeley
UC Berkeley Electronic Theses and Dissertations

Title
Probabilistic Performance Analysis of Fault Diagnosis Schemes

Permalink
https://escholarship.org/uc/item/3ss40068

Author
Wheeler, Timothy Josh

Publication Date
2011

Peer reviewed|Thesis/dissertation

eScholarship.org Powered by the California Digital Library
University of California

Probabilistic Performance Analysis of Fault Diagnosis Schemes

by

Timothy Josh Wheeler

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering–Mechanical Engineering

and the Designated Emphasis

in

Computational Science and Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Andrew K. Packard, Co-chair
Professor Peter J. Seiler, Co-chair
Professor Kameshwar Poolla
Professor Laurent El Ghaoui

Fall 2011

Probabilistic Performance Analysis of Fault Diagnosis Schemes

Copyright © 2011 by Timothy Josh Wheeler

Abstract

Probabilistic Performance Analysis of Fault Diagnosis Schemes

by

Timothy Josh Wheeler

Doctor of Philosophy in Engineering–Mechanical Engineering

University of California, Berkeley

Professor Andrew K. Packard, Co-chair
Professor Peter J. Seiler, Co-chair

The dissertation explores the problem of rigorously quantifying the performance of a fault diagnosis scheme in terms of probabilistic performance metrics. Typically, when the performance of a fault diagnosis scheme is of utmost importance, physical redundancy is used to create a highly reliable system that is easy to analyze. However, in this dissertation, we provide a general framework that applies to more complex analytically redundant or model-based fault diagnosis schemes. For each fault diagnosis problem in this framework, our performance metrics can be computed accurately in polynomial-time.

First, we cast the fault diagnosis problem as a sequence of hypothesis tests. At each time, the performance of a fault diagnosis scheme is quantified by the probability that the scheme has chosen the correct hypothesis. The resulting performance metrics are joint probabilities. Using Bayes rule, we decompose these performance metrics into two parts: marginal probabilities that quantify the reliability of the system and conditional probabilities that quantify the performance of the fault diagnosis scheme. These conditional probabilities are used to draw connections between the fault diagnosis and the fields of medical diagnostic testing, signal detection, and general statistical decision theory.

Second, we examine the problem of computing the performance metrics efficiently and accurately. To solve this problem, we examine each portion of the fault diagnosis problem and specify a set of sufficient assumptions that guarantee efficient computation. In particular, we provide a detailed characterization of the class of finite-state Markov chains that lead to tractable fault parameter models. To demonstrate that these assumptions enable efficient computation, we provide pseudocode algorithms and prove that their running time is indeed polynomial.

Third, we consider fault diagnosis problems involving uncertain systems. The inclusion of uncertainty enlarges the class of systems that may be analyzed with our framework. It also addresses the issue of model mismatch between the actual system and the system used

to design the fault diagnosis scheme. For various types of uncertainty, we present convex optimization problems that yield the worst-case performance over the uncertainty set.

Finally, we discuss applications of the performance metrics and compute the performance for two fault diagnosis problems. The first problem is based on a simplified air-data sensor model, and the second problem is based on a linearized vertical take-off and landing aircraft model.

Soli Deo gloria.

Contents

List of Figures

List of Tables

List of Algorithms

Acknowledgements

When I started writing this dissertation, I felt a chill of loneliness as I stared at the blank page. However, now that I am finished, I clearly see that there were many people in my life contributing to my success, well-being, and happiness. It is not possible to name and thank them all here, so I will attempt to acknowledge some of the more prominent figures.

First, I would like to thank my advisors, Professors Andy Packard and Pete Seiler. It is truly a joy to work with such exceptional minds, and I appreciate all the time and effort they invested in my career. I would also like to thank all the past and present residents of the Berkeley Center for Control & Identification for providing a fun and stimulating work environment. In particular, I would like to thank Eilyan Bitar, who always took the time to be a supportive friend, even when his own work was weighing on him.

On a more personal note, I would like to thank all my friends and family for their love and support over the years. My brothers and sisters at New Church Berkeley have been consistently generous with their prayers and words of encouragement. Although a great physical distance separates me from nearly every member of my family, they have all worked together to keep my spirits lifted and my heart warmed. However, no one has contributed more to my graduate studies or this dissertation than my wife, Ellie. She has faithfully supported me in every way possible, and I hope that I can return even a small portion of her kindness as we share the rest of our lives together.

This work was supported financially by funding from NASA (Grant No. NNX07AC40A, *Reconfigurable Robust Gain-Scheduled Control for Air-Breathing Hypersonic Vehicles*) and by the Department of Mechanical Engineering at the University of California, Berkeley.

Timothy J. Wheeler
Berkeley, California
Fall 2011

Chapter 1

Introduction

In safety-critical applications, a system must not only be highly reliable, but that reliability must be certifiable in some way. For example, the Federal Aviation Administration (FAA) requires designers of civil aircraft to demonstrate that their products will have no more than 10^{-9} catastrophic failures per flight-hour [18]. Such demonstrations are based on two factors: the reliability of the system hardware in a given operating environment and the ability of the system to detect when that hardware has failed. In the aviation industry, both of these issues are addressed by the use of parallel redundant components [18,103,104]. This type of redundancy, known as physical redundancy, ensures the availability of the system, even in the presence of component failures. In a physically redundant configuration, a failed component is detected by directly comparing the behavior of each redundant component. Hence, these schemes tend to detect faults accurately, and their performance is relatively simple to certify using fault trees [41,77].

However, in some applications, such as unmanned aerial vehicles (UAVs), the designer cannot afford the extra size, weight, and power needed to support multiple redundant components. In such situations, the analytical redundancies between dissimilar components can be exploited to detect faults. More specifically, mathematical models of the system are used to establish analytical relationships that hold only when the constituent components of the system are functioning properly. Then, when a component fails, one or more of these relationships is violated and the failure can be detected and diagnosed. This approach, known as model-based fault diagnosis [24,48], certainly reduces the number of individual components needed; however, there are two main drawbacks to consider. First, merely identifying a fault cannot prevent system-wide failure if the failed component is indispensable (i.e. no other components can perform the same critical function). Second, the performance of fault detection schemes based on analytical redundancy can be difficult to quantify if the analytical relationships are dynamic or nonlinear. While the first difficulty is unavoidable, this dissertation addresses the second difficulty.

Although there is a vast body of literature on model-based fault diagnosis (see [9,24,48] and references therein), little attention is given to the rigorous performance analysis of model-based fault diagnosis schemes. In this dissertation, we present a set of probabilis-

tic metrics that rigorously quantify the performance of a reasonably general class of fault diagnosis schemes that includes many model-based schemes. Of course, such metrics are only useful if they are efficiently computable. Monte Carlo methods [79] provide a general-purpose solution to this problem, but it can be difficult to quantify the error present in the results. Moreover, component failures are inherently rare by design, so a thorough Monte Carlo analysis would entail the subtleties and complications of rare-event simulation [1]. In this dissertation, we take a more practical approach—we establish a class of linear systems and fault diagnosis schemes for which the performance metrics can be efficiently computed without resorting to approximations. We also consider the effects of adding uncertainty to various aspects of the fault diagnosis problem. Again, emphasizing the need for computational tractability, we describe a set of uncertainty models for which the worst-case performance can be efficiently and accurately computed without the need for approximation.

1.1 Thesis Overview

The terminology and notation used throughout this dissertation are established in **Chapter 2**. For the sake of brevity, only the most basic concepts of probability and reliability theory are introduced. In addition to the core definitions, we present two probabilistic models for component failure times. In this chapter, we also give a brief survey of the field of fault diagnosis. After defining the key terminology used in fault diagnosis, we present a survey of some of the most popular techniques used to design fault diagnosis schemes, and we discuss some of the strategies used to design more reliable systems. Finally, we present a survey of the existing performance analysis techniques that can be found in the literature.

Chapter 3 examines the quantitative performance analysis of a class of fault diagnosis problems, in which faults affect the system via a stochastic parameter. First, we cast the problem of fault detection as a sequence of hypothesis tests regarding the value of the fault parameter at each time. Building on the vast hypothesis testing literature, we establish a set of joint probabilities that fully quantify the time-varying performance of a given fault diagnosis scheme. Bayes' rule is then used to decompose these performance metrics into two parts: conditional probabilities that characterize the performance of the fault diagnosis scheme and marginal probabilities that characterize the reliability of the underlying system. The receiver operating characteristic, a popular tool in hypothesis testing, medical diagnostic testing, and signal detection theory, is used to develop a set of informative visualizations. Finally, the performance analysis framework is extended to the more general problems of fault isolation and fault identification.

In **Chapter 4**, we examine the computational issues involved in evaluating the performance metrics. By examining each component of the fault diagnosis problem separately, we arrive at a set of sufficient conditions and assumptions, which guarantee that the per-

formance metrics can be computed in polynomial time. In particular, we state and prove a number of theoretical results regarding Markov chains with finite state spaces. In this chapter, we also explore a simplified class of systems, based on independent faults with additive effects, for which the performance metrics can be computed even more efficiently. Finally, we present pseudocode algorithms for computing the performance metrics and we prove that their running time is indeed polynomial, given that the aforementioned conditions are met.

Chapter 5 extends the results of Chapters 3 and 4 by considering fault diagnosis problems with some uncertain aspect. In particular, we examine systems with uncertain inputs, unknown disturbances, uncertain fault signals, and unmodeled or uncertain system dynamics. For each type of uncertainty, we consider the problem of computing the worst-case values of the performance metrics over the given uncertainty set. Hence, these performance analyses take the form of optimization problems. We show that, under some reasonable assumptions, these optimization problems can be written as convex programs, which are readily solved using off-the-shelf numerical optimization packages.

Chapter 6 describes some practical applications of the performance metrics and demonstrates these applications on numerical examples. More specifically, we discuss how the performance metrics can be used in engineering applications such as trade studies, selecting a fault diagnosis scheme, and safety certification. We demonstrate some of these applications using two examples. The first is an air-data sensor system, which measures an aircraft's airspeed and altitude. The second example is a linearized model of the longitudinal dynamics of a fixed-wing vertical take-off and landing (VTOL) aircraft.

Finally, **Chapter 7** summarizes the conclusions drawn from this research work and discusses some avenues for future research.

1.2 Thesis Contributions

1. **Performance of fault detection schemes:** In Chapter 3, we present a rigorous probabilistic framework that can be used to assess the performance of any fault diagnosis scheme applied to a system with a parametric fault model. Unlike existing performance analyses, the performance metrics produced by this framework capture the time-varying nature of the fault-diagnosis problem. Moreover, this framework can be applied to the problems of fault detection, fault isolation, and fault identification.

2. **Time-complexity analysis:** By closely examining the time-complexity of each step in computing the performance metrics, we arrive at a broad class of fault diagnosis problems for which our performance analysis is computationally tractable.

   - *Efficient Algorithms:* We present algorithms for efficiently and accurately computing the performance metrics without resorting to Monte Carlo methods or approxima-

tion.

- *Complexity of Markov Chains:* We establish sufficient conditions on the structure of a finite-state Markov chain, which guarantee that the number of paths with nonzero probability grows polynomially. For time-homogeneous Markov chains, the conditions are necessary, as well as sufficient. In each case, the conditions are easily and efficiently verified using a graph-theoretic test.

3. **Worst-case performance of fault detection schemes with uncertain elements:** We extend our performance analysis by considering systems with uncertain input signals and model uncertainty. The worst-case values of the performance metrics are defined as the optimum points of two optimization problems. We show that, under reasonable assumptions, these optimization problems may be written as convex programs that are easily solved using off-the-shelf numerical optimization routines.

Chapter 2

Background

2.1 Introduction

The purpose of this chapter is to establish the context and background for our discussion of probabilistic fault diagnosis problems. First, we provide a brief summary of the key definitions of probability theory. Then, we review some standard terminology and definitions from reliability theory. Finally, we provide a brief survey of fault diagnosis. This survey includes a list of commonly-used terminology, an outline of the key techniques used to design fault diagnosis schemes, and some comments on existing performance analyses for fault diagnosis problems.

2.2 Probability Theory

In this section, we review the basic definitions of probability theory and establish some notation. A complete survey of probability theory is beyond the scope of this dissertation, and the informal definitions stated here are only meant to clarify the subsequent usage of probability notation. See Rosenthal [81] or Williams [99] for a rigorous measure-theoretic introduction to probability theory, and see Papoulis and Pillai [72] or Jazwinski [50] for an introduction to stochastic processes.

2.2.1 Foundations

Suppose that Ω is a nonempty set called the *sample space*. Each point $\omega \in \Omega$ is an *outcome*. Assume that $\mathscr{F}$ is a *σ-algebra* of subsets of Ω. Each set $E \in \mathscr{F}$ is called a *event*. Let $\mathbf{P}$ be a measure on the measurable space $(\Omega, \mathscr{F})$, such that $\mathbf{P}(\Omega) = 1$. Then, $\mathbf{P}$ is called a *probability measure* and the triple $(\Omega, \mathscr{F}, \mathbf{P})$ is called a *probability space*.

Given a space S, let $\mathscr{T}$ be a topology defined on S. Then, a *Borel set* is any subset of S that can be formed by taking a countable union, a countable intersection, or the complement of open sets in $\mathscr{T}$. The collection of Borel sets in S, denoted $\mathscr{B}(S)$, forms a σ-algebra known as the *Borel σ-algebra*. We use the simpler notation $\mathscr{B}$ when the space S is clear from context.

Given an event $B \in \mathscr{F}$ with $\mathbf{P}(B) > 0$, the *conditional probability* of any event $A \in \mathscr{F}$, given B, is defined as

$$\mathbf{P}(A \mid B) = \frac{\mathbf{P}(A \cap B)}{\mathbf{P}(B)}.$$

Essentially, the function $\mathbf{P}(\bullet \mid B)$ is a probability measure on the space $(B, \mathscr{G})$, where

$$\mathscr{G} := \{A \cap B : A \in \mathscr{F}\} \subset \mathscr{F}.$$

Note that the conditional probability $P(A \mid B)$ is undefined if $\mathbf{P}(B) = 0$.

2.2.2 Random Variables

Given a probability space $(\Omega, \mathscr{F}, \mathbf{P})$ and a measurable space $(S, \mathscr{E})$, a *random variable* is a measurable function $x\colon \Omega \to S$. That is, for all $E \in \mathscr{E}$, the preimage $x^{-1}(E)$ is in $\mathscr{F}$. In this dissertation, we mainly use random variables taking values in the measurable space $\big(\mathbb{R}^n, \mathscr{B}(\mathbb{R}^n)\big)$. Given a random variable x and a measurable set $B \in \mathscr{B}(\mathbb{R}^n)$, the event

$$x^{-1}(B) = \big\{\omega \in \Omega \,:\, x(\omega) \in B\big\}$$

is often written using the informal notation $\{x \in B\}$. The *cumulative distribution function* (CDF) of x is defined for all $c \in \mathbb{R}^n$ as the probability

$$P_x(c) := \mathbf{P}\big(\{x_1 \le c_1\} \cap \{x_2 \le c_2\} \cap \cdots \cap \{x_n \le c_n\}\big).$$

Informally speaking[1], the *probability density function* (PDF) of x is a function $p_x\colon \mathbb{R}^n \to \mathbb{R}_+$, such that

$$\mathbf{P}(x \in B) = \int_{x^{-1}(B)} \mathrm{d}\mathbf{P} = \int_B p_x(s)\,\mathrm{d}s,$$

for any $B \in \mathscr{B}(\mathbb{R}^n)$. If the partial derivatives exists, then p_x can be defined for all $c \in \mathbb{R}^n$ as

$$p_x(c) := \left.\frac{\partial^n P_x}{\partial x_1 \cdots \partial x_n}\right|_{x=c}.$$

If x takes countably many values in $\mathbb{R}^n$, then the *probability mass function* (PMF), defined as

$$p_x(c) = \mathbf{P}(x = c),$$

for all $c \in x(\Omega)$, takes the place of the PDF.

If two random variables are defined on the sample space, then they are said to be *jointly*

[1]Technically, the probability density function of x, if it exists, is defined as the Radon–Nikodym derivative of the measure $\mathbf{P} \circ x^{-1}$ with respect to Lebesgue measure on $\mathbb{R}^n$. Precise conditions for the existence of the Radon–Nikodym derivative can found in [82].

distributed. Let $x\colon \Omega \to \mathbb{R}^m$ and $y\colon \Omega \to \mathbb{R}^n$. The joint CDF of x and y is defined as

$$P_{x,y}(c,d) \coloneqq \mathbf{P}\big(\{x_1 \le c_1\} \cdots \cap \{x_m \le c_m\} \cap \{y_1 \le d_1\} \cdots \cap \{y_n \le d_n\}\big),$$

for any $c \in \mathbb{R}^m$ and $d \in \mathbb{R}^n$, and the joint PDF is a function $p_{x,y}\colon \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}_+$, such that

$$\mathbf{P}(x \in A, y \in B) = \int_{x^{-1}(A)\cap y^{-1}(B)} \mathrm{d}\mathbf{P} = \int_A \int_B p_{x,y}(s,t)\,\mathrm{d}s\,\mathrm{d}t,$$

for any $A \in \mathscr{B}(\mathbb{R}^m)$ and $B \in \mathscr{B}(\mathbb{R}^n)$. If x and y are jointly distributed, then the *marginal density* of y is defined as

$$p_y(d) = \int_{\mathbb{R}^m} p_{x,y}(t,d)\,\mathrm{d}t,$$

for all $d \in \mathbb{R}^n$. The marginal density p_x is similarly defined. The conditional distribution of x given y is defined as

$$p_{x|y}(s \mid t) \coloneqq \frac{p_{x,y}(s,t)}{p_y(t)},$$

for all $s \in \mathbb{R}^m$ and all $t \in \mathbb{R}^n$, such that $p_y(t) > 0$.

2.2.3 Expectation, Mean, and Variance

Given a function $f\colon \mathbb{R}^n \to \mathbb{R}^m$ and a random variable $x\colon \Omega \to \mathbb{R}^n$, the *expected value* of the random variable $f(x)$ is defined as

$$\mathbf{E}\big(f(x)\big) \coloneqq \int_\Omega f\big(x(\omega)\big)\,\mathrm{d}\mathbf{P}(\omega),$$

when the integral exists. If the PDF p_x exists, then $\mathbf{E}\big(f(x)\big)$ may be written as

$$\mathbf{E}\big(f(x)\big) = \int_{\mathbb{R}^n} f(s)\,p_x(s)\,\mathrm{d}s.$$

This integral naturally becomes a sum if p_x is a PMF. The *mean* of x is defined as $\mathbf{E}(x)$ and the *variance* of x is defined as

$$\operatorname{var}(x) \coloneqq \mathbf{E}\Big(\big(x - \mathbf{E}(x)\big)\big(x - \mathbf{E}(x)\big)^T\Big).$$

Given jointly distributed random variables $x\colon \Omega \to \mathbb{R}^m$ and $y\colon \Omega \to \mathbb{R}^n$ and a function $f\colon \mathbb{R}^m \to \mathbb{R}^p$, the *conditional expectation* of $f(x)$ given y can be stated in terms of the conditional density (if it exists) as follows:

$$\mathbf{E}\big(f(x) \mid y\big) = \int_{\mathbb{R}^m} f(s)\,p_{x|y}(s \mid y)\,\mathrm{d}s.$$

Note that $\mathbf{E}\big(f(x) \mid y\big)$ is a random variable taking values in $\mathbb{R}^p$. See [99] for a more rigorous definition of conditional expectation.

2.2.4 Independence

Let $(\Omega, \mathscr{F}, \mathbf{P})$ be a probability space. There are three notions of probabilistic independence:

- Two events, $E_1 \in \mathscr{F}$ and $E_2 \in \mathscr{F}$, are *independent* if $\mathbf{P}(E_1 \cap E_2) = \mathbf{P}(E_1)\mathbf{P}(E_2)$.
- Two σ-algebras $\mathscr{G}_1 \subset \mathscr{F}$ and $\mathscr{G}_2 \subset \mathscr{F}$ are *independent* if, for all $G_1 \in \mathscr{G}_1$ and $G_2 \in \mathscr{G}_2$, $\mathbf{P}(G_1 \cap G_2) = \mathbf{P}(G_1)\mathbf{P}(G_2)$ (i.e., the events G_1 and G_2 are independent).
- Two jointly distributed random variables $x\colon \Omega \to \mathbb{R}^m$ and $y\colon \Omega \to \mathbb{R}^n$ are *independent* if, for all $B_1 \in \mathscr{B}(\mathbb{R}^m)$ and $B_2 \in \mathscr{B}(\mathbb{R}^n)$, the events $x^{-1}(B_1)$ and $y^{-1}(B_2)$ are independent. This independence is denoted as $x \perp\!\!\!\perp y$. Note that $x \perp\!\!\!\perp y$ implies that, for all $a \in \mathbb{R}^m$ and $b \in \mathbb{R}^n$,
$$p_{x,y}(a,b) = p_x(a)\,p_y(b),$$
if these densities exist.

2.2.5 Stochastic Processes

Given an index set T, a *stochastic process* is a function $x\colon T \times \Omega \to \mathbb{R}^n$, such that x_t is a random variable, for all $t \in T$. Alternatively, we could view x as a random variable which takes values in the set of functions mapping T to $\mathbb{R}^n$. We often use the notation $\{x_t\}_{t\in T}$ or simply $\{x_t\}$ to distinguish the stochastic process x from a single random variable. If the index set is $T = [0,\infty)$, then $\{x_t\}$ is called a *continuous-time stochastic process*, and if $T = \mathbb{Z}_+ = \{0,1,\ldots\}$, then $\{x_t\}$ is called a *discrete-time stochastic process*. Given a discrete-time stochastic process $\{x_t\}$, define the notation $x_{i:j} \coloneqq \{x_i, x_{i+1}, \ldots, x_j\}$, for all $i, j \in \mathbb{Z}_+$.

A stochastic process $\{x_t\}$ is called a *Markov process* if

$$p_x(x_{t_n} \mid x_{t_1}, \ldots, x_{t_{n-1}}) = p_x(x_{t_n} \mid x_{t_{n-1}}),$$

for any set of indices $\{t_1, t_2, \ldots, t_n\} \subset T$, such that $t_1 < t_2 < \cdots < t_n$. A *white stochastic process* $\{x_t\}$ is defined as a Markov process, such that

$$p_x(x_{t_2} \mid x_{t_1}) = p_x(x_{t_2}),$$

for all $t_1, t_2 \in T$. A discrete-time Markov process $\{z_k\}$ taking values in some countable set $M \subset \mathbb{R}^m$ is called a *Markov chain*.

Given a stochastic process $\{x_t\colon \Omega \to \mathbb{R}^n\}_{t\in T}$, the *mean function* of x is defined as

$$m_x(t) \coloneqq \mathbf{E}(x_t),$$

for all $t \in T$, the *autocorrelation function* of x is defined as

$$R_x(s,t) \coloneqq \mathbf{E}(x_s x_t^T),$$

for all $s,t \in T$, and the *autocovariance function* of x is defined as

$$C_x(s,t) \coloneqq \mathbf{E}\Big(\big(x_s - m_x(s)\big)\big(x_t - m_x(t)\big)^T\Big),$$

for all $s,t \in T$. The random process $\{x_t\}$ is said to be *strictly stationary* if

$$p(x_{t_1}, x_{t_2}, \ldots, x_{t_m}) = p(x_{t_1+\tau}, x_{t_2+\tau}, \ldots, x_{t_m+\tau})$$

for all finite sets of indices $t_1, t_2, \ldots, t_m \in T$, where $m \in \mathbb{N}$, and all $\tau \geq 0$. The random process $\{x_t\}$ is said to be *wide-sense stationary* (wss) if for some constant $\bar{m} \in \mathbb{R}^n$,

$$m_x(t) = \bar{m},$$

for all $t \in T$, and for any $\tau \in T$,

$$R_x(s+\tau, s) = R_x(t+\tau, t),$$

for all $s,t \in T$. If $\{x_t\}$ is wss, then R_x only depends on the difference between its arguments and we may write $R_x(s+\tau, s) = R_x(\tau)$, for all $s, \tau \in T$. Given a wss process $\{x_t\}$, the *power spectral density* of x is defined as

$$S_x(\xi) \coloneqq \mathbf{F}(R_x)(\xi) = \int e^{-2\pi i \xi \tau} R_x(\tau)\, \mathrm{d}\tau,$$

where $\mathbf{F}$ is the Fourier transform operator.

2.2.6 Common Probability Distributions

1. A *Gaussian random variable* $x \colon \Omega \to \mathbb{R}^n$ with mean $\mu \in \mathbb{R}^n$ and variance $\Sigma \in \mathbb{R}^{n \times n}$, such that $\Sigma \succ 0$, is defined by the PDF

$$p_x(s) \coloneqq \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left(-\frac{1}{2}(s-\mu)^T \Sigma^{-1} (s-\mu)\right).$$

   This distribution is denoted $x \sim \mathcal{N}(\mu, \Sigma)$. If we define $z \coloneqq \Sigma^{-1/2}(x-\mu)$, then $z \sim \mathcal{N}(0, I)$, which is known as the *standard Gaussian distribution*. If z is scalar, then the CDF of z can be written as

$$P_z(c) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{c}{\sqrt{2}}\right)\right),$$

for all $c \in \mathbb{R}$, where

$$\operatorname{erf}(c) \coloneqq \frac{2}{\sqrt{\pi}} \int_0^c e^{-t^2} \, \mathrm{d}t,$$

is known as the *error function*. Similarly, in the scalar case, the CDF of x can be written as

$$P_x(c) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{c-\mu}{\sqrt{2\Sigma}}\right)\right).$$

Although there is no closed-form solution for computing the CDF of a Gaussian, there are many strategies for computing accurate numerical approximations [17, 38].

The following fact is perhaps the most useful property of the Gaussian distribution.

Fact 2.1. *Suppose that $x \sim \mathcal{N}(\mu, \Sigma)$ takes values in $\mathbb{R}^n$. Then, for all $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, the random variable $y = Ax + b$ is also Gaussian with mean $A\mu + b$ and variance $A\Sigma A^T$.*

2. A *Gaussian stochastic process* is a stochastic process $\{x_t\}_{t \in T}$, such that x_t is a Gaussian random variable, for all $t \in T$. If $\{x_t\}$ is also a white process, then

$$C_x(t, s) = Q(t)\delta(t - s),$$

where $Q_t \succeq 0$ for all $t \in T$. Hence, the power spectral density of a white Gaussian process is a constant function.

3. An *exponentially-distributed random variable* $\tau \colon \Omega \to \mathbb{R}_+$ with parameter $\lambda > 0$ has the PDF

$$p_\tau(t) \coloneqq \lambda e^{-\lambda t}$$

and the CDF

$$P_\tau(t) \coloneqq 1 - e^{-\lambda t},$$

for all $t \geq 0$. This distribution is denoted $\tau \sim \operatorname{Exp}(\lambda)$.

4. A *geometrically-distributed random variable* $\kappa \colon \Omega \to \mathbb{Z}_+$ with parameter $q > 0$ has the PMF

$$p_\kappa(k) = (1 - q)^{k-1} q,$$

and the CDF

$$P_\kappa(k) = 1 - (1 - q)^k,$$

for all $k \in \mathbb{Z}_+$. This distribution is denoted $\kappa \sim \operatorname{Geo}(q)$.

2.3 Reliability Theory

In this section, we present a select set of definitions and results from the vast field of reliability theory. The purpose is to establish two useful probabilistic models for the failure time of a system or component. For a thorough treatment of reliability theory, see Rausand and Høyland [77] or Singpurwalla [85].

Let $(\Omega, \mathscr{F}, \mathbf{P})$ be a probability space, and let $\tau\colon \Omega \to \mathbb{R}_+ := [0,\infty)$ be a random variable that represents the time at which some system or component fails. As in the previous section, let P_τ and p_τ denote the cumulative distribution function (CDF) and probability density function (PDF) of τ, respectively.

Definition 2.2. The *mean time to failure* (MTTF) of τ is defined as $\mathbf{E}(\tau)$.

Definition 2.3. The *failure rate* is defined as the expected number of failures in some interval of time, given that no failure has occurred yet. For $\Delta > 0$, the failure rate of τ at time $t \geq 0$ is

$$\rho_\Delta(t) := \frac{\mathbf{P}(t < \tau \leq t + \Delta \mid \tau > t)}{\Delta} = \frac{P_\tau(t+\Delta) - P_\tau(t)}{\Delta\big(1 - P_\tau(t)\big)}.$$

Definition 2.4. The *hazard rate* of τ at time $t \geq 0$ is defined as

$$h(t) := \lim_{\Delta \to 0} \rho_\Delta(t) = \frac{p_\tau(t)}{1 - P_\tau(t)}.$$

Suppose that, for a given sample time $T_s > 0$, the failure time is modeled as a discrete-valued random variable $\kappa\colon \Omega \to \mathbb{Z}_+ := \{0, 1, \ldots\}$. That is, for all $k \in \mathbb{Z}_+$, the event $\{\kappa = k\}$ indicates a failure at time kT_s. In this case, the interval Δ must be a multiple of the sample time T_s, so the hazard rate converges to

$$h(k) = \rho_{T_s}(k) = \frac{P_\kappa(k+1) - P_\kappa(k)}{T_s\big(1 - P_\kappa(k)\big)}.$$

However, there are cases where the discrete failure time κ does not have an underlying sample time. In such cases, the hazard rate is defined as

$$h(k) = \rho_1(k) = \frac{P_\kappa(k+1) - P_\kappa(k)}{1 - P_\kappa(k)}.$$

For many physical systems, the graph of the hazard rate takes the shape of a "bathtub curve", shown in Figure 2.1 [77, 85]. Initially, the system goes through a break-in phase where failures are more likely. If the system survives the break-in phase, the hazard rate remains roughly constant until the systems begins to wear out and failures become more likely again. In modeling physical systems, it is common to assume that the break-in phase has already

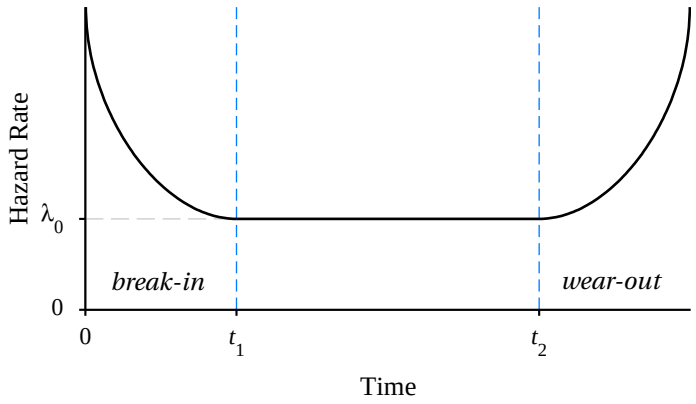


Figure 2.1. "Bathtub" shape of the hazard rate curve for a typical system. Failures are more likely as the component is broken in ($t < t_1$) and as the component wears out ($t > t_2$). In the intermediate period ($t_1 \leq t \leq t_2$), the hazard rate is roughly constant.

taken place, but the wear-out phase has not yet begun. Hence, the class of random variables with a constant hazard function play an important role in reliability theory.

Definition 2.5. A random variable with constant hazard rate is said to be *memoryless*.

Next, we consider two useful probability distributions, one defined on $\mathbb{R}_+$ and one defined on $\mathbb{Z}_+$, that yield memoryless failure times. Verifying these facts is simply a matter of applying the definition of the hazard rate to their respective CDFs and PDFs.

Fact 2.6. *If $\tau \sim \text{Exp}(\lambda)$, then τ is memoryless with $h(t) = \lambda$, for all t.*

Fact 2.7. *If $\kappa \sim \text{Geo}(q)$, then κ is memoryless with $h(k) = \frac{q}{T_s}$, for all $k \in \mathbb{Z}_+$, where $T_s > 0$ is either the underlying sample time of the model or the constant $T_s = 1$.*

Suppose that $\tau \sim \text{Exp}(\lambda)$ models the failure time of some component. For a given sample time $T_s > 0$, it is often useful to define a discrete-valued random variable $\kappa : \Omega \to \mathbb{Z}_+$, such that the CDF P_κ approximates the CDF P_τ. The following fact shows that the geometric distribution provides an ideal discretization of the exponential distribution.

Fact 2.8. *Fix $T_s > 0$, let $\tau \sim \text{Exp}(\lambda)$, and let $\kappa \sim \text{Geo}(q)$, such that $q = 1 - e^{-\lambda T_s}$. Then,*

$$P_\kappa(k) = P_\tau(kT_s),$$

for all k. Moreover, the hazard rate of κ at time step k is

$$h(k) = \lambda - \frac{\lambda^2 T_s}{2} + O(T_s^2),$$

so the hazard rate of κ converges to the hazard rate of τ as $T_s \to 0$.

Proof. For $k \geq 0$, the CDF of κ is

$$P_\kappa(k) = 1 - (e^{-\lambda\Delta})^k = 1 - e^{-\lambda k\Delta} = P_\tau(k\Delta).$$

Since the second-order Taylor approximation of the exponential function is

$$e^{-x} = 1 - x + \frac{x^2}{2} + O(x^3),$$

the hazard rate of κ is approximated by

$$h(k) = \frac{q}{T_s} = \frac{1 - e^{-\lambda T_s}}{T_s} = \lambda - \frac{\lambda^2 T_s}{2} + O(T_s^2)$$

Hence, $h(k) \to \lambda$ as $T_s \to 0$. □

2.4 Fault Diagnosis

This section provides a brief survey of the fault diagnosis literature. To begin, we establish a lexicon of common fault diagnosis terminology. Then, we briefly review some of the existing techniques used to design fault diagnosis schemes. Although this dissertation is focused on performance analysis, rather than design, this survey provides some context for our analysis. Similarly, we survey some of the ways in which redundancy can be used, in conjunction with fault diagnosis schemes, to produce more reliable systems. Finally, we discuss the existing approaches to analyzing the performance of fault diagnosis schemes.

2.4.1 Basic Terminology

Because fault diagnosis research spans many engineering disciplines, there is some disagreement about even the most basic terminology. In the late 1980s, the International Federation of Automatic Control (IFAC) formed the Technical Committee on Fault Detection, Supervision, and Safety of Technical Processes (SAFEPROCESS). One key contribution of the IFAC SAFEPROCESS committee was to establish a set of commonly accepted definitions. The following list, taken directly from [49], is comprised of these definitions:

fault — an unpermitted deviation of at least one characteristic property or parameter of the system from the acceptable/usual/standard condition.

failure — a permanent interruption of a system's ability to perform a required function under specified operating conditions.

malfunction — an intermittent irregularity in the fulfilment of a system's desired function.

disturbance — an unknown (and uncontrolled) input acting on a system.

residual — a fault indicator, based on a deviation between measurements and model-equation-based computations.

fault detection — determination of the faults present in a system and the time of detection.

fault isolation — determination of the kind, location and time of detection of a fault. Follows fault detection.

fault identification — determination of the size and time-variant behaviour of a fault. Follows fault isolation.

fault diagnosis — determination of the kind, size, location and time of detection of a fault. Follows fault detection. Includes fault identification.

reliability — ability of a system to perform a required function under stated conditions, within a given scope, during a given period of time.

safety — ability of a system not to cause danger to persons or equipment or the environment.

availability — probability that a system or equipment will operate satisfactorily and effectively at any point of time.

2.4.2 Brief Survey of Fault Diagnosis

In this section, we present a brief survey of the vast field of fault diagnosis. For a thorough treatment, see Chen and Patton [9], Ding [24], or Isermann [48]. Consider the general fault diagnosis problem in Figure 2.2. The system G is affected by known inputs u, stochastic noises v, unknown deterministic disturbances w, and an exogenous signal f representing a fault. The fault diagnosis scheme is comprised of two parts: a *residual generator* F and a *decision function* δ. The residual generator F uses the known input u and the measured output y to produce a *residual* r, which carries information about the occurrence of faults. The decision function δ evaluates the residual and determines what type of fault, if any, has occurred. The output of the residual generator, d, is called the *decision* issued by the FDI scheme. Typically, d takes values in some finite set of decisions $\mathscr{D}$. This separation of a fault diagnosis scheme into two stages was first proposed in [13].

There are a number of approaches to constructing meaningful residual signals. In a *structured residual set*, the residual r is a vector such that each component r_i is sensitive to a subset of faults. If each residual component r_i is sensitive to a single component f_i of the fault vector, then r is said to be a *dedicated residual set*. Another approach is to make each

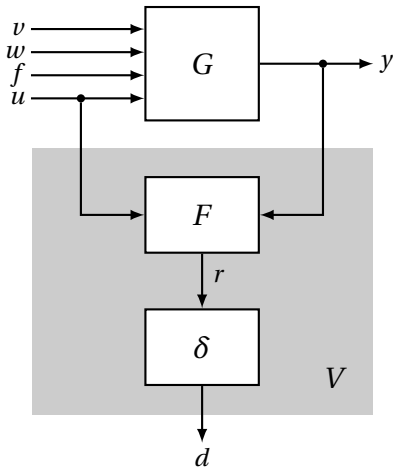


Figure 2.2. General fault diagnosis problem. The plant G is subject to a known deterministic input u, a random input v, a deterministic disturbance w, and a fault input f. The residual generator F uses the plant input u and output y to produce a residual r, and the decision function δ uses the residual r to produce a decision d about the current value of f. Together, F and δ form a fault detection scheme denoted by $V = (F, \delta)$. *Figure adapted from* [9, p. 21].

component r_i sensitive to all faults except f_i, in which case r is called a *generalized residual set*. For all structured residual sets, the occurrence of fault f_i is determined by comparing the components of the residual vector.

Taking a more geometric approach, the residual generator F may be constructed in such a way that when fault f_i occurs (and no other faults occur) the residual r lies in some subspace C_i. Using this approach, faults are detected by determining which subspace C_i is closest to the residual vector r, in some geometric sense. Such residual vectors are called *directional residual vectors* in the literature.

There are many techniques for constructing residual generators. Here, we present a brief survey of some of the most popular methods. Because this dissertation focuses on the performance analysis of FDI schemes, rather than design, this survey is neither exhaustive nor self-contained. The presentation, especially the section on parity equation-based methods, closely follows the survey given in [9, Chap. 2].

Observer-Based Methods

Let the dynamics of G be described by a finite-dimensional ordinary differential equation with state x. In observer-based methods, the residual generator F is an observer that produces an estimate z of some linear function of the output, Ly, where L is chosen by the designer of the fault diagnosis scheme. The residual is defined as

$$r := Q(z - Ly),$$

where the matrix Q is chosen to appropriately weight the estimation errors. The idea behind observer-based methods is to construct the observer F and the weighting matrix Q such that the residual is sensitive to faults. Early presentations of the observer-based method (e.g., [3]) assumed that there were no disturbances or noises affecting the system. For such systems, F consists of a Luenberger observer [64] with weighted estimation error. For systems affected by noises, a Kalman filter [50–53] may be used to obtain an estimate of Ly that minimizes the mean-squared estimation error [67]. For systems affected by a disturbance, an unknown input observer is used to decouple the residual from the effect of the disturbance [10, 58]. Typically, unknown input observers are not full-order and the remaining degrees of freedom may be used to address some other design objective. For example, in systems affected by disturbances and noise, the remaining degrees of freedom may be selected such that mean-squared estimation error is as small as possible [8, 9].

Parity Equation-Based Methods

The parity equation approach is similar to the notion of physical redundancy, in the sense that the residual is formed by comparing the system outputs y. For simplicity, assume that the output $y \in \mathbb{R}^m$ is given by

$$y = Cx + v + f,$$

where v is a noise process and f is a fault signal. Note that parity equation methods typically assume that there are no disturbances affecting the system. The residual is defined as

$$r := Qy,$$

where $Q \neq 0$ is chosen such that $QC = 0$. Hence, the residual can be written as

$$r = Q(v + f) = q_1(v_1 + f_1) + \cdots + q_m(v_m + f_m),$$

where q_i is the ith column of Q. Since each fault f_i enters the residual in the direction of the vector q_i, faults are isolated by choosing the largest component (in magnitude) of the vector $Q^T r$. See [78] for an early survey of parity equation-based methods.

Of course, the requirement that $QC = 0$ can only be met with a nonzero Q when C has a nontrivial null space. For systems where this requirement is not met, a form of temporal redundancy may be used [14,68]. This approach is usually restricted to discrete-time systems with no disturbances or noises. Suppose that the system is of the form

$$\begin{aligned} x_{k+1} &= Ax_k + B_k u_k + R_1 f_k \\ y_k &= Cx_k + D_k u_k + R_2 f_k. \end{aligned}$$

Fix $s \in \mathbb{N}$, and consider the following temporal relations:

$$\underbrace{\begin{bmatrix} y_{k-s} \\ y_{k-s+1} \\ \vdots \\ y_k \end{bmatrix}}_{Y_k} - H \underbrace{\begin{bmatrix} u_{k-s} \\ u_{k-s+1} \\ \vdots \\ u_k \end{bmatrix}}_{U_k} = W x_{k-s} + M \underbrace{\begin{bmatrix} f_{k-s} \\ f_{k-s+1} \\ \vdots \\ f_k \end{bmatrix}}_{\Phi_k},$$

where

$$H := \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{s-1}B & CA^{s-2}B & \cdots & D \end{bmatrix}, \quad M := \begin{bmatrix} R_2 & 0 & \cdots & 0 \\ CR_1 & R_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{s-1}R_1 & CA^{s-2}R_1 & \cdots & R_2 \end{bmatrix}, \quad W := \begin{bmatrix} C \\ CA \\ \vdots \\ CA^s \end{bmatrix}.$$

The residual is defined as

$$r_k := Q(Y_k - HU_k) = QWx_{k-s} + QM\Phi_k.$$

Hence, Q should be chosen such that $QW = 0$ and $QM \neq 0$. By the Cayley–Hamilton Theorem [59], these conditions can always be satisfied if s is large enough [14].

Parameter Estimation-Based Methods

In the parameter estimation approach to fault diagnosis it is assumed that faults cause changes in the physical parameters of the system, which in turn cause changes in the system model parameters [47]. Consider the block diagram shown in Figure 2.3. The system G_θ is parameterized by a vector of model parameters θ taking values in some parameter set Θ. Since faults enter the system G_θ via changes in the parameter θ, no exogenous fault signals are considered. The general idea is to detect faults by observing changes in θ. Since θ is not measured directly, its value must be estimated using the system inputs u and outputs y. If θ_0 is the nominal value of the model parameter and $\hat{\theta}$ is the estimate, then the residual may be defined as

$$r := \hat{\theta} - \theta_0.$$

Another approach to defining the residual is to compare the output of the nominal system (i.e., G_{θ_0}) with the measured output y, in which case the residual is defined as

$$r := y - G_{\theta_0} u.$$

Typically, fault isolation is more difficult using parameter estimation-based methods [9].

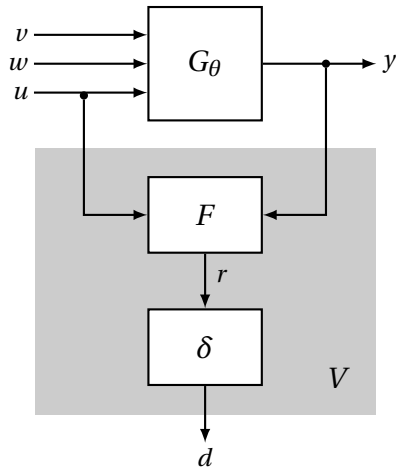


Figure 2.3. General parametric fault diagnosis problem. Here, faults affect the system G via the parameter θ, rather than an exogenous fault signal f, as in Figure 2.2.

2.5 Designing for Reliability

2.5.1 Physical Redundancy

In physically redundant configurations, multiple components performing the same function are used in parallel. A physically redundant system of four sensors is shown in Figure 2.4. Note that each identical sensor S is affected by different noises v_i, disturbances d_i, and faults f_i, making each of the outputs y_i different. The outputs are aggregated into a single measurement $\bar{y}$ using some sort of averaging or voting scheme. To detect a component failure, each output y_i is subtracted from the aggregate output $\bar{y}$ to form a residual r_i.

Advantages of physical redundancy

Generally speaking, physically redundant systems can survive multiple component failures and still perform their prescribed function. For example, a quadruplex system of four components, such as the sensor system in Figure 2.4, can survive two component failures. After one failure, the failed component is taken off-line and the remaining three components function in a triplex configuration. Note that the voting scheme must adapt to this new configuration. If a second failure occurs, the failed component is taken off-line, and the system functions in a duplex configuration. In the event of a third failure, the system is unable to determine which component is healthy and which is failed, rendering the whole system in a failed state.

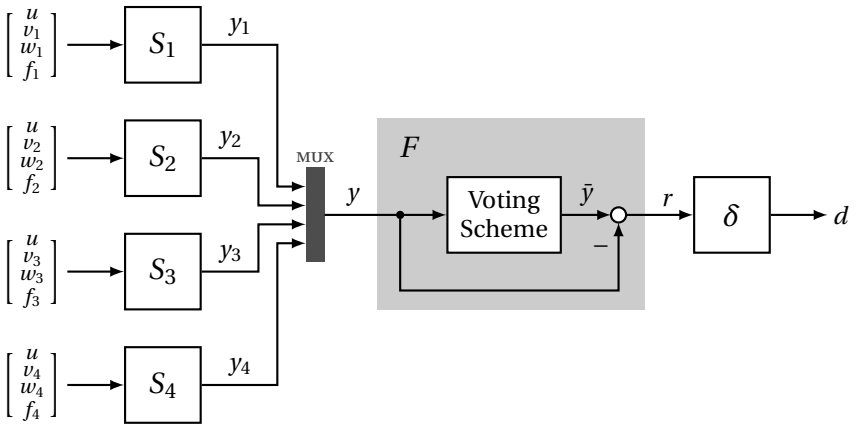


Figure 2.4. System of four physically redundant sensors. Although each sensor S_i is affected by the same input u, each sensor is also affected by a distinct noise v_i, disturbance w_i, and fault signal f_i. The Voting Scheme uses the vector of measurements y to produce a single aggregate output $\bar{y}$. The residual vector r is formed by directly comparing each component of the measured output vector y to the aggregate output $\bar{y}$.

Disadvantages of physical redundancy

The most apparent disadvantage to using physically redundant components is the additional size, weight, power, and cost needed to support multiple copies of the same component. For some systems, such as commercial airliners, the need for reliability justifies the additional cost and physical redundancy is used extensively [18, 69]. However, for other systems, such as Unmanned Aerial Vehicles (UAVs), the use of physically redundant components is less practical.

2.5.2 Analytical Redundancy

An alternative approach to physical redundancy is *analytical redundancy*. In analytically redundant configurations, analytical relationships are used to derive redundant estimates of measured quantities. Consider, for example, the sensor system shown in Figure 2.5. Each of the distinct sensors S_i senses a different physical quantity u_i and produces a different measurement y_i. Suppose that, under ideal conditions (i.e., no noises v_i, disturbances w_i, or faults f_i), the measurements satisfy known analytical relationships:

$$
\begin{aligned}
y_1 &= g_1(y_2, y_3), \\
y_2 &= g_2(y_1, y_4), \\
y_3 &= g_3(y_2, y_4), \\
y_4 &= g_4(y_1, y_3).
\end{aligned}
$$

These relationships can be used to form residual signals. For example,

$$\begin{aligned} r_1 &= y_3 - g_3(y_2, y_4), \\ r_2 &= y_4 - g_4(y_1, y_3), \\ r_3 &= y_2 - g_2(y_1, y_4), \\ r_4 &= y_1 - g_1(y_2, y_3). \end{aligned}$$

For $i = 1,2,3,4$, let $\varepsilon_i > 0$ and define

$$s_i := \begin{cases} 0, & \text{if } |r_i| < \varepsilon_i, \\ 1, & \text{otherwise.} \end{cases}$$

Then, faults can be detected based on the following symptom table [48, §17]:

	Symptoms			
Fault	s_1	s_2	s_3	s_4
1	0	1	1	1
2	1	0	1	1
3	1	1	0	1
4	1	1	1	0

Note that when Sensor i fails (i.e., Fault i occurs), all of the residual except r_i are affected. Hence, this is an example of a generalized residual set. For this example, when two sensors fail, all the symptoms are present and there is no way to determine which faults have occurred.

Advantages of analytical redundancy

The key advantage of using analytical redundancy is the reduced physical complexity of the system. For example, in Figure 2.5, four sensors are used to measure four different quantities y_1, y_2, y_3, and y_4. Thus, each sensor is performing a unique useful task and no extraneous hardware is being used. By moving the redundancy to the software side, the overall system consumes less space, weight, and power.

Disadvantages of analytical redundancy

In general, analytically redundant configurations are less reliable. Since each component performs a unique function, the loss of a single component may compromise an entire subsystem. For example, suppose that Sensor 1 in Figure 2.5 fails. Then, the system no longer has access to a measurement of the quantity y_1. At best, the signal $\hat{y}_1 = g(y_2, y_3)$

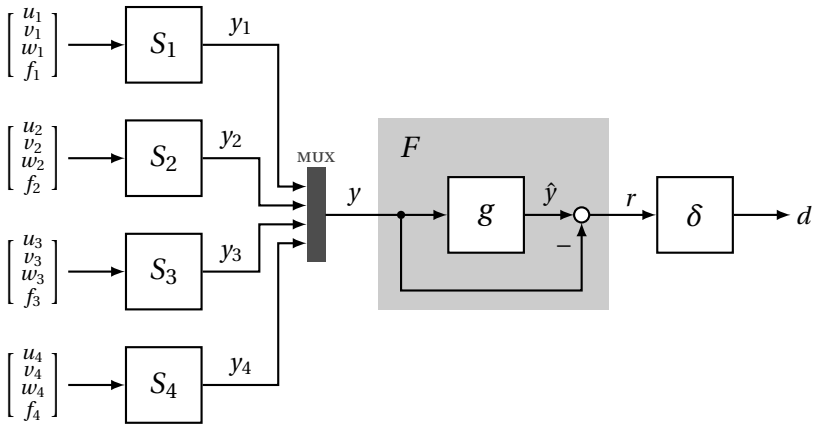


Figure 2.5. System of four analytically redundant sensors. Each sensor S_i is affected by a distinct input u_i, noise v_i, disturbance w_i, and fault signal f_i. The block labeled g represents a set of analytical relationships, which use the vector of measurements y to produce a residual vector r. Then, the decision function δ uses the residual vector r to produce a decision d.

can be used as a substitute. Also, the ability of the system to detect other sensor failures is reduced, because y_1 enters into all four of the residuals.

2.6 Existing Performance Analyses

2.6.1 Standard Approaches

In the fault detection literature, there are two primary ways to assess the performance of a fault detection scheme: simulation and design criteria. The simulation-based approach, used in [12, 15, 25, 32, 42, 62, 89, 107–109], involves simulating a number of realizations of the residual r given that a particular fault occurs at a particular time (see Figure 2.6 for a typical plot of a single simulation). From these simulation data, one can generally get a sense of how well the fault detection scheme detects the fault in question. However, the number of simulations—usually just one—is often too small to say anything statistically meaningful about the performance. Moreover, it is impractical to produce such a plot for every possible fault that may affect the system. By simulating the effect that a particular fault has on the residual, these simulation-based performance assessments assume that either the residual has reached steady-state when the fault occurs or, for some other reason, the time at which the fault occurs is irrelevant. Such assumptions are only meaningful when the residual is completely decoupled from the known inputs, unknown disturbances, and noise signals.

The second approach to assessing the performance of fault detection schemes is to quote the numerical value of design criteria. Examples of design criteria are given in Section 2.4.2. This approach, used in [10, 12, 14, 26, 39], is most useful for comparing fault detection schemes designed using similar criteria. Although it may be possible to produce a scheme using one set of design criteria and then assess their performance with respect to another set,

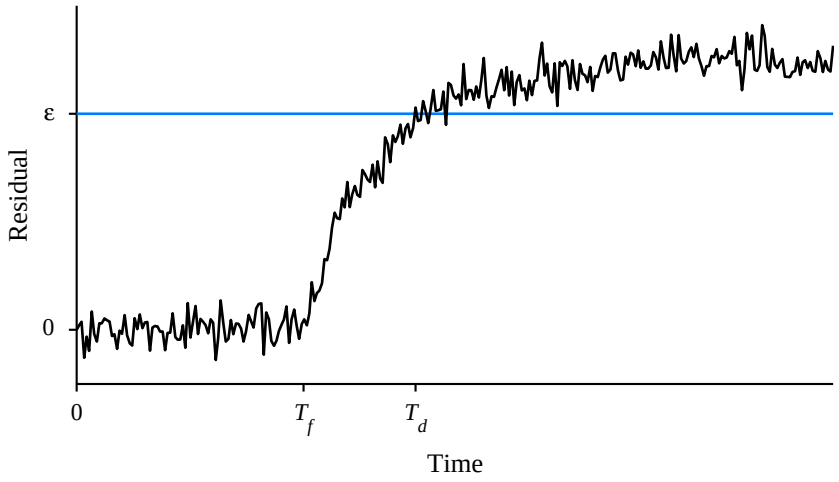


Figure 2.6. Typical plot of the response of the residual to the occurrence of a particular fault at time T_f. The residual crosses the threshold ε at time T_d, giving a detection delay of $T_d - T_f$.

the actual values of the criteria may be hard to interpret in terms of the desired system-level performance (e.g., overall reliability, false alarm rate).

2.6.2 Probabilistic Approaches

Recognizing the need for more rigorous and informative performance metrics, some authors in the fault diagnosis community (e.g., [8, 24, 100]) have proposed the *probability of false alarm* as a performance metric. For a fixed time k, a *false alarm* is defined as the event that the fault detection scheme indicates a fault at time k, given that no fault has occurred at or before time k. Conditional on the event that no fault has occurred, the only source of randomness in the residual $\{r_k\}$ is the noise signal $\{v_k\}$. In many cases, the distribution of the stochastic process $\{r_k\}$ is easily computed, and the probability of a false alarm can be evaluated (or at least bounded above).

However, the probability of false alarm alone cannot characterize the performance of a fault detection scheme. Consider, for example, the trivial decision function defined as $\delta_0 \colon (k, r_k) \mapsto 0$, for all k and r_k. Paired with *any* residual generator F, the fault detection scheme $V = (F, \delta_0)$ will have zero probability of false alarm, but V is incapable of detecting faults. Hence, it is also necessary quantify the *probability of detection*, which is the probability that the fault detection scheme correctly detects a fault when one is present. In general, the probability of detection must be computed for each fault or each class of faults. Performing these computations can be intractable unless special care is taken. For example, the class of fault signals considered in [100] is restricted to the set of randomly occurring biases, which are easily parameterized by the time of occurrence and the magnitude of the bias. More commonly, authors use simulation or design criteria, as in the previous section, to complement the probability of false alarm (e.g., [8]). **One of the main objectives of this**

thesis is to provide a probabilistic framework, in which the probability of detection can be efficiently computed for a large class of random fault signals.

2.6.3 Quickest Detection Problem

A related problem, which lends itself to more rigorous probabilistic analysis, is the *quickest detection problem*. Suppose that we measure a sequence of independent random variables $\{y_k\}_{k\geq 0}$. Initially, the random variables are independent and identically distributed (IID) according to some distribution P_0. Then, at some random time t_f, a change or fault occurs which alters the distribution of the random sequence. After t_f, the sequence $\{y_k\}_{k\geq t_f}$ is still IID, but the distribution is P_1. The goal is to detect that the distribution of $\{y_k\}$ has changed, as quickly as possible, after the fault time t_f. This problem is also known as *statistical change-point detection* or simply *change-point detection*.

A *quickest detection scheme* is a procedure that processes the measurements $\{y_k\}$ and produces an alarm time t_a, which is an estimate of the fault time t_f. Given a quickest detection scheme, the performance is typically assessed by two performance metrics [2, 76, 84]. First, the *mean time between false alarms* is defined as

$$\bar{T} := \mathbf{E}(t_a \mid t_a < t_f),$$

Second, the *mean delay* is defined as

$$\bar{\tau} := \mathbf{E}(t_a - t_f + 1 \mid t_a \geq t_f).$$

Although these metrics quantify the performance of the scheme in a meaningful way, their application to fault diagnosis problems is limited. When the sets of measurements $\{y_k\}_{k<t_f}$ and $\{y_k\}_{k\geq t_f}$ are each IID, these metrics are easy to compute. However, for more complex fault diagnosis problems, as in Figures 2.2 and 2.3, the distribution of y_k changes at each time step and the measurements are usually correlated in time. Hence, computing $\bar{T}$ and $\bar{\tau}$ is intractable for most fault diagnosis problems. Moreover, these metrics do not generalize to the case where many types of changes may occur (i.e., the distribution may change from P_0 to any member of the set $\{P_1, P_2, \ldots, P_m\}$ at time t_f). Despite the strong assumptions required, some authors (e.g., [44]) have successfully applied the quickest detection framework to fault detection problems.

Chapter 3

Probabilistic Performance Analysis

3.1 Introduction

The goal of this chapter is to provide a rigorous probabilistic analysis of fault diagnosis systems. In Section 3.3, fault detection is treated as a type of statistical hypothesis test and the accuracy of the test is analyzed probabilistically. Basic performance metrics, as well as common aggregate measures of performance, are presented. In Section 3.4, the limits of achievable fault detection performance are considered. In Section 3.5, some approaches for certifying and visualizing the time-varying performance of a fault detection system are considered. Finally, Section 3.6 briefly considers some extensions of this analysis to the more general fault isolation problem.

3.2 Problem Formulation

The main objective of this dissertation is to provide a rigorous probabilistic performance analysis of fault diagnosis schemes. Our analysis focuses on the parametric model shown in Figure 3.1. Both the system G_θ and residual generator F are assumed to be discrete-time dynamic systems. The time-varying model parameter $\{\theta_k\}$ is a discrete-time stochastic process taking values in some set Θ, where $\theta_k = 0$ is the nominal value (i.e., no faults or failures). The system G_θ is affected by a known deterministic input $\{u_k\}$, an unknown deterministic disturbance $\{w_k\}$, and a stochastic noise signal $\{v_k\}$. We assume that the distributions of $\{\theta_k\}$ and $\{v_k\}$ are known and that $\{w_k\}$ lies in some convex bounded set.

In the parametric framework, the designer of the fault diagnosis scheme partitions the parameter space into two or more disjoint subsets

$$\Theta = \Theta_0 \sqcup \Theta_1 \sqcup \cdots \sqcup \Theta_q,$$

where $\sqcup$ denotes the disjoint union and $\Theta_0 \coloneqq \{0\}$ is the nominal parameter value. The

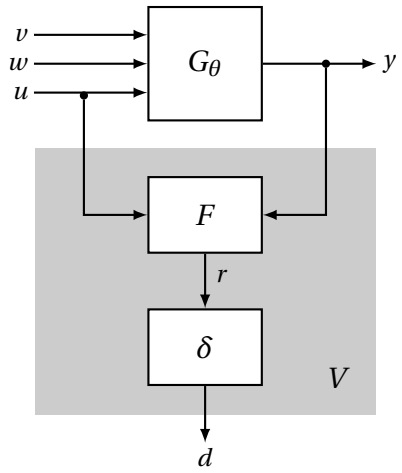


Figure 3.1. General parametric fault diagnosis problem. Faults affect the physical parameters of the system, which in turn affect the system model parameter θ. The plant G is subject to a known deterministic input u, a random input v, and a deterministic disturbance w. The residual generator uses the plant input u and output y to produce a residual r, and the decision function δ uses the residual r to produce a decision d about the current value of θ. Together, F and δ form a fault diagnosis scheme, denoted $V = (F, \delta)$.

corresponding set of possible decisions is defined as

$$\mathscr{D} := \{0, 1, \ldots, q\}.$$

The purpose of the fault diagnosis scheme $V = (F, \delta)$ is to produce a decision $d_k \in \mathscr{D}$, at each time k, indicating which subset $\Theta_{d_k} \subset \Theta$ most likely contains the parameter θ_k. Of course, the scheme V does not have direct access to the parameter. Instead, V must make a decision based on the known input $\{u_k\}$ and the measured output $\{y_k\}$, which is corrupted by the noise signal $\{v_k\}$ and the disturbance $\{w_k\}$. Therefore, the performance of the scheme V is quantified by the probability that the correct decision is made.

The number of partitions q determines what type fault diagnosis problem the scheme V is designed to address. If $q = 1$, the set Θ_1 contains all faulty parameter values, and V is interpreted as a fault detection scheme. If $q > 1$, each subset $\Theta_i \subset \Theta$ represents a different class of faulty behavior, and V is interpreted as a fault isolation scheme. If the parameter space Θ is finite and each partition Θ_i a singleton set, then V achieves fault identification, as well. In Section 3.3, we define probabilistic performance metrics for the fault detection problem ($q = 1$). Then, in Section 3.6, these results are extended to the more general fault isolation problem ($q > 1$).

In this chapter and in Chapter 4, we assume that the deterministic input $\{u_k\}$ is known and fixed, that there is no deterministic disturbance $\{w_k\}$, and that G_θ is a known function of the parameter $\{\theta_k\}$. Chapter 5 extends these results by considering how uncertainty impacts the performance metrics. In particular, Chapter 5 presents some techniques for computing

the worst-case performance under a given uncertainty model.

3.3 Quantifying Accuracy

Our performance analysis of fault detection is rooted in the theory of statistical hypothesis testing. This approach not only allows us to utilize the tools and terminology of hypothesis testing, it also allows us to draw connections between fault detection and other fields, such as signal detection [54, 61, 75, 93], medical diagnostic testing [31, 73, 111], and pattern recognition [34, 57]. For a standard mathematical treatment of statistical hypothesis testing, see Lehmann and Romano [60].

3.3.1 Fault Detection and Hypothesis Testing

For the sake of simplicity, this section focuses on the problem of fault detection, while the more general fault isolation problem is treated in Section 3.6. Hence, the parameter space is partitioned into two sets: the set containing the nominal parameter, $\Theta_0 = \{0\}$, and the set containing all faulty parameter values, $\Theta_1 = \Theta_0^c$. At each time k, define the hypotheses

$$\begin{aligned} \mathscr{H}_{0,k}: &\quad \theta_k \in \Theta_0, \\ \mathscr{H}_{1,k}: &\quad \theta_k \in \Theta_1, \end{aligned}$$

and let $H_{i,k}$ be the event that hypothesis $\mathscr{H}_{i,k}$ is true, for each i. Since exactly one hypothesis is true at each time, the sets $H_{0,k}$ and $H_{1,k}$ form a partition of the sample space Ω. The fault detection scheme V is interpreted as a *test* that decides between the hypotheses $\mathscr{H}_{0,k}$ and $\mathscr{H}_{1,k}$. Although the input data $u_{0:k} = \{u_0, \dots, u_k\}$ are known and deterministic, the distribution of the output data $y_{0:k} = \{y_0, \dots, y_k\}$ clearly depends on which hypothesis is true. Together, $u_{0:k}$ and $y_{0:k}$ are interpreted as a *test statistic*, which is used by the test V to produce a decision d_k in $\mathscr{D} = \{0, 1\}$, at time k. Let $D_{0,k}$ be the event that $d_k = 0$ and let $D_{1,k}$ be the event that $d_k = 1$. Of course, exactly one of these events is true at each time, so the sets $D_{0,k}$ and $D_{1,k}$ form another partition of the sample space Ω.

3.3.2 Probabilistic Analysis

Let the prior probabilities of the hypotheses be denoted

$$\begin{aligned} Q_{0,k} &\coloneqq \mathbf{P}(H_{0,k}), \\ Q_{1,k} &\coloneqq \mathbf{P}(H_{1,k}). \end{aligned}$$

Since exactly one hypothesis is true and exactly one decision is made at each time k, the performance of the test V is characterized by the probability that the events $D_{i,k}$ and $H_{j,k}$

are simultaneously true, for each i and j. The four possible cases are typically given the following names [61, 73]:

$$
\begin{aligned}
D_{0,k} \cap H_{0,k} &\text{ is a } \textit{true negative},\\
D_{1,k} \cap H_{0,k} &\text{ is a } \textit{false positive},\\
D_{0,k} \cap H_{1,k} &\text{ is a } \textit{false negative},\\
D_{1,k} \cap H_{1,k} &\text{ is a } \textit{true positive}.
\end{aligned}
$$

The corresponding probabilities of these events are denoted

$$P_{\text{TN},k} \coloneqq \mathbf{P}(D_{0,k} \cap H_{0,k}), \tag{3.1}$$
$$P_{\text{FP},k} \coloneqq \mathbf{P}(D_{1,k} \cap H_{0,k}), \tag{3.2}$$
$$P_{\text{FN},k} \coloneqq \mathbf{P}(D_{0,k} \cap H_{1,k}), \tag{3.3}$$
$$P_{\text{TP},k} \coloneqq \mathbf{P}(D_{1,k} \cap H_{1,k}). \tag{3.4}$$

In the literature (e.g., [31, 34, 73]), these event are often organized into an array

$$\begin{bmatrix} P_{\text{TN},k} & P_{\text{FN},k} \\ P_{\text{FP},k} & P_{\text{TP},k} \end{bmatrix}, \tag{3.5}$$

called a *confusion matrix* or *contingency table*. Since, for each k, the collection of events $\{D_{i,k} \cap H_{j,k} : i, j \in \mathscr{D}\}$ forms a partition of the sample space, the probabilities (3.1)–(3.4) satisfy the following useful identities:

$$P_{\text{TN},k} + P_{\text{FN},k} = \mathbf{P}(D_{0,k}), \tag{3.6}$$
$$P_{\text{FP},k} + P_{\text{TP},k} = \mathbf{P}(D_{1,k}), \tag{3.7}$$
$$P_{\text{TN},k} + P_{\text{FP},k} = \mathbf{P}(H_{0,k}) = Q_{0,k}, \tag{3.8}$$
$$P_{\text{FN},k} + P_{\text{TP},k} = \mathbf{P}(H_{1,k}) = Q_{1,k}, \tag{3.9}$$
$$P_{\text{TN},k} + P_{\text{FP},k} + P_{\text{FN},k} + P_{\text{TP},k} = 1. \tag{3.10}$$

The identity in equation (3.10) implies that there are only three independent probabilities. In the sequel, we refer to the probabilities $P_{\text{TN},k}$, $P_{\text{FP},k}$, $P_{\text{FN},k}$, and $P_{\text{TP},k}$ as the *performance metrics* for the test V at time k.

Although the probabilities (3.1)–(3.4) quantify every possible state of affairs, with respect to the hypotheses $\mathscr{H}_{0,k}$ and $\mathscr{H}_{1,k}$, the numerical values of these probabilities may be difficult to interpret. For example, suppose that $Q_{1,k} \approx 0$. By equation (3.9), $Q_{1,k} \approx 0$ implies that $P_{\text{FN},k} \approx 0$ and $P_{\text{TP},k} \approx 0$. From the small numerical values of $P_{\text{FN},k}$ and $P_{\text{TP},k}$, it may be difficult to get a sense of how the fault diagnosis scheme will behave in the event that a fault actually occurs. An alternative approach is to consider the relative magnitudes of the probabilities.

For example,

$$\frac{P_{\mathrm{TP},k}}{P_{\mathrm{FN},k}+P_{\mathrm{TP},k}} = \frac{\mathbf{P}(D_{1,k}\cap H_{1,k})}{\mathbf{P}(H_{1,k})} = \mathbf{P}(D_{1,k}\mid H_{1,k}).$$

Hence, we consider the following conditional probabilities:

$$P_{\mathrm{D},k} \coloneqq \mathbf{P}(D_{1,k}\mid H_{1,k}), \tag{3.11}$$

$$P_{\mathrm{F},k} \coloneqq \mathbf{P}(D_{1,k}\mid H_{0,k}). \tag{3.12}$$

Typically, $P_{\mathrm{D},k}$ is called the probability of *detection* and $P_{\mathrm{F},k}$ is called the probability of a *false alarm* [54, 61]. Note that the other conditional probabilities $\mathbf{P}(D_{0,k}\mid H_{1,k})$ and $\mathbf{P}(D_{0,k}\mid H_{0,k})$ are given by $1-P_{\mathrm{D},k}$ and $1-P_{\mathrm{F},k}$, respectively.

Proposition 3.1. *The probabilities* $\{P_{\mathrm{F},k}\}$ *and* $\{P_{\mathrm{D},k}\}$*, together with the prior probabilities* $\{Q_{0,k}\}$*, provide a set of performance metrics that are equivalent to the joint probabilities* (3.1)–(3.4).

Proof. At each time k, the original performance metrics (3.1)–(3.4) are directly computed from $P_{\mathrm{F},k}$, $P_{\mathrm{D},k}$, and $Q_{0,k}$ as follows:

$$\begin{aligned}
P_{\mathrm{TN},k} &= \mathbf{P}(D_{0,k}\mid H_{0,k})\,\mathbf{P}(H_{0,k}) = (1-P_{\mathrm{F},k})\,Q_{0,k},\\
P_{\mathrm{FP},k} &= \mathbf{P}(D_{1,k}\mid H_{0,k})\,\mathbf{P}(H_{0,k}) = P_{\mathrm{F},k}\,Q_{0,k},\\
P_{\mathrm{FN},k} &= \mathbf{P}(D_{0,k}\mid H_{1,k})\,\mathbf{P}(H_{1,k}) = (1-P_{\mathrm{D},k})\,(1-Q_{0,k}),\\
P_{\mathrm{TP},k} &= \mathbf{P}(D_{1,k}\mid H_{1,k})\,\mathbf{P}(H_{1,k}) = P_{\mathrm{D},k}\,(1-Q_{0,k}).
\end{aligned}$$

Also, these equations can be inverted to compute $P_{\mathrm{F},k}$, $P_{\mathrm{D},k}$, and $Q_{0,k}$ as follows:

$$\begin{aligned}
P_{\mathrm{F},k} &= \frac{\mathbf{P}(D_{1,k}\cap H_{0,k})}{H_{0,k}} = \frac{P_{\mathrm{FP},k}}{P_{\mathrm{TN},k}+P_{\mathrm{FP},k}}\\
P_{\mathrm{D},k} &= \frac{\mathbf{P}(D_{1,k}\cap H_{1,k})}{H_{1,k}} = \frac{P_{\mathrm{TP},k}}{P_{\mathrm{FN},k}+P_{\mathrm{TP},k}}\\
Q_{0,k} &= \mathbf{P}(D_{0,k}\cap H_{0,k}) + \mathbf{P}(D_{1,k}\cap H_{0,k}) = P_{\mathrm{TN},k}+P_{\mathrm{FP},k}.
\end{aligned}$$

□

Remark 3.2. Since the sequence $\{Q_{0,k}\}$ quantifies the reliability of the system G_θ, using the conditional probabilities $\{P_{\mathrm{F},k}\}$ and $\{P_{\mathrm{D},k}\}$ as performance metrics decouples the performance of the test V from the underlying system. In the sequel, we will often assume that the system G_θ, as well as the probabilities $\{Q_{0,k}\}$, are fixed, in which case the pair $(P_{\mathrm{F},k}, P_{\mathrm{D},k})$ will completely capture the performance of the test. ◇

3.3.3 Aggregate Measures of Performance

Although the performance metrics $\{P_{\text{TN},k}\}$, $\{P_{\text{FP},k}\}$, $\{P_{\text{FN},k}\}$, and $\{P_{\text{TP},k}\}$ fully characterize the time-varying behavior of the fault detection scheme $V = (F, \delta)$, it is often useful to aggregate these probabilities into a single meaningful quality. In this section, we consider two common aggregate performance measures. These approaches are included to further elucidate the connection between statistical hypothesis testing and performance analysis for fault detection schemes.

Probability of Correctness

The *probability of correctness* of a test V, denoted c_k, is defined as the probability that the decision d_k corresponds to the correct hypothesis. More precisely, for each time k,

$$c_k \coloneqq P_{\text{TN},k} + P_{\text{TP},k} = (1 - P_{\text{F},k})\, Q_{0,k} + P_{\text{D},k}\, Q_{1,k}.$$

Equivalently, one may consider the probability $e_k \coloneqq 1 - c_k$, which is known as the *probability of error* [61].

Bayesian Risk

To generalize the concept of accuracy, we utilize the concepts of loss and risk used in hypothesis testing [60] and general statistical decision theory [4, 22]. Fix a time k. In general, a *loss function* $L_k \colon \Theta \times \mathscr{D} \to \mathbb{R}$ is a nonnegative bounded function that quantifies the loss $L_k(\vartheta_k, d_k)$ incurred by deciding d_k when ϑ_k is the true state of affairs. Since the parameter space is partitioned as $\Theta = \Theta_0 \cup \Theta_1$ and the set of decisions is $\mathscr{D} = \{0, 1\}$, a loss function for the fault detection problem can be expressed as a matrix $L_k \in \mathbb{R}^{2\times 2}$ with nonnegative entries. The value $L_k(i, j)$ can be interpreted as the loss incurred by deciding $d_k = j$ "averaged" over all $\vartheta_k \in \Theta_i$.

The loss matrices $\{L_k\}_{k\geq 0}$ provide a subjective way to quantify the importance of making the correct decision in each possible case. The *Bayesian risk* $R_k(Q, V)$ is defined to be the expected loss incurred by the test V at time k, given that the parameter $\{\theta_k\}$ is distributed according to $Q_k = \{Q_{0,k}, Q_{1,k}\}$. More precisely, for each time k,

$$R_k(Q, V) \coloneqq \mathbf{E}\Big(L(\theta_k, d_k)\Big) = \sum_{i=0}^{1} \sum_{j=0}^{1} L_k(i, j)\mathbf{P}(D_{j,k} \cap H_{i,k}).$$

In terms of the performance metrics, the risk is

$$\begin{aligned} R_k(Q, V) &= L(0,0)\, P_{\text{TN}} + L(1,0)\, P_{\text{FN}} + L(0,1)\, P_{\text{FP}} + L(1,1)\, P_{\text{TP}} \\ &= L(0,0)\, Q_0 + L(1,0)\, Q_1 + \Big(L(0,1) - L(0,0)\Big) P_{\text{F}}\, Q_0 + \Big(L(1,1) - L(1,0)\Big) P_{\text{D}}\, Q_1, \end{aligned} \tag{3.13}$$

where the subscript k has been omitted for the sake of clarity.

Example 3.3 (0-1 Loss). Suppose that the loss matrix

$$L = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

is used for all time. This is typically referred to as "0-1 loss" in the literature [4, 61]. By equation (3.13), the corresponding Bayesian risk of a test V at time k is

$$\begin{aligned} R_k(Q_k, V) &= P_{\text{FP},k} + P_{\text{FN},k} \\ &= P_{\text{F},k}\, Q_{0,k} + (1 - P_{\text{D},k})\, Q_{1,k} \\ &= 1 - c_k. \end{aligned}$$

Thus, placing an upper bound on the 0-1 risk $R_k(Q_k, V)$ is equivalent to placing a lower bound on the probability of correctness c_k. ◇

3.4 Characterizing the Range of Achievable Performance

In Section 3.3, the performance of a test was given in terms of the probabilities $\{P_{\text{F},k}\}$ and $\{P_{\text{D},k}\}$. In this section, we consider the complementary problem of determining what performance values $(P_{\text{F},k}, P_{\text{D},k}) \in [0,1]^2$ are achievable by some test. Again, we draw on the tools of statistical hypothesis testing to address this issue. Namely, we use the Neyman–Pearson Lemma [71] and the receiver operating characteristic (ROC) [57] to characterize the limits of achievable performance. To facilitate our discussion, we first introduce the concept of a randomized test.

3.4.1 Randomized Tests

Up to this point, we have focused our attention on tests $V = (F, \delta)$, where both the residual generator F and the decision function δ are deterministic. However, it is possible to design and implement tests that are nondeterministic. In this section, we introduce nondeterministic or randomized tests and use them to characterize the set of achievable performance points.

Definition 3.4. A hypothesis test V is said to be a *randomized test* if, for a given realization of the test statistic $(\hat{u}_{0:k}, \hat{y}_{0:k})$, the decision $d_k = V(\hat{u}_{0:k}, \hat{y}_{0:k})$ is a random variable.

Define $\mathcal{V}$ to be set of all deterministic and randomized hypothesis tests, and define $\mathcal{W}_k$ to be the set of all performance points $(\alpha, \beta) \in [0,1]^2$ that are achieved by some test $V \in \mathcal{V}$, at time k. The following example shows how to derive randomized tests from the class of

deterministic tests.

Example 3.5. One common way to produce a randomized test is to randomly select a test from some finite collection of deterministic tests $\{V_1, V_2, \ldots, V_m\} \subset \mathcal{V}$ and use the decision produced by that test. More precisely, let p be a point in the simplex

$$S_m \coloneqq \left\{ p \in \mathbb{R}^m : p_i \geq 0, \sum_{i=1}^{m} p_i = 1 \right\},$$

and define λ to be a random variable that takes values in the set $\{1, 2, \ldots, m\}$, such that

$$\mathbf{P}(\lambda = i) = p_i.$$

Let the randomized test V_p be defined by

$$V_p(u_{0:k}, y_{0:k}) \coloneqq V_\lambda(u_{0:k}, y_{0:k}), \tag{3.14}$$

for all k and all $(u_{0:k}, y_{0:k})$. Then, probability of a false alarm for V_p at time k is

$$\begin{aligned} P_{\mathrm{F},k}(V_p) &= \mathbf{P}(D_{1,k} \mid H_{0,k}) \\ &= \sum_{i=1}^{m} \mathbf{P}(D_{1,k} \mid H_{0,k}, \lambda = i)\, \mathbf{P}(\lambda = i) \\ &= \sum_{i=1}^{m} P_{\mathrm{F},k}(V_i)\, p_i. \end{aligned}$$

By a similar calculation, the probability of detection for V_p at time k is

$$P_{\mathrm{D},k}(V_p) = \sum_{i=1}^{m} P_{\mathrm{D},k}(V_i)\, p_i.$$

The case $m = 4$ is shown in Figure 3.2, where the shaded region represents the performance points achieved by the family of randomized tests $\{V_p\}_{p \in S_4}$ obtained using this method. ◇

Fact 3.6. *The set of achievable performance points $\mathcal{W}_k$ is convex.*

Proof. Let (α_1, β_1) and (α_2, β_2) be any two points in $\mathcal{W}_k$, and let V_1 and V_2, respectively, be tests in $\mathcal{V}$ that achieve these performance points at time k. Let $\gamma \in [0, 1]$. To show that $\mathcal{W}_k$ is convex, we must exhibit a test with performance

$$(\alpha, \beta) \coloneqq \gamma(\alpha_1, \beta_1) + (1 - \gamma)(\alpha_2, \beta_2),$$

at time k. Since the point $p \coloneqq (\gamma, 1 - \gamma)$ is in the simplex S_2, we can use the procedure outlined in Example 3.5 to construct a randomized test V_p that utilizes V_1 and V_2. The

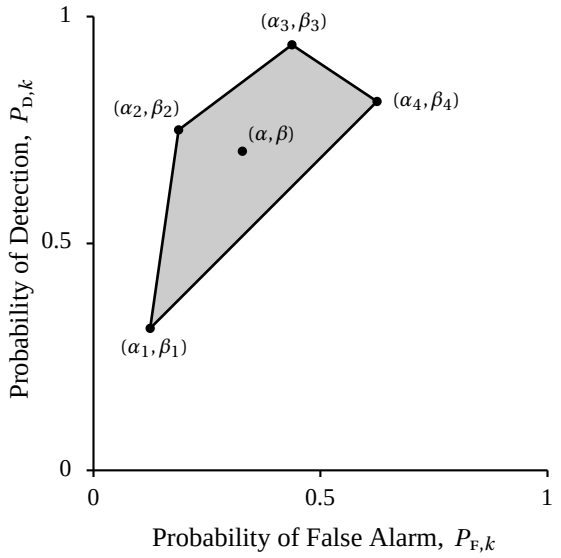


Figure 3.2. Illustration of Example 3.5 showing the range of performance points (shaded region) achievable by randomly selecting the decision made by one of four deterministic tests.

probability of a false alarm for this test is

$$P_{F,k}(V_p) = P_{F,k}(V_1)\gamma + P_{F,k}(V_2)(1-\gamma) = \alpha_1\gamma + \alpha_2(1-\gamma) = \alpha.$$

Similarly, the probability of detection is

$$P_{D,k}(V_p) = P_{D,k}(V_1)\gamma + P_{D,k}(V_2)(1-\gamma) = \beta_1\gamma + \beta_2(1-\gamma) = \beta.$$

Hence, V_p has the desired performance at time k, and $\mathcal{W}_k$ is convex. □

Fact 3.7. *The set $\mathcal{W}_k$ contains the points* $(0,0)$ *and* $(1,1)$.

Proof. Let $V_{\text{no}} \in \mathcal{V}$ be the test makes the decision $d_k = 0$, for all k. Similarly, let $V_{\text{yes}} \in \mathcal{V}$ be the test that makes the decision $d_k = 1$, for all k. The performance of the test V_{no} is clearly $(0,0)$, while the performance of V_{yes} is $(1,1)$. □

Since $\mathcal{W}_k$ is convex and always contains the points $(0,0)$ and $(1,1)$, $\mathcal{W}_k$ also contains the point (γ,γ), for any $\gamma \in (0,1)$. One test that achieves performance (γ,γ), is the randomized test that uses V_{no} with probability $1-\gamma$ and V_{yes} with probability γ. Since such tests make random decisions, independent of the value of the test statistic $(u_{0:k}, y_{0:k})$, they are often called *uninformative tests* [73]. Hence, we are mostly concerned with tests whose performance point is above the diagonal (i.e., $P_{D,k} > P_{F,k}$). However, the following fact shows that a test whose performance point falls below the diagonal can also be useful.

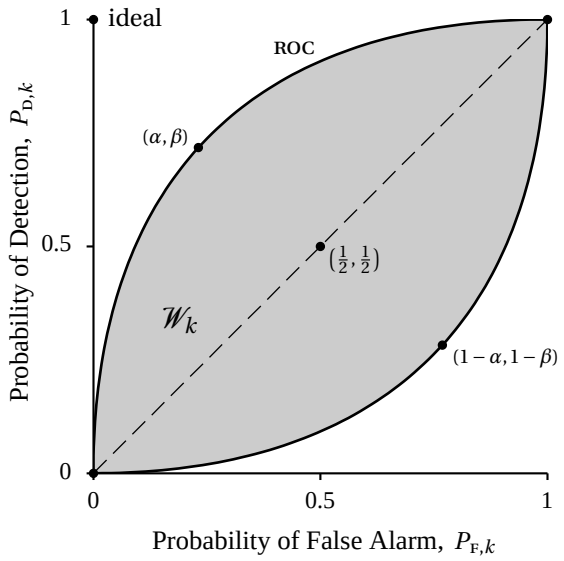


Figure 3.3. Visual summary of Facts 3.6–3.8. At each time k, the set $\mathcal{W}_k$ is convex, it contains the extreme points $(0,0)$ and $(1,1)$, and it is symmetric about the point $\left(\frac{1}{2},\frac{1}{2}\right)$.

Fact 3.8. *The set $\mathcal{W}_k$ is symmetric about the point $\left(\frac{1}{2},\frac{1}{2}\right)$, in the sense that if $(\alpha,\beta)\in\mathcal{W}_k$, then $(1-\alpha,1-\beta)\in\mathcal{W}_k$, as well.*

Proof. Let $(\alpha,\beta)\in\mathcal{W}_k$ and take $V\in\mathcal{V}$ to be a test whose performance, at time k, is given by (α,β). Define $\bar{V}$ to be the test that always decides the opposite of what V decides. Then, the probability of a false alarm for $\bar{V}$ is $1-\alpha$, and the probability of detection for V is $1-\beta$. □

To summarize, at each time k, the set of achievable performance points $\mathcal{W}_k$ is a convex set that is symmetric about the point $\left(\frac{1}{2},\frac{1}{2}\right)$ and contains the extreme points $(0,0)$ and $(1,1)$ (see Figure 3.3). Although Facts 3.6–3.8 are well known and can be found in the literature (e.g., [61]), the brief proofs provided here provide some insight into the structure of the sets $\{\mathcal{W}_k\}_{k\geq 0}$.

3.4.2 Receiver Operating Characteristic

The ideal performance point $(P_{\mathrm{F},k},P_{\mathrm{D},k})=(0,1)$ is achieved by a test that always chooses the correct hypothesis. However, such perfect tests rarely exist, because the test statistic $(u_{0:k},y_{0:k})$ contains only partial information about the parameter θ_k. Indeed, the test statistic is related to the parameter through the dynamics of the system G_θ, which is unlikely to yield a one-to-one relation. Moreover, the exogenous noise process $\{v_k\}$ corrupts the limited information available about θ_k. Therefore, the set $\mathcal{W}_k$ of achievable performance points is separated from the ideal $(0,1)$ by a curve passing through $(0,0)$ and $(1,1)$.

Definition 3.9. The upper boundary between the set $\mathcal{W}_k$ and the ideal point $(0,1)$ is called the *receiver operating characteristic* (ROC) for the set of all tests $\mathcal{V}$.

Since the set $\mathcal{W}_k$ changes with time, the ROC is time-varying, as well. Also, since $\mathcal{W}_k$ is convex (Fact 3.6), the ROC is concave. By Fact 3.8, there is a equivalent convex curve that separates $\mathcal{W}_k$ from the point $(1,0)$. However, the term ROC only refers to the upper boundary.

Characterizing the ROC

Although it may not be possible to compute the ROC for the set of all tests $\mathcal{V}$, the set of tests whose performance points lie on the ROC can be characterized theoretically. For any $\alpha \in (0,1]$, let $\mathcal{V}_\alpha$ be the set of tests for which $P_{\mathrm{F},k} \leq \alpha$, at time k. The set of *Neyman–Pearson* tests are defined as

$$V_{\mathrm{NP}} = \arg\max_{V \in \mathcal{V}_\alpha} \; P_{\mathrm{D},k}(V). \tag{3.15}$$

In general, the set $\mathcal{V}_\alpha$ is too abstract to properly formulate and solve this constrained optimization problem. However, the following lemma shows that V_{NP} is nonempty and explicitly characterizes one element in V_{NP}.

Lemma 3.10 (Neyman–Pearson [71]). *The likelihood ratio test with $P_{\mathrm{F},k} = \alpha$ is in V_{NP}.*

Therefore, the ROC is given by the set of likelihood ratio tests (see [61] for details).

In the optimization problem (3.15), the probability of a false alarm is constrained to be less than some $\alpha \in (0,1]$. However, we can also interpret the ROC in terms of the vector optimization problem

$$\max_{V \in \mathcal{V}} \; (-P_{\mathrm{F},k}, P_{\mathrm{D},k}). \tag{3.16}$$

Since the objective takes values in $[0,1]^2$, it not immediately clear what it means for one point to be better than another. Clearly, the ideal point $(0,1)$ is the best and points on the diagonal are of little use. The notion of Pareto optimality provides one way to compare values of the objective $(-P_{\mathrm{F},k}, P_{\mathrm{D},k})$. We say that a point $(P_{\mathrm{F},k}, P_{\mathrm{D},k}) = (\alpha, \beta)$ is *Pareto optimal* if no other test can simultaneously improve both $P_{\mathrm{F},k}$ and $P_{\mathrm{D},k}$. That is, for any other test with performance $(\alpha', \beta') \neq (\alpha, \beta)$, either $\alpha' > \alpha$ or $\beta' < \beta$. Hence, the ROC can be defined as the set of Pareto optimal points for the vector optimization problem (3.16). One well-known method for generating the set of Pareto optimal points (i.e., the ROC) is to solve the "scalarized" optimization problem

$$\max_{V \in \mathcal{V}} \; -\gamma P_{\mathrm{F},k} + (1-\gamma) P_{\mathrm{D},k} \tag{3.17}$$

for all $\gamma \in [0,1]$ [5, 106]. Since the ROC is concave, a lower bound may be computed by solving (3.17) at a finite collection of points $0 < \gamma_0 < \gamma_1 < \cdots < \gamma_m < 1$ and linearly interpo-

lating between the achieved performance values. (By Fact 3.7, the points $(0,0)$ and $(1,1)$ should also be included in this lower bound.) However, as mentioned above, the set $\mathcal{V}$ is too abstract to make this approach practical. Therefore, in the next section, we consider an extended definition of the ROC that applies to more concrete sets of tests.

Extending the ROC to Specific Families of Tests

In Definition 3.9, the ROC is defined with respect to the set of all tests, including randomized tests. This definition allowed us to characterize the ROC in terms of likelihood ratio tests, via the Neyman–Pearson Lemma (Lemma 3.10), or in terms of Pareto optimality. In practice, however, we want to be able to evaluate the performance of a given test or a given family of tests. For example, consider the parameterized family of fault detection schemes

$$\hat{\mathcal{V}} = \big\{V_\varepsilon \in \mathcal{V} : V_\varepsilon = (F, \delta_\varepsilon) \text{ and } \varepsilon > 0\big\}, \tag{3.18}$$

where the residual generator F is fixed and δ_ε is a threshold function defined as

$$\delta_\varepsilon(r) := \begin{cases} 0, & \text{if } |r| < \varepsilon, \\ 1, & \text{otherwise.} \end{cases}$$

Clearly, $V_\varepsilon \to V_{\text{yes}}$ as $\varepsilon \to 0$, regardless of the choice of F. Similarly, $V_\varepsilon \to V_{\text{no}}$ as $\varepsilon \to \infty$. Hence, the set of achievable performance points is a curve passing through $(0,0)$ and $(1,1)$ (see Figure 3.4). Using randomization, as in Example 3.5, the tests in $\hat{\mathcal{V}}$ can be used to achieve any performance point between this curve and the diagonal (i.e., any point in the convex hull of the curve). Hence, we have the following natural extension of the definition of the ROC.

Definition 3.11. Let $\hat{\mathcal{V}} \subset \mathcal{V}$ be some subset of tests. Define $\hat{\mathcal{W}}_k \subset \mathcal{W}_k$ to be the set of performance points that are achieved by some test in $\hat{\mathcal{V}}$. The upper boundary of the set $\hat{\mathcal{W}}_k$ is called the *receiver operating characteristic* (ROC) for the class of tests $\hat{\mathcal{V}}$ at time k.

3.5 Certifying and Visualizing Performance

3.5.1 Bounds on Performance Metrics

Given a fault detection scheme V, the system G_θ is said to be *available* at time k if no fault has occurred and no false alarm has been issued. Hence, the probability of availability is given by the performance metric $P_{\text{TN},k}$. In a physical system affected by wear and deterioration, $Q_{1,k} \to 1$ as $k \to \infty$, so $P_{\text{TN},k} \to 0$ as $k \to \infty$. Therefore, any bound on $P_{\text{TN},k}$ can only be enforced over a specified time window. Given $N \in \mathbb{N}$ and $a > 0$, one criterion for system

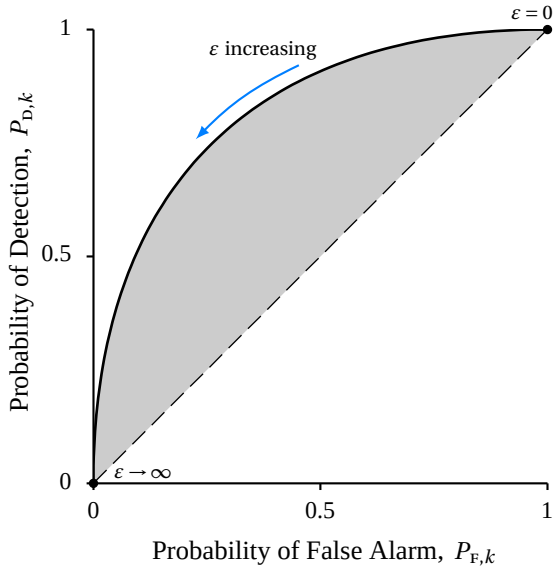


Figure 3.4. Set of performance points achieved by the family of tests given in equation (3.18). Varying the threshold ε yields a curve of performance points passing through $(0,0)$ and $(1,1)$. Randomization can be used to achieve any performance in the convex hull of this curve (shaded region).

availability is to require that

$$P_{TN,k} > a,$$

for $k = 0, 1, \ldots, N$. This type of bound is shown in Figure 3.5(a), where the constraint fails to hold for $k > k_f$. In terms of the performance metrics, the availability may be written as

$$P_{TN,k} = (1 - P_{F,k}) Q_{0,k},$$

for all k. Thus, the lower bound on availability can be translated to a time-varying upper bound on $P_{F,k}$, as follows:

$$P_{F,k} < 1 - \frac{a}{Q_{0,k}},$$

for $k = 0, 1, \ldots, N$. This type of bound is shown in Figure 3.5(b). Note that no fault detection scheme can satisfy the bound on availability once $Q_{0,k} \leq a$.

Given $\beta > \alpha > 0$, another natural performance criterion is to assert that the performance metrics $P_{F,k}$ and $P_{D,k}$ satisfy the constraints

$$P_{F,k} < \alpha \qquad \text{and} \qquad P_{D,k} > \beta,$$

for all k. A visualization of this type of bound is shown in Figure 3.6. In Figure 3.7, this constraint can be visualized in ROC space by plotting the ROC curves at a number of time steps $\{k_0, k_1, \ldots, k_m\}$. Unlike $P_{TN,k}$ which eventually converges to 0, the metrics $P_{F,k}$ and $P_{D,k}$ often converge to steady-state values, so the visualization in Figure 3.7 can depict the

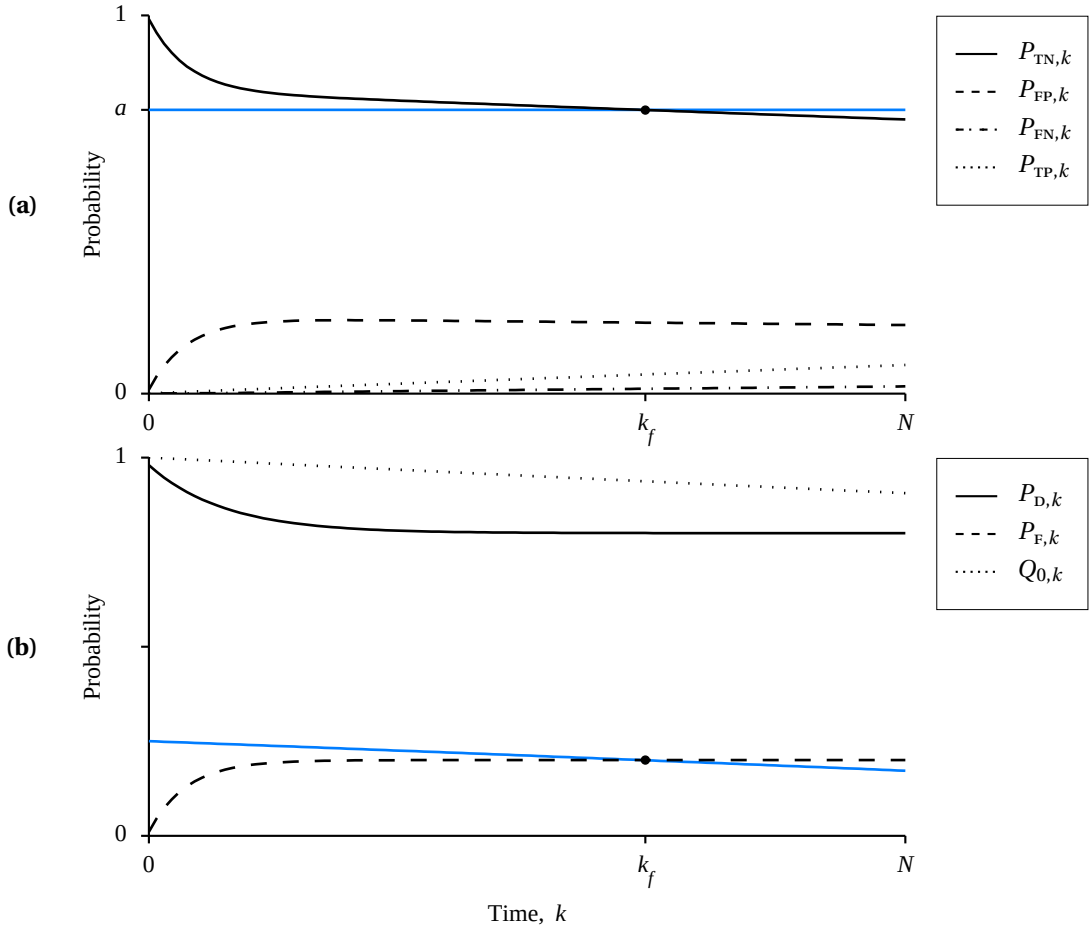


Figure 3.5. Visualization of a constraint on availability. On the top axes **(a)**, the performance metrics $\{P_{\mathrm{TN},k}, P_{\mathrm{FP},k}, P_{\mathrm{FN},k}, P_{\mathrm{TP},k}\}$ are plotted against time, and the constraint on availability is represented by a horizontal blue line. On the bottom axes **(b)**, the corresponding conditional probability metrics $\{P_{\mathrm{D},k}, P_{\mathrm{F},k}\}$, as well as the marginal probability $\{Q_{0,k}\}$, are plotted against time. Note that the lower bound on availability a translates to an upper bound (blue line) on $\{P_{\mathrm{F},k}\}$ that decreases in proportion to $\{Q_{0,k}\}$.

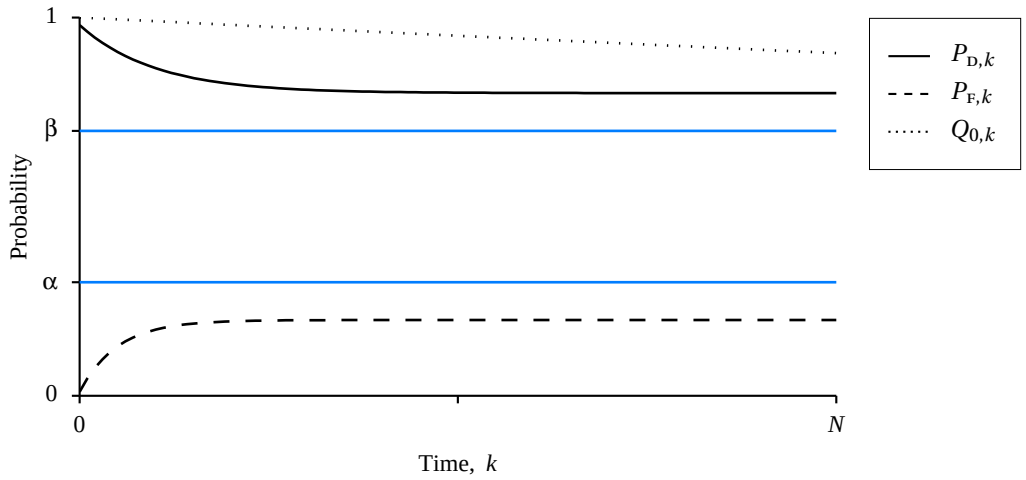


Figure 3.6. Visualization of a constraint on the performance metrics $\{P_{F,k}\}$ and $\{P_{D,k}\}$ over time. Here, the constraint is $P_{D,k} > \beta$ and $P_{F,k} < \alpha$, for $k = 0, 1, \ldots, N$. The marginal probability that the system is in the nominal mode, denoted $\{Q_{0,k}\}$, is shown for reference.

steady-state performance metrics if k_m is large enough.

3.5.2 Bound on Bayesian Risk

As discussed in Section 3.3.3, the Bayesian risk provides a general linear framework for aggregating the performance of a fault detection scheme into a single performance metric. For the sake of simplicity, assume that the loss matrix $L \in \mathbb{R}^2$ is constant for all time. Given a sequence $\{\bar{R}_k\}$, such that $\bar{R}_k > 0$ for all k, the bound on the Bayesian risk at time k is

$$R_k(Q, V) = L_{00}Q_{0,k} + L_{01}Q_{1,k} + (L_{01} - L_{00})P_{F,k}Q_{0,k} + (L_{11} - L_{10})P_{D,k}Q_{1,k} < \bar{R}_k.$$

At each k, the set of performance points $(P_{F,k}, P_{D,k})$ satisfying this bound is the intersection of some half-space in $\mathbb{R}^2$ with the ROC space $[0,1]^2$ (see Figure 3.8). The boundary of this half-space is determined the loss matrix L and the probability $Q_{0,k}$. Clearly, if the ideal performance point $(0,1)$ does not lie in this half-space at time k, then the bound $R_k < \bar{R}_k$ is too stringent.

Note that as $Q_{0,k} \to 1$, the bound on risk approaches

$$L_{00} + (L_{01} - L_{00})P_{F,k} < \bar{R} \quad \Longleftrightarrow \quad P_{F,k} < \frac{\bar{R} - L_{00}}{L_{01} - L_{00}}.$$

Similarly, as $Q_{0,k} \to 0$, the bound approaches

$$L_{01} + (L_{11} - L_{10})P_{D,k} < \bar{R} \quad \Longleftrightarrow \quad P_{D,k} > \frac{L_{01} - \bar{R}}{L_{10} - L_{11}}.$$

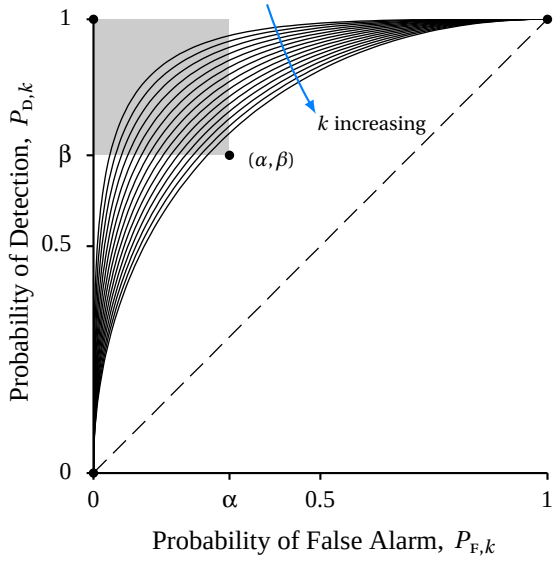


Figure 3.7. Visualization of a constraint on the performance metrics $P_{F,k}$ and $P_{D,k}$ in ROC space. Unlike Figure 3.6, which shows the performance for a single test $V = (F, \delta)$, this visualization shows the performance over an entire family of tests. However, it is less clear in this visualization which curve corresponds to a given point in time.

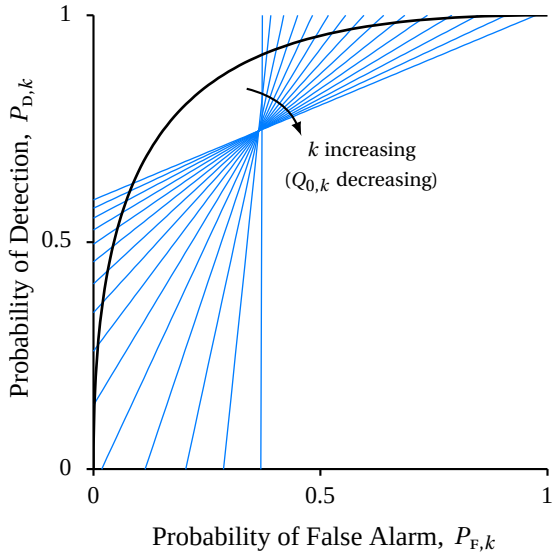


Figure 3.8. Visualization of a constraint on Bayesian risk in ROC space. Each blue line represents the Bayesian risk bound at a different time step. Note that as $Q_{0,k}$ decreases with time, the slope of the bound decreases and the probability of detection $P_{D,k}$ plays a more significant role in satisfying the constraint. A ROC curve corresponding to a single time step is plotted for reference.

In general, as $Q_{0,k}$ decreases, the slope of the boundary line that delineates the set of acceptable performance points also decreases. Hence, from a Bayesian risk perspective, when $Q_{0,k}$ is large and faults are unlikely to occur, it is more important to avoid false alarms. On the other hand, when $Q_{0,k}$ is small and faults are likely to occur, it is more important to detect faults. Figure 3.8 shows a typical plot of the evolution of the Bayesian risk bound through time.

3.6 Extension to Fault Isolation and Identification

In this section, we extend our performance analysis to fault isolation and identification problems. As in the fault detection case, there is a set of joint probabilities that fully characterizes the performance, and a set of conditional probabilities that characterize the performance relative to the marginal probabilities of the hypotheses being considered. We show that these sets of performance metrics are equivalent. We also show how the concept of Bayesian risk is defined in the multi-hypothesis case. Finally, we provide some brief comments on how the ROC curve can be extended, as well.

3.6.1 Quantifying Accuracy

Consider the general fault isolation problem, where the parameter space is partitioned as

$$\Theta = \Theta_0 \sqcup \Theta_1 \sqcup \cdots \sqcup \Theta_q,$$

for some $q > 1$. As in the simpler fault detection case, $\Theta_0 = \{0\}$ represents the nominal parameter value, while the set Θ_i, for $i > 0$, represents the ith class of faulty behavior. If Θ is finite, fault identification can be achieved by taking each Θ_i to be a singleton set. The corresponding set of decisions is

$$\mathscr{D} := \{0, 1, \ldots, q\}.$$

At each time k, define the events

$$D_{i,k} := \{d_k = i\} \qquad \text{and} \qquad H_{j,k} := \{\theta_k \in \Theta_j\},$$

for all $i, j \in \mathscr{D}$. The performance metrics (3.1)–(3.4) are extended to the multi-hypothesis case by the *performance matrix* $J_k \in \mathbb{R}^{(q+1)\times(q+1)}$, which is defined as

$$J_k(i, j) := \mathbf{P}(D_{i,k} \cap H_{j,k}), \qquad i, j \in \mathscr{D}.$$

Hence, J_k can be viewed as a confusion matrix for the multi-hypothesis case. Because $\{D_{0,k}, D_{1,k}, \ldots, D_{q,k}\}$ and $\{H_{0,k}, H_{1,k}, \ldots, H_{q,k}\}$ form partitions of the sample space Ω, the performance matrix satisfies identities analogous to those in equations (3.6)–(3.10). As in

equations 3.6 and 3.7, the ith row-sum of J_k is

$$\sum_{j=0}^{q} J_k(i,j) = \sum_{j=0}^{q} \mathbf{P}(D_{i,k} \cap H_{j,k}) = \mathbf{P}\left(D_{i,k} \cap \bigcup_{j=0}^{q} H_{j,k}\right) = \mathbf{P}(D_{i,k} \cap \Omega) = \mathbf{P}(D_{i,k}).$$

Similarly, the jth column-sum of J_k is

$$\sum_{i=0}^{q} J_k(i,j) = \sum_{i=0}^{q} \mathbf{P}(D_{i,k} \cap H_{j,k}) = \mathbf{P}(H_{j,k}), \tag{3.19}$$

as in equations 3.8 and 3.9. Of course, summing all the entries of J_k gives $\mathbf{P}(\Omega) = 1$, as in equation 3.10. This implies that there are only $(q+1)^2 - 1$ independent performance metrics that need to be evaluated in the multi-hypothesis case.

As in the fault detection case, it is often useful to decouple the issue of test performance from the reliability of the underlying system. Consider the matrix of conditional probabilities $C_k \in \mathbb{R}^{(q+1)\times(q+1)}$ defined as

$$C_k(i,j) \coloneqq \mathbf{P}(D_{i,k} \mid H_{j,k}), \qquad i,j \in \mathscr{D}. \tag{3.20}$$

Also, define the matrix $Q_k \in \mathbb{R}^{(q+1)\times(q+1)}$ of prior probabilities as

$$Q_k \coloneqq \operatorname{diag}\Big\{\mathbf{P}(H_{0,k}), \mathbf{P}(H_{1,k}), \ldots, \mathbf{P}(H_{q,k})\Big\}. \tag{3.21}$$

Proposition 3.12. *The matrix J_k and the pair of matrices (C_k, Q_k) provide equivalent sets of performance metrics.*

Proof. By the definition of conditional probability (see Section 2.2.1),

$$\begin{aligned}
(C_k Q_k)(i,j) &= \sum_{\ell=0}^{q} C_k(i,\ell) Q_k(\ell,j) \\
&= C_k(i,j) Q_k(j,j) \\
&= \mathbf{P}(D_{i,k} \mid H_{j,k})\, \mathbf{P}(H_{j,k}) \\
&= \mathbf{P}(D_{i,k} \cap H_{j,k}) \\
&= J_k(i,j),
\end{aligned}$$

for all $i,j \in \mathscr{D}$, so $J_k = C_k Q_k$. Also, by equation (3.19), the matrix Q_k can be computed

from J_k by taking column-sums. If $Q_k^\dagger$ is the pseudoinverse of Q_k [46], then

$$\begin{aligned}(J_k Q_k^\dagger)(i,j) &= \sum_{\ell=0}^{q} J_k(i,\ell) Q_k^\dagger(\ell,j) \\ &= J_k(i,j) Q_k^\dagger(j,j) \\ &= \begin{cases} \mathbf{P}(D_{i,k} \cap H_{j,k}) \mathbf{P}(H_{j,k})^{-1}, & \text{if } \mathbf{P}(H_{j,k}) \neq 0, \\ 0, & \text{otherwise} \end{cases} \\ &= \mathbf{P}(D_{i,k} \mid H_{j,k}) \\ &= C_k(i,j),\end{aligned}$$

for all $i, j \in \mathscr{D}$, so $C_k = J_k Q_k^\dagger$. Hence, the pair (C_k, Q_k) provides an alternate means of quantifying performance that is numerically equivalent to the performance matrix J_k. □

Remark 3.13. At a high level, evaluating (C_k, Q_k) requires the same amount of effort as evaluating J_k, in the sense that both formulations have the same number of independent quantities to compute. Indeed, the jth column-sum of C_k is

$$\sum_{i=0}^{q} C_k(i,j) = \sum_{i=0}^{q} \mathbf{P}(D_{i,k} \mid H_{j,k}) = \mathbf{P}\left(\bigcup_{i=0}^{q} D_{i,k} \mid H_{j,k} \right) = \mathbf{P}(\Omega \mid H_{j,k}) = 1,$$

so C_k has $(q+1)^2 - (q+1)$ independent entries. Also, the sum of all the elements of Q_k is

$$\sum_{i=0}^{q} Q_k(i,i) = \sum_{i=0}^{q} \mathbf{P}(H_{i,k}) = \mathbf{P}\left(\bigcup_{i=0}^{q} H_{i,k} \right) = \mathbf{P}(\Omega) = 1,$$

so Q_k has q independent entries. Therefore, in total, there are $(q+1)^2 - 1$ quantities that must be computed to obtain C_k and Q_k, which is the same as the number of independent entries of J_k. However, it is often the case that computing a single entry of J_k is more straightforward. ◇

3.6.2 Bayesian Risk

As in the fault detection case, we can define a loss matrix $L \in \mathbb{R}^{(q+1)\times(q+1)}$ with nonnegative entries, such that L_{ij} reflects the subject loss of deciding $d_k = j$ when hypothesis $\mathscr{H}_{i,k}$ is true. The corresponding Bayesian risk is given by

$$R_k(Q,V) = \sum_{i=0}^{q} \sum_{j=0}^{q} L_{ij} \mathbf{P}(D_{j,k} \cap H_{i,k}) = \sum_{i=0}^{q} \sum_{j=0}^{q} L_{ij} J_k(j,i) = \sum_{i=0}^{q} \sum_{j=0}^{q} L_{ij} C_k(j,i) Q_k(i,i).$$

Of course, a different loss matrix L_k can be used at each time step.

3.6.3 ROC Curves for Multiple Hypotheses

Recall that the performance of a fault detection scheme is decoupled from the reliability of the underlying system by considering the conditional probabilities $P_{\mathrm{F},k}$ and $P_{\mathrm{D},k}$. Similarly, the performance of a fault isolation scheme is given by the matrix C_k, which has $q(q+1)$ independent entries. In [33] and [30], the ROC for fault isolation is defined as the set of Pareto optimal values of C_k plotted in the hypercube $[0,1]^{q(q+1)}$. As in the binary case, the set of achievable performance points $\mathscr{W}_k$ is a convex set [88]. The interpretation of the volume of this set is given in [45] for $q=2$ and in [36, 43] for the general case. Unfortunately, it is difficult to visualize the time-varying nature of the multi-hypothesis ROC surface.

Chapter 4

Computational Framework

4.1 Introduction

In this chapter, we discuss the computational issues involved in evaluating the performance metrics defined in Chapter 3. First, we give an overview of these computational issues, which serves as a framework for the remainder of the chapter. Then, we address these issues by imposing assumptions on each component of the fault diagnosis problem: the fault model, the dynamics of the system and residual generator, and the decision function. Together, these assumptions ensure that the performance metrics can be computed efficiently. Finally, we provide algorithms in pseudocode form and prove that the assumptions do indeed result in algorithms with polynomial running time.

Recall that the performance metrics, at time k, are given by the formula

$$J_k(i,j) \coloneqq \mathbf{P}\big(D_{j,k} \cap H_{i,k}\big) = \mathbf{P}\big(D_{j,k} \cap \{\theta_k \in \Theta_i\}\big) = \int_{\Theta_i} \mathbf{P}\big(D_{j,k} \cap \{\theta_k = \vartheta_k\}\big)\ \mathrm{d}\vartheta_k,$$

for each $i, j \in \mathscr{D}$. Because the residual r_k is the output of a dynamic system, each of the random variables $v_0, v_1, \ldots, v_k$ has an impact on r_k, as well as the decision d_k. However, the relationship between r_k and the noise sequence $v_{0:k}$ is not specified unless the entire parameter sequence $\theta_{0:k}$ is known. This issue is addressed by marginalizing over the random variables $\theta_{0:k-1}$ as follows:

$$\begin{aligned}\mathbf{P}\big(D_{j,k} \cap \{\theta_k = \vartheta_k\}\big) &= \int_{\Theta^k} \mathbf{P}\big(D_{j,k} \cap \{\theta_{0:k} = \vartheta_{0:k}\}\big)\ \mathrm{d}\vartheta_{0:k-1} \\ &= \int_{\Theta^k} \mathbf{P}\big(D_{j,k} \mid \theta_{0:k} = \vartheta_{0:k}\big) p_\theta(\vartheta_{0:k})\ \mathrm{d}\vartheta_{0:k-1},\end{aligned}$$

where Θ^k is the k-fold Cartesian product $\Theta \times \cdots \times \Theta$. Thus, the (i,j)th element of the matrix J_k can be written as

$$J_k(i,j) = \int_{\Theta_i} \int_{\Theta^k} \mathbf{P}\big(D_{j,k} \mid \theta_{0:k} = \vartheta_{0:k}\big) p_\theta(\vartheta_{0:k})\ \mathrm{d}\vartheta_{0:k-1}\ \mathrm{d}\vartheta_k. \tag{4.1}$$

Writing the performance metrics in this manner reveals the following computational issues:

1. We must be able to efficiently evaluate the probability density function $p_\theta(\theta_{0:k})$. This issue is addressed by assuming that $\{\theta_k\}$ is a Markov chain with known distribution.

2. The integral must be taken over all $\vartheta_{0:k} \in \Theta^k \times \Theta_i$. Unless a closed-form analytical solution exists, this integral must be evaluated numerically, in which case the high dimensionality makes computation intractable. To address this issue, we make the assumptions necessary to reduce $\Theta^k \times \Theta_i$ to a finite set of manageable size.

3. For each $\vartheta_{0:k} \in \Theta^k \times \Theta_i$, computing the probability

$$\mathbf{P}\big(D_{j,k} \mid \theta_{0:k} = \vartheta_{0:k}\big) = \mathbf{P}(d_k = j \mid \theta_{0:k} = \vartheta_{0:k}) \tag{4.2}$$

   requires knowledge of the conditional density $p_{d|\theta}(d_k \mid \theta_{0:k})$. This issue is addressed in two stages. First, we assume that the system G_θ and the residual generator have a sufficient structure to ensure that $p_{r|\theta}(r_k \mid \theta_{0:k})$ is a Gaussian density with known mean and variance. Then, we consider classes of decision functions, such that the probability in equation (4.2) can be efficiently computed.

4.2 Fault Model

Assume that the fault parameter process $\{\theta_k\}_{k\geq 0}$ is a Markov chain with finite state space

$$\Theta := \{0, 1, \ldots, m\}.$$

At each time k, let $\pi_k \in \mathbb{R}^{m+1}$ be the probability mass function (PMF) of θ_k, and let $\Pi_k \in \mathbb{R}^{(m+1)\times(m+1)}$ be the transition probability matrix. That is,

$$\pi_k(i) := \mathbf{P}(\theta_k = i), \qquad i \in \Theta$$

and

$$\Pi_k(i,j) := \mathbf{P}\big(\theta_{k+1} = j \mid \theta_k = i\big), \qquad i, j \in \Theta.$$

Assume that the initial PMF π_0 and the transition probability matrices $\{\Pi_k\}$ are known. Note that the triple $\big(\Theta, \{\Pi_k\}, \pi_0\big)$ completely defines the probability distribution of the fault parameter sequence $\{\theta_k\}$. We write $\theta \sim \big(\Theta, \{\Pi_k\}, \pi_0\big)$ to denote this fact.

The first computational issue raised in Section 4.1 is the efficient evaluation of the probability mass function $p_\theta(\theta_{0:k})$. The following simple fact about Markov chains indicates that, under mild assumptions, this computation takes only $O(k)$ time.

Fact 4.1. *Given a Markov chain $\theta \sim (\Theta, \{\Pi_k\}, \pi_0)$, let $\ell > 0$ and $\vartheta_{0:\ell} \in \Theta^{\ell+1}$. If $\Pi_k(i,j)$ can be computed or retrieved in $O(1)$ time, for any $k \geq 0$ and any $i, j \in \Theta$, then*

$$p_\theta(\vartheta_{0:\ell}) = \mathbf{P}(\theta_{0:\ell} = \vartheta_{0:\ell})$$

can be computed in $O(\ell)$ time.

Proof. By definition, $\mathbf{P}(\theta_0 = \vartheta_0) = \pi_0(\vartheta_0)$. Let $0 < \tau \leq \ell$. Because $\{\theta_k\}$ is Markov, the probability of the event $\{\theta_{0:\tau} = \vartheta_{0:\tau}\}$ can be factored as

$$\begin{aligned}\mathbf{P}(\theta_{0:\tau} = \vartheta_{0:\tau}) &= \mathbf{P}(\theta_\tau = \vartheta_\tau \mid \theta_{0:\tau-1} = \vartheta_{0:\tau-1})\, \mathbf{P}(\theta_{0:\tau-1} = \vartheta_{0:\tau-1}) \\ &= \mathbf{P}(\theta_\tau = \vartheta_\tau \mid \theta_{\tau-1} = \vartheta_{\tau-1})\, \mathbf{P}(\theta_{0:\tau-1} = \vartheta_{0:\tau-1}) \\ &= \Pi_{\tau-1}(\vartheta_{\tau-1}, \vartheta_\tau)\, \mathbf{P}(\theta_{0:\tau-1} = \vartheta_{0:\tau-1}).\end{aligned}$$

Hence, by induction on τ,

$$\mathbf{P}(\theta_{0:\ell} = \vartheta_{0:\ell}) = \Pi_{\ell-1}(\vartheta_{\ell-1}, \vartheta_\ell)\, \Pi_{\ell-2}(\vartheta_{\ell-2}, \vartheta_{\ell-1}) \cdots \Pi_0(\vartheta_0, \vartheta_1)\, \pi_0(\vartheta_0).$$

Since this computation requires ℓ evaluations of the transition probability matrices and ℓ scalar multiplications, the overall time-complexity is $\ell O(1) + \ell O(1) = O(\ell)$. □

The second computational issue raised in Section 4.1 is the high dimensionality of the integral in equation (4.1). Since the fault parameter space Θ is assumed to be finite, equation (4.1) can be written as

$$J_k(i,j) = \sum_{\vartheta_{0:k} \in \Theta^k \times \Theta_i} \mathbf{P}(D_{j,k} \mid \theta_{0:k} = \vartheta_{0:k})\, \mathbf{P}(\theta_{0:k} = \vartheta_{0:k}), \tag{4.3}$$

for all $i, j \in \mathscr{D}$ and all $k \geq 0$. Of course, exchanging an integral for a summation is of little use if the summation has an intractable number of terms (i.e., the number of terms grows exponentially with k). In general, the summation (4.3) has $m^k m_i$ terms, where $m_i \coloneqq |\Theta_i|$. The following example illustrates the practical implications of this exponential growth.

Example 4.2. Suppose that $\{y_k\}_{k\geq 0}$ is a stochastic process taking values in $\mathbb{R}$ such that the conditional density $p_{y|\theta}(y_k \mid \theta_{0:k} = \vartheta_{0:k})$ is Gaussian for all k and all $\vartheta_{0:k} \in \Theta^{k+1}$. Then, the marginal density of y_k can be written as the sum

$$p_y(y_k) = \sum_{\vartheta_{0:k} \in \Theta^{k+1}} p_{y|\theta}(y_k \mid \theta_{0:k} = \vartheta_{0:k})\, \mathbf{P}(\theta_{0:k} = \vartheta_{0:k}).$$

In this sum, each term is represented by three scalars: the mean and variance of the Gaussian density $p_{y|\theta}(y_k \mid \theta_{0:k} = \vartheta_{0:k})$ and the probability $\mathbf{P}(\theta_{0:k} = \vartheta_{0:k})$. If these data are stored in IEEE single precision (i.e., 32 bits per number), then each term requires $3 \times 32 = 96$ bits or 12 bytes

of storage. In the simplest case, where $\Theta = \{0,1\}$, there are 2^{k+1} terms to store. For example, at $k = 36$, the total storage needed is

$$12 \times 2^{36+1} \approx 1.65 \times 10^{12}\,\text{bytes} > 1\,\text{terabyte!}$$

Since physical systems are often sampled at twice their bandwidth or more, the amount of time represented by 36 discrete samples is small compared to the time-scale of the system. ◇

4.2.1 Limiting Complexity with Structured Markov Chains

Although the number of paths in Θ^k grows exponentially with k, not all of the paths need to be considered in computing equation (4.3), because some paths have zero probability of occurring. That is, some sequences of faults cannot occur under the given model. This section explores, from a theoretical perspective, what properties the Markov chain must have in order to reduce the number of terms in equation (4.3) to a tractable number.

Terminology

Definition 4.3. Given a Markov chain θ taking values in Θ, let $\ell \geq 0$ and $\vartheta_{0:\ell} \in \Theta^{\ell+1}$. If the event $\{\theta_{0:\ell} = \vartheta_{0:\ell}\}$ has nonzero probability, then $\vartheta_{0:\ell}$ is said to be a *possible path* of $\{\theta_k\}$. Otherwise, $\vartheta_{0:\ell}$ is said to be an *impossible path*.

Definition 4.4. A Markov chain is said to be *tractable* if the number of possible paths of length ℓ is $O(\ell^c)$, for some constant c.

Definition 4.5. Let θ be a Markov chain taking values in Θ. A state $\vartheta \in \Theta$ is said to be *degenerate* if $\mathbf{P}(\theta_k = \vartheta) = 0$, for all k (i.e., no possible path ever visits ϑ). A Markov chain with one or more degenerate states is said to be *degenerate*.

Remark 4.6. Our definition of a tractable Markov chain is based on the conventional notion that polynomial-time algorithms are tractable, whereas algorithms requiring superpolynomial time are intractable [19]. This idea is known as *Cobham's Thesis* or the *Cobham–Edmonds Thesis* [16, 29]. ◇

Remark 4.7. Suppose that $\theta \sim \big(\Theta, \{\Pi_k\}, \pi_0\big)$ is a Markov chain with a nonempty set of degenerate states $\bar{\Theta} \subset \Theta$. Let $\hat{\theta}$ be the Markov chain formed by removing the degenerate states from Θ and trimming the matrices $\{\Pi_k\}$ and the PMF π_0 accordingly. Clearly, any possible path of θ is a possible path of $\hat{\theta}$, so the tractability of θ can be determined by analyzing the non-degenerate Markov chain $\hat{\theta}$. ◇

Since the goal is to relate the tractability of Markov chains to properties of directed graphs, we must first establish some definitions from graph theory.

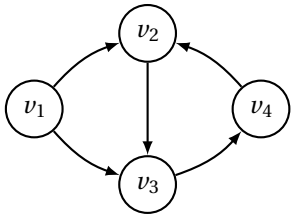


Figure 4.1. Simple example of a directed graph with four vertices and five edges.

Definition 4.8. A *directed graph* is a collection of points, called *vertices*, and ordered pairs of vertices, called *edges*, that begin at one vertex and end at another. More precisely, a graph is a pair (V, E), where the set of vertices V is any nonempty set, and the set of edges $E \subset V \times V$ is such that if $(u, v) \in E$, then the graph contains the edge $u \to v$. The same graph may be represented by the pair (V, A), where $A \in \{0,1\}^{|V|\times|V|}$ is a matrix, such that $(u, v) \in E$ if and only if $A(u, v) = 1$. The matrix A is called the *adjacency matrix* of the graph (V, E).

Definition 4.9. Given a directed graph (V, E), a *cycle* is defined as a sequence of vertices

$$\{v_1, v_2, \ldots, v_m, v_1\},$$

such that

$$v_1 \to v_2 \to \cdots \to v_m \to v_1.$$

That is, $(v_i, v_i + 1) \in E$ for $i = 1, 2, \ldots, m$ and $(v_m, v_1) \in E$. A directed graph with no cycles is said to be *acyclic*.

Example 4.10. Consider the directed graph shown in Figure 4.1. The set of vertices is

$$V = \{v_1, v_2, v_3, v_4\},$$

and the set of edges is

$$E = \big\{(v_1, v_2), (v_1, v_3), (v_2, v_3), (v_3, v_4), (v_4, v_2)\big\}.$$

The corresponding adjacency matrix is

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Note that this graph contains the cycle $\{v_2, v_3, v_4, v_2\}$. ◇

Main Results

The following theorems relate the tractability of Markov chains to easily-verifiable properties of directed graphs.

Theorem 4.11. *Given a non-degenerate, time-homogeneous Markov chain $\theta \sim (\Theta, \Pi, \pi_0)$, define the matrix A as follows:*

$$A(i,j) \coloneqq \begin{cases} 1 & \text{if } i \neq j,\ \Pi(i,j) \neq 0, \\ 0 & \text{otherwise,} \end{cases} \tag{4.4}$$

for all $i, j \in \Theta$. Then, the Markov chain θ is tractable if and only if the directed graph with vertices Θ and adjacency matrix A is acyclic.

Theorem 4.12. *Given a non-degenerate Markov chain $\theta \sim \left(\Theta, \{\Pi_k\}, \pi_0\right)$ with time-varying transition probabilities, define the matrix A as follows:*

$$A(i,j) \coloneqq \begin{cases} 1 & \text{if } i \neq j,\ \Pi_k(i,j) \neq 0 \text{ for some } k \geq 0 \\ 0 & \text{otherwise,} \end{cases} \tag{4.5}$$

for all $i, j \in \Theta$. Then, the Markov chain θ is tractable if the directed graph with vertices Θ and adjacency matrix A is acyclic.

Remark 4.13. Note that Theorem 4.11 gives a necessary and sufficient condition for tractability, while Theorem 4.12 only gives a sufficient condition. Indeed, Example 4.18 (below) shows that the graph-theoretic condition stated in Theorem 4.12 is not necessary for tractability. ◇

Remark 4.14. The presence of cycles in a directed graph $G = (V, E)$ can be determined using the Depth-First Search (DFS) algorithm in $O(|V| + |E|)$ time, where V is the set of vertices and E is the set of edges [19, 21]. For the graphs considered in Theorems 4.11 and 4.12, the number of vertices is $|\Theta| = m + 1$, and the number of edges is no more than $(m+1)^2 - (m+1) = m^2 + m$, since the diagonal entries of A must be 0. Hence, the tractability of a given Markov chain can be verified using DFS in $O(m^2)$ time. ◇

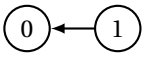
Example 4.15. Suppose that $\Theta = \{0, 1\}$ and

$$\Pi = \begin{bmatrix} 1 & 0 \\ 1-p & p \end{bmatrix},$$

for some $p \in (0,1)$. Then, the corresponding adjacency matrix is

$$A = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}.$$

The graph corresponding to (Θ, A) is



which is clearly acyclic, so (Θ, Π, π_0) is tractable. ◇

Example 4.16. Suppose that $\Theta = \{0,1\}$ and

$$\Pi = \begin{bmatrix} p & 1-p \\ 1-q & q \end{bmatrix},$$

for some $p, q \in (0,1)$. Then, the corresponding adjacency matrix is

$$A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

The graph corresponding to (Θ, A) is

which has the cycles $\{0,1,0\}$ and $\{1,0,1\}$, so (Θ, Π, π_0) is intractable (see Example 4.2). ◇

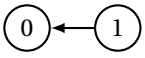
Example 4.17. Suppose that $\Theta = \{0,1\}$ and

$$\Pi_k = \begin{bmatrix} 1 & 0 \\ \max\{0, 1-kp\} & \min\{1, kp\} \end{bmatrix},$$

for some $p \in (0,1)$ and all $k \geq 0$. Then, the corresponding adjacency matrix is

$$A = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}.$$

The graph corresponding to (Θ, A) is



which is clearly acyclic, so $\left(\Theta, \{\Pi_k\}, \pi_0\right)$ is tractable. ◇

Example 4.18. Suppose that $\Theta = \{0,1\}$ and

$$\Pi_k = \begin{bmatrix} p_k & 1-p_k \\ 1-q & q \end{bmatrix},$$

where $q \in (0,1)$ and

$$p_k = \begin{cases} 0.5 & \text{if } k < 10 \\ 1 & \text{otherwise.} \end{cases}$$

Then, the corresponding adjacency matrix is

$$A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

As in Example 4.16, the graph (Θ, A) contains cycles, so Theorem 4.12 does not apply. However, in this simple case, we can see that the Markov chain $\theta \sim \big(\Theta, \{\Pi_k\}, \pi_0\big)$ is tractable. Indeed, consider a path $\vartheta_{0:\ell} \in \Theta^{\ell+1}$, where $\ell \geq 10$. Split the path into two parts, $\vartheta_{0:9}$ and $\vartheta_{10:\ell}$, and let $\hat{\theta}$ be a Markov chain, such that $\hat{\theta}_k = \theta_{k-10}$, for all $k \geq 0$. The first part $\vartheta_{0:9}$ can take 2^{10} different values, while the second part $\vartheta_{10:\ell}$ can be considered as a path of the shifted Markov chain $\hat{\theta}$. Since $\hat{\theta}$ has the same time-homogeneous distribution as the tractable Markov chain in Example 4.15, the number of possible paths of the original Markov chain θ must be polynomial. ◇

Before proving Theorems 4.11 and 4.12, we establish a series of lemmas, each of which is useful in its own right. Then, these lemmas are used to formulate succinct proofs of the main results.

Supporting Lemmas

The first two lemmas state the notion of tractability in terms of the structure of the transition probability matrices.

Lemma 4.19. *Let $\theta \sim \big(\Theta, \{\Pi_k\}, \pi_0\big)$ be a Markov chain, such that Π_k is upper-triangular, for all k. Then, every possible path $\vartheta_{0:\ell} \in \Theta^{\ell+1}$ satisfies the inequalities*

$$\vartheta_0 \leq \vartheta_1 \leq \cdots \leq \vartheta_{\ell-1} \leq \vartheta_\ell.$$

Proof. Let $\vartheta_{0:\ell} \in \Theta^{\ell+1}$ be a possible path. Then, the inequality

$$\Pi_{\ell-1}(\vartheta_{\ell-1}, \vartheta_\ell)\, \Pi_{\ell-2}(\vartheta_{\ell-2}, \vartheta_{\ell-1}) \cdots \Pi_0(\vartheta_0, \vartheta_1)\, \pi_0(\vartheta_0) = \mathbf{P}(\theta_{0:\ell} = \vartheta_{0:\ell}) > 0.$$

implies that $\Pi_i(\vartheta_{i-1}, \vartheta_i) > 0$, for $i = 1, 2, \ldots, \ell$. Since each Π_i is upper triangular, it must be that $\vartheta_{i-1} \leq \vartheta_i$, for $i = 1, 2, \ldots, \ell$. □

Lemma 4.20. *Let $\theta \sim \big(\Theta, \{\Pi_k\}, \pi_0\big)$ be a Markov chain, such that $\Theta = \{0, 1, \ldots m\}$ and Π_k is upper-triangular, for all k. Then, the number of possible paths $\vartheta_{0:\ell} \in \Theta^{\ell+1}$ is*

$$\frac{\ell^m}{m!} + O(\ell^{m-1}).$$

Proof. Let $\vartheta_{0:\ell}$ be a possible path. By Lemma 4.19, $\vartheta_{i-1} \le \vartheta_i$, for $i = 1, \dots, \ell$, so the remainder of the path $\vartheta_{1:\ell}$ makes at most $m - \vartheta_0$ transitions from one state to another. If n such transitions occur, then there are at most $\binom{m-\vartheta_0}{n}$ distinct sets of states that $\vartheta_{1:\ell}$ may visit, and there are no more than $\binom{\ell}{n}$ combinations of times at which these transitions may occur. Therefore, the total number of possible paths up to time ℓ is upper-bounded by

$$C(\ell) \coloneqq \sum_{\vartheta_0=0}^{m} \sum_{n=0}^{m-\vartheta_0} \binom{m-\vartheta_0}{n} \binom{\ell}{n}.$$

The bound

$$\binom{\ell}{n} \coloneqq \frac{\ell(\ell-1)\cdots(\ell-n+1)}{n!} < \frac{\ell^n}{n!},$$

implies that

$$C(\ell) < \sum_{\vartheta_0=0}^{m} \sum_{n=0}^{m-\vartheta_0} \binom{m-\vartheta_0}{n} \frac{\ell^n}{n!} = \frac{\ell^m}{m!} + O(\ell^{m-1}). \qquad \square$$

Of course, the structure of the transition probability matrices $\{\Pi_k\}$ depends on how the states of the Markov chain are labeled. Since a relabeling of the states is affected by a permutation, the following lemma analyzes the relationship between a Markov chain and its permuted counterpart.

Lemma 4.21. *Let $\theta \sim \big(\Theta, \{\Pi_k\}, \pi_0\big)$ be a Markov chain, and let $\sigma\colon \Theta \to \Theta$ be a permutation. Define*

$$\hat{\pi}_0(i) = \pi_0\big(\sigma(i)\big), \qquad i \in \Theta, \tag{4.6}$$

and for all $k \ge 0$ define

$$\hat{\Pi}_k(i,j) = \Pi_k\big(\sigma(i), \sigma(j)\big), \qquad i, j \in \Theta, \tag{4.7}$$

Then, the Markov chain $\hat{\theta} \sim \big(\Theta, \{\hat{\Pi}_k\}, \hat{\pi}_0\big)$ has the same number of possible paths as θ.

Proof. Fix $\ell > 0$ and let $\hat{\vartheta}_{0:\ell}$ be a path of $\hat{\theta}$. For $i = 0, 1, \dots, \ell$, define $\vartheta_i \coloneqq \sigma(\hat{\vartheta}_i)$. Then, the equality

$$\begin{aligned}
\mathbf{P}(\hat{\theta}_{0:\ell} = \hat{\vartheta}_{0:\ell}) &= \hat{\Pi}(\hat{\vartheta}_{\ell-1}, \hat{\vartheta}_\ell) \cdots \hat{\Pi}(\hat{\vartheta}_0, \hat{\vartheta}_1)\, \hat{\pi}(\hat{\vartheta}_0) \\
&= \Pi\big(\sigma(\hat{\vartheta}_{\ell-1}), \sigma(\hat{\vartheta}_\ell)\big) \cdots \Pi\big(\sigma(\hat{\vartheta}_0), \sigma(\hat{\vartheta}_1)\big)\, \pi\big(\sigma(\hat{\vartheta}_0)\big) \\
&= \Pi(\vartheta_{\ell-1}, \vartheta_\ell) \cdots \Pi(\vartheta_0, \vartheta_1)\, \pi(\vartheta_0) \\
&= \mathbf{P}(\theta_{0:\ell} = \vartheta_{0:\ell})
\end{aligned}$$

implies that $\hat{\vartheta}_{0:\ell}$ is a possible path of $\hat{\theta}$ if and only if $\vartheta_{0:\ell}$ is a possible path of θ. Since the permutation σ is a bijection, θ and $\hat{\theta}$ have the same number of possible paths. $\square$

Since relabeling the states of a Markov chain does not alter its tractability, the next step is to seek conditions under which the states can be permuted to achieve upper-triangular transition probability matrices. The following lemmas show that the existence of such permutations can be related to the presence of cycles in the graph (Θ, A).

Lemma 4.22. *Let $\Theta = \{0, 1, \dots, m\}$. Given a matrix $\Pi \in \mathbb{R}^{(m+1)\times(m+1)}$, define the matrix A as in Theorem 4.11. Then, there exists a permutation $\sigma\colon \Theta \to \Theta$, such that the matrix*

$$\hat{\Pi}(i, j) \coloneqq \Pi\big(\sigma(i), \sigma(j)\big), \qquad i, j \in \Theta \tag{4.8}$$

is upper-triangular if and only if the directed graph with vertices Θ and adjacency matrix A is acyclic.

Proof. Suppose that the permutation σ makes $\hat{\Pi}$ upper triangular. Let $n > 0$ and let

$$v_0 \to v_1, \to \cdots \to v_n$$

be a path on the graph (Θ, A). For $i = 1, \dots, n$, the existence of the edge $v_{i-1} \to v_i$ implies that $v_{i-1} \neq v_i$ and

$$\Pi(v_{i-1}, v_i) = \hat{\Pi}\big(\sigma^{-1}(v_{i-1}), \sigma^{-1}(v_i)\big) \neq 0.$$

Since $\hat{\Pi}$ is upper-triangular,

$$\sigma^{-1}(v_0) < \sigma^{-1}(v_1) < \cdots < \sigma^{-1}(v_n).$$

Therefore, no path can never visit the same vertex twice, i.e., the graph (Θ, A) is acyclic.

Now, suppose that the graph (Θ, A) is acyclic. The Depth-First Search (DFS) algorithm generates a pre-visit and post-visit number for each vertex $v \in \Theta$ as it begins exploring v and finishes exploring v, respectively. Because (Θ, A) is acyclic, $A(u, v) = 1$ if and only if

$$\mathrm{pre}(u) < \mathrm{pre}(v) < \mathrm{post}(v) < \mathrm{post}(u) \tag{4.9}$$

(see [21] for details). Define the map $\rho\colon v \mapsto \mathrm{pre}(v)$, for all $v \in \Theta$. Since each vertex has a unique pre-visit number in Θ, the map ρ is a permutation. Let $\sigma \coloneqq \rho^{-1}$ and define $\hat{\Pi}$ as in equation (4.8). Note that for any $i, j \in \Theta$, such that $i > j$,

$$A\big(\sigma(i), \sigma(j)\big) = 0 \implies \Pi\big(\sigma(i), \sigma(j)\big) = \hat{\Pi}(i, j) = 0.$$

Therefore, the permutation σ makes $\hat{\Pi}$ upper-triangular. □

Lemma 4.23. *Let $\Theta = \{0, 1, \dots, m\}$. Given a sequence of matrices $\{\Pi_k\}$ in $\mathbb{R}^{(m+1)\times(m+1)}$, define*

the matrix A as in Theorem 4.12. Then, there exists a permutation $\sigma\colon \Theta \to \Theta$, such that the matrices

$$\hat{\Pi}_k(i,j) \coloneqq \Pi_k\big(\sigma(i),\sigma(j)\big), \qquad i,j \in \Theta,\ k \geq 0, \tag{4.10}$$

are upper-triangular if the directed graph with vertices Θ and adjacency matrix A is acyclic.

Proof. Assume the graph (Θ, A) is acyclic, and run DFS to get the pre-visit number for each vertex. Define the map $\rho\colon v \to \mathrm{pre}(v)$, for all $v \in \Theta$, and define $\sigma \coloneqq \rho^{-1}$. Using σ define the matrices $\{\hat{\Pi}_k\}$, as in equation (4.10). If $i, j \in \Theta$, such that $i > j$, then $A\big(\sigma(i),\sigma(j)\big) = 0$, which implies that

$$\Pi_k\big(\sigma(i),\sigma(j)\big) = \hat{\Pi}_k(i,j) = 0,$$

for all k. Therefore, the permutation σ makes all the matrices $\{\hat{\Pi}_k\}$ upper-triangular. □

Proof of the Main Results

The preceding lemmas provide all the machinery needed to prove Theorems 4.11 and 4.12.

Proof of Theorem 4.11. Suppose that the graph (Θ, A) is acyclic. By Lemma 4.22, there exists a permutation σ, such that the matrix $\hat{\Pi}$, defined in equation (4.8), is upper-triangular. Define $\hat{\pi}_0 = \pi_0 \circ \sigma$. By Lemma 4.20, the Markov chain $\hat{\theta} \sim (\Theta, \hat{\Pi}, \hat{\pi}_0)$ is tractable. Therefore, by Lemma 4.21, the Markov chain θ is also tractable.

Lemma 4.22 states that if the graph (Θ, A) contains a cycle, then there is no permutation σ that makes $\hat{\Pi}$, defined in equation (4.8), upper-triangular. Hence, the proof is complete if we can show that the non-existence of such a permutation implies that θ is not tractable. Suppose that no such permutation exists, and suppose that the graph (Θ, A) has the cycle

$$\vartheta_0 \to \vartheta_1 \to \cdots \to \vartheta_{j-1} \to \vartheta_j = \vartheta_0,$$

for some $\vartheta_0 \in \Theta$ and $j > 0$. Because A is only nonzero where Π is nonzero (see equation (4.5)) and θ is non-degenerate, the cycle $\vartheta_{0:j}$ is a possible path of θ. Hence, θ has a set of possible paths that repeatedly visit ϑ_0 by traversing this cycle. Since a longer cycle would only increase the number of possible paths, it suffices to consider the simplest case where $j = 1$. This case is equivalent to the two-state Markov chain considered in Example 4.2, which was shown to be intractable. □

Proof of Theorem 4.12. Lemma 4.23 states that if the graph (Θ, A) is acyclic, then there exists a permutation σ, such that the matrices $\{\Pi_k\}$, defined in equation (4.10), are all upper-triangular. Define $\hat{\pi}_0 = \pi_i \circ \sigma$. By Lemma 4.20, the Markov chain $\hat{\theta} \sim \big(\Theta, \{\hat{\Pi}_k\}, \hat{\pi}_0\big)$ is tractable. Therefore, by Lemma 4.21, the Markov chain θ is also tractable. □

4.2.2 Special Case: Fault Model Based on Component Failures

Consider a system with L components (e.g., sensors and actuators), and suppose that each component may fail independently of the others. The term fail is used to indicate that the component stops working altogether and *never* resumes normal function. The status of each component (failed or not) at each time k is encoded by a binary variable b, where $b = 0$ indicates that the component has not failed at or before time k, while $b = 1$ indicates otherwise. Thus, the status of all L components at each time k is encoded by a L-bit binary string $b_k \in \{0,1\}^L$. One possible parameter space for this model is the set of 2^L nonnegative integers whose binary representations require no more than L bits. That is,

$$\Theta = \{0, 1, \ldots, 2^L - 1\}.$$

Converting each element of Θ into its binary representation reveals which component failures are encoded by that state.

Proposition 4.24. *Let θ be the stochastic process taking values in Θ, such that θ_k represents which components have failed at or before time k. Then, θ is a Markov chain.*

Proof. Let $k > 0$ and $\vartheta_{0:k} \in \Theta^{k+1}$. Consider the conditional probability

$$\mathbf{P}(\theta_k = \vartheta_k \mid \theta_{0:k-1} = \vartheta_{0:k-1}). \tag{4.11}$$

Let $i_1, i_2, \ldots, i_\ell$ be the indices of the components whose failure is encoded by the state ϑ_{k-1}. Also, let $i_{\ell+1}, i_{\ell+2}, \ldots, i_{\ell+j}$ be the components whose failure is encoded by ϑ_k but not ϑ_{k-1}. Since a failed component must remain in a failed state, the probability (4.11) is determined by the probability that components $i_{\ell+1}, \ldots, i_{\ell+j}$ fail at time k, given $\{\theta_{0:k-1} = \vartheta_{0:k-1}\}$. Although the event $\{\theta_{0:k-1} = \vartheta_{0:k-1}\}$ indicates at what times components $i_1, \ldots, i_\ell$ failed, this information is irrelevant, since the failure times are independent. The only meaningful information contained in the event $\{\theta_{0:k-1} = \vartheta_{0:k-1}\}$ is the fact that components $i_{\ell+1}, \ldots, i_{\ell+j}$ fail at time k, which is also indicated by the event $\{\theta_{k-1} = \vartheta_{k-1}\}$. Therefore,

$$\mathbf{P}(\theta_k = \vartheta_k \mid \theta_{0:k-1} = \vartheta_{0:k-1}) = \mathbf{P}(\theta_k = \vartheta_k \mid \theta_{k-1} = \vartheta_{k-1}),$$

which implies that θ is a Markov chain. □

Proposition 4.25. *The transition probability matrices $\{\Pi_k\}$ for the Markov chain θ are upper-triangular.*

Proof. Suppose that θ transitions from $i \in \Theta$ to $j \in \Theta$ at time k. Let b_i and b_j be the binary representations of i and j, respectively. The transition from i to j has zero probability unless

every 1-bit of b_i is a 1-bit of b_j (i.e., components failures are irreversible). Since $i \neq j$, there must be at least one bit, say the sth bit from the right, such that $b_i(s) = 0$ but $b_j(s) = 1$. Hence,

$$j \geq i + 2^{s-1} > i.$$

In other words, $\Pi_k(i, j)$ is only nonzero where $j \geq i$. □

Corollary 4.26. *The stochastic process θ which encodes the independent irreversible failures of L components is a tractable Markov chain.*

Proof. Propositions 4.24 and 4.25 imply that θ is a Markov chain with upper-triangular transition probability matrices. Hence, by Lemma 4.20, θ is a tractable Markov chain. □

Example 4.27. Consider a system with $L = 2$ components. The corresponding state space is

$$\Theta = \{0, 1, 2, 3\}.$$

If, for example, $\theta_k = 2 = (10)_2$, then component 1 has failed by time k but component 2 has not. Assume that the components fail at random times $\kappa_1 \sim \text{Geo}(q_1)$ and $\kappa_2 \sim \text{Geo}(q_2)$, respectively, where κ_1 and κ_2 are independent. Then, the transition probability matrix for $\{\theta_k\}$ is

$$\Pi = \begin{bmatrix} (1-q_1)(1-q_2) & (1-q_1)q_2 & q_1(1-q_2) & q_1 q_2 \\ 0 & 1-q_1 & 0 & q_1 \\ 0 & 0 & 1-q_2 & q_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Note that Π is upper-triangular, so by Lemma 4.20, the Markov chain is tractable. ◇

4.3 System Dynamics

Recall that the third computational issue presented in Section 4.1 is computing the probability

$$\mathbf{P}(D_{j,k} \mid \theta_{0:k} = \vartheta_{0:k}) = \mathbf{P}(d_k = j \mid \theta_{0:k} = \vartheta_{0:k})$$

for each $j \in \Theta$ and $\vartheta_{0:k} \in \Theta^{k+1}$. The first step toward ensuring that this computation is tractable is to require that the conditional density $p_{r|\theta}(r_k \mid \theta_{0:k})$ is Gaussian with known mean and variance. Conditional on the event $\{\theta_{0:k} = \vartheta_{0:k}\}$, the only source of randomness in the fault detection problem is the noise sequence $\{v_k\}$. Hence, we assume that $\{v_k\}$ is a Gaussian random process. Without loss of generality, we may also assume that $\{v_k\}$ is IID with $v_i \sim \mathcal{N}(0, I)$, for all i [50]. Although it is well-known that linear dynamical systems driven by Gaussian noise have Gaussian outputs [50], we consider the following more general class of systems with conditionally linear dynamics.

Definition 4.28. Let x_0 be a random variable, and let $\{v_k\}$ be a stochastic process. The system G_θ is said to be *conditionally linear* if, conditional on the event $\{\theta_{0:k} = \vartheta_{0:k}\}$, the system output y_k is an affine function of the random variables $x_0, v_0, v_1, \ldots, v_k$, as well as the deterministic input $u_{0:k}$. If x_0 is Gaussian and $\{v_k\}$ is a Gaussian process, then the output $\{y_k\}$ is a Gaussian process and the system G_θ is said to be *conditionally linear-Gaussian* (CLG).

Our approach to ensuring that $p_{r|\theta}(r_k \mid \theta_{0:k})$ is a Gaussian density is to impose certain assumptions on the structure of G_θ and F. The class of CLG systems plays a central role in these assumptions.

4.3.1 Assumptions Regarding the System Dynamics

In this section, we make assumptions about the structure of the system G_θ and the residual generator F. After writing the combined dynamics of the interconnection of these systems, we show that the conditional density $p_{r|\theta}(r_k \mid \theta_{0:k})$ is Gaussian, such that the mean and variance are easily computed by simulating a set of linear recurrences.

Assumed Structure of the System

Let $x_0 \sim \mathcal{N}(\hat{x}_0, \Lambda_{x,0})$, assume that $\{v_k\}$ is Gaussian IID with $v_i \sim \mathcal{N}(0, I)$, and assume that G_θ is given by

$$G_\theta \begin{cases} x_{k+1} = \hat{A}_k(\theta_k)x_k + \hat{B}_{u,k}(\theta_k)u_k + \hat{B}_{v,k}(\theta_k)v_k + \hat{B}_f f_k(\theta_{0:k}), \\ y_k = \hat{C}_k(\theta_k)x_k + \hat{D}_{u,k}(\theta_k)u_k + \hat{D}_{v,k}(\theta_k)v_k + \hat{D}_f f_k(\theta_{0:k}), \end{cases} \tag{4.12}$$

where the sequence of functions

$$\left\{ f_k \colon \Theta^{k+1} \to \mathbb{R}^{n_f} \right\}_{k \geq 0}$$

represents an additive fault signal. Assume that $f_k(0, 0, \ldots, 0) = 0$, for all k, so that $\{f_k\}$ does not affect the system when θ_k remains at the nominal value 0. Conditional on the event $\{\theta_{0:k} = \vartheta_{0:k}\}$, the sequence $\{f_k(\vartheta_{0:k})\}$ may be viewed as another deterministic input driving a linear-Gaussian system. Hence, the system G_θ given by equation (4.12) is CLG.

Remark 4.29. Since $\{\theta_k\}$ is assumed to be a finite-state Markov chain, the CLG model G_θ described by equation (4.12) closely resembles a jump-Markov linear system [20] (also called a state-space regime switching model in finance [56]). However, the inclusion of the additive fault signal $\{f_k\}$ is a departure from the traditional jump-Markov linear framework. We include this additional term, because it facilitates the modeling of sensor and actuator faults and preserves the CLG structure of the system. ◇

Assumed Structure of the Residual Generator

Given the Gaussian assumptions on x_0 and $\{v_k\}$ and CLG structure of the model G_θ, the conditional density $p_{y|\theta}(y_k \mid \theta_{0:k})$ is Gaussian, for all k. To ensure that $p_{r|\theta}(r_k \mid \theta_{0:k})$ is also Gaussian, assume that the residual generator F is a linear time-varying (LTV) system of the form

$$F \begin{cases} \xi_{k+1} = \tilde{A}_k \xi_k + \tilde{B}_{u,k} u_k + \tilde{B}_{y,k} y_k, \\ r_k = \tilde{C}_k \xi_k + \tilde{D}_{u,k} u_k + \tilde{D}_{y,k} y_k. \end{cases} \tag{4.13}$$

Note that this system is unaffected by changes in the parameter $\{\theta_k\}$, except through the measured output $\{y_k\}$.

Combined Dynamics

Assuming that G_θ is CLG and F is linear, the interconnection of the two systems is a single CLG system that takes $\{u_k\}$, $\{v_k\}$, and $\{f_k\}$ as its inputs and outputs the residual $\{r_k\}$. For each k, let $\eta_k \coloneqq (x_k, \xi_k)$ be the combined state of the system. The combined dynamics can be written as

$$\eta_{k+1} = A_k(\theta_k)\eta_k + B_{u,k}(\theta_k)u_k + B_{v,k}(\theta_k)v_k + B_f f_k(\theta_{0:k}), \tag{4.14}$$

$$r_k = C_k(\theta_k)\eta_k + D_{u,k}(\theta_k)u_k + D_{v,k}(\theta_k)v_k + D_f f_k(\theta_{0:k}), \tag{4.15}$$

where

$$A_k(\theta_k) \coloneqq \begin{bmatrix} \hat{A}_k(\theta_k) & 0 \\ \tilde{B}_{y,k}\hat{C}_k(\theta_k) & \tilde{A}_k \end{bmatrix},$$

$$B_{u,k}(\theta_k) \coloneqq \begin{bmatrix} \hat{B}_{u,k}(\theta_k) \\ \tilde{B}_{u,k} + \tilde{B}_{y,k}\hat{D}_{u,k}(\theta_k) \end{bmatrix}, \qquad B_{v,k}(\theta_k) \coloneqq \begin{bmatrix} \hat{B}_{v,k}(\theta_k) \\ \tilde{B}_{y,k}\hat{D}_{v,k}(\theta_k) \end{bmatrix}, \qquad B_f \coloneqq \begin{bmatrix} \hat{B}_f \\ \tilde{B}_{y,k}\hat{D}_f \end{bmatrix},$$

$$C_k(\theta_k) \coloneqq \begin{bmatrix} \tilde{D}_{y,k}\hat{C}_k(\theta_k) & \tilde{C}_k \end{bmatrix},$$

$$D_{u,k}(\theta_k) \coloneqq \tilde{D}_{u,k} + \tilde{D}_{y,k}\hat{D}_{u,k}(\theta_k), \qquad D_{v,k}(\theta_k) \coloneqq \tilde{D}_{y,k}\hat{D}_{v,k}(\theta_k), \qquad D_f \coloneqq \tilde{D}_{y,k}\hat{D}_f.$$

At this point, some remarks about the initial condition of F are in order. Intuitively, the expected value of the residual at time $k = 0$ should be zero. Hence, assuming that $\theta_0 = 0$

almost surely and $\hat{x}_0 = \mathbf{E}(x_0)$, the initial condition ξ_0 should solve the equation

$$\begin{aligned}\mathbf{E}(r_0) &= C_0(0)\begin{bmatrix}\hat{x}_0\\ \xi_0\end{bmatrix} + D_{u,0}(0)u_0\\ &= \tilde{D}_{y,0}\hat{C}_0(0)\hat{x}_0 + \tilde{C}_0\xi_0 + D_{u,0}u_0\\ &= 0.\end{aligned}$$

Since this equation may not always have a solution, a sensible choice is to take ξ_0 to be the minimum-norm solution [23] of the optimization problem

$$\min_{\xi} \|\mathbf{E}(r_0)\|^2 = \min_{\xi} \left\|\tilde{D}_{y,0}\hat{C}_0(0)\hat{x}_0 + \tilde{C}_0\xi + D_{u,0}u_0\right\|^2. \tag{4.16}$$

4.3.2 Computing the Conditional Mean and Variance

If the system G_θ and the residual generator F satisfy the assumptions stated above, it is straightforward to compute the conditional mean and variance of the residual $\{r_k\}$, given a particular parameter sequence. Fix a final time step $N \in \mathbb{N}$ and a parameter sequence $\vartheta_{0:N} \in \Theta^{N+1}$. For all k, define the conditional expected values

$$\hat{\eta}_k(\vartheta_{0:k}) \coloneqq \mathbf{E}(\eta_k \mid \theta_{0:k} = \vartheta_{0:k})$$

and

$$\hat{r}_k(\vartheta_{0:k}) \coloneqq \mathbf{E}(r_k \mid \theta_{0:k} = \vartheta_{0:k}).$$

The simpler notation $\hat{\eta}_k$ and $\hat{r}_k$ will be used when the sequence $\vartheta_{0:k}$ is clear from context. The sequences $\{\hat{\eta}_k\}$ and $\{\hat{r}_k\}$ are given by the linear recurrence

$$\hat{\eta}_{k+1} = A_k(\vartheta_k)\hat{\eta}_k + B_{u,k}(\vartheta_k)u_k + B_f f_k(\vartheta_{0:k}), \tag{4.17}$$

$$\hat{r}_k = C_k(\vartheta_k)\hat{\eta}_k + D_{u,k}(\vartheta_k)u_k + D_f f_k(\vartheta_{0:k}). \tag{4.18}$$

Similarly, define

$$\Lambda_k(\vartheta_{0:k}) \coloneqq \operatorname{var}(\eta_k \mid \theta_{0:k} = \vartheta_{0:k}),$$

and

$$\Sigma_k(\vartheta_{0:k}) \coloneqq \operatorname{var}(r_k \mid \theta_{0:k} = \vartheta_{0:k}).$$

Then, the sequences $\{\Lambda_k\}$ and $\{\Sigma_k\}$ are given by the linear recurrence

$$\Lambda_{k+1} = A_k(\vartheta_k)\Lambda_k A_k^T(\vartheta_k) + B_{v,k}(\vartheta_k)B_{v,k}^T(\vartheta_k), \tag{4.19}$$

$$\Sigma_k = C_k(\vartheta_k)\Lambda_k C_k^T(\vartheta_k) + D_{v,k}(\vartheta_k)D_{v,k}^T(\vartheta_k). \tag{4.20}$$

Therefore, conditional on the event $\{\theta_{0:k} = \vartheta_{0:k}\}$, the residual r_k has the Gaussian distribution $\mathcal{N}(\hat{r}_k, \Sigma_k)$, which is easily computed by simulating equations (4.17)–(4.20).

4.3.3 Special Case: Models with Only Additive Faults

As noted in Remark 4.29, the term $f_k(\theta_{0:k})$ in the CLG model (4.12) facilitates the modeling of additive sensor and actuator failures. In many treatments of fault detection, additive inputs are the only source of faults considered (see [9] and [24] for surveys of fault models used in the literature). As we will see in Section 4.5, this assumption can greatly reduce the amount of computational effort required to evaluate the performance metrics.

Suppose that $\{\theta_k\}$ is a Markov chain representing the independent irreversible failures of L components, as in Section 4.2.2. Hence, the parameter state space is

$$\Theta = \{0, 1, \ldots, 2^L - 1\}.$$

Recall that if the jth component is in a failed state at time k, then the jth bit (from the left) of the binary representation of θ_k is 1. Thus, the time at which component j fails can be determined by examining the realized values of $\{\theta_k\}$. For $j = 1, \ldots, L$, define the map

$$\kappa_j(\vartheta_{0:k}) := \begin{cases} i & \text{if component } j \text{ failed at time } i \le k, \\ \infty & \text{otherwise,} \end{cases}$$

for all $k \in \mathbb{N}$ and $\vartheta_{0:k} \in \Theta^{k+1}$. That is, if the value of $\vartheta_{0:k}$ indicates the failure of the jth component at or before time k, then $\kappa_j(\vartheta_{0:k})$ is the corresponding failure time. Otherwise, $\kappa_j(\vartheta_{0:k})$ just returns ∞.

For $j = 1, \ldots, L$, let the effect of the jth component failure be modeled by a function

$$\varphi_j \colon \{-\infty\} \cup \mathbb{Z} \to \mathbb{R}^{n_f},$$

such that $\varphi_j(z) = 0$, for all $z < 0$. That is, until component j fails, the function φ_j has no effect on the system. For each k, the combined fault signal is defined as

$$f_k(\vartheta_{0:k}) := \sum_{j=1}^{L} \varphi_j\big(k - \kappa_j(\vartheta_{0:k})\big),$$

for all $\vartheta_{0:k} \in \Theta^{k+1}$. In other words, each component failure causes an additive fault signal φ_j to "switch on" at some random time κ_j, which depends on the Markov chain $\{\theta_k\}$.

4.4 Decision Functions

The final step in evaluating the performance metrics is to compute the probabilities

$$\mathbf{P}(d_k = j \mid \theta_{0:k} = \vartheta_{0:k}) = \int_{E_{j,k}} p_{r|\theta}(r_k \mid \theta_{0:k} = \vartheta_{0:k})\, \mathrm{d}r_k, \tag{4.21}$$

where

$$E_{j,k} \coloneqq \{r_k : \delta(k, r_k) = j\}.$$

Assuming that the dynamics are conditionally linear-Gaussian, as in Section 4.3, the conditional distribution $p_{r|\theta}(r_k \mid \theta_{0:k} = \vartheta_{0:k})$ is the Gaussian $\mathcal{N}(\hat{r}_k, \Sigma_k)$. Although these assumptions generally make computation easier, the set $E_{j,k}$ must be simple enough to enable computation of the integral (4.21). In this section, we provide some practical examples of decision functions for which computation is tractable.

4.4.1 Threshold Decision Functions

First, consider the case where $\{r_k\}$ is scalar-valued. One common decision function, used frequently in fault detection [9, 32], is a time-varying threshold function of the form

$$\delta(k, r_k) \coloneqq \begin{cases} 0, & \text{if } |r_k| < \varepsilon_k, \\ 1, & \text{otherwise,} \end{cases}$$

where $\varepsilon_k > 0$, for all k. Hence, $E_{0,k} = [-\varepsilon_k, \varepsilon_k]$, and the integral (4.21) can be written in terms of the density of $\mathcal{N}(\hat{r}_k, \Sigma_k)$ as

$$\mathbf{P}(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k}) = \int_{-\varepsilon_k}^{\varepsilon_k} \frac{1}{\sqrt{2\pi\Sigma_k}} \exp\Big(-\frac{(r_k - \hat{r}_k)^2}{2\Sigma_k}\Big)\, \mathrm{d}r_k. \tag{4.22}$$

Since r_k is scalar, the error function, defined in Section 2.2.6, can be used to write the conditional cumulative distribution function of $r_k \sim \mathcal{N}(\hat{r}_k, \Sigma_k)$ as

$$\mathbf{P}(r_k < c \mid \theta_{0:k} = \vartheta_{0:k}) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{c - \hat{r}_k}{\sqrt{2\Sigma_k}}\right)\right],$$

for all $c \in \mathbb{R}$. Similarly, the integral (4.22) can be written as

$$\mathbf{P}(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k}) = \frac{1}{2}\left[\operatorname{erf}\left(\frac{\varepsilon_k - \hat{r}_k}{\sqrt{2\Sigma_k}}\right) - \operatorname{erf}\left(\frac{-\varepsilon_k - \hat{r}_k}{\sqrt{2\Sigma_k}}\right)\right].$$

Since the error function can be approximated by a rational function with a maximum relative error less than 6×10^{-19} [17], this expression can be evaluated accurately in $O(1)$ time.

In the non-scalar case (i.e., $r_k \in \mathbb{R}^{n_r}$), we define a threshold decision function as follows:

$$\delta(k, r_k) := \begin{cases} 0, & \text{if } \left|(r_k)_i\right| < (\varepsilon_k)_i,\ i = 1, 2, \ldots, n_r \\ 1, & \text{otherwise,} \end{cases}$$

where $\varepsilon_k \in \mathbb{R}_+^{n_r}$ is a vector-valued threshold, for all k. In this case, we must integrate the conditional PDF over the hyper-rectangle

$$E_{0,k} = \left[-(\varepsilon_k)_1, (\varepsilon_k)_1\right] \times \left[-(\varepsilon_k)_2, (\varepsilon_k)_2\right] \times \ldots \times \left[-(\varepsilon_k)_{n_r}, (\varepsilon_k)_{n_r}\right].$$

If the residual is low-dimensional ($n_r < 4$), the integral

$$\mathbf{P}(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k}) = \int_{E_{0,k}} \frac{1}{\sqrt{(2\pi)^{n_r}|\Sigma_k|}} \exp\left(-\frac{1}{2}(r_k - \hat{r}_k)^T \Sigma_k^{-1} (r_k - \hat{r}_k)\right) \mathrm{d}r_k,$$

can be computed using adaptive quadrature methods [37, 38]. Although experimental evidence shows that these methods are typically accurate and fast [37], their running time has not been rigorously characterized. For higher-dimensional residuals ($n_r \geq 4$), there are a number of quasi-Monte Carlo integration methods available [38], which are significantly less accurate than the low-dimensional quadrature methods.

4.4.2 Dynamic Decision Functions

Next, we consider two examples of tractable decision functions that are dynamic. Consider a decision function of the form

$$z_k = g(z_{k-1}, r_k), \tag{4.23}$$

$$d_k = h(z_k), \tag{4.24}$$

where the functions g and h, as well as the initial condition z_{-1}, are known and deterministic. Notice that because z_k is defined in terms of z_{k-1} and r_k, it is possible for the residual r_k to have an immediate effect on the decision d_k. Although equations (4.23) and (4.24) can represent a large class of decision functions, the original goal of computing (4.21) efficiently must still be met. Our approach is to consider cases where $\{z_k\}$ is a Markov chain.

Proposition 4.30. *Suppose that the sequence $\{r_k\}$ is Gaussian and that the initial condition z_{-1} is known and deterministic. The sequence $\{z_k\}$ is a Markov process if and only if the residuals r_i and r_j are uncorrelated, for all $i, j \geq 0$.*

The proof of this well-known proposition can be found in [50, §3.9].

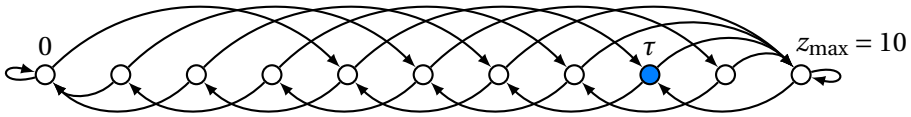


Figure 4.2. State-transition diagram of an up-down counter with parameters $(C_{\mathrm{D}}, C_{\mathrm{U}}, \tau, z_{\max}) = (2, 4, 8, 10)$. The threshold τ is shaded in blue.

Up-Down Counters

The up-down counter provides an intuitive means to improve the performance of an existing decision function δ taking values in $\mathscr{D} = \{0, 1\}$. Let $\{d_k\}_{k\geq 0}$ be the sequence of decisions produced by δ, and assume that, for all $k \geq 0$ and $\vartheta_{0:k} \in \Theta^{k+1}$, the probability

$$\mathbf{P}(d_k = 0 \mid \theta_{0:k} = \vartheta_{0:k})$$

is efficiently computable. The up-down counter produces another sequence of decisions $\{\hat{d}_k\}_{k\geq 0}$, defined by the recurrence

$$z_k = \begin{cases} \min\{z_{\max},\, z_{k-1} + C_{\mathrm{U}}\}, & \text{if } d_k = 1, \\ \max\{0,\, z_k - C_{\mathrm{D}}\}, & \text{otherwise,} \end{cases}$$

$$\hat{d}_k = \begin{cases} 0, & \text{if } z_k < \tau, \\ 1, & \text{otherwise,} \end{cases}$$

where $z_{-1} = 0$ and the parameters C_{D}, C_{U}, τ, $z_{\max}$, and ε_k are scalars, such that

$$0 < C_{\mathrm{D}} \leq C_{\mathrm{U}} \leq \tau \leq z_{\max}.$$

For simplicity, assume that C_{D}, C_{U}, and $z_{\max}$ are all natural numbers, so the state space of the sequence $\{z_k\}$ is

$$\mathscr{Z} := \{0, 1, \ldots, z_{\max}\}.$$

The graph depicted in Figure 4.2 is the state-transition diagram of a simple up-down counter with parameters $(C_{\mathrm{D}}, C_{\mathrm{U}}, \tau, z_{\max}) = (2, 4, 8, 10)$. The arrows indicate which transitions are possible.

Since $z_{-1} = 0$ almost surely, the initial distribution of $\{z_k\}$ is

$$\lambda_{-1}(i) = \mathbb{1}(i = 0), \qquad i \in \mathscr{Z},$$

where $\mathbb{1}$ is the indicator function. Let $\vartheta_{0:k} \in \Theta^{k+1}$ and assume that, conditional on the event $\{\theta_{0:k} = \vartheta_{0:k}\}$, the sequence $\{r_k\}$ is uncorrelated and Gaussian. By Proposition 4.30, the sequence $\{z_k\}$ is conditionally a Markov chain, given $\{\theta_{0:k} = \vartheta_{0:k}\}$, and the transition

probability matrix is given by

$$\begin{aligned}\big(\Lambda_k(\vartheta_{0:k})\big)_{ij} &:= \mathbf{P}(z_k = j \mid z_{k-1} = i, \theta_{0:k} = \vartheta_{0:k}) \\ &= \begin{cases} \mathbf{P}(d_k = 0 \mid \theta_{0:k} = \vartheta_{0:k}), & \text{if } j = \max\{0, i - C_{\mathrm{D}}\}, \\ \mathbf{P}(d_k = 1 \mid \theta_{0:k} = \vartheta_{0:k}), & \text{if } j = \min\{z_{\max}, i + C_{\mathrm{U}}\}, \\ 0, & \text{otherwise,} \end{cases}\end{aligned}$$

for all $i, j \in \mathcal{Z}$. The conditional distribution of z_k, defined as

$$\big(\lambda_k(\vartheta_{0:k})\big)_i = \mathbf{P}(z_k = i \mid \theta_{0:k} = \vartheta_{0:k}), \qquad i \in \mathcal{Z},$$

is computed via the equation

$$\lambda_k^T(\vartheta_{0:k}) = \lambda_{-1}^T \, \Lambda_0(\vartheta_0) \, \Lambda_1(\vartheta_{0:1}) \cdots \Lambda_k(\vartheta_{0:k}).$$

The probability that the up-down counter exceeds the threshold τ is

$$\mathbf{P}(\hat{d}_k = 1 \mid \theta_{0:k} = \vartheta_{0:k}) = \sum_{i=\tau}^{z_{\max}} \mathbf{P}(z_k = i \mid \theta_{0:k} = \vartheta_{0:k}) = \sum_{i=\tau}^{z_{\max}} \big(\lambda_k(\vartheta_{0:k})\big)_i.$$

Suppose that, for some k_1, the underlying decision function δ decides that a fault has occurred in such a way that $d_\ell = 1$, for $\ell \geq k_1$. If $z_{k_1} = 0$, then the decision sequence $\{\hat{d}_k\}$ will remain at 0 until $\lceil \tau / C_{\mathrm{U}} \rceil$ time steps have passed. That is, the up-down counter has an inherent detection delay, specified by the ratio τ / C_{U}. Of course, this delay provides a degree of robustness when the underlying decision function is prone to false alarms. When a false alarm does occur, $\lceil C_{\mathrm{U}} / C_{\mathrm{D}} \rceil$ time steps with no further false alarms must pass before the counter state $\{z_k\}$ falls below its original value. Hence, the ratio $C_{\mathrm{U}} / C_{\mathrm{D}}$ specifies how long it takes for a spurious up-count to be "forgotten."

Similarly, suppose that for some k_2, the effect of a fault subsides and $d_\ell = 0$, for all $\ell \geq k_2$. If z_{k_2} happens to be at $z_{\max}$, then the decision sequence $\{\hat{d}_k\}$ will not return to 0 until $\lceil (z_{\max} - \tau) / C_{\mathrm{D}} \rceil$ time steps have elapsed. As in the previous scenario, the up-down counter has an inherent delay, specified by the ratio $(z_{\max} - \tau) / C_{\mathrm{D}}$. This particular delay provides a degree of robustness against missed detections.

Although the up-down counter seems to have inherent delays in these idealized scenarios, the robustness provided by the up-down counter can actually lead to a more responsive fault detection scheme. Figures 4.3(a) and 4.3(b) show the realizations of the counter state $\{z_k\}$ and the residual $\{r_k\}$, respectively, for a typical up-down counter based on a ε-threshold decision function. In this particular simulation, a fault occurs at time k_1 and subsides at time k_2. The delay in the up-down counter can clearly be seen in Figure 4.3(a). However, the original decision function has a large number of false alarms. If the threshold ε is increased

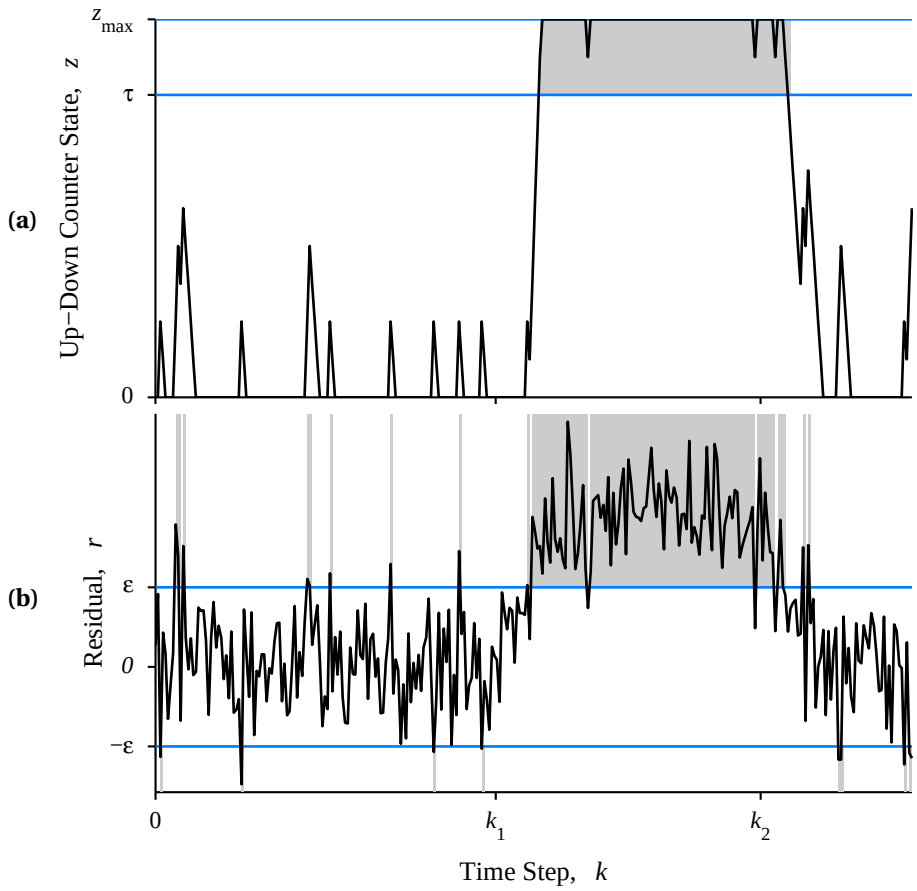


Figure 4.3. Comparison of the behavior of an up-down counter **(a)** and the behavior of the underlying threshold decision function **(b)**. The horizontal blue lines indicated the threshold regions, and the vertical shaded bands indicate the ranges of time where the respective decision function signals that a fault has occurred. The actual fault starts at time k_1 and stops at time k_2.

to the point where the number of false alarms is reasonable, the delay of the original threshold decision function would be even greater. Therefore, in this case, the up-down counter actually responds more quickly.

Note that for $\alpha > 0$, the parameters $(C_D\alpha, C_U\alpha, \tau\alpha, z_{\max}\alpha)$ define an equivalent up-down counter with state space

$$\mathcal{Z}_\alpha := \{0, \alpha, 2\alpha, \ldots, z_{\max}\alpha\}.$$

In the special case where

$$C_D = C_U = \tau = z_{\max},$$

the decisions produced by the up-down counter are identical to those produced by the original decision function (i.e., $\hat{d}_k = d_k$, for all k).

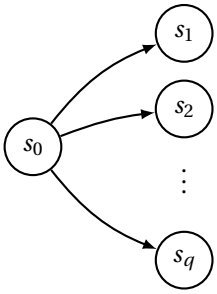


Figure 4.4. State-transition diagram for a system that reconfigures when a fault occurs. The state s_0 represents the nominal configuration, while state s_i, $i \neq 0$, represents the configuration that is used when $d_k = i$. Since the fault diagnosis problem essentially restarts when a reconfiguration occurs, only one level of reconfiguration is shown.

Systems that Reconfigure when a Fault is Detected

Thus far, we have considered fault diagnosis problems in which the decision sequence $\{d_k\}$ may be nonzero at one instant and then return to zero at the next. Sometimes, however, it is useful to consider the case where some action is taken once $\{d_k\}$ is no longer zero. In particular, we consider the case where the system is reconfigured when $d_k \neq 0$. For example, if $d_k = i$ indicates that component i has failed at or before time k, then the system G_θ should be reconfigured to no longer use that component. Similarly, the fault diagnosis scheme $V = (F, \delta)$ must also be reconfigured. Once the system G_θ and scheme V have been reconfigured, a new fault diagnosis problem begins. In this section, we demonstrate that such reconfigurations can be modeled by a dynamic decision function, so that the property of being in a given configuration can be computed efficiently using our performance analysis framework.

Suppose that $V = (F, \delta)$ is a fault diagnosis scheme designed for the plant G_θ in its nominal configuration, such that $d_k = \delta(k, r_k)$ takes values in the set $\mathscr{D} = \{0, 1, \ldots, q\}$. Let s_0 denote the original configuration of G_θ and V. Similarly, for $i = 1, \ldots, q$, let s_i denote the reconfiguration of the system and scheme that takes place when $d_k = i$. Assume that, after reconfiguration, there is no returning to the original configuration s_0. Hence, the set of possible reconfigurations is governed by the state-transition diagram shown in Figure 4.4.

Let the sequence $\{z_k\}$ represent the configuration at each time step, and let $\{\hat{d}_k\}$ be a new sequence of decisions that is given by the recurrence

$$z_k = \begin{cases} \delta(k, r_k) & \text{if } z_{k-1} = 0, \\ z_{k-1} & \text{otherwise,} \end{cases}$$
$$\hat{d}_k = z_k,$$

where $z_{-1} = 0$. This recurrence defines a dynamic decision function that decides which configuration is in use at each point in time. Note that the state space of $\{z_k\}$ is $\mathscr{Z} = \{0, 1, \ldots, q\}$.

If we assume that the system G_θ and the residual generator F meet the assumptions of Section 4.3, then given a particular mode sequence $\{\vartheta_k\}$, the conditional distribution of the residual $\{r_k\}$ is Gaussian, at each k. Hence, $\{z_k\}$ is a stochastic process, and by Proposition 4.30, $\{z_k\}$ is a Markov chain if and only if the sequence $\{r_k\}$ is uncorrelated in time. Otherwise, if $\{r_k\}$ is correlated, then

$$\mathbf{P}(z_k = 0 \mid \theta_{0:k} = \vartheta_{0:k}) = \mathbf{P}\big(\delta(k, r_k) = 0, \delta(k-1, r_{k-1}) = 0, \ldots, \delta(0, r_0) = 0 \mid \theta_{0:k} = \vartheta_{0:k}\big).$$

for all k. Clearly, as k becomes large, the joint probability on the right hand side becomes intractable to compute numerically.

Assume that the sequence $\{r_k\}$ is Gaussian and uncorrelated. Since $\{z_k\}$ is a Markov chain conditional on the event $\{\theta = \vartheta\}$, the probability distribution of $\{z_k\}$ is given by the initial distribution and transition probability matrices. Since $z_{-1} = 0$ almost surely, the initial distribution is

$$\lambda_{-1}(i) = \mathbb{1}(i = 0), \qquad i \in \mathcal{Z},$$

where $\mathbb{1}$ is the indicator function. Given $\{\theta_{0:k} = \vartheta_{0:k}\}$, the transition probability matrix at time k is

$$\begin{aligned}\big(\Lambda_k(\vartheta_{0:k})\big)_{ij} &\coloneqq \mathbf{P}\big(z_k = j \mid z_{k-1} = i, \theta_{0:k} = \vartheta_{0:k}\big) \\ &= \begin{cases} \mathbf{P}\big(\delta(k, r_k) = j \mid \theta_{0:k} = \vartheta_{0:k}\big) & \text{if } i = 0, \\ 1 & \text{if } i = j,\ 1 \le i \le q, \\ 0 & \text{otherwise,} \end{cases}\end{aligned}$$

for all $i, j \in \mathcal{Z}$. The conditional distribution of z_k, defined as

$$\big(\lambda_k(\vartheta_{0:k})\big)_i \coloneqq \mathbf{P}(z_k = i \mid \theta_{0:k} = \vartheta_{0:k}), \qquad i \in \mathcal{Z},$$

is computed via the equation

$$\lambda_k^T(\vartheta_{0:k}) = \lambda_{-1}^T \, \Lambda_0(\vartheta_0) \, \Lambda_1(\vartheta_{0:1}) \cdots \Lambda_k(\vartheta_{0:k}).$$

Therefore, the main challenge in computing $\lambda_k(\vartheta_{0:k})$ for a given $\vartheta_{0:k} \in \Theta^{k+1}$ is computing the probability

$$\mathbf{P}\big(\delta(k, r_k) = j \mid \theta_{0:k} = \vartheta_{0:k}\big),$$

for all $i \in \mathcal{Z}$. Section 4.4.1 demonstrates how this probability is computed for threshold decision functions.

If we define the event $\hat{D}_{i,k} = \{\hat{d}_k = i\}$, for each $i \in \mathcal{Z}$ and each $k \ge 0$, then the performance

metrics at time k are defined as

$$\hat{J}_k(i,j) \coloneqq \mathbf{P}(\hat{D}_{j,k} \cap H_{i,k}), \qquad i,j \in \mathscr{D}.$$

For each k, the value $\hat{J}_k(i,j)$ is the probability that the system is in configuration s_i when it should be in configuration s_j. Note that the event $\hat{D}_{j,k} \cap H_{i,k}$ may or may not represent a safe state of affairs, depending on the values of i and j. For example, when the jth fault occurs (i.e., $\{\theta_k\}$ enters the set Θ_j), the system is designed to reconfigure to a back-up mode s_j. Hence, it would be unsafe to continue operation in the nominal configuration s_0 when the jth fault occurs. In any case, the probability that the system is in a safe configuration at time k can be computed by summing the appropriate entries of $\hat{J}_k$.

4.5 Algorithms for Computing Performance

In this section, we present high-level algorithms for computing the performance metrics. First, we consider systems that satisfy the restrictions discussed in Sections 4.2–4.4. Then, we consider a special case, based on Sections 4.2.2 and 4.3.3, that consists of an LTV system with L independent additive faults. Finally, this special case is further simplified by assuming that the dynamics are LTI. For each system class, the time-complexity of computing the performance metrics is analyzed.

4.5.1 Sufficiently Structured Systems

Suppose that the fault parameter sequence θ is a tractable Markov chain satisfying the conditions of Theorem 4.11 or 4.12. Also, assume that the combined CLG dynamics of G_θ and F can be written in the form of equation (4.12), and assume that the decision function δ is such that the probability

$$\mathbf{P}(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k})$$

can be computed in $O(1)$ time. The most common class of decision functions meeting this last criterion is the class of threshold functions.

If all these assumptions hold, then the joint probability performance metrics $\{P_{\text{TN},k}\}$, $\{P_{\text{FP},k}\}$, $\{P_{\text{FN},k}\}$, and $\{P_{\text{TP},k}\}$ are computed using Algorithm 4.1. This algorithm consists of two nested *for*-loops. The outer loop (Lines 1–21) considers all possible mode sequences, while the inner loop (Lines 2–20) updates the performance metrics at each time step. The inner loop can be divided into three parts, as follows:

- Lines 3–7 compute the probability of the fault parameter sequence $\vartheta_{0:N}$.
- Lines 8–11 update the recurrences for the mean $\hat{r}_k$ and variance Σ_k of the residual, conditional on the event $\{\theta_{0:k} = \vartheta_{0:k}\}$.

Algorithm 4.1. General procedure for computing the performance metrics, where the decision function δ is a time-varying threshold.

Require: A final time $N \in \mathbb{N}$, a Gaussian initial state $\eta_0 \sim \mathcal{N}(\hat{\eta}_0, \Lambda_0)$, a sequence of thresholds $\{\varepsilon_k\}$ such that $\varepsilon_i > 0$, and a fault model $\theta \sim \big(\Theta, \{\Pi_k\}, \pi_0\big)$.

---

1 **for all** $\vartheta_{0:N} \in \Theta^{N+1}$ with nonzero probability **do**
2 **for** $k = 0, 1, \ldots, N$ **do**
3 **if** $k = 0$ **then**
4 $\mathbf{P}(\theta_0 = \vartheta_0) = \pi_0(\vartheta_0)$
5 **else**
6 $\mathbf{P}(\theta_{0:k} = \vartheta_{0:k}) = \Pi_{k-1}(\vartheta_{k-1}, \vartheta_k)\, \mathbf{P}(\theta_{0:k-1} = \vartheta_{0:k-1})$
7 **end if**
8 $\hat{\eta}_{k+1} = A_k(\vartheta_k)\hat{\eta}_k + B_{u,k}(\vartheta_k)u_k + B_f f_k(\vartheta_{0:k})$
9 $\hat{r}_k = C_k(\vartheta_k)\hat{\eta}_k + D_{u,k}(\vartheta_k)u_k + D_f f_k(\vartheta_{0:k})$
10 $\Lambda_{k+1} = A_k(\vartheta_k)\Lambda_k A_k^T(\vartheta_k) + B_{v,k}(\vartheta_k)B_{v,k}^T(\vartheta_k)$
11 $\Sigma_k = C_k(\vartheta_k)\Lambda_k C_k^T(\vartheta_k) + D_{v,k}(\vartheta_k)D_{v,k}^T(\vartheta_k)$
12 Compute $\mathbf{P}\big(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k}\big)$
13 **if** $\vartheta_k \in \Theta_0$ **then**
14 $P_{\text{TN},k} = P_{\text{TN},k} + \mathbf{P}\big(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k}\big)\, \mathbf{P}\big(\theta_{0:k} = \vartheta_{0:k}\big)$
15 $P_{\text{FP},k} = P_{\text{FP},k} + \Big(1 - \mathbf{P}\big(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k}\big)\Big)\, \mathbf{P}\big(\theta_{0:k} = \vartheta_{0:k}\big)$
16 **else**
17 $P_{\text{FN},k} = P_{\text{FN},k} + \mathbf{P}\big(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k}\big)\, \mathbf{P}\big(\theta_{0:k} = \vartheta_{0:k}\big)$
18 $P_{\text{TP},k} = P_{\text{TP},k} + \Big(1 - \mathbf{P}\big(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k}\big)\Big)\, \mathbf{P}\big(\theta_{0:k} = \vartheta_{0:k}\big)$
19 **end if**
20 **end for**
21 **end for**

- Line 12 computes the conditional probability $\mathbf{P}(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k})$, and then Lines 13–19 use this probability to update the performance metrics. Note that Line 18 is technically superfluous, because the performance metrics must sum to one.

Remark 4.31. While most of the computation is straightforward, Line 1 is the most difficult portion of this algorithm, as it requires all possible parameter sequences to be generated. One option is to generate and store all the sequences in an array. However, this size of such an array would be prohibitively large. Another option is to dynamically generate the sequences while bookkeeping which sequences have already been considered. This is the approach taken with the special cases in Sections 4.5.2 and 4.5.3. However, we have not yet discovered a practical implementation for this portion of the algorithm. ◇

Proposition 4.32. *Let N be the final time used in Algorithm 4.1, and let $\Theta = \{0, 1, \ldots, m\}$. In additions to the assumptions on $\{\theta_k\}$, G_θ, F, and δ made above, assume that the fault input $f_k(\vartheta_{0:k})$ can be computed in $O(1)$ time, for any k and $\vartheta_{0:k}$. Then, the total running time of Algorithm 4.1 is $O(N^{m+1})$.*

Proof. Because θ is assumed to be a tractable Markov chain, the *for all*-loop over possible sequences $\vartheta_{0:N}$ executes $O(N^m)$ times. Line 4 is a simple look-up and Line 6 is a single multiplication, so Lines 3–7 take $O(1)$ time to compute. Since $f_k(\vartheta_{0:k})$ can be computed in $O(1)$ time, Lines 8–11 can be computed in $O(1)$ time, as well. By assumption, the decision function δ is such that Line 12 can be computed in $O(1)$ time. Clearly, the remaining computations (Line 13–19) can also be computed in $O(1)$ time. Since each individual line takes $O(1)$ time, we conclude that each iteration of the *for*-loop over k takes $O(1)$ time. Therefore, the total running time of Algorithm 4.1 is $O(N^{m+1})$. □

4.5.2 LTV Special Case Based on Component Failures

In this section we present a special system structure, based on Sections 4.2.2 and 4.3.3, that permits a more straightforward implementation of Algorithm 4.1. Suppose that the system consists of L components that fail independently at random, and assume that system is only affected by additive faults. Hence, the combined dynamics of the system G_θ and the residual generator F are given by

$$\eta_{k+1} = A_k \eta_k + B_{u,k} u_k + B_{v,k} v_k + B_f \sum_{j=1}^{L} \varphi_j\big(k - \kappa_j(\theta_{0:k})\big),$$

$$r_k = C_k \eta_k + D_{u,k} u_k + D_{v,k} v_k + D_f \sum_{j=1}^{L} \varphi_j\big(k - \kappa_j(\theta_{0:k})\big),$$

where $\kappa_j(\theta_{0:k})$ is the random time at which the jth component fails. Because $\theta_{0:k}$ only affects the system via the random failure times, specifying a particular parameter sequence $\vartheta_{0:N}$ is equivalent to specifying the corresponding failure times $\hat{\kappa}_j := \kappa_j(\vartheta_{0:N})$, for $j = 1, 2, \ldots, L$.

Another important feature of this special case is the additive structure of the fault input. Since each φ_j enters additively, the portion of the residual due to each φ_j can be computed separately and then combined using the principle of superposition. Similarly, the portion of the residual due to the initial condition η_0 and the known input $u_{0:N}$ can be computed separately. Because φ_j has no effect until the jth component fails (i.e., $\varphi_j(k - \hat{\kappa}_j) = 0$, for $k < \hat{\kappa}_j$), we only need to compute the portion of the residual due to φ_j for $k \geq \hat{\kappa}_j$.

The procedure for computing the performance metrics for this special case is split into two parts: Algorithm 4.2 computes each portion of the residual, while Algorithm 4.3 computes the performance metrics. Although Algorithm 4.2 applies to any system of L components, Algorithm 4.3 focuses on the case $L = 2$. This greatly simplifies the presentation

of the algorithm, and it is a straightforward matter to write a version of Algorithm 4.3 for any finite number of components. Algorithm 4.2 consists of two parts:

- Lines 1–7 simulate the portion of the conditional mean of the residual due to the initial condition η_0 and the known input $u_{0:N}$. Lines 1–7 also simulate the conditional variance of the residual, which does not depend on the fault input $\sum_j \varphi_j(k-\kappa_j)$.
- Lines 8–16 simulate the portion of the conditional mean of the residual due to each component failing at each possible time.

Algorithm 4.3, on the other hand, consists of four parts:

- Lines 2–4 compute the performance metrics $P_{\text{TN},k}$ and $P_{\text{FP},k}$.
- Lines 5–10 update the performance metrics $P_{\text{FN},k}$ and $P_{\text{TP},k}$ by considering all possible cases where component 1 fails but component 2 does not.
- Lines 11–16 update the performance metrics $P_{\text{FN},k}$ and $P_{\text{TP},k}$ by considering all possible cases where component 2 fails but component 1 does not.
- Lines 17–24 update the performance metrics $P_{\text{FN},k}$ and $P_{\text{TP},k}$ by considering all possible cases where both components fail.

Proposition 4.33. *Assume that the probability $\mathbf{P}(\kappa_j = k)$ can be computed in $O(1)$ time, for all j and k. Also, assume that the decision function δ is such that $\mathbf{P}(D_{0,k} \mid \theta_{0:k} = \vartheta_{0:k})$ can be computed in $O(1)$ time for any $\vartheta_{0:N} \in \Theta^{N+1}$ and all $k \geq 0$. Then, the running time of Algorithm 4.2 is $O(LN^2)$ and the running time of Algorithm 4.3 is $O(LN^L)$. Therefore, computing the performance metrics requires a total of $O\big(LN^{\max\{2,L\}}\big)$ time.*

Proof. First, we show that the running time of Algorithm 4.2 is $O(LN^2)$. Since updating the recurrences in Lines 3–6 takes $O(1)$ time, Lines 2–7 take $O(N+1)$ time to compute. Similarly, Lines 12–13 take $O(1)$ time to compute. The number of times that Lines 12–13 must be executed is

$$\begin{aligned}\sum_{j=1}^{L}\sum_{\hat{\kappa}_j=1}^{N}\sum_{k=\hat{\kappa}_j}^{N} 1 &= \sum_{j=1}^{L}\sum_{\hat{\kappa}_j=1}^{N} N-\hat{\kappa}_j+1 \\ &= \sum_{j=1}^{L}\frac{N(N+1)}{2} \\ &= O(LN^2).\end{aligned}$$

Therefore, Lines 8–16 take $O(LN^2)$ to compute, and the total running time of Algorithm 4.2 is $O(LN^2)$.

Algorithm 4.2. Procedure for computing the components of the mean and variance of the residual for the LTV special case.

Require: A final time $N \in \mathbb{N}$ and a Gaussian initial state $\eta_0 \sim \mathcal{N}(\hat{\eta}_0, \Lambda_0)$.

Let $\hat{\eta}_0^{(0,0)} = \hat{\eta}_0$
for $k = 0, 1, \ldots, N$ **do**
    $\hat{\eta}_{k+1}^{(0,0)} = A_k \hat{\eta}_k^{(0,0)} + B_{u,k} u_k$
    $\hat{r}_k^{(0,0)} = C_k \hat{\eta}_k^{(0,0)} + D_{u,k} u_k$
    $\Lambda_{k+1} = A_k \Lambda_k A_k^T + B_{v,k} B_{v,k}^T$
    $\Sigma_k = C_k \Lambda_k C_k^T + D_{v,k} D_{v,k}^T$
end for
for $j = 1, 2, \ldots, L$ **do**
    for $\hat{\kappa}_j = 1, 2, \ldots, N$ **do**
        Let $\hat{\eta}_0^{(j,\kappa_j)} = 0$
        for $k = \hat{\kappa}_j, \hat{\kappa}_j + 1, \ldots, N$ **do**
            $\hat{\eta}_{k+1}^{(j,\hat{\kappa}_j)} = A_k \hat{\eta}_k^{(j,\hat{\kappa}_j)} + B_f \varphi_j(k - \hat{\kappa}_j)$
            $\hat{r}_k^{(j,\hat{\kappa}_j)} = C_k \hat{\eta}_k^{(j,\hat{\kappa}_j)} + D_f \varphi_j(k - \hat{\kappa}_j)$
        end for
    end for
end for

Algorithm 4.3. Procedure for computing the performance metrics for the LTV special case with two components.

Require: A final time $N \in \mathbb{N}$, a sequence of thresholds $\{\varepsilon_k\}$ such that $\varepsilon_i > 0$, the conditional variance of the residual $\{\Sigma_k\}$, and the components of the conditional mean of the residual $\hat{r}_k^{(0,0)}$, $\hat{r}_k^{(1,s)}$, and $\hat{r}_k^{(2,s)}$, for $k = 0, 1, \ldots, N$ and $s = 1, 2, \ldots, N$.

---

for $k = 0, 1, \ldots, N$ **do**
  Compute $\mathbf{P}\big(D_{0,k} \mid \kappa_1 > k, \kappa_2 > k\big)$
  $P_{\text{TN},k} = \mathbf{P}\big(D_{0,k} \mid \kappa_1 > k, \kappa_2 > k\big)\, \mathbf{P}\big(\kappa_1 > k\big)\, \mathbf{P}\big(\kappa_2 > k\big)$
  $P_{\text{FP},k} = \Big(1 - \mathbf{P}\big(D_{0,k} \mid \kappa_1 > k, \kappa_2 > k\big)\Big)\, \mathbf{P}\big(\kappa_1 > k\big)\, \mathbf{P}\big(\kappa_2 > k\big)$
  for $s = 1, 2, \ldots, k$ **do**
    $\hat{r}_k = \hat{r}_k^{(0,0)} + \hat{r}_k^{(1,s)}$
    Compute $\mathbf{P}\big(D_{0,k} \mid \kappa_1 = s, \kappa_2 > k\big)$
    $P_{\text{FN},k} = P_{\text{FN},k} + \mathbf{P}\big(D_{0,k} \mid \kappa_1 = s, \kappa_2 > k\big)\, \mathbf{P}\big(\kappa_1 = s\big)\, \mathbf{P}\big(\kappa_2 > k\big)$
    $P_{\text{TP},k} = P_{\text{TP},k} + \Big(1 - \mathbf{P}\big(D_{0,k} \mid \kappa_1 = s, \kappa_2 > k\big)\Big)\, \mathbf{P}\big(\kappa_1 = k\big)\, \mathbf{P}\big(\kappa_2 > k\big)$
  end for
  for $t = 1, 2, \ldots, k$ **do**
    $\hat{r}_k = \hat{r}_k^{(0,0)} + \hat{r}_k^{(2,t)}$
    Compute $\mathbf{P}\big(D_{0,k} \mid \kappa_1 > k, \kappa_2 = t\big)$
    $P_{\text{FN},k} = P_{\text{FN},k} + \mathbf{P}\big(D_{0,k} \mid \kappa_1 > k, \kappa_2 = t\big)\, \mathbf{P}\big(\kappa_1 > k\big)\, \mathbf{P}\big(\kappa_2 = t\big)$
    $P_{\text{TP},k} = P_{\text{TP},k} + \Big(1 - \mathbf{P}\big(D_{0,k} \mid \kappa_1 > k, \kappa_2 = t\big)\Big)\, \mathbf{P}\big(\kappa_1 > k\big)\, \mathbf{P}\big(\kappa_2 = t\big)$
  end for
  for $s = 1, 2, \ldots, k$ **do**
    for $t = 1, 2, \ldots, k$ **do**
      $\hat{r}_k = \hat{r}_k^{(0,0)} + \hat{r}_k^{(1,s)} + \hat{r}_k^{(2,t)}$
      Compute $\mathbf{P}\big(D_{0,k} \mid \kappa_1 = s, \kappa_2 = t\big)$
      $P_{\text{FN},k} = P_{\text{FN},k} + \mathbf{P}\big(D_{0,k} \mid \kappa_1 = s, \kappa_2 = t\big)\, \mathbf{P}\big(\kappa_1 = s\big)\, \mathbf{P}\big(\kappa_2 = t\big)$
      $P_{\text{TP},k} = P_{\text{TP},k} + \Big(1 - \mathbf{P}\big(D_{0,k} \mid \kappa_1 = s, \kappa_2 = t\big)\Big)\, \mathbf{P}\big(\kappa_1 = s\big)\, \mathbf{P}\big(\kappa_2 = t\big)$
    end for
  end for
end for

Second, we show that the running time of the L-component version of Algorithm 4.3 is $O(LN^L)$. For $i = 0, 1, \ldots, L$, we must consider all cases in which i components fail at or before time N. There are $\binom{L}{i}$ ways to choose which i components fail, and each component can fail at any time $\kappa \in \{1, 2, \ldots, N\}$. By the binomial theorem [40], the total number of cases to consider is

$$\sum_{i=0}^{L} \binom{L}{i} N^i = (1+N)^L = O(N^L).$$

In Algorithm 4.3, Lines 2-4, 6–9, 12–15, and 19–22 are essentially identical. In general, these four lines must be executed for each possible case. By assumption, the probabilities of the form

$$\mathbf{P}(D_{0,k} \mid \kappa_j = s_j, j = 1, \ldots L),$$

as well as the component failure probabilities $\mathbf{P}(\kappa_j = s_j)$ and $\mathbf{P}(\kappa_j > k)$, can be evaluated in $O(1)$ time. Since we must compute L such component failure probabilities in each possible case, the running time of Algorithm 4.3 is $O(LN^L)$. Therefore, the total time required to compute the performance metrics is $O(LN^2) + O(LN^L) = O\big(LN^{\max\{2,L\}}\big)$. □

Remark 4.34. At first glance, the combined running time of Algorithms 4.2 and 4.3, seems little better than the polynomial running time of the general procedure given in Algorithm 4.1. However, as shown in Section 4.2.2, a system with L components leads to a Markov chain with state space $\Theta = \{0, 1, \ldots, 2^L - 1\}$. Therefore, the running time of Algorithm 4.1 would be $O\big(N^{2^L-1}\big)$, which is significantly worse than $O(LN^L)$ for practical values of L and N. ◇

4.5.3 LTI Special Case Based on Component Failures

The special case considered in the previous section can be simplified further by assuming that the dynamics are time-invariant. That is, we assume the combined dynamics are of the form

$$\eta_{k+1} = A\eta_k + B_u u_k + B_v v_k + B_f \sum_{j=1}^{L} \varphi_j\big(k - \kappa_j(\theta_{0:k})\big),$$

$$r_k = C\eta_k + D_u u_k + D_v v_k + D_f \sum_{j=1}^{L} \varphi_j\big(k - \kappa_j(\theta_{0:k})\big),$$

As in the LTV case, superposition is used to reduce the amount of computation required. However, because the system is now LTI, the portion of the conditional mean of the residual due to component j failing at time κ_j can be obtained by time-shifting the portion due to component j failing at time 1. For all $n \in \mathbb{N}$, let the n-shift operator z^n be defined by

$$z^n \colon x_{0:N} \mapsto \{\underbrace{0, \ldots, 0}_{n \text{ zeros}}, x_0, x_1, \ldots, x_{N-n}\},$$

Algorithm 4.4. Procedure for computing the components of the mean and variance of the residual for the LTI special case.

Require: A final time $N \in \mathbb{N}$ and a Gaussian initial state $\eta_0 \sim \mathcal{N}(\hat{\eta}_0, \Lambda_0)$.

1 Let $\hat{\eta}_0^{(0,0)} = \hat{\eta}_0$
2 **for** $k = 0, 1, \ldots, N$ **do**
3 $\quad \hat{\eta}_{k+1}^{(0,0)} = A\hat{\eta}_k^{(0,0)} + B_u u_k$
4 $\quad \hat{r}_k^{(0,0)} = C\hat{\eta}_k^{(0,0)} + D_u u_k$
5 $\quad \Lambda_{k+1} = A\Lambda_k A^T + B_v B_v^T$
6 $\quad \Sigma_k = C\Lambda_k C^T + D_v D_v^T$
7 **end for**
8 **for** $j = 1, 2, \ldots, L$ **do**
9 $\quad$ Let $\hat{\eta}_0^{(j,1)} = 0$
10 $\quad$ **for** $k = 0, 1, \ldots, N$ **do**
11 $\quad\quad \hat{\eta}_{k+1}^{(j,1)} = A\hat{\eta}_k^{(j,1)} + B_f \varphi_j(k - \kappa_j)$
12 $\quad\quad \hat{r}_k^{(j,1)} = C\hat{\eta}_k^{(j,1)} + D_f \varphi_j(k - \kappa_j)$
13 $\quad$ **end for**
14 **end for**

for all $x_{0:N}$. Then, using the notation established in Algorithms 4.2 and 4.3,

$$\hat{r}_{0:N}^{(j,\kappa_j)} = z^{\kappa_j - 1}\big(\hat{r}_{0:N}^{(j,1)}\big), \tag{4.25}$$

for all j, k, and κ_j.

The procedure for computing the conditional mean and variance of the residual for the LTI special case is given in Algorithm 4.4, which is the LTI analogue of Algorithm 4.2. The analogue of Algorithm 4.3 for the LTI case (not shown here) is obtained by applying the formula (4.25) to each term $\hat{r}_k^{(j,\kappa_j)}$.

Proposition 4.35. *The running time of Algorithm 4.4 is $O(LN)$.*

Proof. Lines 3–6 each take $O(1)$ time to compute. Thus, Lines 1–7 require $O(N)$ time in total. Similarly, Lines 11-12 take $O(1)$ time to compute, so Lines 8–14 require $O(LN)$ time in total. Therefore, the overall running time of Algorithm 4.4 is $O(LN)$. □

The process of time-shifting the simulation results of Algorithm 4.4 can be done using careful array indexing, so we assume that the time-shifing process does not increase the complexity of evaluating the performance metrics. Hence, we have the following corollary

Table 4.1. Time-complexity of computing the performance metrics using Algorithms 4.1–4.4. The column labeled "Simulations" indicates the number of times the recurrence for the conditional mean of the residual (equation (4.17)) must be simulated.

Problem Type	**Simulations**	**Total Complexity**	**Algorithm**
General	$O\big((m+1)^{N+1}\big)$	$O\big(N(m+1)^{N+1}\big)$	4.1
Structured	$O(N^m)$	$O(N^{m+1})$	4.1
LTV Special Case	$O(LN^2)$	$O(LN^L)$	4.2 & 4.3
LTI Special Case	$O(LN)$	$O(LN^L)$	4.4 & 4.3 (shifted)

to Proposition 4.35.

Corollary 4.36. *The time to compute the performance metrics for the* LTI *special case using Algorithm 4.4 and a time-shifted version of Algorithm 4.3 is* $O(LN^L)$.

Proof. By Proposition 4.33, the running time of the time-shifted version of Algorithm 4.3 is $O(LN^L)$, which dominates the running time of Algorithm 4.4. □

The time-complexity results established in Propositions 4.32–4.35 and Corollary 4.36 are summarized in Table 4.1.

4.6 Comments on Continuous-Time Models

In Chapter 3, as well as the present chapter, the model G_θ and the residual generator F are assumed to be discrete-time dynamic systems. Generally speaking, there is no reason to assume that the model G_θ is discrete. Indeed, continuous-time jump-Markov linear systems are treated in detail in [65] and [66], and more general hybrid stochastic differential equations are considered in [105]. The biggest difficulty in using continuous-time models is extending the Markov chain $\{\theta_k\}$ to the more general class of jump processes [105]. In practice, however, the residual generator F only has access to discrete observations $\big\{y(t_k)\big\}_{k\geq 0}$ of the output signal, where $\{t_k\}_{k\geq 0}$ is a sequence of discrete observation times. Hence, the problem is greatly simplified by assuming that G_θ is a discrete-time system, as well.

Chapter 5

Worst-Case Performance Analysis

5.1 Introduction

In this chapter, we consider the performance of a fault detection scheme under uncertain conditions. First, we establish some notation and discuss the various types of uncertainty under consideration. Next, we formulate well-defined optimization problems that characterize the worst-case performance in terms of the probability of false alarm and the probability of detection. Since these optimization problems are, in general, intractable, we impose additional assumptions on the fault diagnosis problem, which yield much simpler optimization problems. Using these assumptions, we consider two classes of optimization problems: those with uncertain signals and those with model uncertainty. Finally, for each class of problems, we show how the worst-case probability of false alarm and the worst-case probability of detection can be formulated as convex programs that can be solved using readily-available numerical optimization software. The results in this section are restricted to fault detection problems involving scalar-valued residuals and threshold decision functions.

5.1.1 Notation

Up to this point, we have used the notation $\{u_k\}_{k\geq 0}$ to denote a discrete-time signal or stochastic process. To simplify notation, we represent sequences by a single letter (e.g., $u = \{u_k\}$) and the action of a dynamic system is represented in more compact operator notation. For example, if the system G maps the input $\{u_k\}$ to the output $\{y_k\}$, we write $y = Gu$. Let $\mathscr{S}^n$ be the set of one-sided deterministic sequences taking values in $\mathbb{R}^n$. For $p \in [1,\infty)$, define

$$\ell_p^n := \left\{ u \in \mathscr{S}^n \,:\, \|u\|_p := \left(\sum_{k=0}^{\infty} \|u_k\|_p^p \right)^{\frac{1}{p}} < \infty \right\}.$$

In the case where $p = \infty$, define

$$\ell_\infty^n := \left\{ u \in \mathscr{S}^n \,:\, \|u\|_\infty := \sup_{k\geq 0} \|u_k\|_\infty < \infty \right\}.$$

For $p \in [1,\infty]$, the ℓ_p-norm ball centered at $u^\circ \in \ell_p^n$ with radius $\gamma > 0$ is defined as

$$B_p^n(u^\circ, \gamma) := \{u + u^\circ \in \mathscr{S}^n \,:\, \|u\|_p < \gamma\}.$$

We may write $B_p(u^\circ, \gamma)$ when the dimension of the sequence is clear from context or of little significance. Given an input-output operator $G\colon \ell_p^n \to \ell_p^m$, with $p \in [1,\infty]$, define the induced norm

$$\|G\|_{ip} := \sup_{u \neq 0} \frac{\|Gu\|_p}{\|u\|_p}.$$

For $p \in [1,\infty]$ and $\gamma > 0$, define the set of norm-bounded operators

$$\mathbf{\Delta}_p^{m\times n}(\gamma) := \{\Delta\colon \mathscr{S}^n \to \mathscr{S}^m \,:\, \|\Delta\|_{ip} < \gamma\}.$$

Similarly, for $p \in [1,\infty]$, $\gamma > 0$, and $q \in \mathbb{N}$, define the set of block-structured norm-bounded operators

$$\hat{\mathbf{\Delta}}_p^{m\times n}(\gamma) := \left\{ \Delta = \operatorname{diag}\{\Delta_1, \Delta_2, \ldots, \Delta_q\} \;:\; \Delta_i \in \mathbf{\Delta}_p^{m_i \times n_i}(\gamma),\ \sum_{i=1}^{q} m_i = m,\ \sum_{i=1}^{q} n_i = n \right\}.$$

We may write $\mathbf{\Delta}_p(\gamma)$ or $\hat{\mathbf{\Delta}}_p(\gamma)$ when the dimension of the operator is clear from context or of little significance.

For each $s \in \mathbb{N}$, define the *s-step truncation operator*

$$\tau_s\colon \mathscr{S}^n \to \mathscr{S}^n\colon u \mapsto \{u_0, u_1, \ldots, u_{s-1}, u_s, 0, 0, \ldots\}.$$

The *one-step shift operator* z is defined as

$$z\colon \mathscr{S}^n \to \mathscr{S}^n\colon u \mapsto \{0, u_0, u_1, \ldots\}.$$

An operator $G\colon \mathscr{S}^n \to \mathscr{S}^m$ is said to be *time-invariant* if

$$Gz = zG.$$

Otherwise, G is said to be *time-varying*.

5.1.2 Types of Uncertainty Considered

Although there are many distinct ways to include uncertainty in the fault detection problem, we consider the following four types of uncertainty:

1. ***Families of Inputs:*** In Chapters 3 and 4, the performance metrics are computed for a single fixed input sequence u. Since this input sequence affects the values of

the performance metrics, a comprehensive performance analysis would consider all possible values of u, which is clearly not feasible. One reasonable compromise is to compute the worst-case performance over a specified family of inputs. To this end, we consider families of inputs that have the following form:

$$B_p^{n_u}(u^\circ, \gamma) = \{u + u^\circ \in \mathscr{S}^{n_u} \,:\, \|u\|_p < \gamma\},$$

where $u^\circ \in \ell_p^{n_u}$ is a fixed nominal input, $p \in [1, \infty]$ specifies the ℓ_p-norm, and $\gamma > 0$ is the desired bound.

2. ***Bounded Disturbances:*** Thus far, we have assumed that the system G_θ is affected by a noise signal v. It is also useful to consider the case where a deterministic signal w, called a *disturbance*, affects the system in such a way that the fault diagnosis scheme cannot use w to generate a residual. We consider disturbances in the bounded set

$$B_p^{n_w}(0, \gamma) = \{w \in \mathscr{S}^{n_w} \,:\, \|w\|_p < \gamma\},$$

where $p \in [1, \infty]$ specifies the ℓ_p-norm, and $\gamma > 0$ is the desired bound.

3. ***Uncertain Fault Signals:*** In Chapters 3 and 4, it is assumed that the fault signal f_k at time k is a known, fixed function of the fault parameter sequence $\theta_{0:k}$. While this approach may work for certain types of faults, it often useful to consider the worst-case performance of a fault diagnosis scheme over a set of possible fault signals. Hence, for a given parameter sequence ϑ, we assume the fault signal lies in a bounded set of the form

$$B_p^{n_f}\big(f(\vartheta)^\circ, \gamma\big) = \{f + f^\circ(\vartheta) \in \mathscr{S}^{n_f} \,:\, \|f\|_p < \gamma\},$$

where $f^\circ(\vartheta) \in \ell_p^{n_f}$ is the nominal value of the fault signal, $p \in [1, \infty]$ specifies the ℓ_p-norm, and $\gamma > 0$ is the desired bound.

4. ***Model Uncertainty:*** In model-based fault diagnosis schemes, the residual generator is usually designed according to the nominal system model G_0. However, it useful to consider cases where G_0 does not perfectly model the system or the designer of the residual generator does not have accurate knowledge of the true model. Both of these cases are addressed by assuming that the parameterized system G_θ is uncertain. In particular, we assume that the system consists of an interconnection of the system G_θ and an uncertain operator Δ. We consider two classes of uncertain operators. First, we consider the class norm-bounded linear time-invariant uncertainties

$$\mathbf{\Delta}_{2,\text{LTI}}(\gamma) := \{\Delta \in \mathbf{\Delta}_2(\gamma) \,:\, \Delta \text{ is LTI, causal, stable}\},$$

where $\gamma > 0$ is the desired bound. Second, we consider the class of norm-bounded

linear time-varying uncertainties

$$\mathbf{\Delta}_{2,\text{LTV}}(\gamma) \coloneqq \{\Delta \in \mathbf{\Delta}_2(\gamma) \,:\, \Delta \text{ is LTV, causal, stable}\},$$

We may also assume that the uncertain operator Δ is block-structured, in which case the uncertainty sets are

$$\hat{\mathbf{\Delta}}_{2,\text{LTI}}(\gamma) \coloneqq \{\Delta \in \hat{\mathbf{\Delta}}_2(\gamma) \,:\, \Delta \text{ is LTI, causal, stable}\},$$

and

$$\hat{\mathbf{\Delta}}_{2,\text{LTV}}(\gamma) \coloneqq \{\Delta \in \hat{\mathbf{\Delta}}_2(\gamma) \,:\, \Delta \text{ is LTV, causal, stable}\}.$$

The overall uncertainty in the fault diagnosis problem depends on which of these four types of uncertainty are included in the model. For simplicity, we consider two classes of problems. The first class has no model uncertainty, and the overall uncertainty set is

$$\mathscr{P}_s = \Big\{\big(u, w, f(\vartheta)\big) \,:\, u \in B_p(u^\circ, \gamma_1),\, w \in B_p(0, \gamma_2),\, f(\vartheta) \in B_p\big(f^\circ(\vartheta), \gamma_3\big)\Big\},$$

where u°, ϑ and $f^\circ(\vartheta)$ are fixed signals and $\gamma_1, \gamma_2, \gamma_3 > 0$ are fixed bounds. The second class only has model uncertainty, and the overall uncertainty set $\mathscr{P}_\Delta$ is either $\mathbf{\Delta}_{2,\text{LTI}}$ or $\mathbf{\Delta}_{2,\text{LTV}}$ (or one of their block-structured counterparts, $\hat{\mathbf{\Delta}}_{2,\text{LTI}}$ or $\hat{\mathbf{\Delta}}_{2,\text{LTV}}$).

For a given point ρ in either $\mathscr{P}_s$ or $\mathscr{P}_\Delta$, the fault diagnosis problem is well-defined and we can compute the performance metrics. Hence, the goal is to determine which value of ρ leads to the worst-case performance in some well-defined sense.

5.1.3 Worst-case Optimization Problems

In order to find the worst-case value of an uncertain signal or operator, we must establish quantitative criteria that lead to well-defined optimization problems. More precisely, we must establish a meaningful way to transform the sequences $\{P_{\text{F},k}\}$ and $\{P_{\text{D},k}\}$ into scalar-valued objective functions. Because the procedure is the same for both uncertainty sets, $\mathscr{P}_s$ and $\mathscr{P}_\Delta$, we let $\mathscr{P}_{(\bullet)}$ represent the unspecified uncertainty set. From the outset, we assume that the residual is scalar-valued and that δ is a time-varying threshold function.

Maximizing the Probability of a False Alarm

For any $\rho \in \mathscr{P}_{(\bullet)}$, the probability of false alarm at time k is

$$\begin{aligned} P_{\text{F},k}(\rho) &= \mathbf{P}\big(|r_k(\rho)| \geq \varepsilon_k \mid \theta_{0:k} = 0_{0:k}\big) \\ &= 1 - \mathbf{P}\big(|r_k(\rho)| < \varepsilon_k \mid \theta_{0:k} = 0_{0:k}\big), \end{aligned}$$

where $0_{0:k}$ denotes the sequence of $k+1$ zeros. Clearly, uncertainty has a negative impact on performance when the probability of false alarm increases. Hence, a worst-case parameter $\rho^\star \in \mathscr{P}_{(\bullet)}$, with respect to the probability of a false alarm, is defined as an optimum point of the following optimization problem:

$$\begin{aligned} P_{\mathrm{F}}^\star &= \max_{\rho\in\mathscr{P}_{(\bullet)}} \max_{0\le k\le N} P_{\mathrm{F},k}(\rho) \\ &= 1 - \min_{\rho\in\mathscr{P}_{(\bullet)}} \min_{0\le k\le N} \mathbf{P}\big(|r_k(\rho)| < \varepsilon_k \mid \theta_{0:k} = 0_{0:k}\big), \end{aligned} \tag{5.1}$$

where $N \ge 0$ is a fixed final time.

Minimizing the Probability of Detection

We analyze the effect of uncertainty conditional on the occurrence of particular fault. Fix a final time N, and let $\vartheta_{0:N} \in \Theta^{N+1}$ be a possible fault parameter sequence, such that $\vartheta_N \neq 0$. Define

$$k_f := \min\{k \ge 0 : \vartheta_k \neq 0\}. \tag{5.2}$$

That is, the fault represented by the sequence $\vartheta_{0:N}$ occurs at time k_f. For any $\rho \in \mathscr{P}_{(\bullet)}$, the probability of detecting the fault at time k is

$$\begin{aligned} P_{\mathrm{D},k}(\rho, \vartheta_{0:N}) &= \mathbf{P}\big(|r_k(\rho)| \ge \varepsilon_k \mid \theta_{0:k} = \vartheta_{0:k}\big) \\ &= 1 - \mathbf{P}\big(|r_k(\rho)| \le \varepsilon_k \mid \theta_{0:k} = \vartheta_{0:k}\big) \end{aligned}$$

With respect to the probability of detecting the fault parameterized by $\vartheta_{0:N}$, a worst-case parameter $\rho^\star \in \mathscr{P}_{(\bullet)}$ is defined as an optimum point of the following optimization problem:

$$\begin{aligned} P_{\mathrm{D}}^\star(\vartheta_{0:N}) &= \min_{\rho\in\mathscr{P}_{(\bullet)}} \max_{k_f\le k\le N} P_{\mathrm{D},k}(\rho, \vartheta_{0:N}) \\ &= 1 - \max_{\rho\in\mathscr{P}_{(\bullet)}} \min_{k_f\le k\le N} \mathbf{P}\big(|r_k(\rho)| < \varepsilon_k \mid \theta_{0:k} = \vartheta_{0:k}\big). \end{aligned} \tag{5.3}$$

In other words, a worst-case parameter $\rho^\star \in \mathscr{P}_{(\bullet)}$ diminishes the effect of the fault parameterized by $\vartheta_{0:N}$ as much as or more than any other parameter $\rho \in \mathscr{P}_{(\bullet)}$.

5.2 Formulating Tractable Optimization Problems

Both optimization problems (5.1) and (5.3) involve the expression

$$\min_{k_f\le k\le N} \mathbf{P}\big(|r_k(\rho)| < \varepsilon_k \mid \theta_{0:k} = \vartheta_{0:k}\big), \tag{5.4}$$

for the appropriate choice of k_f and $\vartheta_{0:N}$. The chief difficulty in solving (5.1) and (5.3) is expressing the minimum (5.4) as a function of ρ, which can then be minimized or maximized to compute $P_{\mathrm{F}}^\star$ or $P_{\mathrm{D}}^\star$, respectively. To properly address this difficulty, we must make some additional assumptions about the sequence $\{r_k(\rho)\}$. Then, under these assumptions, we develop a heuristic that allows us to write the minimization (5.4) in a more tractable form.

5.2.1 Simplifying Assumptions

Fix $\rho \in \mathscr{P}_{(\bullet)}$, and let $\hat{r}_k(\rho, \vartheta_{0:k})$ and $\Sigma_k(\rho, \vartheta_{0:k})$ be the mean and variance, respectively, of the residual $r_k(\rho)$ conditional on the event $\{\theta_{0:N} = \vartheta_{0:N}\}$. To make the minimization (5.4) tractable, we make the following assumptions:

Assumption 1. The variance $\{\Sigma_k\}$ does not depend on the uncertain parameter ρ.

Assumption 2. The variance $\{\Sigma_k\}$ does not depend on the sequence $\vartheta_{0:N}$.

Assumption 3. The threshold ε_k is chosen in proportion to the variance Σ_k. That is, for some fixed $\nu > 0$, $\varepsilon_k = \nu\Sigma_k$, for all k.

Remark 5.1. The purpose of Assumption 1 is to simplify the relationship between the uncertain parameter ρ and the function being minimized in (5.4). Similarly, Assumption 3 simplifies the minimization (5.4) by removing the effect of the time-varying threshold $\{\varepsilon_k\}$. Because the sequence of thresholds $\{\varepsilon_k\}$ must be chosen *a priori*, Assumption 3 is only possible when Assumptions 1 and 2 hold. An important special case where Assumptions 1 and 2 hold is the case where the noise signal v is added directly to the system output y before it enters the residual generator F. ◇

Proposition 5.2. *Let $\rho \in \mathscr{P}_{(\bullet)}$, $0 \le k_f < N$, and $\vartheta_{0:N} \in \Theta^{N+1}$. If Assumptions 1–3 hold, then*

$$\operatorname*{arg\,min}_{k_f \le k \le N} \mathbf{P}\big(|r_k(\rho)| < \varepsilon_k \mid \theta_{0:k} = \vartheta_{0:k}\big) = \operatorname*{arg\,max}_{k_f \le k \le N} \frac{\big|\hat{r}_k(\rho, \vartheta_{0:k})\big|}{\sqrt{\Sigma_k}}.$$

To facilitate the proof of this proposition, we first establish the following lemma:

Lemma 5.3. *Let the function $\mathcal{L}\colon [0,\infty) \times \mathbb{R} \to [0,1)$ be defined as*

$$\mathcal{L}(\nu, \mu) \coloneqq \int_{-\nu}^{\nu} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(s-\mu)^2}{2}\right) \mathrm{d}s.$$

For any $\nu > 0$ and all $\mu_1, \mu_2 \in \mathbb{R}$,

$$|\mu_1| < |\mu_2| \iff \mathcal{L}(\nu, \mu_1) > \mathcal{L}(\nu, \mu_2).$$

Proof. Using the error function, defined in Section 2.2.6, we can write the function $\mathcal{L}$ as

$$\mathcal{L}(\nu,\mu) = \frac{1}{2}\left[\operatorname{erf}\left(\frac{\nu-\mu}{\sqrt{2}}\right) + \operatorname{erf}\left(\frac{\nu+\mu}{\sqrt{2}}\right)\right].$$

Since the map $\mu \mapsto \mathcal{L}(\nu,\mu)$ is clearly even, it suffices to consider $0 \le \mu_1 < \mu_2$. We prove the claim by showing that $\mu \mapsto \mathcal{L}(\nu,\mu)$ is monotonically decreasing on $[0,\infty)$. The derivative of $\mathcal{L}$ at $\mu_0 \ge 0$ is

$$\begin{aligned}
\left.\frac{\partial \mathcal{L}(\nu,\mu)}{\partial \mu}\right|_{\mu=\mu_0} &= \frac{1}{2}\frac{\partial}{\partial \mu}\left[\operatorname{erf}\left(\frac{\nu-\mu}{\sqrt{2}}\right) + \operatorname{erf}\left(\frac{\nu+\mu}{\sqrt{2}}\right)\right]_{\mu=\mu_0} \\
&= \frac{1}{2}\left[\frac{2}{\sqrt{\pi}}\exp\left(-\frac{(\nu-\mu_0)^2}{2}\right)\left(\frac{-1}{\sqrt{2}}\right) + \frac{2}{\sqrt{\pi}}\exp\left(-\frac{(\nu+\mu_0)^2}{2}\right)\left(\frac{1}{\sqrt{2}}\right)\right] \\
&= \frac{1}{\sqrt{2\pi}}\left[\exp\left(-\frac{(\nu+\mu_0)^2}{2}\right) - \exp\left(-\frac{(\nu-\mu_0)^2}{2}\right)\right].
\end{aligned}$$

Since $\mu_0 \ge 0$,

$$(\nu-\mu_0)^2 \le (\nu+\mu_0)^2,$$

with equality if and only if $\mu_0 = 0$. This inequality, together with the fact that the map $x \mapsto e^{-x}$ is monotonically decreasing, implies that

$$\left.\frac{\partial \mathcal{L}(\nu,\mu)}{\partial \mu}\right|_{\mu=\mu_0} \le 0,$$

with equality if and only if $\mu_0 = 0$. □

Proof of Proposition 5.2. Define the "scaled" residual

$$\mu_k(\rho) \coloneqq \frac{r_k(\rho)}{\sqrt{\Sigma_k}},$$

and let $\nu > 0$ be such that $\varepsilon_k = \nu\Sigma_k$, for all k. Note that the conditional mean of $\mu_k(\rho)$ is

$$\hat{\mu}_k(\rho,\vartheta_{0:k}) \coloneqq \mathbf{E}\big(\mu_k(\rho) \mid \theta_{0:k} = \vartheta_{0:k}\big) = \frac{\hat{r}_k(\rho,\vartheta_{0:k})}{\sqrt{\Sigma_k}},$$

and the conditional variance of $\mu_k(\rho)$ is

$$\mathbf{E}\Big(\big(\mu_k(\rho) - \hat{\mu}_k(\rho,\vartheta_{0:k})\big)^2 \;\Big|\; \theta_{0:k} = \vartheta_{0:k}\Big) = \frac{1}{\Sigma_k}\mathbf{E}\Big(\big(r_k(\rho) - \hat{r}_k(\rho,\vartheta_{0:k})\big)^2 \;\Big|\; \theta_{0:k} = \vartheta_{0:k}\Big) = 1.$$

Hence, it is straightforward to show that

$$\begin{aligned}\mathbf{P}\big(|r_k(\rho)| < \varepsilon_k \mid \theta_{0:k} = \vartheta_{0:k}\big) &= \mathbf{P}\left(\left|\frac{r_k(\rho)}{\sqrt{\Sigma_k}}\right| < \frac{\varepsilon_k}{\sqrt{\Sigma_k}} \;\middle|\; \theta_{0:k} = \vartheta_{0:k}\right) \\ &= \mathbf{P}\big(|\mu_k(\rho)| < \nu \mid \theta_{0:k} = \vartheta_{0:k}\big) \\ &= \mathcal{L}\big(\nu, \hat{\mu}_k(\rho, \vartheta_{0:k})\big).\end{aligned}$$

Let $k_1, k_2 \in \mathbb{N}$ be any two time points in the interval $[k_f, N]$. By Lemma 5.3,

$$\mathbf{P}\big(|r_{k_1}(\rho)| < \varepsilon_{k_1} \mid \theta_{0:k_1} = \vartheta_{0:k_1}\big) < \mathbf{P}\big(|r_{k_2}(\rho)| < \varepsilon_{k_2} \mid \theta_{0:k_2} = \vartheta_{0:k_2}\big)$$

if and only if

$$\big|\hat{\mu}_{k_1}(\rho, \vartheta_{0:k_1})\big| > \big|\hat{\mu}_{k_2}(\rho, \vartheta_{0:k_2})\big|. \qquad \square$$

5.2.2 Simplified Worst-case Optimization Problems

The section demonstrates how Assumptions 1–3 and Proposition 5.2 are applied to the problems of computing $P_{\mathrm{F}}^{\star}$ and $P_{\mathrm{D}}^{\star}$.

Maximizing the Probability of False Alarm

Suppose that Assumptions 1–3 hold and assume that no faults have occurred (i.e., $\vartheta = 0$). The worst-case probability of false alarm is

$$P_{\mathrm{F}}^{\star} = 1 - \min_{\rho \in \mathscr{P}_{(\bullet)}} \min_{0 \le k \le N} \mathbf{P}\big(|r_k(\rho)| < \varepsilon_k \mid \theta_{0:k} = 0_{0:k}\big)$$

By Proposition 5.2, optimum values of ρ and k are obtained by solving

$$\hat{\mu}^{\star} = \max_{\rho \in \mathscr{P}_{(\bullet)}} \max_{0 \le k \le N} \frac{|\hat{r}_k(\rho)|}{\sqrt{\Sigma_k}} = \max_{0 \le k \le N} \max_{\rho \in \mathscr{P}_{(\bullet)}} \frac{|\hat{r}_k(\rho)|}{\sqrt{\Sigma_k}}.$$

Because Σ_k does not depend on ρ, this optimization may be solved in two separate stages. First, for $k = 0, 1, \ldots, N$, solve the optimization

$$\hat{r}_k^{\star} = \max_{\rho\, \mathscr{P}_{(\bullet)}} |\hat{r}_k(\rho)|, \tag{5.5}$$

and then compute

$$\hat{\mu}^{\star} = \max_{0 \le k \le N} \frac{\hat{r}_k^{\star}}{\sqrt{\Sigma_k}}.$$

At this point, we must consider what additional assumptions are needed to ensure that the optimization (5.5) can be written as a convex program. Because the residual is

scalar-valued, we can write $\hat{r}_k^\star$ as the solution of the optimization

$$\hat{r}_k^\star = -\min\left\{ \min_{\rho \in \mathscr{P}_{(\bullet)}} -\hat{r}_k(\rho),\ \min_{\rho \in \mathscr{P}_{(\bullet)}} \hat{r}_k(\rho) \right\}.$$

This problem is convex if $\mathscr{P}_{(\bullet)}$ is a convex set and both $\hat{r}_k(\rho)$ and $-\hat{r}_k(\rho)$ are convex functions of ρ (i.e., $\hat{r}_k(\rho)$ is affine in ρ). Once optimum values $k^\star$ and $\rho^\star$ have been obtained, the worst-case probability of false alarm is given by

$$P_{\mathrm{F}}^\star = 1 - \mathbf{P}\big(|r_{k^\star}(\rho^\star)| < \varepsilon_{k^\star} \mid \theta_{0:k^\star} = 0_{0:k^\star}\big).$$

To summarize, the problem of computing $P_{\mathrm{F}}^\star$ is a convex optimization if $\mathscr{P}_{(\bullet)}$ is a convex set and $\hat{r}_k$ is affine in ρ, for all k.

Minimizing the Probability of Detection

Suppose that Assumptions 1–3 hold. Let ϑ be a fault parameter sequence such that $\vartheta_N \neq 0$, and let k_f be the fault time, as defined in equation (5.2). The worst-case probability of detection is

$$P_{\mathrm{D}}^\star = 1 - \max_{\rho \in \mathscr{P}_{(\bullet)}} \min_{k_f \leq k \leq N} \mathbf{P}\big(|r_k(\rho)| < \varepsilon_k \mid \theta_{0:k} = \vartheta_{0:k}\big).$$

By Proposition 5.2, optimum values of ρ and k are obtained by solving

$$\hat{\mu}^\star = \min_{\rho \in \mathscr{P}_{(\bullet)}} \max_{k_f \leq k \leq N} \frac{|\hat{r}_k(\rho)|}{\sqrt{\Sigma_k}}.$$

If we define the vector

$$\hat{R}(\rho) := \begin{bmatrix} \hat{r}_{k_f}(\rho) \\ \hat{r}_{k_f+1}(\rho) \\ \vdots \\ \hat{r}_N(\rho) \end{bmatrix}$$

and the diagonal matrix

$$W := \operatorname{diag}\left\{ \frac{1}{\Sigma_{k_f}}, \frac{1}{\Sigma_{k_f+1}}, \ldots, \frac{1}{\Sigma_N} \right\}, \tag{5.6}$$

then we may write

$$\hat{\mu}^\star = \min_{\rho \in \mathscr{P}_{(\bullet)}} \left\| W^{1/2} \hat{R}(\rho) \right\|_\infty.$$

Since the matrix W is fixed, taking the ∞-norm is equivalent to taking the weighted pointwise maximum of $\hat{r}_{k_f}(\rho), \ldots, \hat{r}_N(\rho)$. Because the pointwise maximum of convex functions is convex [5], computing $P_{\mathrm{D}}^\star$ is a convex optimization if $\mathscr{P}_{(\bullet)}$ is convex and each $\hat{r}_k$ is a convex function of ρ, for $k = k_f, \ldots, N$. Once an optimum value $\rho^\star$ has been computed, let $k^\star$ be

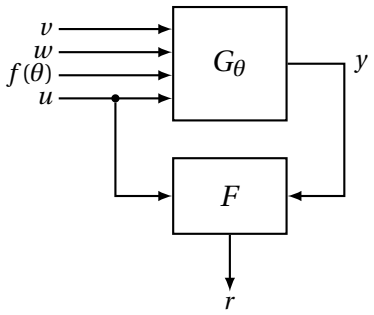


Figure 5.1. Uncertain fault diagnosis problem with uncertain signals but no model uncertainty. The signals u, w, and $f(\theta)$ are constrained to lie in some bounded, convex uncertainty set.

such that $k_f \le k^\star \le N$ and

$$\frac{|\hat{r}_{k^\star}(\rho^\star)|}{\sqrt{\Sigma_{k^\star}}} = \hat{\mu}^\star.$$

Then, the worst-case probability of detection is given by

$$P_{\mathrm{D}}^\star = 1 - \mathbf{P}\big(|r_{k^\star}(\rho^\star)| < \varepsilon_{k^\star} \mid \theta_{0:k^\star} = \vartheta_{0:k^\star}\big).$$

To summarize, the optimization to compute $P_{\mathrm{D}}^\star$ can be written as a convex program if $\mathscr{P}_{(\bullet)}$ is a convex set and $\hat{r}_k$ is a convex function of ρ, for $k = k_f, \ldots, N$.

5.3 Problems with No Model Uncertainty

First, we consider the class of problems with no model uncertainty. Fix a parameter sequence $\theta = \vartheta$, an ℓ_p-norm with $p \in [1,\infty]$, and constants $\gamma_1, \gamma_2, \gamma_3 > 0$. The uncertainty set under consideration is

$$\mathscr{P}_s = \Big\{\big(u, w, f(\vartheta)\big) \,:\, u \in B_p(u^\circ, \gamma_1),\, w \in B_p(0, \gamma_2),\, f(\vartheta) \in B_p\big(f^\circ(\vartheta), \gamma_3\big)\Big\},$$

where u° and $f^\circ(\vartheta)$ are fixed. Decompose the input and fault signals into nominal and uncertain parts, as follows:

$$u = u^\circ + \tilde{u} \qquad f(\vartheta) = f^\circ(\vartheta) + \tilde{f}.$$

If the system G_ϑ is partitioned as

$$G_\vartheta = \begin{bmatrix} G_{1,\vartheta} & G_{2,\vartheta} & G_{3,\vartheta} & G_{4,\vartheta} \end{bmatrix},$$

then the system output can be written as

$$y = G_{1,\vartheta} u + G_{2,\vartheta} v + G_{3,\vartheta} w + G_{4,\vartheta} f(\vartheta).$$

If the residual generator is partitioned as

$$F = \begin{bmatrix} F_1 & F_2 \end{bmatrix},$$

then the residual can be written as

$$\begin{aligned} r &= F_1 y + F_2 u \\ &= (F_1 G_{1,\vartheta} + F_2) u + F_1 G_{2,\vartheta} v + F_1 G_{3,\vartheta} w + F_1 G_{4,\vartheta} f(\vartheta) \\ &= (F_1 G_{1,\vartheta} + F_2)(u^\circ + \tilde{u}) + F_1 G_{2,\vartheta} v + F_1 G_{3,\vartheta} w + F_1 G_{4,\vartheta}\big(f^\circ(\vartheta) + \tilde{f}\big). \end{aligned}$$

Divide the residual into the sum of its nominal, uncertain, and random parts as follows:

$$r = r^{\mathrm{nom}} + r^{\mathrm{unc}} + r^{\mathrm{rnd}},$$

where

$$\begin{aligned} r^{\mathrm{nom}} &= (F_1 G_{1,\vartheta} + F_2) u^\circ + F_1 G_{4,\vartheta} f^\circ(\vartheta), \\ r^{\mathrm{unc}} &= (F_1 G_{1,\vartheta} + F_2) \tilde{u} + F_1 G_{3,\vartheta} w + F_1 G_{4,\vartheta} \tilde{f}, \\ r^{\mathrm{rnd}} &= F_1 G_{2,\vartheta} v. \end{aligned}$$

Since v is zero-mean by assumption, the conditional mean of the residual at time k is

$$\hat{r}_k = \mathbf{E}(r_k \mid \theta_{0:k} = \vartheta_{0:k}) = r_k^{\mathrm{nom}} + r_k^{\mathrm{unc}},$$

and the conditional variance at time k is

$$\Sigma_k = \mathbf{E}\big((r_k - \hat{r}_k)^2\big) = \mathbf{E}\Big(\big(r_k^{\mathrm{rnd}}\big)^2\Big).$$

Note that Assumption 1 holds because the variance Σ is not affected by any of the uncertain signals $\tilde{u}$, w, or $\tilde{f}$. However, Assumption 2 only holds if the operator $G_{2,\vartheta}$ does not depend on the fault parameter ϑ. That is,

$$G_\vartheta = \begin{bmatrix} G_{1,\vartheta} & G_2 & G_{3,\vartheta} & G_{4,\vartheta} \end{bmatrix}.$$

A convenient choice is to take $G_2 = I$, which corresponds to additive measurement noise injected between the plant G_ϑ and the residual generator F.

Maximizing the Probability of False Alarm

Assume that no faults have occurred (i.e., $\vartheta = 0$). The worst-case probability of false alarm is

$$P_{\mathrm{F}}^{\star} = 1 - \min_{(u,w)\in\mathscr{P}_s} \min_{0\le k\le N} \mathbf{P}\big(|r_k| < \varepsilon_k \mid \theta_{0:k} = 0_{0:k}\big).$$

As explained in Section 5.2.2, the crux of computing $P_{\mathrm{F}}^{\star}$ is computing

$$\hat{r}_k^{\star} = \max_{(u,w)\in\mathscr{P}_s} \left| r_k^{\mathrm{nom}} + r_k^{\mathrm{unc}} \right|,$$

for $k = 0, 1, \ldots, N$. More formally, this optimization can be written as

$$\begin{aligned} \hat{r}_k^{\star} = \underset{\tilde{u},w}{\text{maximize}} \quad & \left| r_k^{\mathrm{nom}} + r_k^{\mathrm{unc}} \right| \\ \text{subject to} \quad & r^{\mathrm{nom}} = (F_1 G_{1,0} + F_2)\, u^{\circ}, \\ & r^{\mathrm{unc}} = (F_1 G_{1,0} + F_2)\, \tilde{u} + F_1 G_{3,0}\, w, \\ & \|\tilde{u}\|_p < \gamma_1, \\ & \|w\|_p < \gamma_2, \end{aligned}$$

for $p \in [1,\infty]$ and $\gamma_1, \gamma_2 > 0$. Note that the signal r^{nom} is fixed. Since r^{unc} is a linear function of $\tilde{u}$ and w, the mean of the residual $\hat{r}_k = r_k^{\mathrm{nom}} + r_k^{\mathrm{unc}}$ is an affine function of the decision variables $\tilde{u}$ and w. For $p \in [1,\infty]$, the norm bounds on the decision variables are convex constraints. Therefore, this optimization can be written as a convex program, for all k. In particular, if $p \in \{1,\infty\}$, this optimization can be written as a pair of linear programs (LP), and if $p = 2$, this optimization can be written as a pair of second-order cone programs (SOCP). Both LPs and SOCPs are readily solved with optimization packages, such as SeDuMi [90].

Minimizing the Probability of Detection

Let ϑ be a fault parameter sequence such that $\vartheta_N \neq 0$, and let k_f be the fault time, as defined in equation (5.2). The worst-case probability of detection is

$$P_{\mathrm{D}}^{\star} = 1 - \max_{(u,w,f(\vartheta))\in\mathscr{P}_s} \min_{k_f\le k\le N} \mathbf{P}\big(|r_k| < \varepsilon_k \mid \theta_{0:k} = \vartheta_{0:k}\big).$$

By Proposition 5.2, optimum values of u, w, f, and k are obtained by solving

$$\hat{\mu}^{\star} = \min_{(u,w,f(\vartheta))\in\mathscr{P}_s} \max_{k_f\le k\le N} \frac{|\hat{r}_k|}{\sqrt{\Sigma_k}}.$$

As in Section 5.2.2, if the matrix W is defined by equation (5.6), then this optimization may be written more formally as

$$\begin{aligned}
\hat{\mu}^\star = \underset{\tilde{u},\, w,\, \tilde{f}}{\text{minimize}} \quad & \left\| W^{1/2} \hat{R} \right\|_\infty \\
\text{subject to} \quad & \hat{R}_i = r^{\text{nom}}_{k_f+i-1} + r^{\text{unc}}_{k_f+i-1}, \qquad i = 1, 2, \ldots N - k_f + 1, \\
& r^{\text{nom}} = (F_1 G_{1,\vartheta} + F_2)\, u^\circ + F_1 G_{4,\vartheta} f^\circ, \\
& r^{\text{unc}} = (F_1 G_{1,\vartheta} + F_2)\, \tilde{u} + F_1 G_{3,\vartheta} w + F_1 G_{4,\vartheta} \tilde{f}, \\
& \|\tilde{u}\|_p < \gamma_1, \\
& \|w\|_p < \gamma_2, \\
& \left\|\tilde{f}\right\|_p < \gamma_3,
\end{aligned}$$

for $p \in [1, \infty]$ and $\gamma_1, \gamma_2, \gamma_3 > 0$. Since the signal r^{nom} is fixed, $\hat{R}_k$ is an affine function of the decision variables $\tilde{u}$, w, and $\tilde{f}$, for each k. Since the pointwise maximum of convex functions is convex [5] and the matrix W is fixed, the objective function is convex. For $p \in [1, \infty]$ the norm bounds on $\tilde{u}$, w, and $\tilde{f}$ are convex constraints. Therefore, this optimization is a convex program. In particular, if $p \in \{1, \infty\}$, this optimization is a linear program (LP), and if $p = 2$, this optimization is a second-order cone program (SOCP). Both LPs and SOCPs are readily solved with optimization packages, such as SeDuMi [90].

5.4 Problems with Model Uncertainty

In this section, we consider systems of the form shown in Figure 5.2, where the linear operator Δ represents model uncertainty and the signals u and f are known. Note that this system is not affected by a disturbance w. If the system G_θ is partitioned as

$$G_\theta = \begin{bmatrix} G_{11,\theta} & G_{12,\theta} & G_{13,\theta} & G_{14,\theta} \\ G_{21,\theta} & G_{22,\theta} & G_{23,\theta} & G_{24,\theta} \end{bmatrix},$$

then the signals labeled in Figure 5.2 are related as follows:

$$\begin{aligned}
\beta &= \Delta \alpha, \\
\alpha &= G_{11,\theta} \beta + G_{12,\theta} v + G_{13,\theta} f(\theta) + G_{14,\theta} u, \\
y &= G_{21,\theta} \beta + G_{22,\theta} v + G_{23,\theta} f(\theta) + G_{24,\theta} u.
\end{aligned}$$

Recall that Proposition 5.2 only applies if Assumptions 1–3 of Section 5.2.1 hold. Since the residual generator F is a known linear operator with no uncertainty, the validity of these assumptions depends on the manner in which the noise v affects the system output y.

Let $T_{v \to y}$ denote the map from v to y. If the interconnection shown in Figure 5.2 is

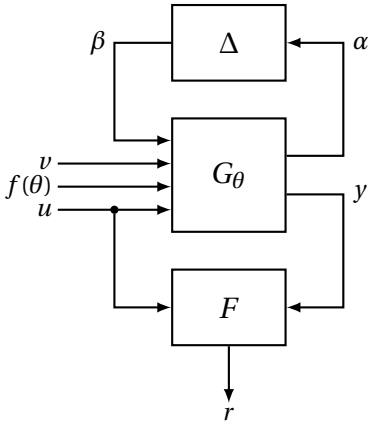


Figure 5.2. Uncertain fault diagnosis problem with model uncertainty. The uncertain operator Δ is constrained to lie in some bounded, convex uncertainty set. For simplicity, we assume that the signals u and $f(\theta)$ are known.

well-posed (i.e., the inverse of $I - G_{11,\theta}\Delta$ exists for all $\vartheta \in \Theta$ and all admissible Δ), then

$$\alpha = (I - G_{11,\theta}\Delta)^{-1}\big(G_{12,\theta}\, v + G_{13,\theta} f(\theta) + G_{14,\theta} u\big),$$

which implies that

$$T_{v\to y} = G_{21,\theta}\Delta(I - G_{11,\theta}\Delta)^{-1}G_{12,\theta} + G_{22,\theta}.$$

Therefore, Assumptions 1 and 2 hold if the noise v does not pass through the uncertain operator Δ (i.e., $G_{12,\theta} = 0$), and the map $G_{22,\theta}$ does not depend on the parameter θ. That is,

$$G_\theta = \begin{bmatrix} G_{11,\theta} & 0 & G_{13,\theta} & G_{14,\theta} \\ G_{21,\theta} & G_{22} & G_{23,\theta} & G_{24,\theta} \end{bmatrix}.$$

The important special case $G_{22} = I$ corresponds to additive measurement noise.

Fix a parameter sequence ϑ and an input u. Assuming that $G_{12,\theta} = 0$ and $\theta = \vartheta$, the signals α and β are given by the equations

$$\begin{aligned} \alpha &= (I - G_{11,\theta}\Delta)^{-1}\big(G_{13,\theta} f(\vartheta) + G_{14,\theta} u\big), \\ \beta &= \Delta\alpha = \Delta\big(G_{13,\theta} f(\vartheta) + G_{14,\theta} u\big). \end{aligned} \tag{5.7}$$

Since the signals $f(\vartheta)$ and u are known and Δ is constrained to be a member of the set $\mathscr{P}_\Delta$, these equations can be interpreted as a constraint on the signal β. Hence, our approach to computing the worst-case performance is to compute the worst-case β, such that equation (5.7) is satisfied by some $\Delta \in \mathscr{P}_\Delta$. The theoretical results that yield such constraints on β can be found in the literature on interpolation theory and model invalidation.

5.4.1 Interpolation Results

The general problem of interpolation consists of finding an operator Δ in some prescribed set $\mathscr{P}_\Delta$, such that Δ maps some fixed input data $\alpha_{0:N}$ to some fixed output data $\beta_{0:N}$. This section states, without proof, a number of relevant results from interpolation theory. The key feature of these results is that, for a given $\alpha_{0:N}$, an interpolating operator exists if and only if $\beta_{0:N}$ lies in some convex set. Therefore, these results can be used as convex constraints on $\beta_{0:N}$ in the previously-defined worst-case optimization problems.

First, we establish some useful notation. For any $a \in \mathscr{S}^m$ and any $\ell > 0$, define the block-Toeplitz matrix

$$T_\ell(a) := \begin{bmatrix} a_0 & 0 & 0 & \cdots & 0 \\ a_1 & a_0 & 0 & \cdots & 0 \\ a_2 & a_1 & a_0 & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ a_\ell & a_{\ell-1} & a_{\ell-2} & \cdots & a_0 \end{bmatrix} \in \mathbb{R}^{m(\ell+1)\times(\ell+1)}.$$

Let $M\colon \mathscr{S}^m \to \mathscr{S}^n$ be a causal linear operator with the impulse response

$$\big\{M[i,j] \in \mathbb{R}^{n\times m} : i \geq j \geq 0\big\}.$$

That is, if $y = Mu$, then

$$y_k = \sum_{j=0}^{k} M[k,j]\,u_j,$$

for all $k \geq 0$. For any such M and any $\ell > 0$, define the block lower-triangular matrix

$$\mathscr{T}_\ell(M) = \begin{bmatrix} M[0,0] & 0 & 0 & \cdots & 0 \\ M[1,0] & M[1,1] & 0 & \cdots & 0 \\ M[2,0] & M[2,1] & M[2,2] & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ M[\ell,0] & M[\ell,1] & M[\ell,2] & \cdots & M[\ell,\ell] \end{bmatrix} \in \mathbb{R}^{n(\ell+1)\times m(\ell+1)}.$$

Note that if M is time-invariant and $y = Mu$, then the matrix $\mathscr{T}_\ell(M)$ is block-Toeplitz and

$$T_\ell(y) = \mathscr{T}_\ell(M)\,T_\ell(u),$$

for all $\ell \geq 0$.

Now, we are ready to state some key results from interpolation theory. These results are summarized at the end of this section in Table 5.1. The following extension of the Carathéodory–Fejér Theorem [80] is due to Fedčina [35] and is used in a number of model-invalidation studies [11, 74, 87].

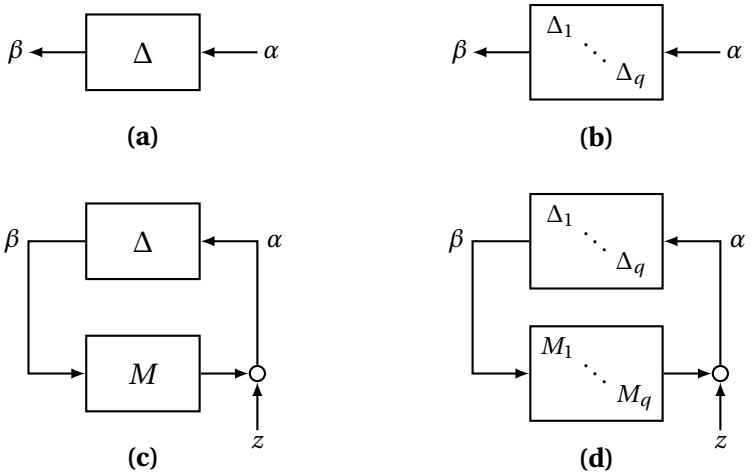


Figure 5.3. Block diagrams for the interpolation results. Theorems 5.4 and 5.9 apply to diagram **(a)**. Corollaries 5.5 and 5.10 apply to diagram **(b)**. Theorem 5.6 applies to diagram **(c)** and Theorem 5.7 applies to diagram to diagram **(d)**.

Theorem 5.4. *Given sequences $\alpha \in \ell_2^n$ and $\beta \in \ell_2^m$ and constants $\gamma > 0$ and $N \in \mathbb{N}$, there exists an operator $\Delta \in \mathbf{\Delta}_{2,\mathrm{LTI}}(\gamma)$, such that*

$$\tau_N \beta = \tau_N \Delta \alpha$$

if and only if

$$T_N^*(\beta) T_N(\beta) \preceq \gamma^2 T_N^*(\alpha) T_N(\alpha).$$

For many applications, it is appropriate to impose additional structure on the interpolating operator Δ. One structure that appears frequently in the robust control literature [28, 86, 110] is the class of block-diagonal operators, which we denote $\hat{\mathbf{\Delta}}_p(\gamma)$. As shown in [11], Theorem 5.4 is extended to operators in set $\hat{\mathbf{\Delta}}_{2,\mathrm{LTI}}(\gamma)$ by simply treating each block-partition separately. Hence, we state this extension as a corollary of Theorem 5.4.

Corollary 5.5. *Given sequences $\alpha \in \ell_2^n$ and $\beta \in \ell_2^m$ and constants $\gamma > 0$ and $N \in \mathbb{N}$, there exists an operator $\Delta = \mathrm{diag}\{\Delta_1, \ldots, \Delta_q\} \in \hat{\mathbf{\Delta}}_{2,\mathrm{LTI}}(\gamma)$, such that*

$$\tau_N \beta = \tau_N \Delta \alpha$$

if and only if

$$T_N^*(\beta_i) T_N(\beta_i) \preceq \gamma^2 T_N^*(\alpha_i) T_N(\alpha_i),$$

for $i = 1, 2, \ldots, q$, where α and β are partitioned such that $\beta_i = \Delta_i \alpha_i$.

The following extension of Theorem 5.4, due to Chen and Wang [11], is useful when the interpolating operator Δ is in a feedback interconnection with another operator.

Theorem 5.6. *Consider the feedback interconnection shown in Figure 5.3(c), in which*

$$\beta = \Delta(I - M\Delta)^{-1} z.$$

Fix $\gamma > 0$ and let $M\colon \ell_2^m \to \ell_2^n$ be a linear time-invariant operator, such that $\|M\|_{i2} \leq \frac{1}{\gamma}$. Then, given sequences $z \in \ell_2^n$ and $\beta \in \ell_2^m$ and $N \in \mathbb{N}$, there exists an operator $\Delta \in \mathbf{\Delta}_{2,\mathrm{LTI}}(\gamma)$, such that

$$\tau_N \beta = \tau_N \Delta(I - M\Delta)^{-1} z$$

if and only if

$$\begin{bmatrix} T(\beta)^T \mathcal{T}(M)^T T(z) + T(z)^T \mathcal{T}(M) T(\beta) + T(z)^T T(z) & T(\beta)^T \\ T(\beta) & \left(\frac{1}{\gamma^2} I - \mathcal{T}(M)^T \mathcal{T}(M)\right)^{-1} \end{bmatrix} \succeq 0,$$

where the subscript N on the operators T and $\mathcal{T}$ has been omitted for clarity.

As in Corollary 5.5, Theorem 5.6 can be extended to the case where Δ is block-diagonal. However, for general M, the matrix inequality in Theorem 5.6 becomes a nonconvex constraint on β, and there is no computationally tractable way to check for the existence of a block-diagonal interpolating operator [11, 92]. However, Chen and Wang [11] show that this matrix inequality is convex in β if M is sufficiently structured. The necessary structure is stated in the following theorem.

Theorem 5.7. *Consider the feedback interconnection shown in Figure 5.3(d), in which*

$$\beta = \Delta(I - M\Delta)^{-1} z.$$

Fix $\gamma > 0$, assume $\Delta = \mathrm{diag}\{\Delta_1, \ldots, \Delta_q\}$, and let $M\colon \ell_2^m \to \ell_2^n$ be a linear time-invariant operator, such that

$$M = \mathrm{diag}\{M_1, M_2, \ldots, M_q\},$$

where the dimensions of M_i are compatible with Δ_i. Further, assume that $\|M_i\|_{i2} \leq \frac{1}{\gamma}$, for all i. Then, given sequences $z \in \ell_2^n$ and $\beta \in \ell_2^m$, there exists an operator $\Delta \in \hat{\mathbf{\Delta}}_{2,\mathrm{LTI}}(\gamma)$, such that

$$\tau_N \beta = \tau_N \Delta(I - M\Delta)^{-1} z$$

if and only if

$$\begin{bmatrix} T(\beta_i)^T\mathscr{T}(M_i)^T T(z_i) + T(z_i)^T\mathscr{T}(M_i)T(\beta_i) + T(z_i)^T T(z_i) & T(\beta_i)^T \\ T(\beta_i) & \left(\frac{1}{\gamma^2}I - \mathscr{T}(M_i)^T\mathscr{T}(M_i)\right)^{-1} \end{bmatrix} \succeq 0,$$

for $i = 1,2,\ldots,q$, where β and z are partitioned compatibly with Δ and M.

Remark 5.8. The statement and proof of Theorems 5.6 and 5.7 involves the relationship

$$T_N(\alpha) = \mathscr{T}_N(M)T_N(\beta) + T_N(z),$$

which only holds when M is time-invariant. To the best of our knowledge, there is no extension of these results in which M is time-varying. ◇

The following time-varying extension of Theorem 5.4 is due to Poolla et al. [74] and used in the model-invalidation context by [27, 87, 92].

Theorem 5.9. *Given sequences $\alpha \in \ell_2^n$ and $\beta \in \ell_2^m$ and constants $\gamma > 0$ and $N \in \mathbb{N}$, there exists an operator $\Delta \in \mathbf{\Delta}_{2,\mathrm{LTV}}(\gamma)$, such that*

$$\tau_N\beta = \tau_N\Delta\alpha$$

if and only if

$$\|\tau_k\beta\|_2 \le \gamma\|\tau_k\alpha\|_2,$$

for $k = 0,1,\ldots,N$.

As in Corollary 5.5, this result is easily extended to the case where Δ is block-diagonal by considering each block-partition separately. Hence, we have the following corollary of Theorem 5.9.

Corollary 5.10. *Given sequences $\alpha \in \ell_2^n$ and $\beta \in \ell_2^m$ and constants $\gamma > 0$ and $N \in \mathbb{N}$, there exists an operator $\Delta = \mathrm{diag}\{\Delta_1,\ldots,\Delta_q\} \in \hat{\mathbf{\Delta}}_{2,\mathrm{LTV}}(\gamma)$, such that*

$$\tau_N\beta = \tau_N\Delta\alpha$$

if and only if

$$\|\tau_k\beta_i\|_2 \le \gamma\|\tau_k\alpha_i\|_2,$$

for $k = 0,1,\ldots,N$ and $i = 1,2,\ldots,q$, where α and β are partitioned such that $\beta_i = \Delta_i\alpha_i$.

Remark 5.11. The condition $\tau_N\beta = \tau_N\Delta\alpha$ used in these interpolation theorems implies that the values α_j and β_j are irrelevant for $j > N$. In the model invalidation literature, this

Table 5.1. Summary of interpolation results for linear operators with and without feedback. The column labeled *Diagram* indicates which part of Figure 5.3 applies.

Result	Diagram	Uncertainty Set	Feedback Operator
Theorem 5.4	(a)	$\mathbf{\Delta}_{2,\text{LTI}}(\gamma)$	
Corollary 5.5	(b)	$\hat{\mathbf{\Delta}}_{2,\text{LTI}}(\gamma)$	
Theorem 5.6	(c)	$\mathbf{\Delta}_{2,\text{LTI}}(\gamma)$	M LTI, $\|M\|_{i2} < \frac{1}{\gamma}$
Theorem 5.7	(d)	$\hat{\mathbf{\Delta}}_{2,\text{LTI}}(\gamma)$	M LTI, $M = \text{diag}\{M_1, \ldots, M_q\}$, $\|M_i\|_{i2} < \frac{1}{\gamma}$
Theorem 5.9	(a)	$\mathbf{\Delta}_{2,\text{LTV}}(\gamma)$	
Corollary 5.10	(b)	$\hat{\mathbf{\Delta}}_{2,\text{LTV}}(\gamma)$	

condition is imposed because only a finite amount of data can be used to invalidate the model. Although the theorems may be more naturally stated in terms of finite sequences $\alpha_{0:N}$ and $\beta_{0:N}$, the truncation operator τ_N is more compatible with the operator-theoretic notation used throughout this chapter. ◇

Remark 5.12. In some instances, the time-invariance assumption of Theorems 5.4 and 5.6 is too restrictive and the time-varying assumption of Theorem 5.9 is too conservative. In the model invalidation literature [91, 101], similar theorems are stated for a time-varying operator Δ such that the *rate of variation* ν, defined as

$$\nu(\Delta) := \left\| z^{-1}\Delta - \Delta z^{-1} \right\|_{i2},$$

is bounded. However, these theorems are stated in the frequency-domain and cannot be used to formulate worst-case optimization problems using Proposition 5.2. To the best of our knowledge, there are no time-domain interpolation results that take into account the rate of variation. ◇

5.4.2 Using the Interpolation Results to Find Worst-case Performance

Having established a variety of interpolation results, we now consider how these results are used as constraints in the worst-case optimization problems. For the sake of simplicity, we only treat the cases where the uncertain operator Δ is unstructured. In each case, the extension to the block-diagonal case is straightforward.

Suppose that Assumptions 1 and 2 of Section 5.2.1 are met by taking $G_{12,\theta} = 0$ and letting G_{22} be independent of the fault parameter θ. Then, the system output is given by

$$\begin{aligned}
\beta &= \Delta\alpha \\
\alpha &= G_{11,\theta}\beta + G_{13,\theta} f(\theta) + G_{14,\theta} u \\
y &= G_{21,\theta}\beta + G_{22} v + G_{23,\theta} f(\theta) + G_{24,\theta} u
\end{aligned}$$

Fix a parameter sequence $\theta = \vartheta$ and let the residual generator F be partitioned as $F = \begin{bmatrix} F_1 & F_2 \end{bmatrix}$. Divide the residual into its non-random and random parts, as follows:

$$r = r^{\text{unc}} + r^{\text{rnd}},$$

where

$$\begin{aligned} r^{\text{unc}} &= F_2 G_{21,\theta}\beta + F_2 G_{23,\theta} f(\theta) + (F_2 G_{24,\theta} + F_1)u \\ r^{\text{rnd}} &= F_2 G_{22} v. \end{aligned}$$

Since v is zero-mean by assumption, the conditional mean of the residual at time k is

$$\hat{r}_k = \mathbf{E}(r_k \mid \theta_{0:k} = \vartheta_{0:k}) = r_k^{\text{unc}},$$

and the conditional variance at time k is

$$\Sigma_k = \mathbf{E}\big((r_k - \hat{r}_k)^2\big) = \mathbf{E}\Big(\big(r_k^{\text{rnd}}\big)^2\Big).$$

Note that, as desired, the sequence $\{\Sigma_k\}$ does not depend on β or θ.

Maximizing the Probability of False Alarm

Assume that no faults have occurred ($\vartheta = 0$). Recall that the worst-case probability of false alarm is

$$P_{\text{F}}^\star = 1 - \min_{\Delta \in \mathscr{P}_\Delta} \min_{0 \le k \le N} \mathbf{P}\big(|r_k| < \varepsilon_k \mid \theta_{0:k} = 0_{0:k}\big).$$

As explained in Section 5.2.2, the crux of computing $P_{\text{F}}^\star$ is solving

$$\hat{r}_k^\star = \max_{\Delta \in \mathscr{P}_\Delta} \big|r_k^{\text{unc}}\big|,$$

for $k = 0, 1, \ldots, N$. There are two cases to consider: $\mathscr{P}_\Delta = \mathbf{\Delta}_{2,\text{LTI}}(\gamma)$ and $\mathscr{P}_\Delta = \mathbf{\Delta}_{2,\text{LTV}}(\gamma)$.

Case 1. Suppose that Δ belongs to the set $\mathbf{\Delta}_{2,\text{LTI}}(\gamma)$ and assume that $G_{11,0}$ is an LTI operator with $\|G_{11,0}\|_{i2} < \frac{1}{\gamma}$. Then, for $k = 0, 1, \ldots, N$, applying Theorem 5.6 yields the following optimization:

$$\begin{aligned} \hat{r}_k^\star = \underset{\beta}{\text{maximize}} \quad & \big|r_k^{\text{unc}}\big| \\ \text{subject to} \quad & r^{\text{unc}} = F_2 G_{21,0}\beta + (F_2 G_{24,0} + F_1)u \\ & z = G_{13,0} f(0) + G_{14,0} u \\ & \mathscr{J}(\beta) \succeq 0, \end{aligned}$$

where

$$\mathscr{J}(\beta) := \begin{bmatrix} T(\beta)^T \mathscr{T}(G_{11,0})^T T(z) + T(z)^T \mathscr{T}(G_{11,0}) T(\beta) + T(z)^T T(z) & T(\beta)^T \\ T(\beta) & \left(\frac{1}{\gamma^2} I - \mathscr{T}(G_{11,0})^T \mathscr{T}(G_{11,0})\right)^{-1} \end{bmatrix}.$$

Note that the subscript N has been omitted from the operators T and $\mathscr{T}$ for clarity.

Since u and $f(0)$ are known, r^{unc} is an affine function of β. Also, the signal z is fixed, so the function $\mathscr{J}(\beta)$ is linear in β, and the constraint $\mathscr{J}(\beta) \succeq 0$ is a linear matrix inequality (LMI). Therefore, this optimization can be cast as a semidefinite program (SDP), which is a type of convex program that is readily solved with numerical optimization packages, such as SeDuMi [90].

Case 2. Suppose that Δ belongs to the set $\mathbf{\Delta}_{2,\text{LTV}}(\gamma)$ and assume that $G_{11,0} = 0$ (i.e., Δ does not experience feedback). Then, for $k = 0, 1, \ldots, N$, applying Theorem 5.9 yields the following optimization:

$$\begin{aligned} \hat{r}_k^\star = \underset{\beta}{\text{maximize}} \quad & \left| r_k^{\text{unc}} \right| \\ \text{subject to} \quad & r^{\text{unc}} = F_2 G_{21,0} \beta + F_2 G_{23,0} f(0) + (F_2 G_{24,0} + F_1) u \\ & \alpha = G_{13,0} f(0) + G_{14,0} u \\ & \|\tau_\ell \beta\|_2 \le \gamma \|\tau_\ell \alpha\|_2, \qquad \ell = 0, 1, \ldots, k. \end{aligned}$$

As in Case 1, r_k^{unc} is affine in β. Since the $k+1$ inequality constraints are quadratic in $\beta_{0:N}$, this optimization problem is a SOCP. As previously mentioned, SOCPs are readily solved with numerical optimization packages.

Minimizing the Probability of Detection

Let ϑ be a fault parameter sequence such that $\vartheta_N \neq 0$, and let k_f be the fault time, as defined in equation (5.2). Recall that the worst-case probability of detection is

$$P_{\text{D}}^\star = 1 - \max_{\Delta \in \mathscr{P}_\Delta} \min_{k_f \le k \le N} \mathbf{P}\big(|r_k| < \varepsilon_k \mid \theta_{0:k} = \vartheta_{0:k}\big).$$

By Proposition 5.2, the optimum values of Δ and k are obtained by solving

$$\hat{\mu}^\star = \min_{\Delta \in \mathscr{P}_\Delta} \max_{k_f \le k \le N} \frac{|\hat{r}_k|}{\sqrt{\Sigma_k}}.$$

As in Section 5.2.2, if the matrix W is defined as in equation 5.6 and the vector $\hat{R}$ is defined as

$$\hat{R} = \begin{bmatrix} r^{\text{unc}}_{k_f} \\ r^{\text{unc}}_{k_f+1} \\ \vdots \\ r^{\text{unc}}_{N} \end{bmatrix},$$

then this optimization may be written as

$$\hat{\mu}^\star = \min_{\Delta \in \mathscr{P}_\Delta} \|W^{1/2}\hat{R}\|_\infty.$$

There are two cases to consider: $\mathscr{P}_\Delta = \mathbf{\Delta}_{2,\text{LTI}}(\gamma)$ and $\mathscr{P}_\Delta = \mathbf{\Delta}_{2,\text{LTV}}(\gamma)$.

Case 1. Suppose that Δ belongs to the set $\mathbf{\Delta}_{2,\text{LTI}}(\gamma)$ and assume that $G_{11,\vartheta}$ is an LTI operator with $\|G_{11,\vartheta}\|_{i2} < \frac{1}{\gamma}$. Then, applying Theorem 5.6 yields the following optimization:

$$\begin{aligned} \hat{\mu}^\star = \underset{\beta}{\text{maximize}} \quad & \|W^{1/2}\hat{R}\|_\infty \\ \text{subject to} \quad & \hat{R}_i = r^{\text{unc}}_{k_f+i-1}, \qquad i = 1,\dots,N-k_f+1, \\ & r^{\text{unc}} = F_2 G_{21,\vartheta}\beta + F_2 G_{23,\vartheta} f(\vartheta) + (F_2 G_{24,\vartheta} + F_1)u \\ & z = G_{13,\vartheta} f(\vartheta) + G_{14,\vartheta} u \\ & \mathscr{J}(\beta) \succeq 0, \end{aligned}$$

where

$$\mathscr{J}(\beta) := \begin{bmatrix} T(\beta)^T \mathscr{T}(G_{11,\vartheta})^T T(z) + T(z)^T \mathscr{T}(G_{11,\vartheta}) T(\beta) + T(z)^T T(z) & T(\beta)^T \\ T(\beta) & \left(\frac{1}{\gamma^2} I - \mathscr{T}(G_{11,\vartheta})^T \mathscr{T}(G_{11,\vartheta})\right)^{-1} \end{bmatrix}.$$

Note that the subscript N has been omitted from the operators T and $\mathscr{T}$ for clarity.

Since the matrix W is fixed, the objective function is a weighted pointwise maximum of $r^{\text{unc}}_{k_f},\dots,r^{\text{unc}}_N$. Of course, r^{unc} is an affine function of β, so the objective is convex in β. Since z is fixed, $\mathscr{J}(\beta)$ is linear in β, and the constraint $\mathscr{J}(\beta) \succeq 0$ is a LMI. Therefore, this optimization is a SDP.

Case 2. Suppose that Δ belongs to the set $\mathbf{\Delta}_{2,\text{LTV}}(\gamma)$ and assume that $G_{11,\vartheta} = 0$ (i.e., Δ does not experience feedback). Then, applying Theorem 5.9 yields the following optimization:

$$\begin{aligned}
\hat{\mu}^\star = \underset{\beta}{\text{maximize}} \quad & \|W^{1/2}\hat{R}\|_\infty \\
\text{subject to} \quad & \hat{R}_i = r^{\text{unc}}_{k_f+i-1}, \qquad i = 1,\dots,N-k_f+1, \\
& r^{\text{unc}} = F_2 G_{21,\vartheta}\beta + F_2 G_{23,\vartheta} f(\vartheta) + (F_2 G_{24,\vartheta} + F_1)u \\
& \alpha = G_{13,\vartheta} f(\vartheta) + G_{14,\vartheta} u \\
& \|\tau_\ell \beta\|_2 \le \gamma \|\tau_\ell \alpha\|_2, \qquad \ell = 0,1,\dots,k.
\end{aligned}$$

As in Case 1, the objective is a weighted pointwise maximum of affine functions of β, which implies that it is convex. Since the signal α is fixed, each of the $k+1$ inequality constraints is quadratic in β and the optimization problem is a SOCP.

Chapter 6

Applications

6.1 Introduction

In this chapter, we explore various applications of the performance analysis framework developed in the preceding chapters. To begin, we examine, from a high level, the various usages of the performance metrics. Then, we demonstrate how the performance metrics are computed for two aerospace examples. The first example is a simplified air-data sensor system consisting of a pitot-static probe and a flight path angle measurement. The second example is a linearized model of a vertical take-off and landing (VTOL) fixed-wing aircraft. For the first example, we consider the effects of uncertain signals, and for the second example, we consider the effects of additive model uncertainty.

6.2 Types of Studies

Although there are many ways to interpret the performance metrics, the following types of studies stand out as natural applications of our performance analysis framework:

1. **Selecting a fault detection scheme:** Given a fixed system G_θ, the performance metrics can be used to select the best fault diagnosis scheme from a finite set of schemes

$$\left\{V^{(i)} = (F^{(i)}, \delta^{(i)}) : i = 1, 2, \ldots, m\right\}.$$

   This type of application is most useful when the fault diagnosis schemes are designed using disparate methodologies with incomparable design criteria.

2. **Trade studies:** Given a collection of systems

$$\left\{G_\theta^{(i)} : i = 1, 2, \ldots, m\right\}$$

   and a collection of fault diagnosis schemes

$$\left\{V^{(i)} = (F^{(i)}, \delta^{(i)}) : i = 1, 2, \ldots, m\right\},$$

let c_i be the cost of implementing the system $G_\theta^{(i)}$ with the scheme $V^{(i)}$, for all i. A *trade study* examines the trade-off between the cost c_i and the performance of the scheme $V^{(i)}$, with respect to the system $G_\theta^{(i)}$, for each i. For example, each system $G_\theta^{(i)}$ may consist of a different combination of sensors and components, in which case a trade study may be used to decide if it is more beneficial, from a fault diagnosis standpoint, to use higher-quality components or to use redundant copies of a lower-quality component. In addition to size, weight, and monetary costs, c_i may also include a measure of how difficult it is to compute the performance metrics for the fault diagnosis problem given by $G_\theta^{(i)}$ and $V^{(i)}$.

3. **Certifying system safety:** Suppose that when a fault is detected, the system G_θ and the fault diagnosis scheme V are reconfigured, as in Section 4.4.2. Recall that in Section 4.4.2, we showed that

$$\hat{J}_k(i,j) = \mathbf{P}(\hat{D}_{j,k} \cap H_{i,k})$$

is the probability of the system being in configuration s_j when it should be in configuration s_j. Note that for some (i,j) pairs, the event $\hat{D}_{j,k} \cap H_{i,k}$ is safe, while for other pairs it is not. For example, it is safe to be in the nominal mode when no faults have occurred, but it is unsafe to be in the nominal mode when a critical sensor has failed. Therefore, by computing and analyzing the matrices $\{\hat{J}_k\}$, we can quantitatively certify that the probability that system is in a safe configuration, at time k, is within some acceptable range $[1-\alpha, 1]$.

6.3 Air-Data Sensor Example

Nearly all aircraft use a pitot-static probe to determine airspeed V and altitude h. Because these data are essential for flying, the pitot-static probe is integrated into the flight control feedback loop. These sensors are prone to a number of failures, such as icing and blockage, that cause them to produce incorrect values. If such a failure goes undetected, the autopilot system or the pilot may use the erroneous values to issue commands that cause the aircraft to crash. To avoid such disasters, large commercial aircraft, such as the Boeing 777 [103,104], have multiple pitot-static probes in different locations. However, most aircraft designers have developed a set of standard operating procedures that allow safe recovery of the aircraft when a pitot-static probe failure is detected [6]. In this application we explore the detection of such faults by exploiting the analytical redundancy between airspeed, altitude, and flight path angle. This example was also studied less extensively in the conference papers [97,98].

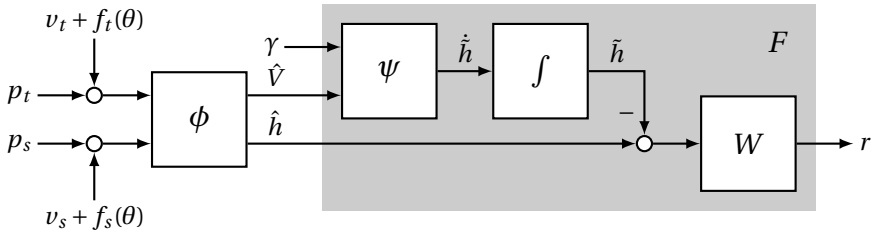


Figure 6.1. Block diagram of a pitot-static probe with a fault detection scheme based on analytical redundancy. The map ϕ (shown graphically in Figure 6.2) represents the system G, while the shaded region, labeled F, is the residual generator.

6.3.1 Problem Formulation

Consider the fault detection problem shown in Figure 6.1, in which a pitot tube measures the total pressure p_t, and a static port measures the static pressure p_s. These measurements are corrupted by adding Gaussian white noise processes, v_t and v_s, and randomly occurring faults, f_t and f_s. From the measured pressures, airspeed and altitude are derived using the relations

$$\begin{bmatrix} V \\ h \end{bmatrix} = \phi(p_t, p_s) := \begin{bmatrix} \operatorname{sign}(p_t - p_s)\, c_3 \left(\left| \left(\frac{p_t - p_s}{p_0} + 1 \right)^{c_4} - 1 \right| \right)^{\frac{1}{2}} \\ c_1 \left(1 - \left(\frac{p_s}{p_0} \right)^{c_2} \right) \end{bmatrix}, \tag{6.1}$$

where the constants

$$\begin{aligned} c_1 &= 44.331\,\text{km}, \\ c_2 &= 0.1903, \\ c_3 &= 760.427\,\text{m/s}, \\ c_4 &= 2/7, \\ p_0 &= 101.325\,\text{kPa} \end{aligned}$$

model the troposphere (up to 17 km) [18]. These equations are plotted in Figure 6.2 for subsonic flight in the troposphere. We use the notation $\hat{V}$ for the derived airspeed and $\hat{h}$ for the derived altitude to indicate that these quantities are corrupted by random disturbances and faults. Note ϕ actually gives the *indicated* airspeed, which is the airspeed that would be measured if the sensors were at standard atmospheric conditions. To obtain the *true* airspeed, we would also need a measurement of the outside air temperature [18]. However, we ignore this issue for the sake of simplicity.

The fault signals are randomly-occurring biases, defined as

$$f_t(t) := b_t \mathbb{1}(t \geq \tau_t) \quad \text{and} \quad f_s(t) := b_s \mathbb{1}(t \geq \tau_s),$$

for $t \geq 0$, where b_t and b_s are known, fixed bias magnitudes, and τ_t and τ_s are independent

exponential random variables $\tau_t \sim \text{Exp}(\lambda_t)$ and $\tau_s \sim \text{Exp}(\lambda_s)$.

The dynamic portion of the fault detection scheme F is contained in the shaded region of Figure 6.1. The input γ is the flight path angle of the aircraft, which we assume is measured exactly with no noises or faults. Consider the following analytical relationship between V, h, and γ:

$$\begin{aligned} h(t) &= h(0) + \int_0^t \psi\big(V(s), \gamma(s)\big)\,\mathrm{d}s \\ &= h(0) + \int_0^t V(s)\sin\gamma(s)\,\mathrm{d}s, \end{aligned}$$

which is used to derive $\tilde{h}$ from γ and $\hat{V}$. The fault detection scheme attempts to detect the faults f_t and f_s by analyzing the difference $\hat{h} - \tilde{h}$. However, as the noisy signal $\psi(\hat{V}, \gamma)$ passes through the integrator, the noise accumulates and $\tilde{h}$ diverges from $\hat{h}$. To counteract this effect, a high-pass or "washout" filter of the form

$$W(s) = \frac{s}{s+a}, \qquad a > 0,$$

is applied to the difference $\hat{h} - \tilde{h}$ to produce the residual r. The drawback of using this filter is that it removes the steady-state or "DC" component from the signal $\hat{h} - \tilde{h}$. We assume that the decision function (not depicted in Figure 6.1) is a threshold function with threshold $\varepsilon > 0$.

6.3.2 Applying the Framework

To apply the computational framework developed in Chapter 4, the system G must be LTV. As shown in Figure 6.2, the map ϕ is only mildly nonlinear for modest changes in differential pressure $p_d := p_t - p_s$ and static pressure p_s, so we take the first-order approximation

$$\phi\big(p_t + v_t + f_t(\theta),\ p_s + v_s + f_s\big) \approx \phi(p_t, p_s) + \Phi \begin{bmatrix} v_t \\ v_s \end{bmatrix} + \Phi \begin{bmatrix} f_t(\theta) \\ f_s \end{bmatrix},$$

where $\Phi := (D\phi)(p_t, p_s)$ is the Jacobian linearization of ϕ. Then, the linearized system G is given by the static equation

$$y = \hat{D}_u u + \hat{D}_v v + \hat{D}_f f,$$

where

$$y = \begin{bmatrix} \hat{V} \\ \hat{h} \end{bmatrix}, \qquad u = \phi(p_t, p_s), \qquad v = \begin{bmatrix} v_t \\ v_s \end{bmatrix}, \qquad f = \begin{bmatrix} f_t \\ f_s \end{bmatrix},$$

and $\hat{D}_u = I$ and $\hat{D}_v = \hat{D}_f = \Phi$. Note that for a given flight path angle γ, the map ψ can be interpreted as a linear function of V. Hence, the residual generator F can be written as the

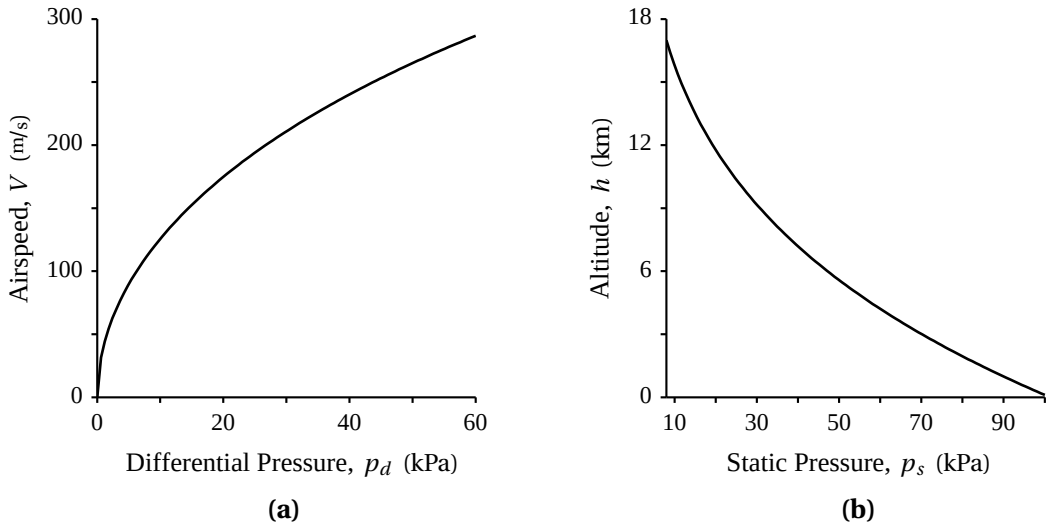


Figure 6.2. Visualization of the air-data sensor equations. Plot of **(a)** the (indicated) airspeed V as a function of differential pressure $p_d = p_t - p_s$ and **(b)** the altitude h as a function of static pressure p_s. The values plotted here are typical for subsonic flight in the troposphere.

linear system

$$F \begin{cases} \dot{\xi} = \tilde{A}\xi + \tilde{B}_y y, \\ r = \tilde{C}\xi + \tilde{D}_y y, \end{cases}$$

where

$$\tilde{A} = -a, \qquad \tilde{B}_y = \begin{bmatrix} \sin(\gamma) & a \end{bmatrix}, \qquad \tilde{C} = -1, \qquad \tilde{D}_y = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

The final step in applying our performance analysis framework is to convert everything to discrete time. Let $T_s > 0$ be a fixed sample time, and let $N \in \mathbb{N}$ be the final time step (i.e., NT_s is the time horizon considered). We use the "zero-order hold" method [7] to discretize the continuous-time dynamics. For each $k \geq 0$, define the input $u_k := \phi\big(p_t(t_k), p_s(t_k)\big)$, where $t_k = kT_s$. Since the discrete-time analogue of Gaussian white noise is an IID Gaussian sequence [50,72], we define the IID sequences $\{v_{t,k}\}_{k\geq 0}$ and $\{v_{s,k}\}_{k\geq 0}$ with $v_{t,i} \sim \mathcal{N}(0, \sigma_t^2)$ and $v_{s,i} \sim \mathcal{N}(0, \sigma_s^2)$, respectively, for all i. The fault signals are represented in discrete-time by $f_{t,k} = b_t \mathbb{1}(k \geq \kappa_t)$ and $f_{s,k} = b_s \mathbb{1}(k \geq \kappa_s)$, for all k, where $\kappa_t \sim \text{Geo}(q_t)$ and $\kappa_s \sim \text{Geo}(q_s)$ are geometric random variables. As shown in Fact 2.8, the best discrete-time model is achieved when $q_t = 1 - e^{-\lambda_t T_s}$ and $q_s = 1 - e^{-\lambda_s T_s}$.

6.3.3 Numerical Results

First, we compute the joint probability and conditional probability performance metrics defined in Chapter 3. For these simulations, the following parameters values are used:

- Sample time: $T_s = 0.05\,\text{s}$
- Time horizon: $N = 72{,}000$ (i.e., $NT_s = 1\,\text{hour}$)
- Flight path angle: 0.5° (constant)
- Airspeed: $V = 45\,\text{m/s}$ (constant)
- Initial Altitude: $h(0) = 200\,\text{m}$
- Noise standard deviations: $\sigma_t = 2.5\,\text{Pa}$, $\sigma_s = 2.5\,\text{Pa}$
- Bias fault magnitudes: $b_t = -0.04\,\text{kPa}$, $b_s = 0.05\,\text{kPa}$
- Continuous failure time models: $\lambda_t = \lambda_s = 0.001\,\text{hr}^{-1} = 2.78 \times 10^{-7}\,\text{s}^{-1}$
- Discrete failure time models: $q_t = q_s = 1.389 \times 10^{-7}$
- Filter pole: $a = 0.003$ (before discretization)
- Threshold: $\varepsilon = 2\,\text{m}$

The resulting performance metrics are plotted in Figures 6.3(a) and 6.3(b). Note that, in this case, the component failure rates are so small that the plots of $\{P_{\text{FN},k}\}$ and $\{P_{\text{TP},k}\}$ are barely distinguishable from zero.

Next, we plot the ROC curves as the threshold ε varies from 0.1 m to 50 m. The curves shown in Figure 6.4 correspond to times ranging from 1 minute to 1 hour. Note that, in Figure 6.3(b), the probability of detection $\{P_{\text{D},k}\}$ dips at about 7 minutes. Hence, some of the ROC curves in Figure 6.4 cross over one another. However, the general trend is that the ROC curves pass closer to the ideal point $(0,1)$ as time increases.

For our third numerical experiment, we observe that the probability of detection, plotted in Figure 6.3(b), converges to a steady-state value. To better understand the effects of changing the washout filter pole a and the noise standard deviation σ, we compute the steady-state values of $\{P_{\text{D},k}\}$ as a and σ vary. In Table 6.1, these steady-state values are tabulated for a ranging from 0.0005 to 0.004 and σ ranging from 2 Pa to 10 Pa. Note that the value of a listed in Table 6.1 corresponds to the continuous-time washout filter W before discretization. Also, the same standard deviation σ is used for both noise signals, $\{v_{t,k}\}$ and $\{v_{s,k}\}$. All other parameters remain the same as in the previous experiments.

In our fourth experiment, we seek to find the worst-case flight path, with respect to the probability of false alarm. For these optimizations, we use the values $a = 0.003$ and $\sigma_t = \sigma_s = 2.5\,\text{Pa}$, as in the first two experiments. We assume that there is no disturbance w or model uncertainty Δ affecting the system. The class of uncertain inputs considered is

$$B_2(u^\circ, \gamma) = \left\{\tilde{u} + u^\circ \,:\, \|\tilde{u}\|_2 < \gamma\right\},$$

where $u^\circ = (V, h)$ is the flight path described in the first experiment. Since we only consider additive input faults, the conditional variance of the residual, $\{\Sigma_k\}$ does not depend on the fault parameter sequence θ or the uncertain input. Hence, we can fulfill Assumptions 1–3 of

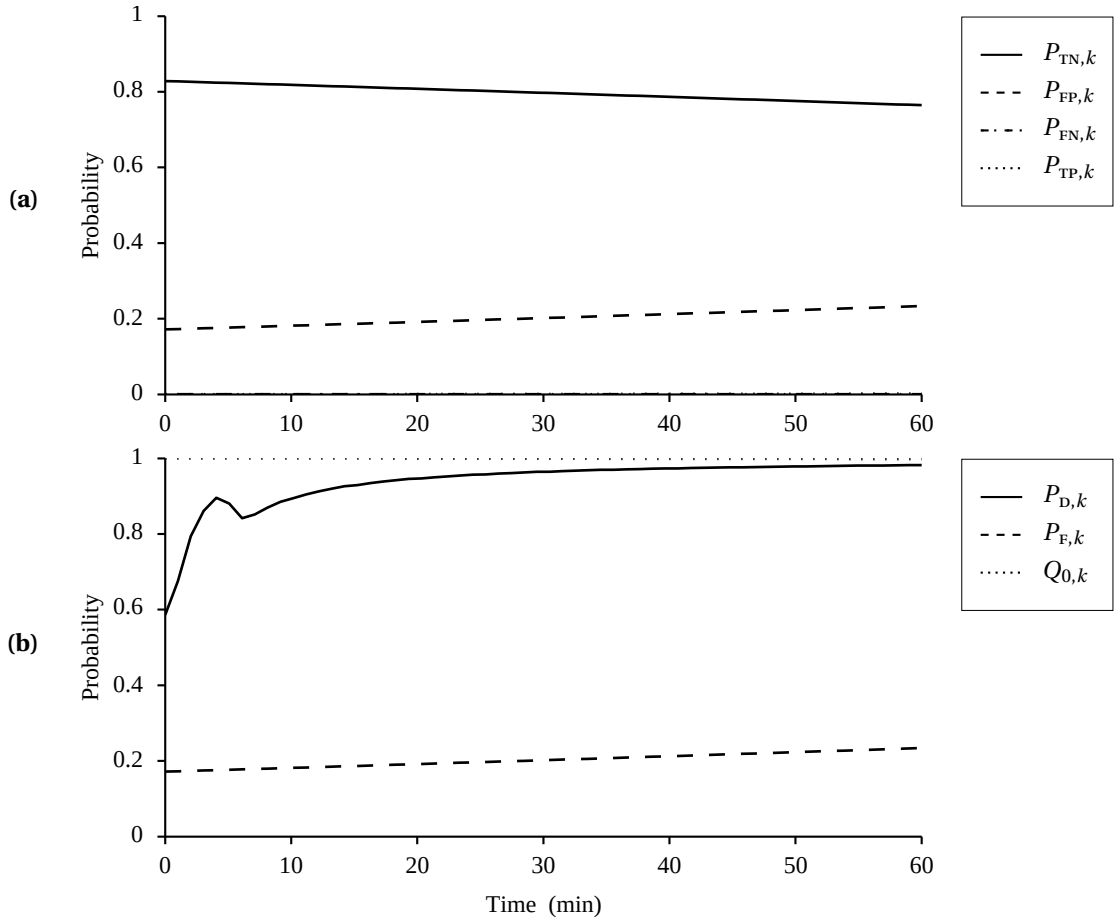


Figure 6.3. Performance metrics for the air-data sensor system. Plot **(a)** shows the joint probability performance metrics, and plot **(b)** shows the conditional probability performance metrics. Note that the sequences $\{P_{\mathrm{FN},k}\}$ and $\{P_{\mathrm{TP},k}\}$ have small values and are barely distinguishable from zero.

Table 6.1. Steady-state performance of the air-data sensor system for various values of the washout filter pole a and the noise standard deviation σ. Note that the values of the pole a refer to the continuous-time dynamics before discretization, but the standard deviation σ refers to the discretized IID Gaussian noise sequences (i.e., $\sigma_s = \sigma_t = \sigma$).

	Noise Standard Deviation, σ (Pa)				
Pole, a	2	4	6	8	10
0.0005	0.9742	0.9482	0.9216	0.8943	0.8662
0.001	0.9739	0.9469	0.9183	0.8875	0.8534
0.0015	0.9736	0.9454	0.9137	0.8756	0.8211
0.002	0.9732	0.9435	0.9064	0.8423	0.7303
0.0025	0.9729	0.9410	0.8879	0.7631	0.5968
0.003	0.9725	0.9373	0.8427	0.6517	0.4692
0.0035	0.9720	0.9291	0.7687	0.5387	0.3680
0.004	0.9715	0.9104	0.6790	0.4411	0.2933

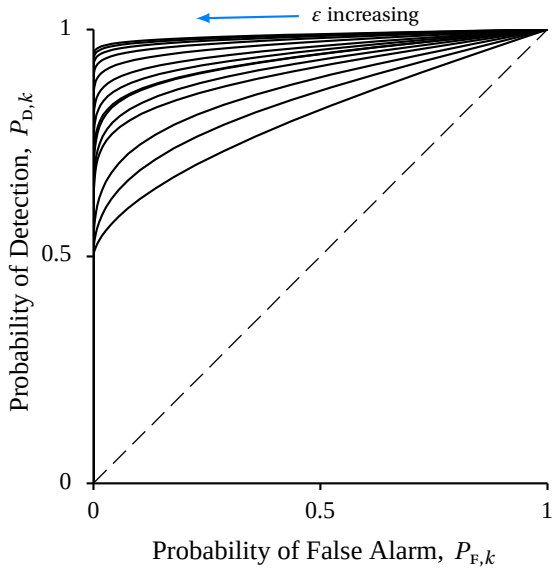


Figure 6.4. Performance metrics for the air-data sensor system plotted in ROC space. Each ROC curve represents the performance of the fault detection scheme shown in Figure 6.1 at a particular time step as the threshold ε on the decision function δ is varied.

Section 5.2.1 by using the proportional threshold

$$\varepsilon_k = \nu\sqrt{\Sigma_k},$$

where $\nu = 2.25$. We use the YALMIP interface [63] to SeDuMi [90] to solve the optimization problem. The resulting worst-case values $P_F^\star(\gamma)$ are plotted in Figure 6.5 for γ ranging from 0 to 10.

Finally, we compute the worst-case fault signal, with respect to the probability of detection. For this computation, we assume that there are no other sources of uncertainty. Let ϑ be the fault parameter sequence in which both sensors fail at $k = 18,000$ (15 minutes). The class of uncertain fault signals considered is

$$B_2(f^\circ, \gamma) = \{\tilde{f} + f^\circ(\vartheta) \,:\, \|\tilde{f}\|_2 < \gamma\},$$

where $f^\circ(\vartheta)$ is the nominal bias fault with magnitudes b_s and b_f defined above. The time horizon of the simulation is shortened to 17 minutes (i.e., $N = 20,400$ time steps). Hence, the signal $\tilde{f}$ must decrease the probability of detection (i.e., suppress the effect of the nominal fault $f^\circ(\vartheta)$) over a 2 minute interval. Again, we use the YALMIP interface [63] to formulate the optimization problem and SeDuMi [90] to solve it. The resulting worst-case values $P_D^\star(\gamma)$ are plotted in Figure 6.6 for γ ranging from 1.5 to 2.0. Note that, for each γ, the value of $P_D^\star(\gamma)$ would increase as the number of time steps N is increased, because the perturbation $\tilde{f}$ would have to suppress the effect of $f^\circ(\vartheta)$ over a longer time span. That is, increasing N

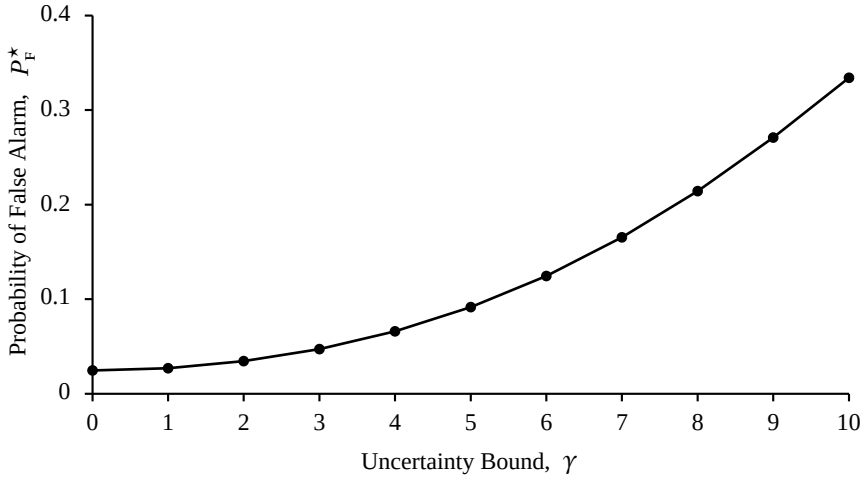


Figure 6.5. Worst-case probability of false alarm for the air-data sensor system with an uncertain input of the form $u = u^\circ + \tilde{u}$, where u° is a fixed nominal input and $\|\tilde{u}\|_2 < \gamma$.

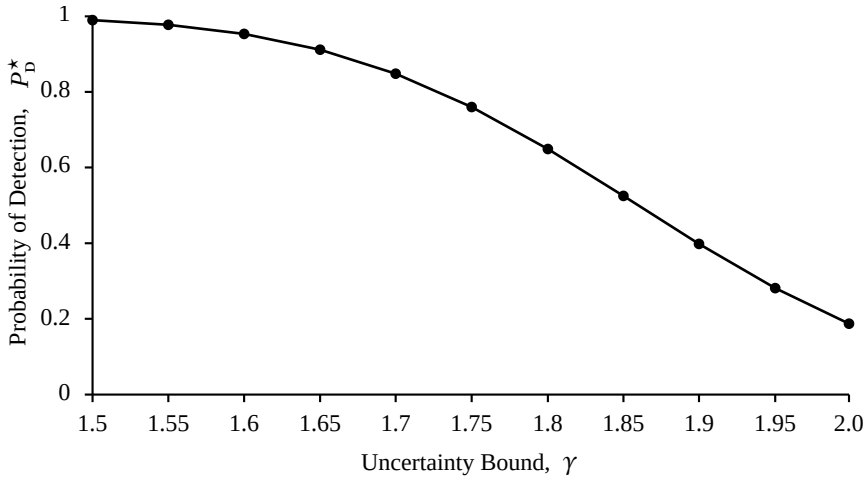


Figure 6.6. Worst-case probability of detection for the air-data sensor system with an uncertain fault signal of the form $f(\vartheta) = f(\vartheta)^\circ + \tilde{f}$, where ϑ is a fixed fault parameter sequence, $f(\vartheta)$ is a fixed nominal fault signal, and $\|\tilde{f}\|_2 < \gamma$.

and decreasing γ have a similar effect on the worst-case performance. The relatively short time span (2 minutes) used for these simulations was chosen to keep the computations manageable.

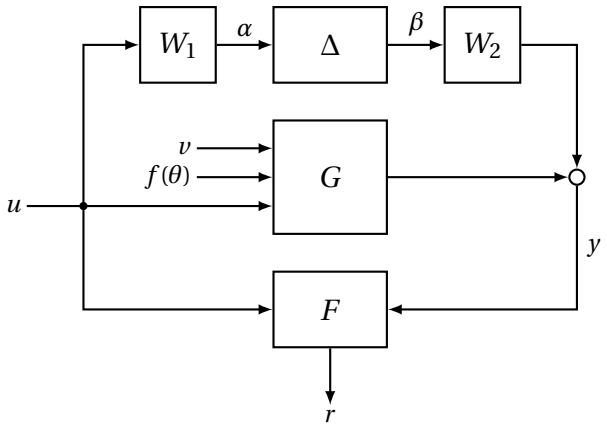


Figure 6.7. Block diagram of a linearized vertical take-off and landing (VTOL) aircraft model with additive model uncertainty.

6.4 VTOL Aircraft Example

In this section, we examine the effects of additive model uncertainty on the performance of an observer-based fault detection scheme. The system under consideration is is a linearized model of the longitudinal dynamics of a vertical take-off and landing (VTOL) aircraft. The original modeling and linearization of this system are due to Narendra and Tripathi [70]. Since the publication of [70], variants of this model have been used in a number of fault detection studies (e.g., [83, 94–96, 102]).

6.4.1 Problem Formulation

Consider the block diagram shown in Figure 6.7. The additive uncertainty affects the map from the input u to the output y. Assume that both W_1 and W_2 are fixed square matrices, and assume that $\Delta \in \mathbf{\Delta}_{2,\mathrm{LTV}}(\gamma)$. The continuous-time dynamics of the system are of the form

$$\begin{aligned} \dot{x} &= Ax + B_u u + B_v v + B_f f(\theta), \\ y &= Cx + \big(D_u + W_2 \Delta W_1\big) u + D_v v, \end{aligned}$$

where the states and inputs are defined as

$$x = \begin{bmatrix} \text{horizontal velocity (knots)} \\ \text{vertical velocity (knots)} \\ \text{pitch rate } (^{\text{deg}}/_{\text{s}}) \\ \text{pitch angle (deg)} \end{bmatrix}, \qquad u = \begin{bmatrix} \text{collective pitch control} \\ \text{longitudinal pitch control} \end{bmatrix}$$

The following matrices correspond to the linearized VTOL model at an airspeed of 135 knots:

$$A = \begin{bmatrix} -9.9477 & -0.7476 & 0.2632 & 5.0337 \\ 52.1659 & 2.7452 & 5.5532 & -24.4221 \\ 26.0922 & 2.6361 & -4.1975 & -19.2774 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$$B_u = \begin{bmatrix} 0.4422 & 0.1761 \\ 3.5446 & -7.5922 \\ -5.5200 & 4.4900 \\ 0 & 0 \end{bmatrix}, \quad B_v = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad B_f = B_u,$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}, \quad D_u = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad D_v = \begin{bmatrix} 0 & 0.2 \\ 0 & 0.1 \\ 0.3 & 0 \\ 0 & 0 \end{bmatrix}, \quad D_f = D_u.$$

Residual Generator

The residual generator is based on a Luenberger observer [64] with the observer gain $L \in \mathbb{R}^{4\times 4}$. Hence, the continuous-time dynamics of the residual generator F are of the form

$$F \begin{cases} \dot{\xi} = A\xi + B_u u + L(y - \hat{y}), \\ \hat{y} = C\xi + D_u u, \\ r = M(y - \hat{y}). \end{cases}$$

To obtain a scalar-valued residual, we take M to be

$$M = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}.$$

We consider the following observer gain matrices:

1. Gain proposed by Wei and Verhaegen [96]:

$$L_1 = \begin{bmatrix} 0.6729 & -1.4192 & -0.0396 & 1.7178 \\ 5.0829 & 0.0881 & 0.2018 & -1.5150 \\ -5.0978 & 10.5595 & 3.4543 & -11.2687 \\ 0.5041 & -1.0298 & -0.0012 & 1.0785 \end{bmatrix}$$

2. Gain proposed by Wang, Wang, and Lam [95]:

$$L_2 = \begin{bmatrix} 4.3021 & -10.0144 & -3.5587 & 4.8599 \\ 6.3561 & -1.6791 & -0.9140 & -2.4219 \\ -21.1044 & 47.6843 & 17.6497 & -22.7378 \\ 2.9567 & -6.7268 & -2.7124 & 3.4869 \end{bmatrix}$$

3. Gain proposed by Wang and Yang [94]:

$$L_3 = \begin{bmatrix} 0.6953 & -1.3907 & 0 & 1.7402 \\ 4.9745 & 0.0509 & 0 & -1.6751 \\ -5.1998 & 10.3996 & 3.3333 & -11.3239 \\ 0.5100 & -1.0201 & 0 & 1.0781 \end{bmatrix}$$

The resulting residual r is passed to a threshold decision function δ.

Input Signals

For the system input u, we use the signals defined in [96], where u is the output of a controller K. It is difficult to obtain the exact form of u without also implementing K, which would add unnecessary complexity to our example. However, the plots of u shown in [96] can be closely approximated by the following continuous-time signal:

$$u(t) = \begin{bmatrix} 1.5 - 0.03(t \bmod 100) + 0.25\sin\left(\frac{2\pi t}{3}\right) \\ -0.75 + 0.03(t \bmod 50) \end{bmatrix}, \tag{6.2}$$

where the terms of the form $n(t \bmod m)$ are due to the "sawtooth wave" reference command used in [96].

For the fault model, we assume that there are two components that fail independently at random. For the sake of simplicity, we follow [96] and take the faults to be randomly occurring biases:

$$f(t) = \begin{bmatrix} b_1 \mathbb{1}(t \geq \tau_1) \\ b_2 \mathbb{1}(t \geq \tau_2) \end{bmatrix}$$

where $\tau_1 \sim \text{Exp}(\lambda_1)$ and $\tau_2 \sim \text{Exp}(\lambda_2)$ are the random failure times. Section 4.2.2 demonstrates that the discrete-time version of this fault model (see Fact 2.8) can be parameterized by a tractable Markov chain θ.

Finally, we assume that the noise signal v is a Gaussian white noise process.

6.4.2 Applying the Framework

The main task in applying our computational framework is to convert the continuous-time VTOL aircraft model to a discrete-time system. For a fixed sample time $T_s > 0$, we use the "zero-order hold" method [7] to discretize the system and the residual generator. The input signal in equation (6.2) is sampled to obtain $u_k = u(kT_s)$, for all $k \geq 0$. Using Fact 2.8, we convert the random failure times τ_1 and τ_2 to discrete failure times $\kappa_1 \sim \text{Geo}(q_1)$ and $\kappa_2 \sim \text{Geo}(q_2)$, respectively, where $q_i = 1 - e^{-\lambda_i T_s}$. Finally, we assume that the noise $\{v_k\}$ is an IID Gaussian process with $v_i \sim \mathcal{N}(0, \sigma^2 I)$, for all i.

6.4.3 Numerical Results

First, we compute the joint probability and conditional probability performance metrics defined in Chapter 3. For these simulations, the following parameter values are used:

- Sample time: $T_s = 0.05\,\text{s}$
- Time horizon: $N = 72{,}000$ (i.e., $NT_s = 1\,\text{hour}$)
- Noise standard deviation: $\sigma = 5$
- Bias fault magnitudes: $b_1 = 2$, $b_2 = -2$.
- Continuous failure time models: $\lambda_1 = \lambda_2 = 0.002\,\text{hr}^{-1} = 5.56 \times 10^{-7}\,\text{s}^{-1}$
- Discrete failure time models: $q_1 = q_2 = 2.778 \times 10^{-8}$
- Threshold: $\varepsilon_k = \nu\sqrt{\Sigma_k}$, $\nu = 2.25$

Note that the threshold ε_k is proportional to the residual standard deviation $\sqrt{\Sigma_k}$, for all k. This choice, which fulfils Assumption 3 in Section 5.2.1, is possible because the noise v does not pass through the uncertain operator and the map from v to the output y does not depend on the fault parameter θ (see Section 5.4). The performance metrics generated with observer gain L_1 are plotted in Figures 6.8(a) and 6.8(b). Since the component failure rates are so small, the plotted values of $\{P_{\text{FN},k}\}$ and $\{P_{\text{TP},k}\}$ are barely distinguishable from zero.

Next, we compare the performance of the three residual generators parameterized by the observer gain matrices L_1, L_2, and L_3. Because the performance metrics plotted in Figures 6.8(a) and 6.8(b) converge to steady-state values, we compare the performance of the residual generators by examining their values at the final time step N. The resulting steady-state performance metrics are listed in Table 6.2. Note that the probability of false alarm is the same for all three cases. This is because the residual is zero-mean when no faults occur and the threshold is proportional to the noise standard deviation $\sqrt{\Sigma_k}$. Thus, the parameter ν can be chosen to achieve a desired false alarm probability.

Our next experiment involves finding the worst-case additive uncertainty, with respect to the probability of false alarm. The uncertainty set considered here is

$$\mathbf{\Delta}_{2,\text{LTV}}(\gamma) = \left\{\Delta\colon \ell_2^2 \to \ell_2^4 \,:\, \Delta \text{ LTV, causal stable }, \|\Delta\|_{i2} < \gamma\right\},$$

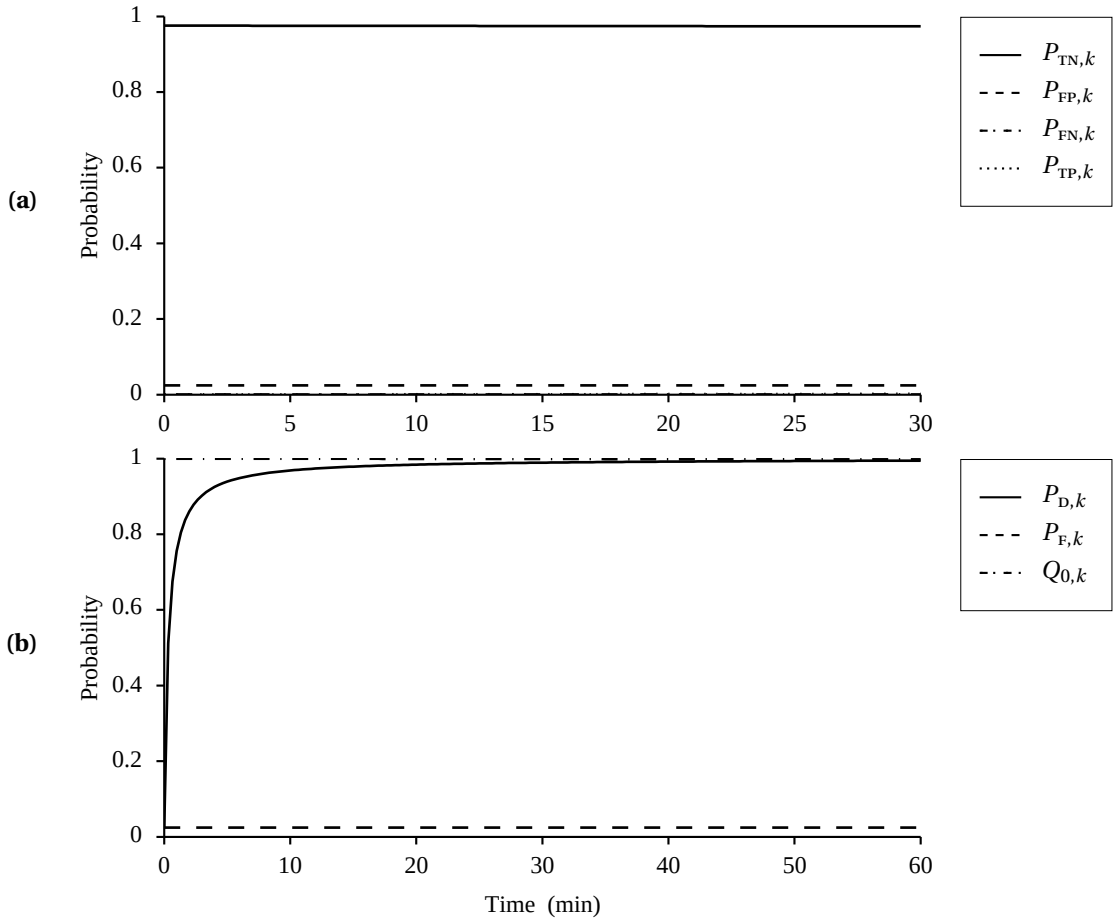


Figure 6.8. Performance metrics for the VTOL aircraft example with observer gain L_1. Plot **(a)** shows the joint probability performance metrics, and plot **(b)** shows the conditional probability performance metrics. Note that the sequences $\{P_{\text{FN},k}\}$ and $\{P_{\text{TP},k}\}$ are barely distinguishable from zero.

Table 6.2. Steady-state values of the performance metrics for the VTOL aircraft example. For each observer gain L_i, the steady-state value is taken to be the value achieved at the final time step N.

	Performance Metrics					
Gain	$P_{\text{TN},N}$	$P_{\text{FP},N}$	$P_{\text{FN},N}$	$P_{\text{TP},N}$	$P_{\text{F},N}$	$P_{\text{D},N}$
L_1	0.9735	0.02439	1.082×10^{-5}	1.998×10^{-3}	0.02444	0.9946
L_2	0.9735	0.02439	4.496×10^{-5}	1.965×10^{-3}	0.02444	0.9776
L_3	0.9735	0.02439	1.082×10^{-5}	1.998×10^{-3}	0.02444	0.9946

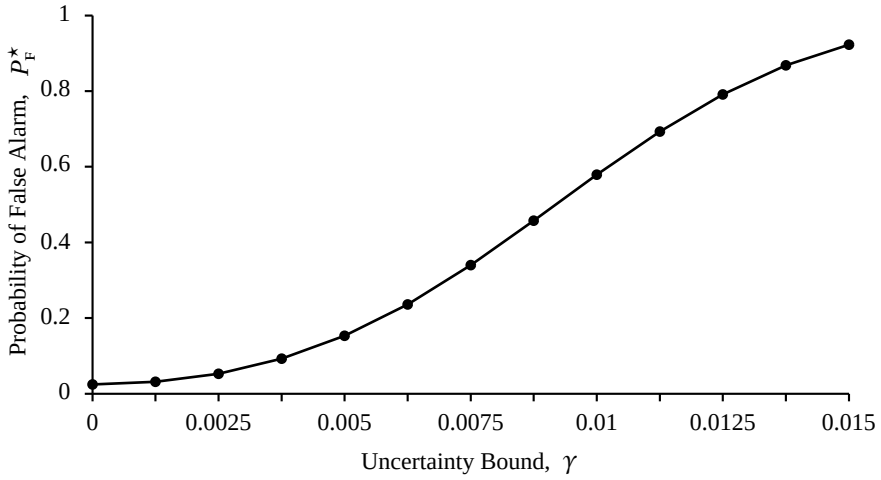


Figure 6.9. Worst-case probability of false alarm $P_F^\star$ for the VTOL aircraft example with additive model uncertainty (see Figure 6.7), where $\Delta \in \mathbf{\Delta}_{2,\text{LTV}}(\gamma)$.

and the weight matrices are $W_1 = I_{2\times 2}$ and $W_2 = I_{4\times 4}$. Because the worst-case optimization problems involve a large number of constraints and decision variables, we shorten the time horizon to 1 minute (i.e., $N = 1,200$). We use the residual generator based on the observer gain L_1. The MATLAB toolbox YALMIP [63] is used to formulate the optimization problem, which is solved by SeDuMi [90]. The resulting worst-case values are plotted in Figure 6.9 for γ ranging from 0 to 0.015.

Finally, we consider the problem of finding the worst-case additive uncertainty, with respect to the probability of detection. As in the previous experiment, we assume that $W_1 = I$, $W_2 = I$, and Δ lies in the set

$$\mathbf{\Delta}_{2,\text{LTV}}(\gamma) = \left\{\Delta\colon \ell_2^2 \to \ell_2^4 \,:\, \Delta \text{ LTV, causal stable }, \|\Delta\|_{i2} < \gamma\right\}.$$

Again, to keep the size of the optimization problem manageable, we reduce the time horizon to 1 minute. Let ϑ be the fault parameter sequence in which both faults occur at $t = 10\,\text{s}$ or $k = 200$. Again, YALMIP [63] is used to formulate the optimization problem in MATLAB, and SeDuMi [90] is used to compute an optimal solution. The resulting worst-case values are plotted in Figure 6.10 for γ ranging from 0.11 to 0.18. As in the air-data sensor example (Section 6.3), computing the worst-case probability of detection $P_D^\star$ is a matter of suppressing the nominal fault signal. In this case, it becomes increasingly difficult to find a Δ with bounded induced 2-norm to suppress the effect of the fault signal as the simulation time horizon NT_s is increased or as the norm-bound γ is decreased.

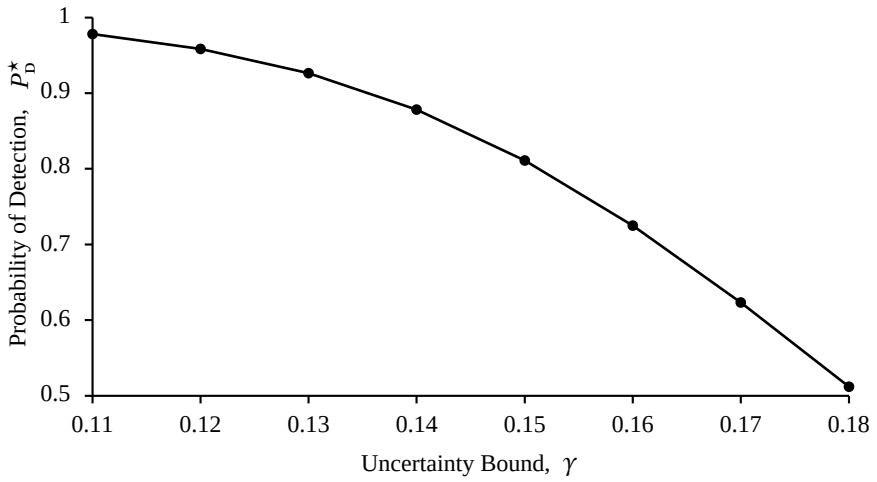


Figure 6.10. Worst-case probability of detection $P_D^\star$ for the VTOL aircraft example with additive model uncertainty (see Figure 6.7), where $\Delta \in \mathbf{\Delta}_{2,\mathrm{LTV}}(\gamma)$.

Chapter 7

Conclusions & Future Work

This dissertation considers the problem of rigorously quantifying the performance of a fault diagnosis scheme using accurate and efficient numerical algorithms. In Chapter 3, we established a set of quantitative performance metrics, based on a sequence of hypothesis tests, that apply to the class of parametric fault diagnosis problems. We also showed how these performance metrics can be decoupled into two parts: one quantifying the reliability of the underlying system and the other quantifying the performance of the fault diagnosis scheme. Throughout the dissertation, we emphasized simpler problems with exact solutions over more complex problems with approximate solutions. Hence, in Chapter 4, we established a set of sufficient assumptions, which limit the class of fault diagnosis problems in such a way that the performance metrics can be computed efficiently and accurately. To make these assumptions less restrictive, and to address the common problem of modeling errors, we considered the effects of uncertainty in Chapter 5. For various types of uncertainty, we formulated convex optimization problems that define the worst-case performance of a given fault diagnosis scheme. Finally, in Chapter 6 we demonstrated the application of our framework on two aerospace examples.

The framework developed in this dissertation is just a preliminary step toward a more rigorous approach to the design and analysis of fault diagnosis schemes. Although there are many avenues open for future research, the following issues seem to provide natural extensions to the results presented here.

1. **Tractable Markov chains:** As discussed in Remark 4.13, the graph-theoretic condition stated in Theorem 4.12 is sufficient but not necessary for a Markov chain with time-varying transition probability matrices to be tractable. The simple case considered in Example 4.18 seems to suggest that there may be more complex conditions involving multiple graphs that are indeed necessary for tractability. Finding such a necessary condition would make it possible to study the tractability of a wider class of non-time-homogeneous Markov chains.

2. **Decision functions:** Although threshold decision functions are commonly found in the fault diagnosis literature, there are a number of other popular decision functions

that deserve equal attention.

- *Likelihood ratio tests:* As stated in Chapter 3, likelihood ratio tests provide the highest probability of detection for a given probability of false alarm (see Lemma 3.10). A decision function based on a likelihood ratio test between two hypotheses $H_{0,k}$ and $H_{1,k}$ can be written as

$$\delta(k, r_{0:k}) = \begin{cases} 0 & \text{if } \Lambda(r_{0:k}) > \varepsilon_k \\ 1 & \text{otherwise,} \end{cases}$$

  where the likelihood ratio test statistic is defined as

$$\Lambda(r_{0:k}) \coloneqq \frac{p_r(r_{0:k} \mid H_{0,k})}{p_r(r_{0:k} \mid H_{1,k})}.$$

  Note that, at each time k, the decision function δ depends on the entire sequence of residuals $r_{0:k}$. Therefore, δ must be written in terms of a dynamic decision function with a state that "remembers" the past values of r_k, or the decision function must become increasingly complex with each time step.

- *Decision functions based on norms:* There are a number of decision functions in the fault detection literature that are based on taking some norm of the residual signal. For example, when the residual is vector-valued, the decision function may be of the form

$$\delta(k, r_k) \coloneqq \mathbb{1}\big(\|r_k\|_2 > \varepsilon_k\big),$$

  where $\mathbb{1}$ is the indicator function. Similarly, one may define a norm over some time window T, as follows:

$$\|r_{0:k}\|_{2,T} \coloneqq \left(\frac{1}{T} \sum_{\ell=\max\{0,k-T+1\}}^{k} \|r_\ell\|_2^2\right)^{1/2}$$

  The corresponding decision function is

$$\delta(k, r_{0:k}) \coloneqq \mathbb{1}\big(\|r_{0:k}\|_{2,T} > \varepsilon_k\big).$$

  Both of these norm-based decision functions can be found in the literature (see [24] and references therein); however, neither of them fit the computational framework presented here.

- *Dynamic decision functions applied to correlated residuals:* Recall that in Section 4.4.2, the state of the dynamic decision function is a Markov chain if and only if the residuals are Gaussian and uncorrelated in time. This strong assumption usually only occurs when the noise signal is added directly to the system output as

measurement noise. Hence, the applicability of dynamic decision functions would be significantly increased if the Gaussian residuals were allowed to be correlated in time. Even if exact results cannot be obtained, bounds on the performance metrics could still be useful in most applications.

3. **Model uncertainties:** In Section 5.4.1, we present interpolation results in which the induced 2-norm of the interpolating operator Δ is bounded. Then, in Section 5.4.2, we show how these results can be used to form convex optimization problems that yield the worst-case performance. Using a similar approach, we may also consider uncertainties with bounded induced ∞-norm. Indeed, in [74], Poolla *et al.* prove an interpolation result, where Δ is LTI casual, stable, and

$$\|\Delta\|_{i\infty} := \sup_{\alpha \neq 0} \frac{\|\Delta\alpha\|_\infty}{\|\alpha\|_\infty} < \gamma,$$

for some $\gamma > 0$. The necessary and sufficient conditions for the existence of such an interpolating operator are stated in terms of the feasibility of a linear program (LP). The linear constraints in this LP are readily incorporated into our worst-case optimization problems. The LTV version of this result, due to Khammash and Pearson [55], can also be used as constraints in our worst-case optimization problems.

4. **Approximations:** Although the emphasis throughout this dissertation has been placed on exact computation, there is considerable value in computing approximate solutions with known error bounds. Such approximate algorithms would fulfill the same practical purpose of their more exact counterparts while saving a great deal of computation time. Indeed, such algorithms could be used for preliminary analyses to determine which input and fault signals are most interesting. Then, the exact algorithms could be used to refine the approximate solutions.

References

[1] S. Asmussen and P. W. Glynn, *Stochastic Simulation: Algorithms and Analysis*, Springer, New York, 2007.

[2] M. Basseville and I. V. Nikiforov, *Detection of Abrupt Changes: Theory and Application*, PTR Prentice Hall, Englewood Cliffs, NJ, 1993.

[3] R. V. Beard, *Failure Accommodation in Linear Systems Through Self-Reorganization*, Ph.D. Thesis, Massachusetts Intitute of Technology, 1971.

[4] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*, Springer-Verlag, New York, 2nd ed., 1985.

[5] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, New York, 2004.

[6] D. Carbaugh, D. Forsythe, and M. McIntyre, *Erroneous flight instrument information*, AERO Magazine, 8 (1998), pp. 10–21.

[7] C.-T. Chen, *Linear System Theory and Design*, Oxford University Press, New York, 3 ed., 1999.

[8] J. Chen and R. J. Patton, *Optimal filtering and robust fault diagnosis of stochastic systems with unknown disturbances*, IEE Proceedings–Control Theory and Applications, 143 (1996), pp. 31–36.

[9] J. Chen and R. J. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems*, Kluwer Academic, Boston, MA, 1999.

[10] J. Chen, R. J. Patton, and H.-Y. Zhang, *Design of unknown input observers and robust fault detection filters*, International Journal of Control, 63 (1996), pp. 85–105.

[11] J. Chen and S. Wang, *Validation of linear fractional uncertain models: Solutions via matrix inequalities*, IEEE Transactions on Automatic Control, 41 (1996), pp. 844–849.

[12] R. H. Chen, D. L. Mingori, and J. L. Speyer, *Optimal stochastic fault detection filter*, Automatica, 39 (2003), pp. 377–390.

[13] E. Y. Chow and A. S. Willsky, *Issues in the development of a general design algorithm for reliable failure detection*, in Proceedings of the 19th IEEE Conference of Decision and Control, Albuquerque, NM, Dec. 1980, pp. 1006–1012.

[14] ——, *Analytical redundancy and the design of robust failure detection systems*, IEEE Transactions on Automatic Control, AC-29 (1984), pp. 603–614.

[15] W. H. Chung and J. L. Speyer, *A game theoretic fault detection filter*, IEEE Transactions on Automatic Control, 43 (1998), pp. 143–161.

[16] A. Cobham, *The intrinsic computational difficulty of functions*, in Proceedings of the 1964 Congress for Logic, Methodology, and the Philosophy of Science, Jerusalem, Aug. 1964, pp. 24–30.

[17] W. J. Cody, *Rational Chebyshev approximations for the error function*, Mathematics of Computation, 23 (1969), pp. 631–637.

[18] R. P. G. Collinson, *Introduction to Avionics Systems*, Kluwer Academic, Boston, MA, 2nd ed., 2003.

[19] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, MIT Press, Cambridge, MA, 3rd ed., 2009.

[20] O. L. V. Costa, M. D. Fragoso, and R. P. Marques, *Discrete-Time Markov Jump Linear Systems*, Springer-Verlag, London, 2005.

[21] S. Dasgupta, C. Papadimitriou, and U. Vazirani, *Algorithms*, McGraw–Hill, Boston, MA, 2008.

[22] M. H. DeGroot, *Optimal Statistical Decisions*, McGraw–Hill, New York, 1970.

[23] J. W. Demmel, *Applied Numerical Linear Algebra*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997.

[24] S. X. Ding, *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*, Springer-Verlag, Berlin, Jan. 2008.

[25] D. A. dos Santos and T. Yoneyama, *A Bayesian solution to the multiple composite hypothesis testing for fault diagnosis in dynamic systems*, Automatica, 47 (2011), pp. 158–163.

[26] R. K. Douglas and J. L. Speyer, *Robust fault detection filter design*, Journal of Guidance, Control, and Dynamics, 19 (1996), pp. 214–218.

[27] G. Dullerud and R. Smith, *A nonlinear functional approach to LFT model validation*, Systems & Control Letters, 47 (2002), pp. 1–11.

[28] G. E. Dullerud and F. Paganini, *A Course in Robust Control Theory: A Convex Approach*, Springer, New York, 2000.

[29] J. Edmonds, *Paths, trees, and flowers*, Canadian Journal of Mathematics, 17 (1965), pp. 449–467.

[30] D. C. Edwards, C. E. Metz, and M. A. Kupinski, *Ideal observers and optimal ROC hypersurfaces in N-class classification.*, IEEE Transactions on Medical Imaging, 23 (2004), pp. 891–895.

[31] J. P. Egan, *Signal Detection Theory and ROC Analysis*, Academic Press, New York, 1975.

[32] A. Emami-Naeini, M. M. Akhter, and S. M. Rock, *Effect of model uncertainty on failure detection: The threshold selector*, IEEE Transactions on Automatic Control, 33 (1988), pp. 1106–1115.

[33] R. Everson and J. Fieldsend, *Multi-class ROC analysis from a multi-objective optimisation perspective*, Pattern Recognition Letters, 27 (2006), pp. 918–927.

[34] T. Fawcett, *An introduction to ROC analysis*, Pattern Recognition Letters, 27 (2006), pp. 861–874.

[35] I. P. Fedčina, *A criterion for the solvability of the Nevanlinna–Pick interpolation problem*, Matematicheskie Issledovaniya, 7 (1972), pp. 213–227.

[36] C. Ferri, J. Hernández-Orallo, and M. A. Salido, *Volume under the ROC Surface for multi-class problems*, in Machine Learning: ECML 2003. Proceedings of the 14th European Conference on Machine Learning, Cavtat-Dubrovnik, Sept. 2003, pp. 108–120.

[37] A. Genz, *Numerical computation of rectangular bivariate and trivariate normal and t probabilities*, Statistics and Computing, 14 (2004), pp. 251–260.

[38] A. Genz and F. Bretz, *Computation of Multivariate Normal and t Probabilities*, Springer-Verlag, Berlin, 2009.

[39] J. Gertler and Q. Luo, *Robust isolable models for failure diagnosis*, AIChE Journal, 35 (1989), pp. 1856–1868.

[40] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics: A Foundation for Computer Science*, Addison–Wesley, Reading, MA, 2nd ed., 1994.

[41] M. S. Hamada, A. G. Wilson, C. S. Reese, and H. F. Martz, *Bayesian Reliability*, Springer, New York, 2008.

[42] F. Hamelin and D. Sauter, *Robust fault detection in uncertain dynamic systems*, Automatica, 36 (2000), pp. 1747–1754.

[43] D. J. Hand and R. J. Till, *A simple generalisation of the area under the ROC curve for multiple class classification problems*, Machine Learning, 45 (2001), pp. 171–186.

[44] S. Hansen, M. Blanke, and J. Adrian, *Diagnosis of UAV pitot tube failure using statistical change detection*, in Proceedings of the 7th IFAC Symposium on Intelligent Autonomous Vehicles, Lecce, Italy, Sept. 2010.

[45] X. He and E. C. Frey, *The meaning and use of the volume under a three-class ROC surface (VUS)*, IEEE Transactions on Medical Imaging, 27 (2008), pp. 577–588.

[46] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, New York, 1985.

[47] R. Isermann, *Process fault detection based on modeling and estimation methods—A survey*, Automatica, 20 (1984), pp. 387–404.

[48] ——, *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, Springer-Verlag, Berlin, 2006.

[49] R. Isermann and P. Ballé, *Trends in the application of model-based fault detection and diagnosis of technical processes*, Control Engineering Practice, 5 (1997), pp. 709–719.

[50] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, Academic Press, New York, 1970.

[51] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*, Prentice Hall, Upper Saddle River, NJ, 2000.

[52] R. E. Kálmán, *A new approach to linear filtering and prediction problems*, Transactions of the ASME, Series D: Journal of Basic Engineering, 82 (1960), pp. 35–45.

[53] R. E. Kálmán and R. S. Bucy, *New results in linear filtering and prediction theory*, Transactions of the ASME, Series D: Journal of Basic Engineering, 83 (1961), pp. 95–107.

[54] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume II – Detection Theory*, Prentice Hall PTR, Upper Saddle River, NJ, 1998.

[55] M. Khammash and J. B. Pearson, *Performance robustness of discrete-time systems with structured uncertainty*, IEEE Transactions on Automatic Control, 36 (1991), pp. 398–412.

[56] C.-J. Kim and C. R. Nelson, *State-Space Models with Regime Switching: Classical and Gibbs-Sampling Approaches with Applications*, MIT Press, Cambridge, MA, 1999.

[57] W. Krzanowski and D. Hand, *ROC Curves for Continuous Data*, Chapman and Hall/CRC, May 2009.

[58] P. Kudva, N. Viswanadham, and A. Ramakrishna, *Observers for linear systems with unknown inputs*, IEEE Transactions on Automatic Control, AC-25 (1980), pp. 113–115.

[59] P. D. Lax, *Linear Algebra and Its Applications*, John Wiley & Sons, Hoboken, NJ, 2nd ed., 2007.

[60] E. L. Lehmann and J. P. Romano, *Testing Statistical Hypotheses*, Springer, New York, 3rd ed., 2005.

[61] B. C. Levy, *Principles of Signal Detection and Parameter Estimation*, Springer, New York, 2008.

[62] X. Li and K. Zhou, *A time domain approach to robust fault detection of linear time-varying systems*, Automatica, 45 (2009), pp. 94–102.

[63] J. Löfberg, *YALMIP: A toolbox for modeling and optimization in MATLAB*, in Proceedings of the 2004 IEEE International Symposium on Computer Aided Control Systems Design, Taipei, Taiwan, Sept. 2004, pp. 284–289.

[64] D. G. Luenberger, *Observing the state of a linear system*, IEEE Transactions on Military Electronics, 8 (1964), pp. 74–80.

[65] M. Mariton, *Detection delays, false alarm rates and the reconfiguration of control systems*, International Journal of Control, 49 (1989), pp. 981–992.

[66] M. Mariton, *Jump Linear Systems in Automatic Control*, Marcel Dekker, New York, 1990.

[67] R. K. Mehra and J. Peschon, *An innovations approach to fault detection and diagnosis in dynamic systems*, Automatica, 7 (1971), pp. 637–640.

[68] L. A. Mironovski, *Functional diagnosis of linear dynamical systems*, Automation and Remote Control, 40 (1979), pp. 1198–1205.

[69] I. Moir and A. G. Seabridge, *Civil Avionics Systems*, Professional Engineering Publishing, London, 2003.

[70] K. S. Narendra and S. S. Tripathi, *Identification and optimization of aircraft dynamics*, Journal of Aircraft, 10 (1973), pp. 193–199.

[71] J. Neyman and E. S. Pearson, *On the problem of the most efficient tests of statistical hypotheses*, Philosophical Transactions of the Royal Society. Series A, Containing Papers of a Mathematical or Physical Character, 231 (1933), pp. 289–337.

[72] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, McGraw–Hill, Boston, MA, 4th ed., 2002.

[73] M. S. Pepe, *The Statistical Evaluation of Medical Tests for Classification and Prediction*, Oxford University Press, Oxford, 2003.

[74] K. Poolla, P. Khargonekar, A. Tikku, J. Krause, and K. Nagpal, *A time-domain approach to model validation*, IEEE Transactions on Automatic Control, 39 (1994), pp. 951–959.

[75] H. V. Poor, *An Introduction to Signal Detection and Estimation*, Springer-Verlag, New York, 2nd ed., 1994.

[76] H. V. Poor and O. Hadjiliadis, *Quickest Detection*, Cambridge University Press, Cambridge, 2009.

[77] M. Rausand and A. Høyland, *System Reliability Theory: Models, Statistical Methods, and Applications*, Wiley-Interscience, 2nd ed., 2004.

[78] A. Ray and R. Luck, *An introduction to sensor signal validation in redundant measurement systems*, IEEE Control Systems Magazine, 11 (1991), pp. 44–49.

[79] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer, New York, 2nd ed., 2004.

[80] M. Rosenblum and J. Rovnyak, *Hardy Classes and Operator Theory*, Oxford University Press, New York, 1985.

[81] J. S. Rosenthal, *A First Look at Rigorous Probability Theory*, World Scientific, Hackensack, NJ, 2nd ed., 2006.

[82] H. L. Royden and P. M. Fitzpatrick, *Real Analysis*, Prentice Hall, Boston, MA, 4th ed., 2010.

[83] M. Saif and Y. Guan, *A new approach to robust fault detection and identification*, IEEE Transactions on Aerospace and Electronic Systems, 29 (1993), pp. 685–695.

[84] A. N. Shiryaev, *On optimum methods in quickest detection problems*, Theory of Probability and Its Applications, VIII (1963), pp. 22–46.

[85] N. D. Singpurwalla, *Reliability and Risk: A Bayesian Perspective*, John Wiley & Sons, Chichester, Aug. 2006.

[86] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, John Wiley & Sons, Chichester, 2nd ed., Dec. 2005.

[87] R. Smith, G. Dullerud, S. Rangan, and K. Poolla, *Model validation for dynamically uncertain systems*, Mathematical Modelling of Systems, 3 (1997), pp. 43–58.

[88] A. Srinivasan, *Note on the location of optimal classifiers in N-dimensional ROC space*, Technical Report (PRG-TR-2-99), Programming Research Group, Oxford University Computing Laboratory, 1999.

[89] J. Stoustrup, H. Niemann, and A. la Cour-Harbo, *Optimal threshold functions for fault detection and isolation*, in Proceedings of the 2003 American Control Conference, Denver, CO, June 2003, pp. 1782–1787.

[90] J. F. Sturm, *Using SeDuMi 1.02, A MATLAB toolbox for optimization over symmetric cones*, Optimization Methods and Software, 11 (1999), pp. 625–653.

[91] A. Tikku and K. Poolla, *Robust performance against slowly-varying structured perturbations*, in Proceedings of the 32nd IEEE Conference on Decision and Control, San Antonio, TX, Dec. 1993, pp. 990–995.

[92] O. Toker and J. Chen, *On computational complexity of invalidating structured uncertainty models*, Systems & Control Letters, 33 (1998), pp. 199–207.

[93] H. L. Van Trees, *Detection, Estimation, and Modulation Theory. Part 1: Detection, Estimation, and Linear Modulation Theory*, John Wiley & Sons, New York, 2001.

[94] H. Wang and G.-H. Yang, *Fault detection observer design in low frequency domain*, in Proceedings of the 15th IEEE International Conference on Control Applications, Oct. 2007, pp. 976–981.

[95] H. B. Wang, J. L. Wang, and J. Lam, *Worst-case fault detection observer design: Optimization approach*, Journal of Optimization Theory and Applications, 132 (2007), pp. 475–491.

[96] X. Wei and M. Verhaegen, *Robust fault detection observer design for linear uncertain systems*, International Journal of Control, 84 (2011), pp. 197–215.

[97] T. J. Wheeler, P. Seiler, A. K. Packard, and G. J. Balas, *Performance analysis of fault detection systems based on analytically redundant linear time-invariant dynamics*, in Proceedings of the 2011 American Control Conference, San Francisco, CA, June 2011, pp. 214–219.

[98] ——, *Performance analysis of LTV fault detection systems with additive faults*, in Proceedings of the 50th IEEE Conference on Decision and Control, Orlando, FL, Dec. 2011.

[99] D. Williams, *Probability with Martingales*, Cambridge University Press, New York, 1991.

[100] A. S. Willsky and H. L. Jones, *A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems*, IEEE Transactions on Automatic Control, 21 (1976), pp. 108–112.

[101] G. Wolodkin and K. Poolla, *Spectral power distribution using time-varying operators*, in Proceedings of the 1994 American Control Conference, Baltimore, MD, June 1994, pp. 3147–3151.

[102] Y. Xiong and M. Saif, *Robust fault isolation observer design*, in Proceedings of the 1999 American Control Conference, San Diego, CA, June 1999, pp. 2077–2081.

[103] Y. C. Yeh, *Triple-triple redundant 777 primary flight computer*, in Proceedings of the 1996 IEEE Aerospace Applications Conference, Aspen, CO, Feb. 1996, pp. 293–307.

[104] ——, *Safety critical avionics for the 777 primary flight controls system*, in Proceedings of the 20th Digital Avionics Systems Conference, Daytona Beach, FL, Oct. 2001, pp. 1.C.2.1–1.C.2.11.

[105] G. G. Yin and C. Zhu, *Hybrid Switching Diffusions: Properties and Applications*, Springer, New York, 2010.

[106] L. A. Zadeh, *Optimality and non-scalar-valued performance criteria*, IEEE Transactions on Automatic Control, AC-8 (1963), pp. 59–60.

[107] M. Zhong, Q. Ding, and P. Shi, *Parity space-based fault detection for Markovian jump systems*, International Journal of Systems Science, 40 (2009), pp. 421–428.

[108] M. Zhong, S. X. Ding, and E. L. Ding, *Optimal fault detection for linear discrete time-varying systems*, Automatica, 46 (2010), pp. 1395–1400.

[109] M. Zhong, J. Lam, S. X. Ding, and P. Shi, *Robust fault detection of Markovian jump systems*, Circuits, Systems & Signal Processing, 23 (2004), pp. 387–407.

[110] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice Hall, Englewood Cliffs, NJ, 1995.

[111] X.-H. Zhou, N. A. Obuchowski, and D. K. McClish, *Statistical Methods in Diagnostic Medicine*, John Wiley & Sons, Hoboken, NJ, 2011.